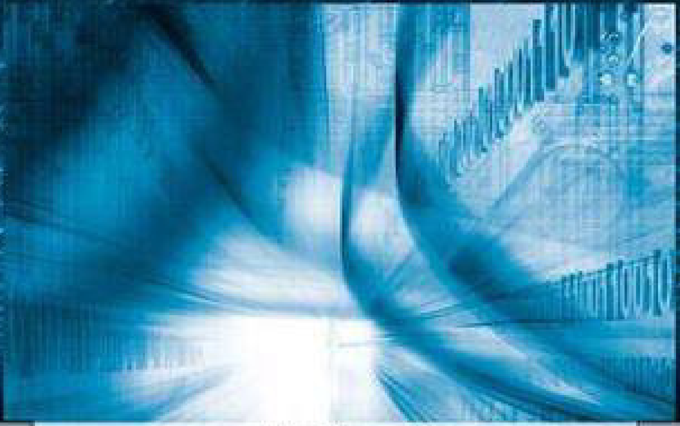
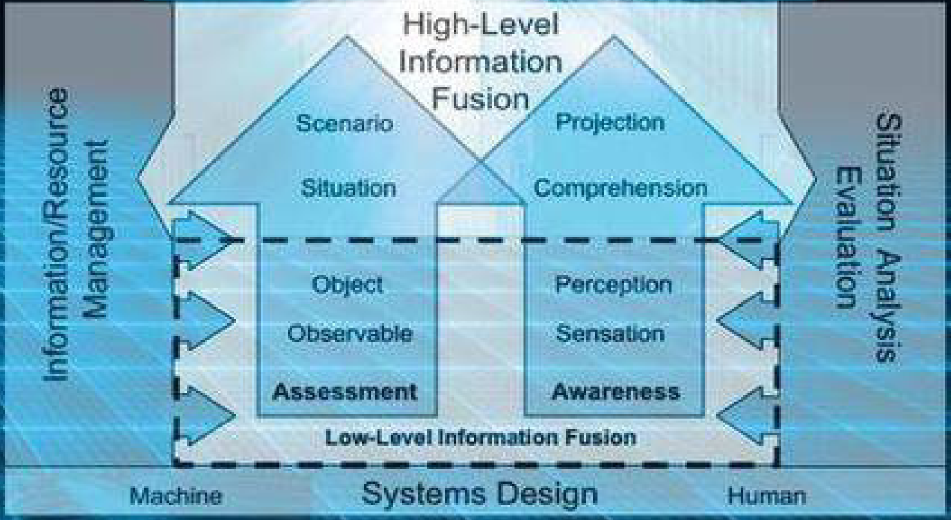
HIGH-LEVEL INFORMATION FUSION

MANAGEMENT AND SYSTEMS DESIGN





ERIK BLASCH

ÉLOI BOSSÉ

DALE A. LAMBERT

High-Level Information Fusion Management and Systems Design

High-Level Information Fusion Management and Systems Design

Erik Blasch
Éloi Bossé
Dale A. Lambert

Editors

ARTECH
HOUSE

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the U.S. Library of Congress.

British Library Cataloguing in Publication Data
A catalog record for this book is available from the British Library.

ISBN-13: 978-1-60807-151-7

Cover design by Vicki Kane

10 9 8 7 6 5 4 3 2 1

To Doctors Bruce, Barbara, and Jitka Blasch
who always encouraged me to explore my interests
Erik Blasch

To my three children: Étienne, Julianne, and Samuel Éloi Bossé

To Keetza for lovingly tolerating all those squiggly symbols
Dale A. Lambert

Contents

CHAPTER 1

Introduction

1.1 High-Level Information Fusion (HLIF) Challenges

1.2 Book Structure

  1.2.1 Perspectives from Australian Contributions

  1.2.2 Perspectives from Canadian Contributions

  1.2.3 Perspectives from United States Contributions

1.3 A Science of High-Level Information Fusion

  References

PART I

Information Fusion Concepts

CHAPTER 2

Situation Assessment and Situation Awareness

2.1 Introduction

2.2 Situation Awareness and Situation Assessment Defined

2.3 Situation Awareness (SAW) Models

  2.3.1 Endsley's SAW Model

  2.3.2 Recognition Primed Decision (RPD) Making Model

2.4 Situation Assessment Models

  2.4.1 Data Fusion Information Group Model

  2.4.2 Situational Assessment Models for the User

2.5 Situational Assessment Model Based on Activities of Interest

  2.5.1 Syntactic algorithms and Semantic Synonyms in Information Fusion Analysis

  2.5.2 Definition of a Situation

  2.5.3 The Situation Awareness Reference Model

2.6 Current Information Fusion Situation Assessment Reference Model for

Information Fusion
2.7 Discussion
  2.7.1 Situation Assessment Representations and Theory
  2.7.2 Situation Assessment Metrics
  2.7.3 SA/SAW Issues and Challenges
2.8 Conclusions
  References

CHAPTER 3

The State Transition Data Fusion Model

3.1 Information Revolution
  3.1.1 Situation Awareness
  3.1.2 Data Fusion
  3.1.3 Renaissance
3.2 State Transitions
  3.2.1 Classification
  3.2.2 States
  3.2.3 Transitions
  3.2.4 JDL States in the World
3.3 The STDF Fusion Process
  3.3.1 Prediction, Observation, and Explanation
  3.3.2 The General Form of a Fusion Process
  3.3.3 JDL Assessments
3.4 Level 0 Fusion
  3.4.1 Level 0 Signal Fusion
  3.4.2 Level 0 Textual Fusion
3.5 Level 1 Fusion
  3.5.1 Level 1 Signal Fusion
  3.5.2 Level 1 Textual Fusion
3.6 Level 2 Fusion
3.7 Level 3 Fusion
  References

CHAPTER 4

Formalization of Situation Analysis Through Interpreted Systems Semantics

4.1 Introduction
  4.1.1 Formal Models of Higher Levels of Information Fusion
  4.1.2 Situations in State Spaces
4.2 Background
  4.2.1 Interpreted Systems
  4.2.2 Different Kinds of Interpreted Systems
4.3 Formalization of the Situation Analysis Process
  4.3.1 Situation
  4.3.2 Situation Awareness
  4.3.3 Situation Perception and Comprehension
  4.3.4 Situation Analysis
4.4 Illustrations on a Surveillance Scenario
  4.4.1 Situation
  4.4.2 Situation Awareness
  4.4.3 Belief, Revision, and Update
  4.4.4 Situation Analysis
4.5 Conclusions
  References

PART II

Distributed Information Fusion and Management

CHAPTER 5

The Role of Information Management to Support High-Level Fusion

5.1 Introduction: What Is Information Management and Why Do We Care?
5.2 Model of Information Management
  5.2.1 Managed Information Objects
  5.2.2 Actors
  5.2.3 Service Layers
  5.2.4 Information Spaces
  5.2.5 Utility of the Information Management (IM) Model

5.3 Information Management Challenges in a Coalition Environment
5.4 Information Management Best Practices
    5.4.1 Information Sharing
    5.4.2 Reducing Complexity
    5.4.3 Control and Flexibility
5.5 Information Management Support to Information Fusion
    5.5.1 Information Lifecycle
    5.5.2 Syntactic and Semantic Interoperability
    5.5.3 Management and Exploitation of Contextual Information
    5.5.4 Management and Exploitation of Unstructured Information
    5.5.5 Information Management as a Service
    5.5.6 Workflow
5.6 Information Management from an Agent Perspective
5.7 Conclusions
    References

CHAPTER 6

Coalition Distributed Information Fusion Testbed

6.1 Models of Collaboration
    6.1.1 Technology Showcase
    6.1.2 Technology Demonstration
    6.1.3 Technology Evaluation
    6.1.4 Technology Sharing
    6.1.5 Joint Development
    6.1.6 Joint Ownership
6.2 Requirements
    6.2.1 Provide Simulated Information Feeds
    6.2.2 Real-Time Performance
    6.2.3 Distributed Architecture
    6.2.4 Integrate Heterogeneous Systems
    6.2.5 Loose Coupling Between Components
    6.2.6 Dynamic Resource Management and Process Control
6.3 CoAX (Collaboration 2002 Experiment)

6.4 Architecture
    6.4.1 Simulation Layer
    6.4.2 Information Management Layer
    6.4.3 Information Fusion Layer
    6.4.4 Resource Management Layer
    6.4.5 Human-Machine Interface Layer
6.5 Conclusion
    References

CHAPTER 7

Information Fusion and Resource Management Testbed

7.1 Introduction
7.2 INFORM Lab architecture
    7.2.1 OODA Agent Components
    7.2.2 Platforms
    7.2.3 Default Communicator
    7.2.4 Goals
    7.2.5 Situation Evidence
    7.2.6 Agent Affliations and Relationships
    7.2.7 Services
    7.2.8 Extension Mechanisms
7.3 INFORM Lab Implementation
7.4 Tests and Validation
7.5 Conclusion
    References

CHAPTER 8

The Legal Agreement Protocol

8.1 Conceptualization
    8.1.1 Decentralization
    8.1.2 Ubiquity
    8.1.3 Automation
    8.1.4 Integration

8.2 Formalization
    8.2.1 Contract Formation
    8.2.2 Contract Performance
    8.2.3 Contract Remedies
8.3 Computation
    8.3.1 Contract Formation
    8.3.2 Contract Performance
    8.3.3 Contract Remedies
8.4 Sample Vignette
    References

PART III

Human-System Interaction

CHAPTER 9

User-Defined Operating Picture (UDOP)

9.1 Introduction
9.2 The Need for a New Picturing Capability: UDOP
    9.2.1 Challenges with Picturing Capabilities
    9.2.2 Potential Universality of Picturing Challenges and Issues
    9.2.3 Impact of Picturing Challenges and Issues
    9.2.4 Defining Users and User Needs
    9.2.5 Current Abilities to Define Own Pictures
    9.2.6 Purposes of Picturing Capabilities
9.3 Characteristics of a UDOP
9.4 Realizing a Future UDOP Capability
    9.4.1 Developing an Understanding of Components and Architectures
    9.4.2 Providing Guidance for Exploitation of UDOP Visualizations
    9.4.3 Feasibility of UDOP
    9.4.4 Way Forward
9.5 A Few Examples of Remaining Issues
    9.5.1 Awareness of Information Sources
    9.5.2 Selecting Information Sources

  9.5.3 Dealing with Remaining Need-to-Know Constraints
  9.5.4 Catering for Varying End User Expertise
9.6 Conclusions
  Acknowledgments
  References

CHAPTER 10

User Information Fusion Decision Making Analysis with the C-OODA Model

10.1 Introduction
10.2 Decision Making Models
  10.2.1 DFIG and OODA Loop
  10.2.2 Multiplayer OODA
10.3 The Cognitive OODA Loop
  10.3.1 Situation Assessment Models
  10.3.2 SHOR Model for Action
  10.3.3 The Skills-Rules-Knowledge Model
  10.3.4 The Modular OODA (M-OODA)
  10.3.5 The Cognitive Process Included in the C-OODA
10.4 Simulation
10.5 Discussions and Conclusions
  References

PART IV

Scenario-Based Design

CHAPTER 11

Scenario-Based Design for Situation Analysis

11.1 Introduction
11.2 Findings on SBD Methodology
  11.2.1 The Proposed SBD Framework for Military C2
  11.2.2 Specifics of the Military Strike in Atlantis Vignette
11.3 Scenario-Based Design Process Based on Atlantis Problem Scenario
11.4 Conclusion

References

CHAPTER 12

A Coalition Approach to High-Level Information Fusion

12.1 Introduction
  12.1.1 Vision
  12.1.2 Content
12.2 Scenario
12.3 CDIFT
12.4 Platforms, Sensor Models, and Trackers
  12.4.1 Redland Warships
  12.4.2 Convoy
  12.4.3 Commercial Air Corridors
  12.4.4 Blueland Ground-Based Radars
  12.4.5 Events and Order of Battle ORBAT
12.5 Fusion 2+
12.6 Indicators of Collective Behaviour
  12.6.1 Indicators of Collective Behaviour Algorithm
  12.6.2 Identifying Candidate Clusters
  12.6.3 Assessing Confidence
  12.6.4 Inferring Intent
  12.6.5 CDIFT Application
12.7 STDF Model
  12.7.1 State Representation
  12.7.2 Observation
  12.7.3 Prediction and Explanation
12.8 Higher COP
12.9 Urban Operations
12.10 Combat Search and Rescue (CSAR)
12.11 Conclusion
  References

CHAPTER 13

Operating Condition Scenario Modeling for Information Fusion Assessment

13.1 Introduction
  13.1.1 Sensor-Based Classifier Operating Conditions
  13.1.2 Scenario-Based Evaluation
  13.1.3 Design of Experiments for Scenarios
13.2 Operating Condition Model Terminology
  13.2.1 Direct Versus Indirect OCs
  13.2.2 Derived OCs
  13.2.3 Standard OCs Versus Extended OCs
13.3 Operating Condition Model Design
  13.3.1 Bayes Model
  13.3.2 Bayes Fusion from Real World (Scenario) Analysis
13.4 Example Operating Conditions
  13.4.1 Target OCs
  13.4.2 Environmental OCs
  13.4.3 Sensor OCs
  13.4.4 ATC Training OCs
  13.4.5 OC Model
13.5 Conditioning on Operating Conditions
13.6 Conclusions
  Acknowledgments
  References

PART V

Measures of Effectiveness

CHAPTER 14

A Toolbox for the Evaluation of Surveillance Strategies Based on Interpreted Systems

14.1 Introduction
14.2 Situations Generated By Motion And Sensing Strategies
  14.2.1 Visibility-Based Pursuit-Evasion in Graphs
  14.2.2 Sensor Placement Problem
  14.2.3 Exploration

- 14.3 Situation Analysis Toolbox
  - 14.3.1 Countersmuggling Vignette
  - 14.3.2 The Discretization Toolbox
  - 14.3.3 The State Generator
  - 14.3.4 The State Searching Toolbox
  - 14.3.5 The Behavior Simulation Toolbox
  - 14.3.6 The Visualization Toolbox
- 14.4 Conclusions
  - Acknowledgments
  - References

CHAPTER 15

Measuring the Worthiness of Situation Assessment

- 15.1 Introduction
- 15.2 The Situation Assessment Concept
  - 15.2.1 Situation Awareness Reference Model
  - 15.2.2 Activities of Interest Snapshot in Time
  - 15.2.3 Data Information Ratio
- 15.3 Metrics
  - 15.3.1 AOI Score
  - 15.3.2 Measuring How Well We Are Doing
- 15.4 Example
  - 15.4.1 Calculated Example with Few Activities
  - 15.4.2 Simulated Example with Numerous Activities
- 15.5 Conclusions
  - References

CHAPTER 16

Measures of Effectiveness for High-Level Information Fusion

- 16.1 Introduction
- 16.2 Background
  - 16.2.1 Low-Level Versus High-Level Information Fusion
  - 16.2.2 High-Level Information Fusion as a Form of Reasoning

- 16.2.3 Information Fusion Systems Evaluation
- 16.2.4 Quality of Service/Information Research
- 16.2.5 Metric Standardization

16.3 Information Fusion Quality Measures
- 16.3.1 Quality of Service
- 16.3.2 Information Quality

16.4 Information Fusion MOEs
- 16.4.1 Low-Level Information Fusion MOEs
- 16.4.2 High-Level Information Fusion MOEs
- 16.4.3 Organizational Effectiveness
- 16.4.4 Information Gain

16.5 Situation Awareness Example

16.6 Conclusions

References

CHAPTER 17

Summary

17.1 Current Trends

17.2 Future

References

About the Editors

Contributors

Index

CHAPTER 1

Introduction

Dale A. Lambert (Australia), Erik P. Blasch (United States), and Éloi Bossé (Canada)

The promise afforded by the Information Age is matched by the challenge of processing large quantities of imperfect information effectively and efficiently. Information fusion seeks to meet this challenge through the semiautomation of the functionalities of sensation, perception, cognition, comprehension, and projection that is otherwise performed by people for situation awareness (SAW). This book presents the results of *high-level information fusion* (HLIF) research, which deals with the automation of comprehension for situation assessment (SA) and the automation of projection for impact assessment (IA). The book is a collaborative international effort stemming from the Technical Cooperation Program's (TTCP) Command, Control, Communications, and Information Systems Group's (C3I) technical panel on Information Fusion (TP1), and builds upon *Concepts, Models, and Tools for Information Fusion*, previously published for TP1 by Artech House [1]. The remainder of this chapter outlines the grand HLIF challenges, provides the structure of the book with perspectives from the international contributions, and delivers an overview of the approaches to HLIF theory and representations, distributed information management, and systems design issues including user interactions, scenario-based analysis, and evaluation.

High-Level Information Fusion (HLIF) Challenges

Over the past decade, authors have posited grand challenges [2] for HLIF. These serve as a construct for understanding this book.

- *Paradigm Challenge:* How should the interdependency between the sensor fusion and information fusion paradigms be managed?
- *Semantic Challenge:* What symbols should be used and how do those symbols acquire meaning?
- *Epistemic Challenge:* What information should we represent and how should it be represented and processed within the machine?
- *Interface Challenge:* How do we interface people to complex symbolic information stored within machines to provide decision support?
- *System Challenge:* How should we manage information fusion systems formed from combinations of people and machines?

In this text, the System Challenge is further decomposed to consider [3]:

- *Design Challenge:* How should we design information fusion systems formed from combinations of people and machines?
- *Evaluation Challenge:* How should we evaluate the effectiveness of information fusion systems?

Common themes emerging from these challenges include: HLIF theories and paradigms; ontologies and semantic representations; agent and processing models; human-machine interfaces; information management architectures and testbeds; scenario-based design and alternative systems-based design approaches; and information fusion metrics.

Book Structure

The first TTCP Information Fusion book [1] presented the state of the art in high-level information fusion (HLIF) by canvassing the various options available to fusion practitioners. HLIF includes situation and impact assessment, together with their process refinement and user refinement; while low-level information fusion (LLIF) comprises subobject and object assessment [4]. Developments in information fusion stem from the seminal book in data fusion [5] and include LLIF signal, text and image processing, LLIF tracking and identification [6], and elements of high level data fusion [7], with contemporary analysis of knowledge management [8] and human-centered designs [9]. Figure 1.1 discloses the matrix of relationships by paralleling the human process of SAW with the machine process of machine information fusion, and highlighting the use of LLIF products and services to support HLIF outcomes.

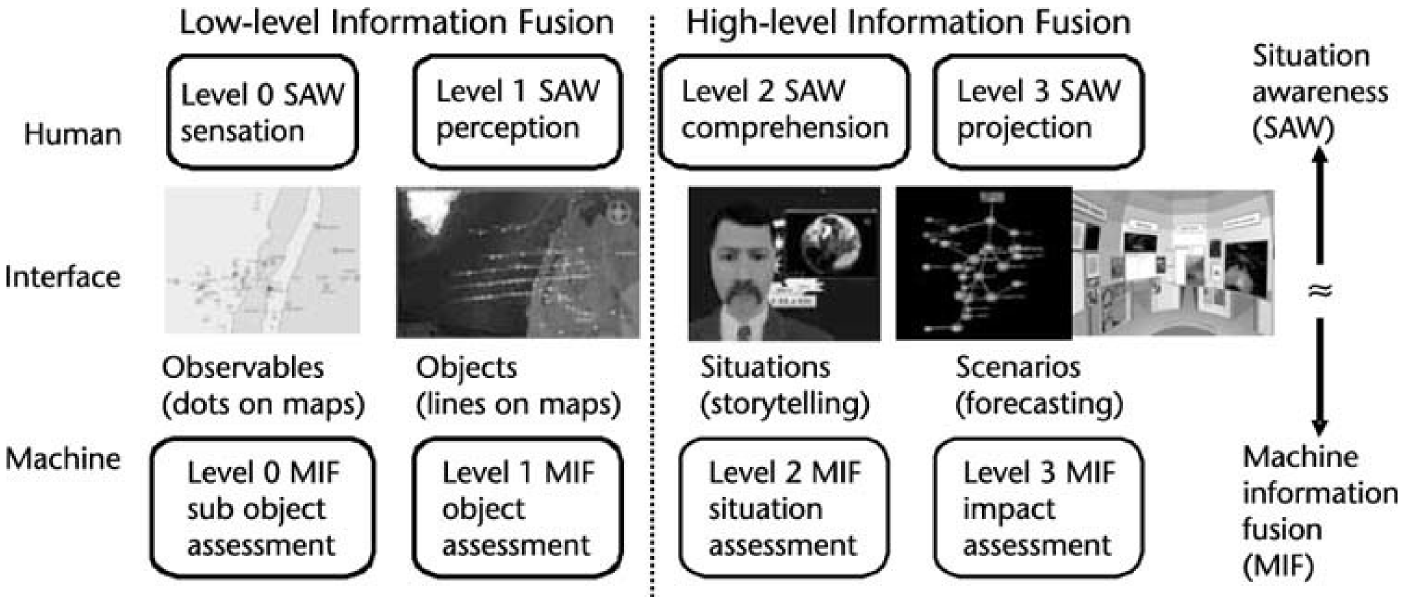


Figure 1.1 Relations between LLIF and HLIF as well as SAW and IF. [10].

This second book extends beyond [1] by identifying which options have been chosen by the TTCP representatives from different nations. To reflect this, the second book is structured around a collection of information fusion themes that expose the similarities and differences in HLIF research from these independent perspectives. The book is divided into five parts, as shown in Figure 1.2: (1) information fusion concepts and representations, (2) distributed information fusion and management, (3) human systems interaction, (4) scenario-based design, and (5) information fusion evaluation.

- Part 1 “Information Fusion Concepts” presents underlying conceptualizations of information fusion including Situation Awareness using the information fusion situation assessment (IFSA) model (Chapter 2), the state transition data fusion model (STDF) (Chapter 3), and interpreted systems (IS) for situation analysis (Chapter 4).

- Part 2 “Distributed Information Fusion and Management” explores information management (Chapter 5), implementations in coalition testbeds (Chapter 6 and Chapter 7), and a framework for dynamically distributing information fusion processes across multiple fusion systems (Chapter 8).
- Part 3 exposes human-system interaction implementations for combining human and machine capabilities of information fusion, such as the user defined operating picture (UDOP) (Chapter 9), and the cognitive observe-orient-decide-act model (C-DODA) (Chapter 10).
- Part 4 reflects upon scenario-based design (SBD) by disclosing the role of crafted scenarios in designing information fusion systems (Chapter 11 and Chapter 12) and elements of operating conditions (Chapter 13).
- Part 5 “Measures of Effectiveness” considers measures of effectiveness for information fusion with an example of situational analysis (Chapter 14), HLIF measures of worthiness (Chapter 15), and measures of effectiveness for SA (Chapter 16).

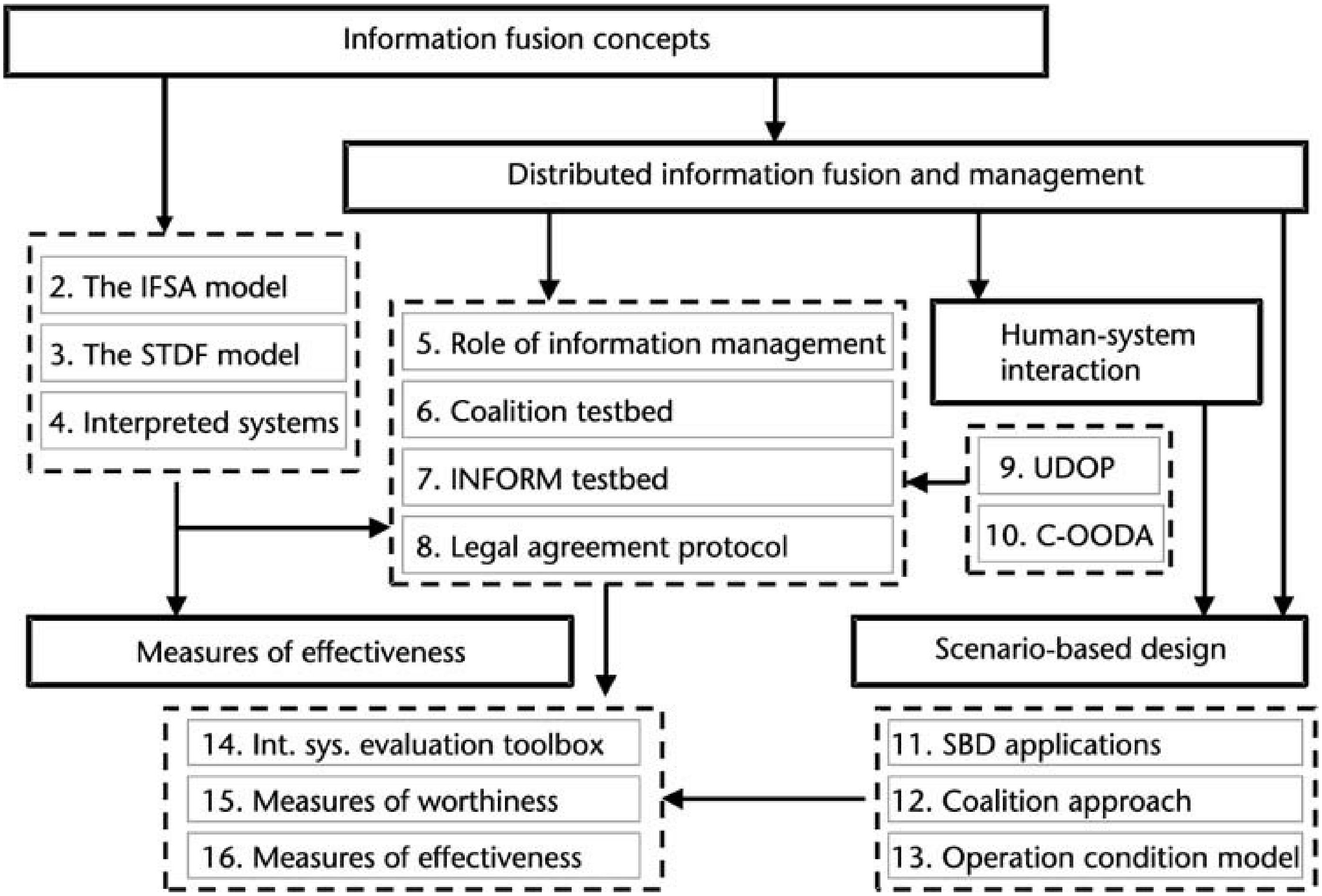


Figure 1.2 Overview of book.

Contributed chapters from multi-national TTCP C3I Technical Panels (TP) are also a feature in the book. They include: Chapter 12, which was prepared by TP1 Information Fusion; Chapter 9, which was formulated by TP2 Command Information Interfaces; Chapter 5, which was written by TP3 Information Management; and Chapter 7, which was drafted by TP4 Dynamic Planning and Scheduling.

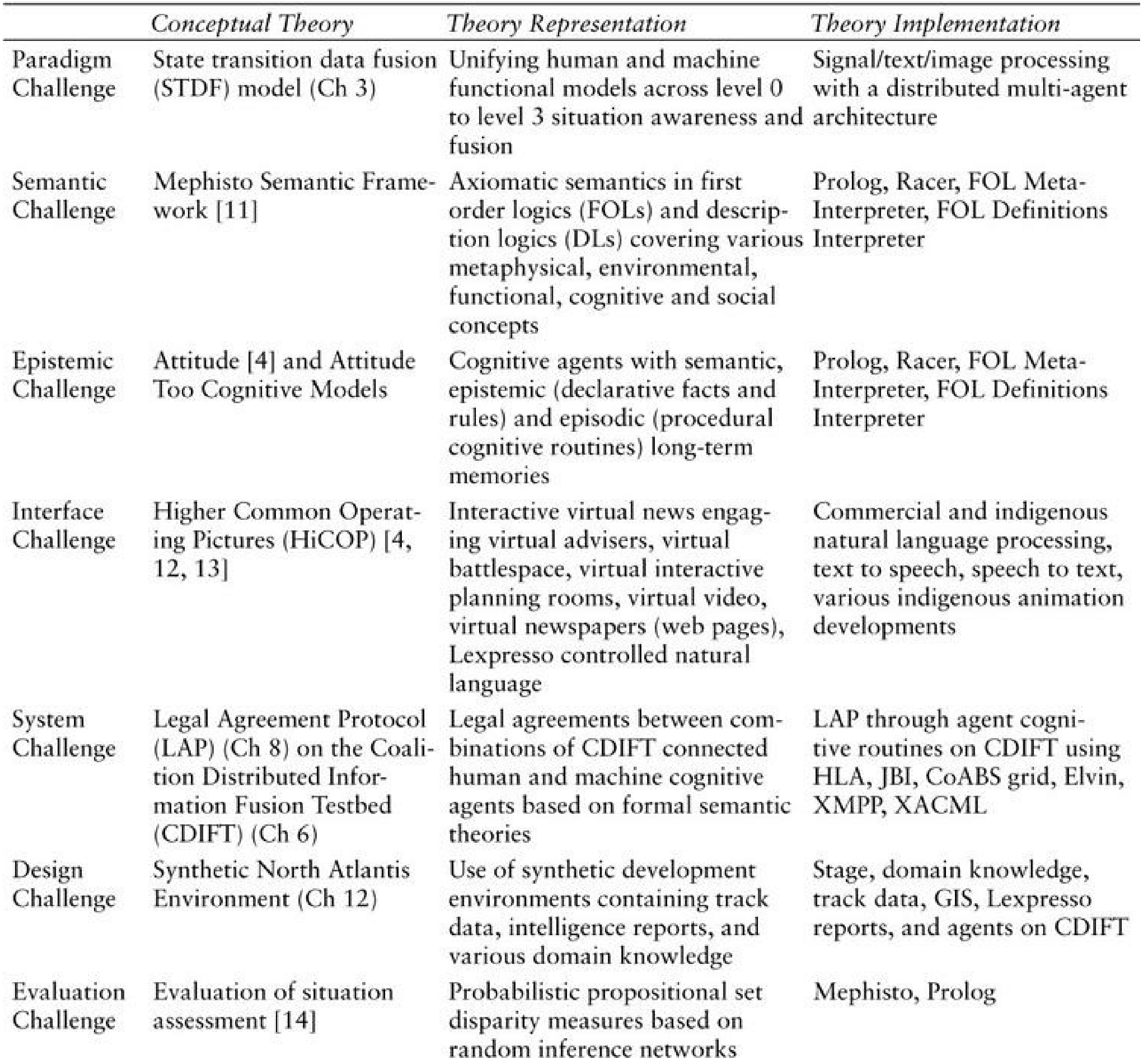
The themed structure of this book is designed to highlight the independent choices made by representatives from different nations. The following subsections summarize how each of the different national perspectives address the grand challenges. Where the approach taken is included within the book, the relevant book chapters are identified. Where the approach taken is not included within the book, previous publications are cited instead.

1.2.1 Perspectives from Australian Contributions

Australia has an integrated program across the HLIF themes of the book. The Information Fusion Concepts and Representations of Part 1 relies upon the State Transition Data Fusion (STDF) model of Chapter 3, which presents a unifying systems model for information fusion across the sensation, perception, comprehension, and projection components. At the higher-levels of fusion, the STDF model assumes the existence of semantic representations implemented through the Mephisto semantic framework [11], while automated reasoning with these semantic representations is conducted through agents complying with the ATTITUDE [4] and ATTITUDE TOO cognitive models. These agents operate on the Coalition Distributed Information Fusion Testbed (CDIFT), as outlined in Part 2, Chapter 6. CDIFT delivers the underlying distributed information fusion and management infrastructure developed by TTCP C3I TP1 to facilitate experimentation with nations' fusion systems. Chapter 8 presents a second aspect of Part 2, the legal agreement protocol, which facilitates the dynamic formation of fusion architectures from agents operating on the CDIFT infrastructure. Part 3 (on human-system interaction) does not contain an Australian chapter, though elements of Australia's Higher Common Operating Pictures (HiCOP) concept are evident in both Chapter 6 and Chapter 12. HiCOP includes implemented interaction between human and machine agent systems through interfaces, like virtual advisers and virtual battlespaces, as seen in Figure 1.1 [4, 12, 13]. Chapter 12 is a collective piece written by members of TTCP C3I TP1. Chapter 12 summarizes the Part 4 scenario-based design approach used by TTCP C3I TP1 to develop a coalition approach to higher-level fusion and reflects the panel session demonstration undertaken by TTCP C3I TP1 at the 12th *International Information Fusion Conference* held in Seattle in 2009. An Australian approach to Part

5 “Measures of Effectiveness” is absent from this book, though the interested reader can consult [14] for an Australian approach to probabilistic propositional set disparity measures. Table 1.1 summarizes the Australian contributions to the HLIF challenges.

Table 1.1 Australian Contributions to HLIF Challenges

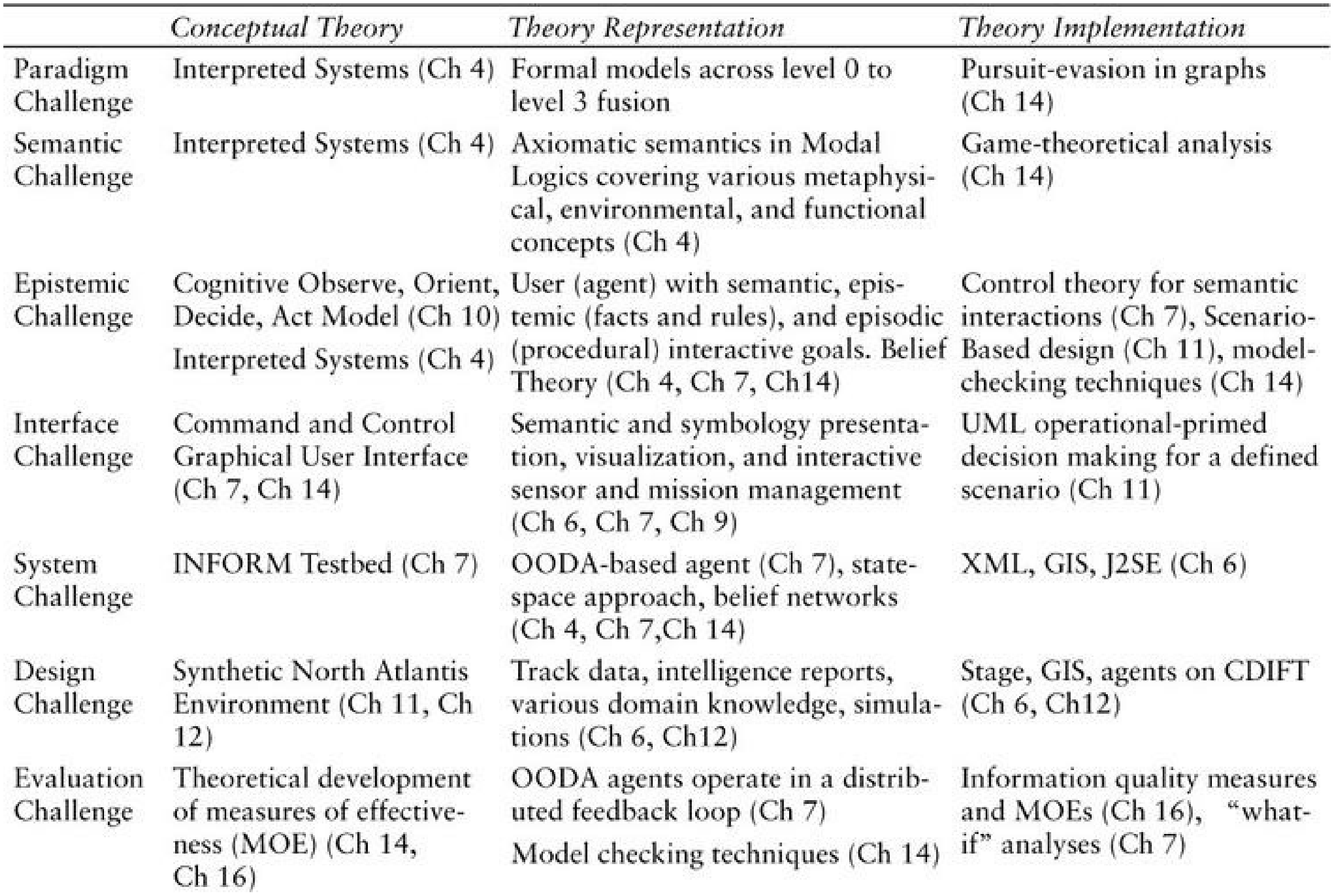


	Conceptual Theory	*Theory Representation*	*Theory Implementation*
Paradigm Challenge	State transition data fusion (STDF) model (Ch 3)	Unifying human and machine functional models across level 0 to level 3 situation awareness and fusion	Signal/text/image processing with a distributed multi-agent architecture
Semantic Challenge	Mephisto Semantic Framework [11]	Axiomatic semantics in first order logics (FOLs) and description logics (DLs) covering various metaphysical, environmental, functional, cognitive and social concepts	Prolog, Racer, FOL Meta-Interpreter, FOL Definitions Interpreter
Epistemic Challenge	Attitude [4] and Attitude Too Cognitive Models	Cognitive agents with semantic, epistemic (declarative facts and rules) and episodic (procedural cognitive routines) long-term memories	Prolog, Racer, FOL Meta-Interpreter, FOL Definitions Interpreter
Interface Challenge	Higher Common Operating Pictures (HiCOP) [4, 12, 13]	Interactive virtual news engaging virtual advisers, virtual battlespace, virtual interactive planning rooms, virtual video, virtual newspapers (web pages), Lexpresso controlled natural language	Commercial and indigenous natural language processing, text to speech, speech to text, various indigenous animation developments
System Challenge	Legal Agreement Protocol (LAP) (Ch 8) on the Coalition Distributed Information Fusion Testbed (CDIFT) (Ch 6)	Legal agreements between combinations of CDIFT connected human and machine cognitive agents based on formal semantic theories	LAP through agent cognitive routines on CDIFT using HLA, JBI, CoABS grid, Elvin, XMPP, XACML
Design Challenge	Synthetic North Atlantis Environment (Ch 12)	Use of synthetic development environments containing track data, intelligence reports, and various domain knowledge	Stage, domain knowledge, track data, GIS, Lexpresso reports, and agents on CDIFT
Evaluation Challenge	Evaluation of situation assessment [14]	Probabilistic propositional set disparity measures based on random inference networks	Mephisto, Prolog

1.2.2 Perspectives from Canadian Contributions

A Canadian approach to the HLIF themes of the book is articulated through chapters in each part of the book. Part 1 on information fusion concepts and representations contains formalization of situation analysis through interpreted systems semantics (Chapter 4). Interpreted systems semantics formalizes state transition systems as temporal trajectories potentially formed from multi-agent action. The Canadian-led chapter in Part 2 “Distributed Information Fusion and Management” is the TTCP C3I T P 4 Chapter 7. It discusses the Information Fusion and Resource Management Laboratory (INFORM Lab) testbed for experimenting with higher-level distributed information fusion and dynamic resource and configuration management, given multiple constraints on resource and communication networks. Chapter 9 presents the TTCP C3I TP2 contribution to Part 3 “Human-Systems Interaction,” by articulating the need for, and nature of User Defined Common Operating Pictures (UDOPs) for interfacing higher-level fusion systems to users. Scenario-based design for situation analysis serves as Chapter 11 in Part 4 “Scenario-Based Design”. Chapter 11 presents the use of scenarios as a basis for designing HLIF systems, with particular reference to the North Atlantis problem scenario. Finally, Part 5 “Measures of Effectiveness” introduces a toolbox for the evaluation of surveillance strategies based on interpreted systems in Chapter 14, which outlines the situation analysis toolbox as an implementation of the interpreted systems approach to building situations and analysing them. Part 5 also includes Chapter 16 with an application to maritime domain awareness. Table 1.2 summarizes the Canadian contributions to the HLIF challenges.

Table 1.2 Canadian Contributions to HLIF Challenges



	Conceptual Theory	*Theory Representation*	*Theory Implementation*
Paradigm Challenge	Interpreted Systems (Ch 4)	Formal models across level 0 to level 3 fusion	Pursuit-evasion in graphs (Ch 14)
Semantic Challenge	Interpreted Systems (Ch 4)	Axiomatic semantics in Modal Logics covering various metaphysical, environmental, and functional concepts (Ch 4)	Game-theoretical analysis (Ch 14)
Epistemic Challenge	Cognitive Observe, Orient, Decide, Act Model (Ch 10) Interpreted Systems (Ch 4)	User (agent) with semantic, epistemic (facts and rules), and episodic (procedural) interactive goals. Belief Theory (Ch 4, Ch 7, Ch14)	Control theory for semantic interactions (Ch 7), Scenario-Based design (Ch 11), model-checking techniques (Ch 14)
Interface Challenge	Command and Control Graphical User Interface (Ch 7, Ch 14)	Semantic and symbology presentation, visualization, and interactive sensor and mission management (Ch 6, Ch 7, Ch 9)	UML operational-primed decision making for a defined scenario (Ch 11)
System Challenge	INFORM Testbed (Ch 7)	OODA-based agent (Ch 7), state-space approach, belief networks (Ch 4, Ch 7,Ch 14)	XML, GIS, J2SE (Ch 6)
Design Challenge	Synthetic North Atlantis Environment (Ch 11, Ch 12)	Track data, intelligence reports, various domain knowledge, simulations (Ch 6, Ch12)	Stage, GIS, agents on CDIFT (Ch 6, Ch12)
Evaluation Challenge	Theoretical development of measures of effectiveness (MOE) (Ch 14, Ch 16)	OODA agents operate in a distributed feedback loop (Ch 7) Model checking techniques (Ch 14)	Information quality measures and MOEs (Ch 16), “what-if” analyses (Ch 7)

1.2.3 Perspectives from United States Contributions

A United States perspective is also outlined in each part of the book. In Part 1, Chapter 2 situation awareness and situation assessment canvass a number of issues relating to a variety of models for HLIF. TTCP C3I TP3 developed Part 2, Chapter 5, presents a layered information management model relating actors, producers, consumers, managers, and federates to deliver trusted, accurate, understandable, timely, and properly characterized information. Chapter 10 addresses Part 3 (Human-System Interaction) by proffering a Cognitive Observe-Orient-Decide-Act (C-OODA) control theoretic model for information fusion decision making. Part 4 "Scenario-Based Design" includes Chapter 13, which considers real-world operating condition modeling for probabilistic automatic target classification, based on designed experiments using scenarios of collected information to enable elements of SA. Chapter 15 rounds off the United States contribution in Part 5 "Measures of Effectiveness." Chapter 15 highlights the Information Fusion Situation Assessment model that brings together the Data Fusion Information Group and Situation Awareness Reference process models to identify activities of interest, and then considers various measures of performance of a situation assessment system, relative to an activities of interest worthiness assessment. Table 1.3 summarizes the United States contributions to the HLIF challenges.

Table 1.3 United States Contributions to HLIF Challenges

	Conceptual Theory	*Theory Representation*	*Theory Implementation*
Paradigm Challenge	Information Fusion Situation Assessment (IFSA) Model (Ch 2)	Operational process models across level 0 to level 6 fusion	Signal/Text/Image processing with a SA/SAW architecture (Ch 15)
Semantic Challenge	Development of IFSA taxonomy (Ch 2)	Operational semantics of computational models to infer meaning over environmental, functional, cognitive and social concepts (Ch 2, Ch13, Ch 15)	Numeric and Language fusion integration in an image (Ch 10) a cyber system (Ch 15)
Epistemic Challenge	Information Management Model (Ch 5)	Agents for workflow and service-based semantic, epistemic (facts and rules) and episodic (procedures) information processing (Ch 5)	Agent routines in CDIFT using HLA, JBI, CoABS grid, XML, XACML (Ch 6)
Interface Challenge	User Defined Operating Pictures (UDOP) (Ch 9) with operational conditional assessment (Ch 13)	Visualizations for a Common Operational Picture (COP) with symbologies, information management, and collaboration tools (Ch 9). User refinement support to fusion methods with cognitive theory (Ch 10)	Visualization tools to support SA for maritime surveillance (Ch 7), image analysis (Ch 10), target classification (Ch 13) and cyber threats (Ch 15)
System Challenge	Information Management Model (Ch 5) for the Coalition Distributed Information Fusion Testbed (Ch 6)	Use of ontologies and workflow/service/human agents for the CDIFT. Coordination of user/machine fusion methods based on information needs and tools (Ch 10, Ch 13, Ch 16)	Agent routines in CDIFT (**Ch 6**) using HLA, JBI, CoABS grid, XML, XACML and user refinement (Ch 10, Ch 13, Ch 15, Ch 16)
Design Challenge	Synthetic North Atlantis Environment (Ch 12)	Track data, intelligence reports, various domain knowledge (Ch 6, Ch 12)	Stage, GIS, agents on CDIFT (Ch 6, Ch 12)
Evaluation Challenge	Development of theoretical measures of effectiveness (MOE) (Ch 16)	Bayes networks to measure probabilistic variations from Operational Conditions (Ch 13) and derivation of MOEs from performance measures (Ch 10)	Development of MOEs for cyber analysis, (Ch 15) and coastal surveillance (Ch 16)

A Science of High-Level Information Fusion

The previous sections deliver a catalogue of approaches to the grand challenges of HLIF, which have primarily been formulated independently by representatives of different nations. Although the independence has engendered some obvious differences in these approaches, the striking observation is that, for each grand challenge, there is also a coalescence of ideas across the nations. This is suggestive of the birth of a new science of HLIF.

- The *Paradigm Challenge* juxtaposes the *independently* formulated United States IFSA framework (Chapter 2); the Australian STDF framework (Chapter 3); and the Canadian IS framework (Chapter 4). All three promote situations as a fundamental construct of the world, and all three utilize the machine interpretation of situations and the machine prediction of situations in the world. All three represent situations in machines through states and time stepped transitions between states. However, there are differences in how these manifest. Situations are represented very formally under the IS and STDF frameworks and less formally under the IFSA framework. The machine processing of situations is characterized by formal logics under the IS framework and by functional architecture process models under the STDF and IFSA frameworks.
- The *Semantic Challenge* considers how meaning is realized within machines. In all three cases, states are implemented as knowledge representations within the machine that can express sophisticated concepts well beyond sensed characteristics. In all three cases transitions between states are understood as graphs. But the independent origins also foster variations in the three approaches. The IS (Chapters 4 and 14) and IFSA (Chapter 2) approaches implement states through state vectors with operational semantics, while the STDF (Chapter 3) Mephisto [11] approach engages propositional formulae with axiomatic semantics. The IFSA and IS models both represent state transitions within the machine as directed graphs. Potential state transitions are represented in STDF by graphs, expressed as regular expression cognitive routines with procedural semantics (Chapter 12), but actual state transitions are simply expressed through a knowledge base content.
- The *Epistemic Challenge* features prominently in each framework. In all three cases, there was an acknowledgement of a significant shift in complexity from LLIF to HLIF. The HLIF world incorporating social relationships and political intentions [12] is far more complex than the LLIF world of signals and objects. All three methods foster a realization that the processing emphasis shifts from the

world to the machine as we move from LLIF fusion to HLIF. The LLIF processing emphasis is on the machine extracting content from information sensed in the world. The HLIF processing emphasis is on the machine imposing content on the sensed information. Thus in all three cases, HLIF machines are termed agents. For example, a HLIF agent can only infer that a sensed airborne object poses a threat if it imposes background knowledge about alliances, possible targets, etc. However, the independent origins again lead to variations in the details. For example, the ATTITUDE TOO Cognitive Model is the Australian agent construct whereas the Canadians employ the control theoretic C-OODA model (Chapter 10).

- The *Interface Challenge* is prominent under all three approaches because there is a strong emphasis on pairing machine fusion with human situation awareness. In all three cases, this pairing involves interfaces across the different levels of fusion and so interface technology moves beyond the traditional "dots on maps" and "lines on maps" technology of LLIF (UDOP in Chapter 9, command and control graphical user interface in Chapter 7 and Hi-COP in [4, 12, 13]). The independent origins again promote diversity. The United States framework (Chapter 2) introduces additional fusion levels to accommodate different aspects, with level 5 for human-machine interaction while the Australian framework (Chapter 3) applies the same four level fusion framework to both people and machines, and the Canadian framework (Chapter 4) applies the same modal logic framework to both people and machines. The emphasis is also slightly different in each case. The United States framework has a focus on obtaining and utilizing human situation awareness; Canadian attention is directed toward decision support; and the Australian framework remains agnostic toward what is performed by humans and what is performed by machines.
- The *System Challenge* is to not only exploit this agreed pairing of human situation awareness and machine information fusion, but to also do so in the presence of distributed collections of people and distributed clusters of machines. In all three nations, information management is deemed fundamental (Chapter 5, TTCP C3I TP3), where a distributed infrastructure is used to facilitate interaction between clusters of fusion machines (CDIFT in Chapter 6 and INFORM in Chapter 7). Recognition of the need for a distributed infrastructure led TTCP C3I TP1 to establish CDIFT as a common HLIF testbed. It combines American, Australian, and Canadian infrastructure components to support interoperable fusion products. In all three nations, a coordination strategy is promoted to manage multi-agent engagements across the infrastructure, but their nature differs. The Canadian IS framework (Chapter 14) uses a game theoretic model for agent interaction; the Australian LAP framework (Chapter 8) employs an agreement protocol for agent

interaction; and the United States control of agent-based systems (CoABS) framework (Ch 6) employs the knowledge acquisition in automated specification (KAoS) system to resource constrain distributed agents.

- The *Design Challenge* is another consensus issue for the members of TTCP C3I TP1. In all three cases, the HLIF processing emphasis on the agent imposing content on the sensed information led to the promotion of a scenario-based approach to the development of HLIF systems. The design of a HLIF system cannot occur without a rich context of the world in mind, as noted above in relation to a sensed airborne object under the Epistemic Challenge. This leads to a scenario-based approach to design. Part 4 includes the North Atlantis scenario-based design developed by TP1 for multi-national collaboration to support efforts of information management and information fusion systems design.
- The *Evaluation Challenge* is currently the least progressed aspect of HLIF, but a coalesce of ideas is nonetheless evident across the three nations. In each case, measures of content similarity or disparity have been proposed to evaluate assessments. Independence again engenders different approaches. The Canadian approach highlights evidential reasoning to measure probabilistic relations (Chapter 7) and game theory in interpreted systems to measure action trade-offs (Chapter 14). The United States contribution includes a number of situation measures of performance based on activities (Chapter 15) and information theory for situation measures of effectiveness (Chapter 16). The Australian offering [14] promotes probabilistic measures of the disparity between sets of propositions.

Through use of the grand challenges, this book distils a number of important commonalities that define HLIF from a collection of independent multinational implementations of HLIF systems. The collective analysis is suggestive of a science of HLIF including theories and representations, information management, and systems design methods of user interactions, sceneario assessment, and evaluation.

rences

[1] Bossé, É., J. Roy, and S. Wark, *Concepts, Models, and Tools for Information Fusion*, Artech House, Inc., Norwood, MA, 2007.

[2] Lambert, D. A., "Grand Challenges of Information Fusion," *International Conference on Information Fusion*, 2003.

[3] Blasch, E. P., J. Llinas, D. A. Lambert, P. Valin, S. Das, C.-Y. Chong, M. M. Kokar, and E. Shahbazian, "High Level Information Fusion Developments, Issues, and Grand Challenges – Fusion10 Panel Discussion," *International Conference on Information Fusion*, 2010.

[4] Lambert, D. A., "A Blueprint for Higher-Level Fusion Systems," *Journal of Information Fusion*, Vol. 10, No. 1, pp. 6 – 24, 2009.

[5] Waltz, E., and J. Llinas, *Multisensor and Data Fusion*, Artech House, Norwood, MA, 1990.

[6] Blackman, S., R. and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, MA, 1999.

[7] Das, S., *High-Level Data Fusion*, Artech House, Norwood, MA, 2008.

[8] Waltz, E., *Knowledge Management in the Intelligence Enterprise*, Artech House, Norwood, MA, 2003.

[9] Hall, D. L., and J. M. Jordan, *Human-Centered Information Fusion*, Artech House, Norwood, MA, 2010.

10] Lambert, D. A., "Unifcation of Sensor and Higher-Level Fusion," *International Conference on Information Fusion*, 2006.

11] Lambert, D. A., and C. Nowak, "The Mephisto Conceptual Framework," *DSTO Technical Report DSTO-TR-2162*, 2008.

12] Wark, S., D. A. Lambert, M. Nowina-Krowicki, A. Zschorn, and D. Pang, "Situational Awareness: Beyond Dots On Maps To Virtually Anywhere," *SimTecT*, Adelaide AUS, 2009.

13] Wark, S., and D. A. Lambert, "Presenting The Story Behind The Data: Enhancing Situational Awareness Using Multimedia Narrative," *IEEE MILCOM*, 2007.

14] Lingard, D., and D. A. Lambert, "Evaluation of the Effectiveness of Machine-based Situation Assessment," *Fourth Australian Conference on Artifcial Life*, Melbourne Australia. 2009.

15] Waltz, E., *Human Social Cultural Behavior Modeling and Fusion* – Tutorial, Oct. 2011.

Part I
Information Fusion Concepts

CHAPTER 2

Situation Assessment and Situation Awareness

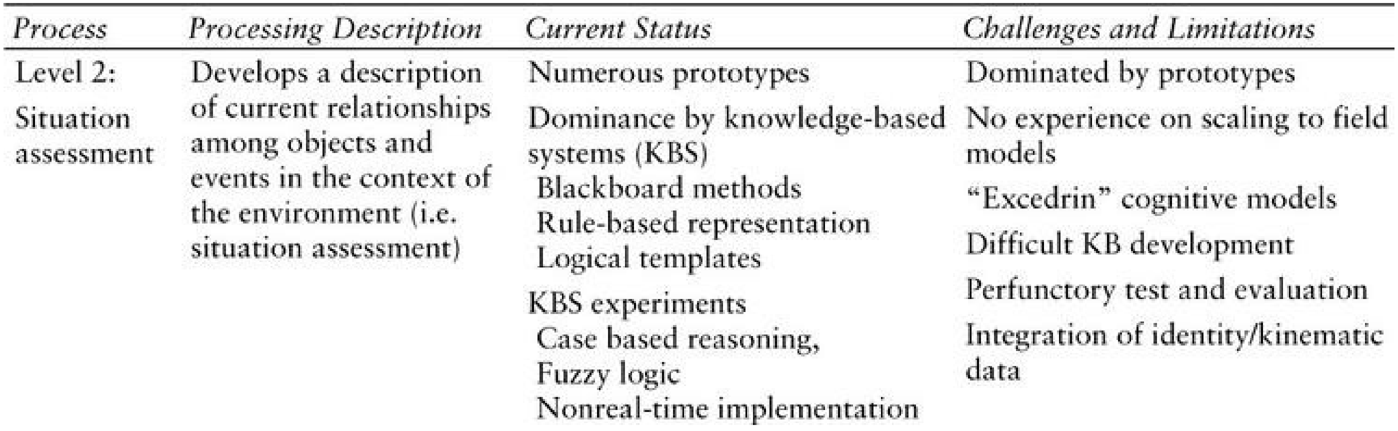
Erik P. Blasch (United States)

Situation assessment (SA) involves deriving relations among entities, such as the aggregation of object states (i.e., classification and location) and situational awareness (SAW) entails the mental state of a user. While SA/SAW has been recognized in the information fusion and human factors literature, there still exist open questions regarding situation and knowledge representation and theoretical reasoning methods to afford SA/SAW. For instance, while a great amount of data is collected over a region of interest, how does this information get aggregated and presented to an attention constrained user? Information overload can deteriorate cognitive reasoning, so a pragmatic solution to information representation and theoretical constructs are needed for effective and efficient situation understanding. In this chapter, we present issues associated with Level 2 (situation assessment) information fusion including (1) challenges, (2) overview of SAW/SA models for user perception, reasoning representation, and theoretical knowledge discovery process models, (3) syntactic and semantic representations, (4) user-fusion interaction through performance metrics, and (5) a combined SA/SAW information fusion reference model. While a definitive conclusion is not the aim of the chapter, many critical issues of models are proposed in order to characterize future successful strategies towards knowledge representation and theoretical reasoning strategies for situation assessment.

Introduction

Situation assessment (SA) is an important part of the information fusion (IF) process because it (1) is the purpose for the use of IF to synthesize the multitude of information, (2) provides an interface between the user and the automation, and (3) focuses data collection and management. Hall and Llinas have listed a variety of SA challenges for real systems implementation shown in Table 2.1 [1]. Since the late 1990s, there have been updates in the progress of SA methods, but there are still remaining issues that result from technology advances. Since low-level information fusion (LLIF) of target tracking and identification is well explored, there is an ongoing discussion about the needs for high-level information fusion (HLIF) [2–4] that includes situation assessment.

Table 2.1 SA Challenges and Limitations: Hall and Llinas



Process	*Processing Description*	*Current Status*	*Challenges and Limitations*
Level 2: Situation assessment	Develops a description of current relationships among objects and events in the context of the environment (i.e. situation assessment)	Numerous prototypes Dominance by knowledge-based systems (KBS) Blackboard methods Rule-based representation Logical templates KBS experiments Case based reasoning, Fuzzy logic Nonreal-time implementation	Dominated by prototypes No experience on scaling to field models "Excedrin" cognitive models Difficult KB development Perfunctory test and evaluation Integration of identity/kinematic data

From: [1].

As the research has progressed on SA and situational awareness (SAW), new developments press the need for understanding the components of the SA/SAW research. A summary of contemporary needs includes [2]:

Common Issues

- User: SA processes include perceptual, interactive, and human control;
- Process models: updating behavioral models (e.g., Bayes Nets, procedural/logical, perceptual, learning);
- Context: operational situation (i.e., dependent on the current state of the environment);
- Meaning: semantics and syntax issues (e.g., formal theory, methods, ontologies);
- Metrics: develop a standard set of measures (e.g., trust, bounds, uncertainty).

Common Challenges

- Explanation of process for evidence accumulation and contradiction detection in reasoning;
- Visualization displays to facilitate inferential chains, collaborative interaction, and knowledge representation;
- Interactive control for corrections and utility assessment for knowledge management.

Throughout this book, various representations and theories are presented to assess the needs of HLIF. In this chapter we explore the developments of situation assessment as providing relations among entities to provide awareness to both the analyst and the information fusion system. Section 2.2 discusses the basis of SA/SAW from the needs of a system and the user. Section 2.3 describes various SAW models advances from the previous text [5] . Section 2.4 introduces SA models and Section 2.5 combines the current SA/SAW models. Section 2.6 provides a discussion on metrics and current challenges, and Section 2.7 provides a summary.

Situation Awareness and Situation Assessment Defined

There are two main groups addressing situational information: the engineering information fusion (IF) community (i.e., SA) and the human factors community (i.e., SAW). SAW is a mental state while SA supports (e.g., fusion products) that state. For the IF community, there are many leading authors proposing different aspects of SAW research. Research categorizes developments, but another way is by applications. There are many application communities looking at SAW research, including military, medical, aviation, security, and environmental. Each might have differences, but the commonality rests in the fact that a multitude of data needs to be efficiently synthesized into a single operating picture (discussed in Chapter 9) [6] to assist a user in completing their mission tasks.

Level 2 information fusion, SA is the estimation and prediction of relations among entities, to include force structure and force relations, communications, etc. which require adequate user inputs to define entities. Other definitions of SA coordinate with threat and impact assessment.

> *Situation assessement* (SA) as “a quantitative evaluation of the situation that has to do with the notions of judgment, appraisal, and relevance.” [5] (defined as Level 2 information fusion).
> *Threat assessment* (TA) as “an expression of intention to inflict evil, injury, or damage. The focus of threat analysis is to assess the likelihood of truly hostile actions and, if they were to occur, projected possible outcomes…” [5] (defined as Level 3 information fusion).

From these definitions, we see that the role of SA is to provide an evaluation of the environment and provide information about entities, relations, and objects (e.g., people, assets, or networks). Given the presented situational picture, there is a need to characterize and display the information to provide awareness. SAW is a mental state about the situation that utilizes the displayed information and affords the user/groups the ability to process, reason, and act on the information.

Situation awareness is an important concept of how people become aware of things happening in their environment. SAW, from a human factor’s point of view determines what effects contribute to a mental picture of action. The *HQ USAF AFISC/SE Safety Investigation Workbook* [7] defines SAW as:

> keeping track of prioritized significant events and the condition’s in one’s environment.

SAW was further refined [8] to incorporate the meaning and forecasting of events as:

> SAW is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future.

One of the premier models for SAW and user decision-making is the extended Observe-Orient-Decide-Act (OODA) model, which is also termed as Boyd's control loop [9], as shown in Figure 2.1. In the "orient" analysis, there is additional information that a user engages in analyzing and synthesizing a situation (e.g., cultural traditions, heritage, and previous experience). Key to the "Orient" phase is the incorporation of new information. New information is combined with previous information to update the situation as well as provide answers to questions posed by the user. The OODA loop model has been widely used to represent decision-making in military environments. The OODA applications for user modeling include information fusion [10], military systems [11], target recognition [12], cultural modeling [13], and recently, semiautomated decision making. [14]

However, the classical or extended versions of the OODA loop suffered from a lack of details to sufficiently support the design of systems. For instance, the model depicted in Figure 2.1 proposes a more detailed version of the orient process to the detriment of the others. Key to the developments and instantiations of the OODA models include: application relevant decision-making based on context, time of analysis, and uncertainty representation. Several developments include the modular M-OODA [15], the cognitive C-OODA [16], and the technology, emotion, culture, and knowledge (TECK-OODA) [13]. Further developments of the OODA extensions are provided in Chapter 10. The development of the OODA analysis is popularized in many domains and the elements of the subsequent SA/SAW models are natural extensions to the basic tenets, but reflecting the change in technology (e.g., collection and processing observations), the developments of information fusion (e.g., synthesis of observations), and applications to different domains other than military targeting. We next explore the SAW models as development of the needs for SA.

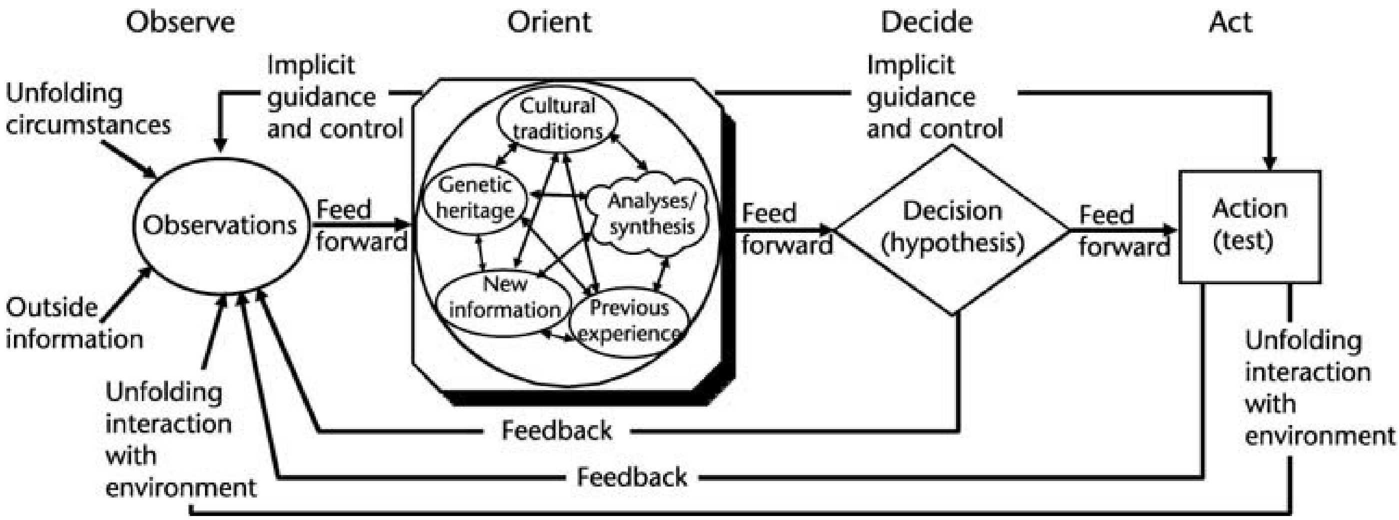


Figure 2.1 The extended OODA loop. [9].

Situation Awareness (SAW) Models

2.3.1 Endsley's SAW Model

The human in the loop (HIL) of a semiautomated system must be given adequate SAW. According to the Endsley model [8] shown in Figure 2.2, SAW translates into three levels:

- Level 1 SAW—Perception of elements in the environment;
- Level 2 SAW—Comprehension of the current situation;
- Level 3 SAW—Projection of future states.

Operators of dynamic systems use their SAW in determining their actions. The three levels correspond to the OODA Loop. To optimize decision making, SA should be as precise as possible about the observed objects in the environment (Level 1 SAW). An SA approach should present a fused representation of the data (Level 2 SAW) to orient a user and provide support for the operator's projection needs (Level 3 SAW) in order to facilitate operator's decision making and action. Boyd [9] emphasized the need reduce the OODA time. From the SAW model presented in Figure 2.2, workload is a key component of the model that reflects the user decision and reaction time. To instantiate the SAW model, we note that the user must be primed for situations (the OODA Loop "orient" function of experience) to be able to operate faster and more effectively.

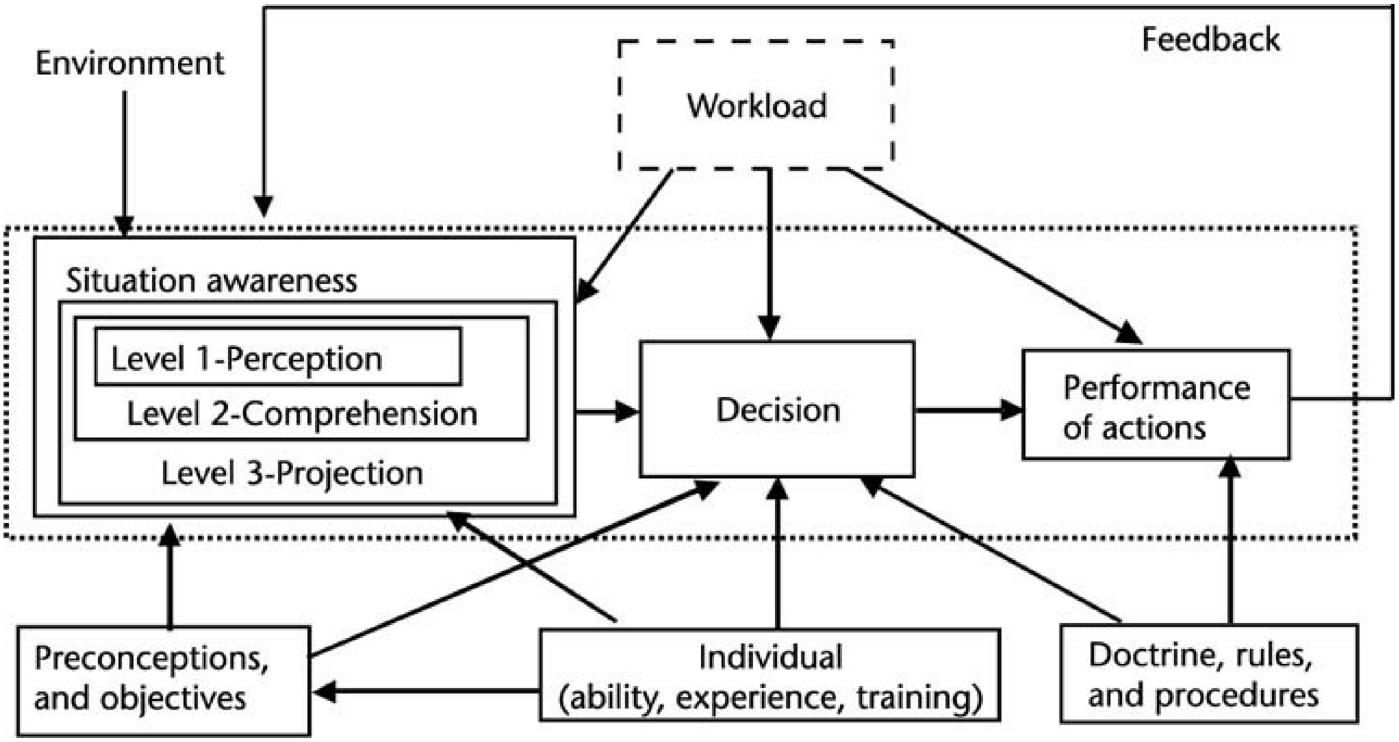


Figure 2.2 Endsley's SAW model.

2.3.2 Recognition Primed Decision (RPD) Making Model

To understand how the human uses the situation perspective to refine SAW, the RPD model illustrates contextual reasoning. The RPD model [17] shown in Figure 2.3 develops the user decision making capability based on the current situation and past experiences and highlights user goals, requests, and salient cues. The RPD model allows a system designer to capture the reduction in reaction time and increase in accuracy for the cases in which the user cues an information fusion system (IFS) and when the IFS cues the user. According to Breton [16], SAW and RPD processes are parallel and are further developed in the Chapter 10.

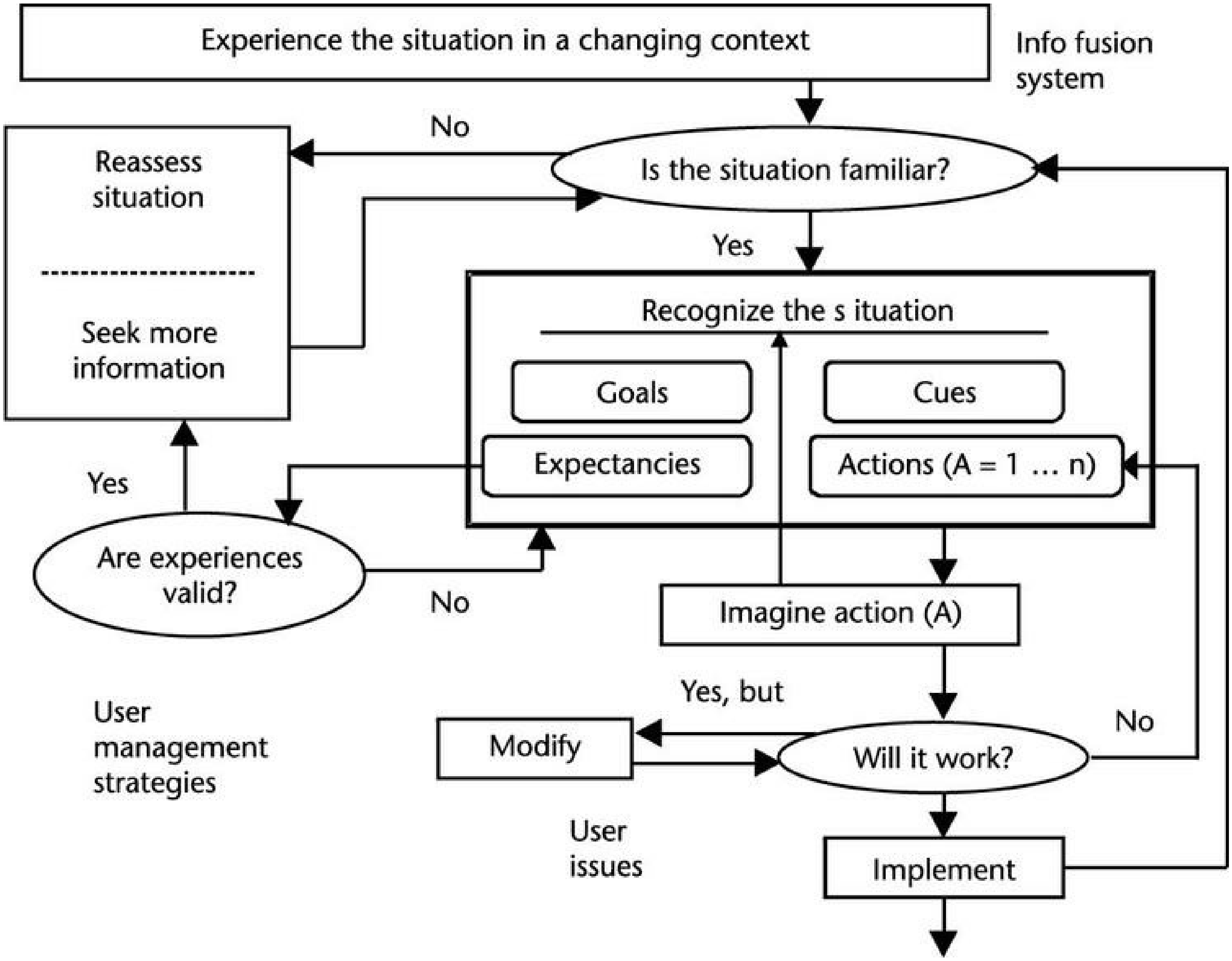


Figure 2.3 Recognition-primed decision (RPD) making model for situation awareness.

The key to SAW models is that an information fusion system must adequately represent the situation, compose a situation assessment theory, and prioritize information needs. Information priority is related to the information desired to orient and make decisions related to user tasks. The user must have the ability to choose or select the objects of interest and the processes from which raw data is converted to the fused information. Finally, situation understanding includes knowledge representation of data origins, sensor uncertainties, and information pedigrees.

To utilize the OODA construct, SAW model extensions, and the RPD framework, we next explore the models that focus on situation assessment.

Situation Assessment Models

2.4.1 Data Fusion Information Group Model

The Data Fusion Information Group (DFIG) model [2, 18] was proposed a an update to the Joint Director Laboratories (JDL) model [19]. In terms of the DFIG model, Observe is Level 1 fusion of object assessment, and Orient is Level 2 fusion of situation assessment. Both Observe and Orient have also been referred to as situation awareness. Decide is Level 5 fusion of user refinement and Act is Level 4 fusion of process refinement. Act and Decide are considered decision-making functions.

The DFIG model, shown in Figure 2.4, separates the data fusion and management functions. Management functions are divided into sensor control, platform placement, and user selection to meet mission objectives. Level 2 and 3 fusion includes tacit functions that are inferred from Level 1 explicit fusion representations of object assessment. Since the unobserved aspects of the SA problem cannot be processed by a computer, user knowledge and reasoning is necessary. The current definitions, based on the revised JDL fusion model [19], include:

Level 0—Data assessment: estimation and prediction of signal/object observable states on the basis of pixel/signal level data association (e.g., information systems collections);

Level 1—Object assessment: estimation and prediction of entity states on the basis of data association, continuous state estimation and discrete state estimation (e.g., data processing);

Level 2—Situation assessment: estimation and prediction of relations among entities, to include force structure and force relations, communications, and so forth (e.g., information processing);

Level 3—Threat/impact assessment: estimation and prediction of effects on situations of planned or estimated actions by the participants; to include interactions between action plans of multiple players (e.g., assessing threat actions to planned actions and mission requirements, performance evaluation);

Level 4—Process refinement (an element of resource management): adaptive data acquisition and processing to support sensing objectives (e.g., sensor management and information systems dissemination, command/control);

Level 5—User refinement (an element of knowledge management): adaptive determination of who queries information and who has access to information (e.g., information operations) and adaptive data retrieved and displayed to support cognitive decision making and actions (e.g., human computer interface);

Level 6—Mission management (an element of platform management): adaptive determination of spatial-temporal control of assets (e.g., airspace operations) and route planning and goal determination to support team decision making and actions (e.g., theater operations) over social, economic, and political constraints.

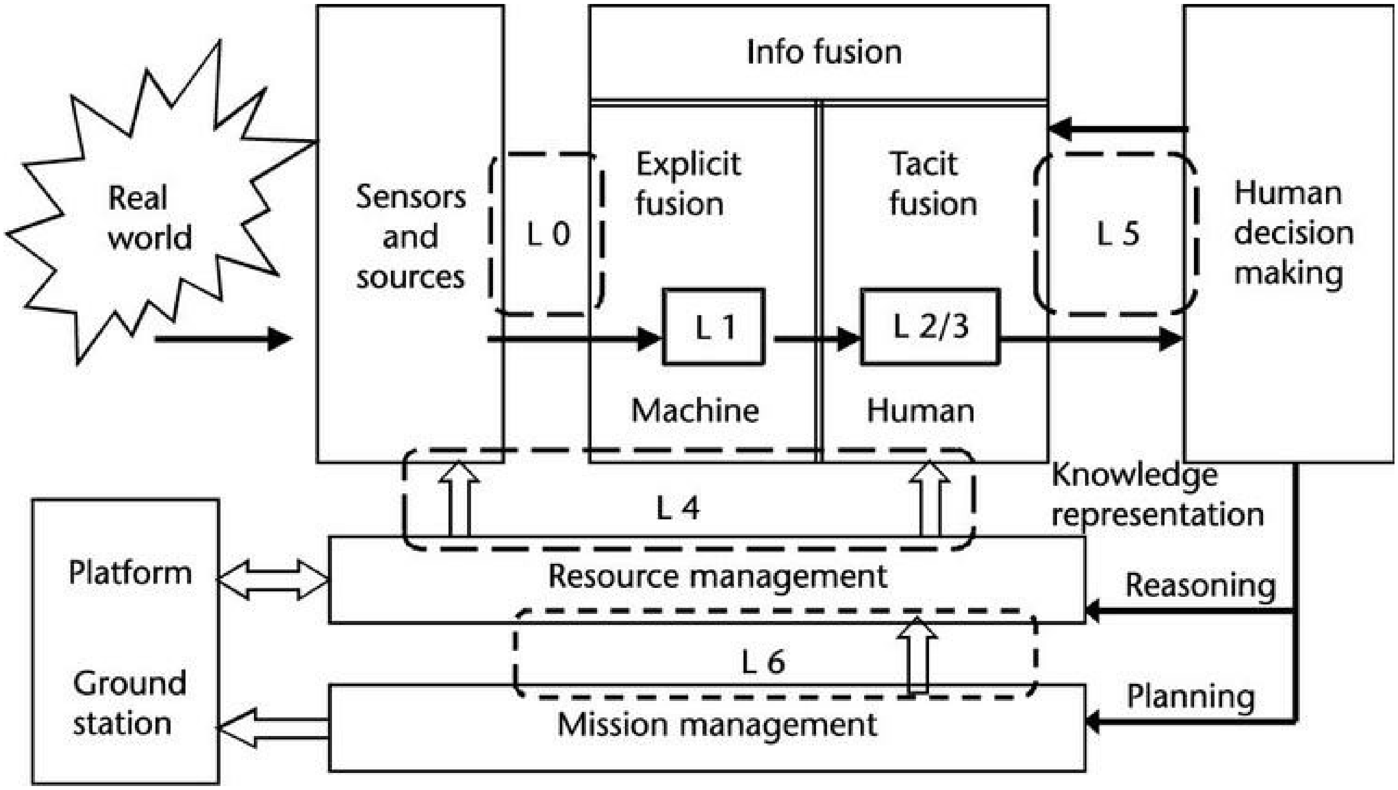


Figure 2.4 Data Fusion Information Group (DFIG) model.[2].

For SA, the user must (1) prioritize information needs to the fusion manager, (2) require reliable and validated information, and (3) seek patterns [2]. The information priority is based on the information desired. The user must have the ability to select the objects of interest and the processes from which the raw data is converted to the fused data. Users have individual differences for knowledge representation (KR) and thus, the coordination between the user and the machine needs to be flexible to satisfy SA information needs requirements. Information needs can be satisfied by capturing the attention, workload, and trust parameters with a set of standard metrics, such as timeliness, throughput, confidence, and accuracy [2]. Information needs are based on the theoretical KR methods, which need rigorous testing in experimental designs to define SA products. Additionally, dynamic updating of knowledge delivery for planning requires timely and reliable data for reasoning. It is important to note that reliability and validity are two different concepts. A piece of information can be 100% reliable and either totally diagnostic (100% validity) or undiagnostic (0% validity) in predicting information. However, the less reliable the information, the less valid it is because of the inherent uncertainty (i.e., error) in the information itself. Next we relate DFIG as a function of the theoretical metrics captured in information needs.

2.4.2 Situational Assessment Models for the User

SAW is based on the user's need to perceive and act on the environment as well as the interactions with the SA products. A fusion system must satisfy the user's functional needs and extend their sensory capabilities. A user fuses data and information over time and space and acts through their world mental model—whether it be in the head or with graphical displays, tools, and techniques. The user-fusion model [20, 21], shown in Figure 2.5, captures the coordination between the user and the DFIG levels as related to metrics (see Chapter 16).

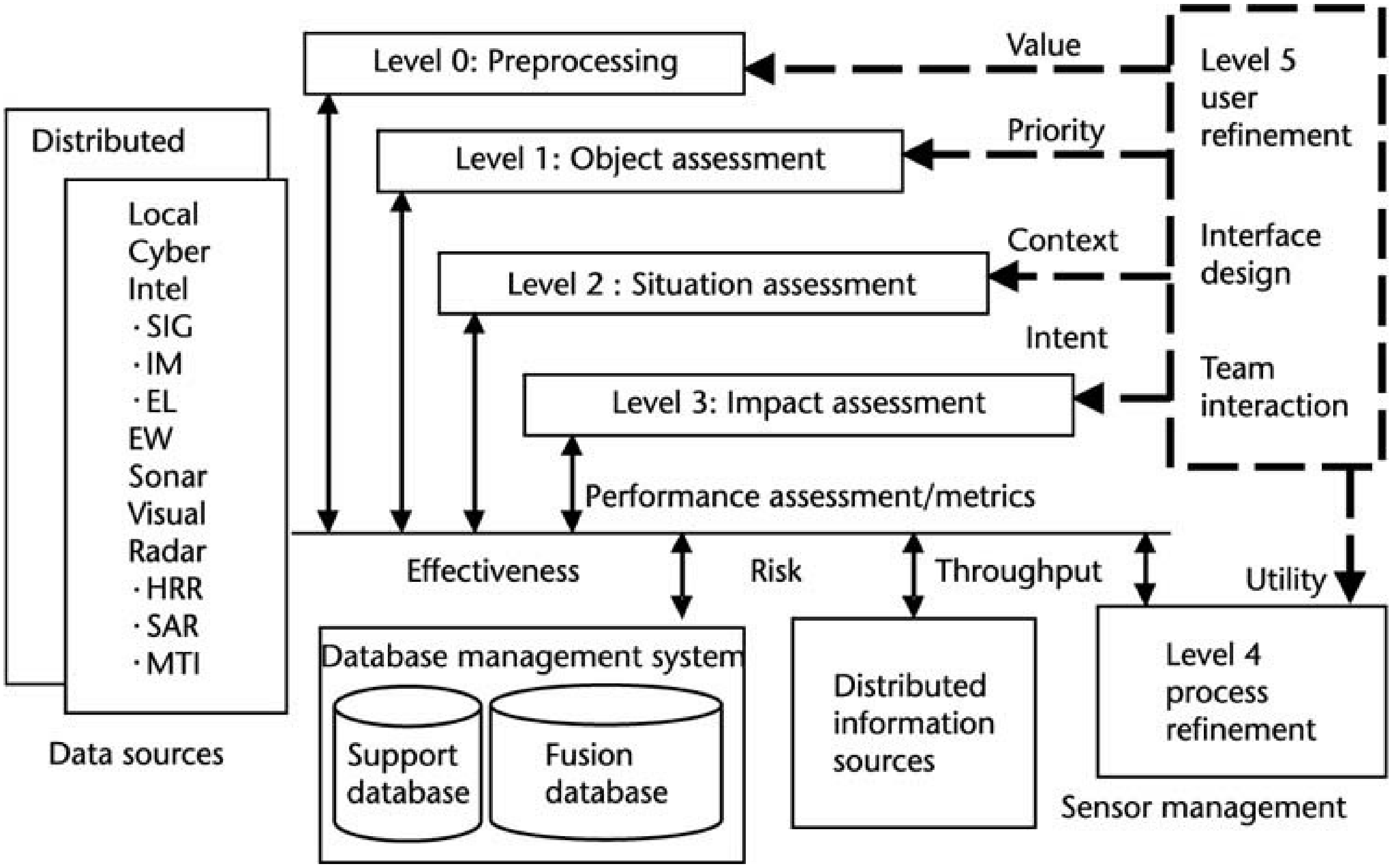


Figure 2.5 User fusion model. [22].

The key for SA is the user's mental model [22]. The mental model is the representation of the world as aggregated through the data gathering, The user's perception [23], IF design, and adaptations to contextual economic, social, and political situations vary based on the activities of interest (see Chapter15).

Situational Assessment Model Based on Activities of Interest

The SA fusion model components [24] by Roy, summarized in the previous text [5], show the various information needs to provide the user with an appropriate SA/SAW. SAW has received increased attention due to its diverse applications in a number of problem domains including: asymmetric threat, cyber, and homeland security. [25, 26] Salerno et al. [27], proposes an architecture that combines the Endsley and the information fusion models [5] (shown in Figure 2.6), and have applied their model to various strategic, operational, and tactical applications [28].

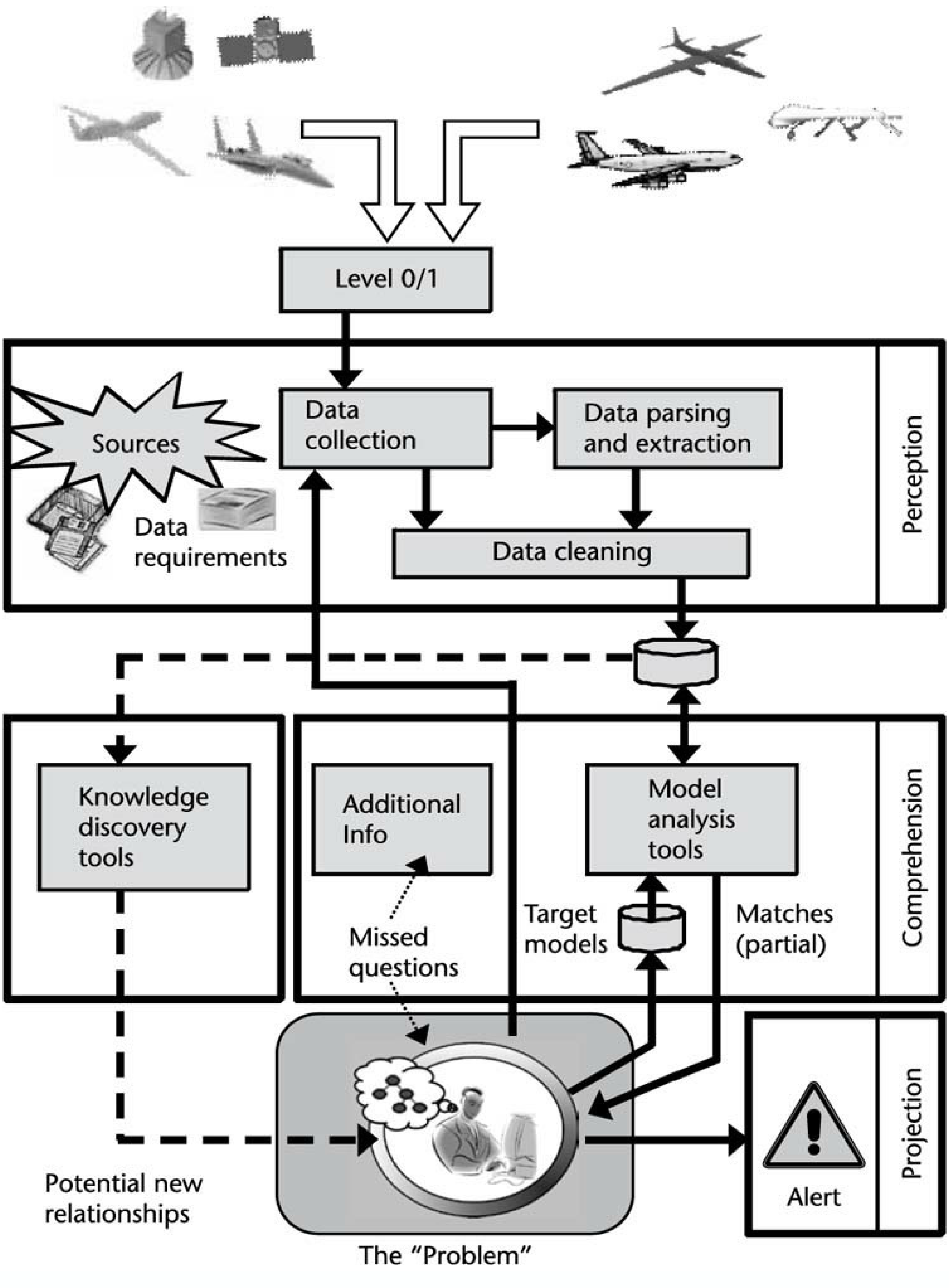

Level 0/1
Sources
Data collection
Data parsing and extraction
Data requirements
Data cleaning
Perception
Knowledge discovery tools
Additional Info
Model analysis tools
Target models
Missed questions
Matches (partial)
Comprehension
Potential new relationships
The "Problem"
Alert
Projection

Figure 2.6 SA framework [2].

Through a display, a user can (1) build a model by either editing an existing template/model or create a new one, (2) activate/deactivate existing models, or (3) view active models and any evidence that has been associated with the model over time. Different political, military, economic, social, infrastructure, and information models can be accessed and the result published (or subscribed) to aid SA. Key to the model display are semantic notions of a situation (see Chapters 3 and 4).

2.5.1 Syntactic Algorithms and Semantic Synonyms in Information Fusion Analysis

Issues encountered in SA tool development mainly pertaining to evidence access, storage, usage, and providing a priori knowledge. In order to resolve any semantic issues of context and value, we need to normalize the data before we can use it. Data normalization involves converting different formats of the same data into a common representation. Dealing with semantic inconsistencies is much more difficult. In these cases, we need to resolve synonyms both in what is represented and what the value itself represents. Two different labels can have the same meaning, or two aliases can represent the same entity. Finally, what level of a priori data is needed depends on the context of operation.

There continues to be a debate as to what LLIF and HLIF represent. One belief is that LLIF deals only with the tracking and identification of individual objects while HLIF is the aggregation of the objects into groups or units. For example, LLIF objects could be various equipments, such as cars, people, and assets. At HLIF, equipment along with personnel can be aggregated into a unit based on time and space. Current developments in the World Wide Web, news sources, and political concepts require new definitions for tracking non-physical objects and situation descriptions. Questions arise, such as how does the IF system acquire the necessary a priori knowledge (or relationships) to perform aggregation? What is the difference between formal models for identifying an object, a group, or an activity? To answer these questions, we first need to define a situation and then explore HLIF requirements for activity projection.

2.5.2 Definition of a Situation

Elements of a situation include entities, objects, groups, and events, that lead to elements of an activity [2]. In Chapter 15, we will explore the activity of interest (AOI) score. Here, we use the elements of a situation state that are similar to Level 1 tracking and identification with the addition of relations among entities and its correspondence to an activity. It is noted that the Dale Lambert [29] uses the state transition data fusion (STDF) model to capture objects, relations (for SA), and effects (for impact). Lambert generates possible event analysis through a probability measure, which we also use (see Chapter 3).

An *entity* [30] is defined as "something that has a distinct, separate existence, though it need not be a material existence. In particular, abstractions and legal fictions are usually regarded as entities." For example, an object [31] is a physical entity.

A *group* is "a number of things (entities, to include individuals) being in some relation to each other," while an *event* is "something that takes place; an occurrence at an arbitrary point in time; something that happens at a given place and time." Both entities and groups can be associated with a specific event or events. An *activity* [32] is "something done as an action or a movement." Activities are composed of entities/groups related by one or more events over time and/or space.

Thus by definition an event, group, and activity can be considered as a more complex entity (e.g., object) and can be tracked and identified. By using the definitions presented above, we argue that activities and the aggregation of these activities (which we refer to as "the situation") is both a part and a result of LLIF. Models or a priori knowledge are necessary for LLIF to be capable of identifying the object, group or activity. This a priori knowledge (i.e., the relationships or associations) can be learned through knowledge discovery and validated by an operator or provided directly. Here, we note that knowledge discovery techniques use statistically relevant occurrences for pattern recognition and data mining. As such, new or novel ideas cannot be learned and require knowledge elicitation.

2.5.3 The Situation Awareness Reference Model

Figure 2.7 combines the elements of the situation assessment (SA) and the user analysis (SAW) to provide forecasting of future events, situation refinement through information collection, and knowledge representation between players. Level 2 SA/SAW includes the interpretation or meaning of what is happening with respect to context and time while Level 3 (impact assessment) is the determination of whether there exists a threat —is there an entity, group, event or activity that we should care about? For the Level 3 IA, we need to define the multiple players as "us" and "them," and the knowledge representations that construct the player actions.

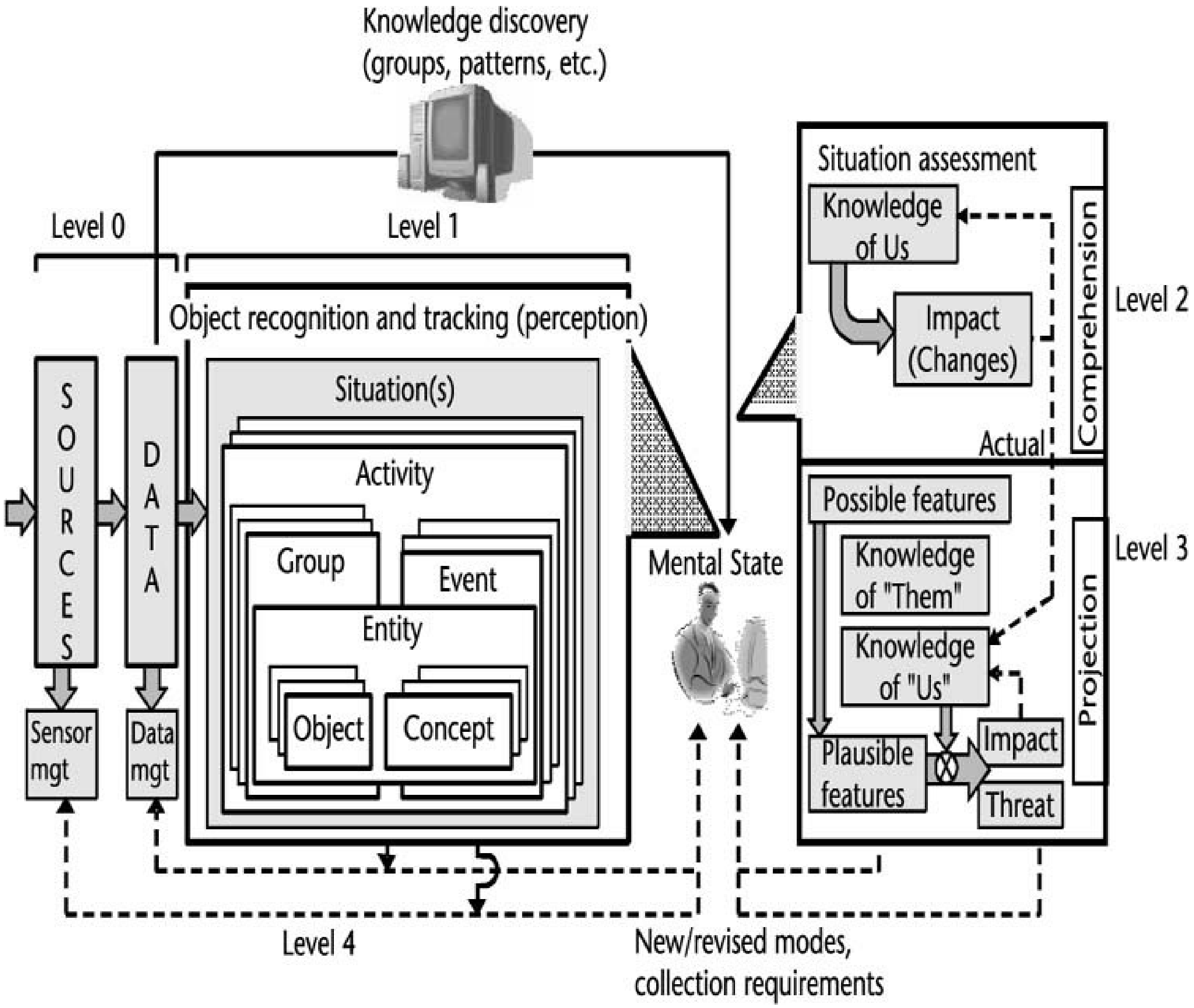


Figure 2.7 SA reference model [33].

Specifically, situation assessment is a quantitative evaluation of the environment including judgment, appraisal, and relevance. Roy [25] provides a description of a number of questions/products that are developed under what they call SA. Information aggregation from a knowledge representation of "us" is essentially situation comprehension. LLIF attempts to answer analysis questions of existence and size (How many?), identity (What/Who?), kinematics (Where?), and time (When?), while HLIF provides analysis of behavior (What is the object doing?), activity (Build up, draw down?), intent (Why?), salience (What makes it important?), and capability/capacity (What could they/it do?).

After an SA assessment, the next step would be to forecast the current situation and threat into the future. We call this function *projection*. It is true, however, that to perform projection one would rely on the current situation and forecast not only those salient activities into the future but also any future impacts or threats. Thus, to summarize, a situation is a snapshot of the aggregated activities at time t. Comprehensions are the elements of the situation. Projection takes the situation and projects or forecasts it to time, $t + n$, where n represents some number of time steps.

In this chapter, we have developed the various methods used to analyze the process of SAW, SA, and the user and developed extensions and clarifications based on previous work to relate to physical and non-physical entities. Next, we bring together our ideas in a common SA/SAW reference model, which we call the IF-SA reference model, shown in Figure 2.8. The model will be explored in Chapter 15 for evaluation. It is noted that the user role and their needs for SAW are captured in the combined model to afford both the machine and the user situation assessment. The user has qualitative analysis strengths of prior experiences, mission objectives, and contextual knowledge, whereas the machine has quantitative analysis strengths of large databases, rapid computations, and means of visual analytics.

Current Information Fusion Situation Assessment Reference Model for Information Fusion

Given the developments of SAW and SA, we combine the ideas into an integrated information fusion situation assessment (IFSA) model in which the role of SA stratifies the object/event analysis. The IFSA captures the elements of the SAW reference model with the proposed changes to the JDL model that includes the DFIG elements of a combined L2/L3 analysis as well as the coordination with the user in Level 5 information fusion. The IFSA model is presented in Figure 2.8.

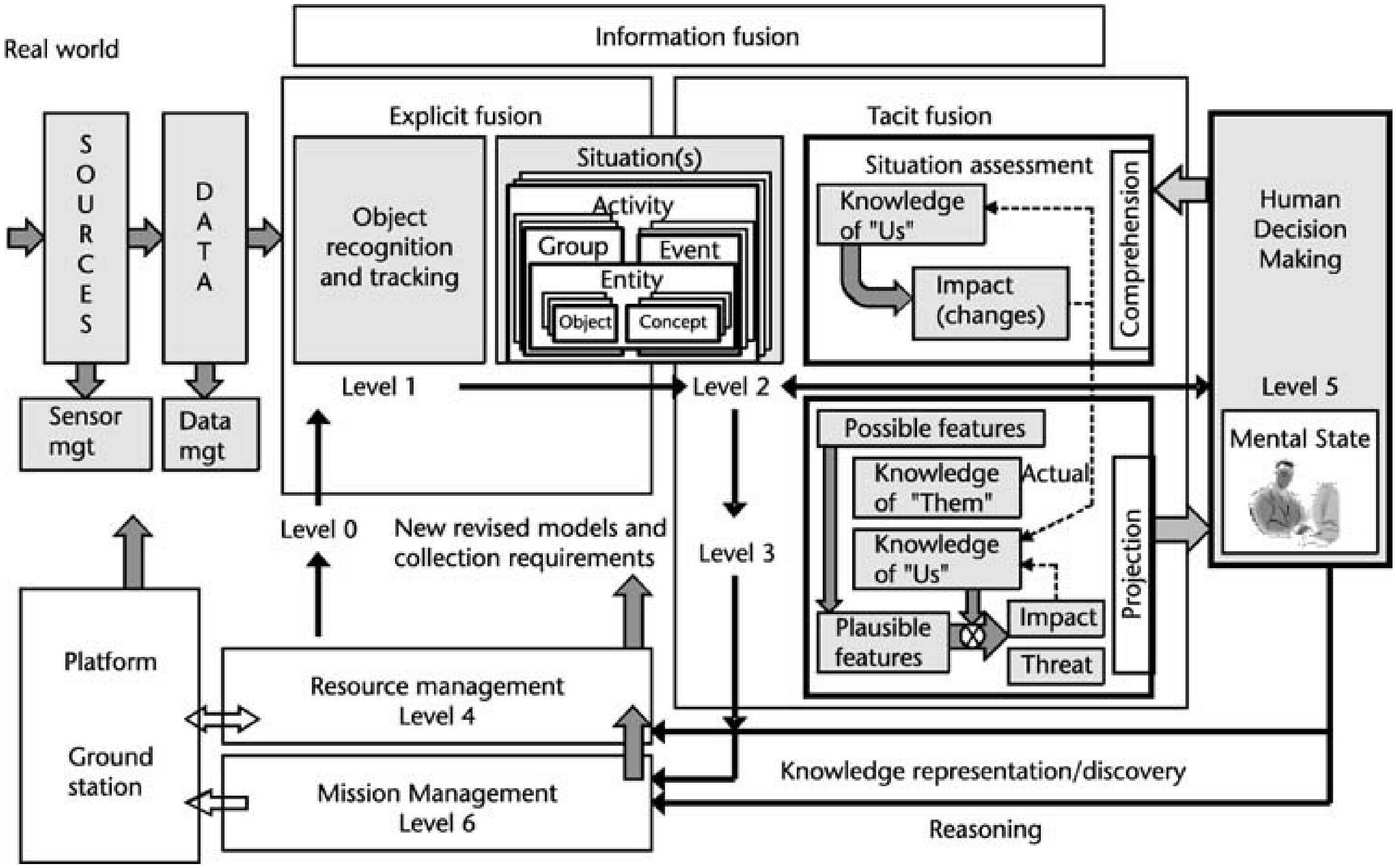


Figure 2.8 Information fusion stituation assessment (IFSA) model.

One question to ask is, why another model? First, the goal is to capture all the previous models into a comprehensive model that can be applied to any domain and set of users. As technology advances, there is a need to revisit the process models that characterize the elements of the process. We explain the model based on a top-down versus a bottom-up strategy. We choose to describe the model in a top-down fashion as the user has to attend to the data that is given before any meaningful analysis is conducted. Starting with Boyd's control loop, the right side of Figure 2.8 captures the needs of the user and their ability to observe and orient themselves to the information.

As the users request information for their SAW, they must regress over the data they have, what they need, and the control actions. The information fusion system provides the elements of the information from the left side of Figure 2.8, which provides alerts that call to the attention of situations of interest (denoted as explicit fusion). The user can coordinate with any of these levels to update the SAW and control the collections for SA through tacit fusion. Finally, we note that there are needs between resource management (e.g., airborne assets and web pages) versus that of mission management (e.g., goals, policies, and doctrines) as shown in the bottom of Figure 2.8.

While the IFSA model captures our current assessment of the domain of SA and SAW, other issues need to be considered such as metrics for evaluation operating conditions analysis (see Chapter 13) and the challenges of situation model refinement for practical use. Other models in the book also explore representations and theories to instantiate SA products that are useful to the community.

Discussion

There is a need for process models for SA representation, mathematical methods for SA theory, and metrics to standardize SA testing and validation to help to bring together the SA/SAW research for information fusion automation to aid users. Throughout this chapter, we highlighted process models to set the stage for information fusion SA/SAW representation. SAW supports the user's mental state and SA supports the state estimation, so we focus on the theory of SA and the methods used to process the information.

2.7.1 Situation Assessment Representations and Theory

Developments of SA representation and theory require contextual knowledge management [34]. Waltz and Llinas [6] detailed various issues in defining SA, but allude to limited SA representations except for military applications and lack of a formal SA theory. Techniques presented included expectation template-based techniques such as event-activity profiling, knowledge-based tools, and expert systems. Blackman and Popoli [35] suggest fuzzy methods for reasoning about SA and present examples in tracking and identifying targets with rule-based approaches using knowledge about target behaviors to infer and predict a situation.

A summary of various SA functions and methods for SA refinement is presented in Figure 2.9, which is a categorization of the information techniques from [34, 35] for SA. The figure is not a complete list, however, it details many of the methods explored in SA analysis that leads to a formal theory. We note that there is a need for knowledge representation, an inference engine for the theory, and elements of control to process the information.

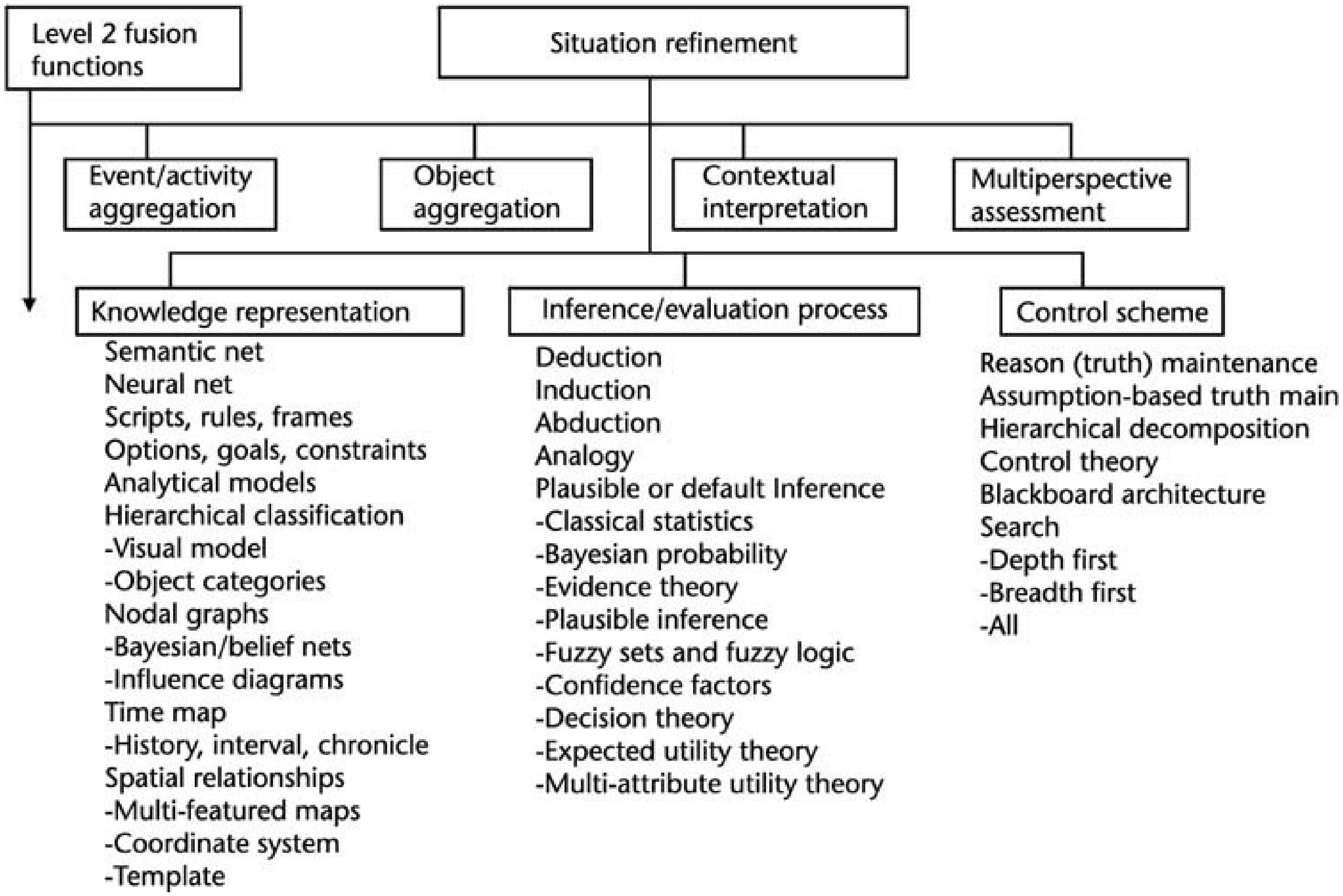


Figure 2.9 SA functions and methods to refne the SA estimate.

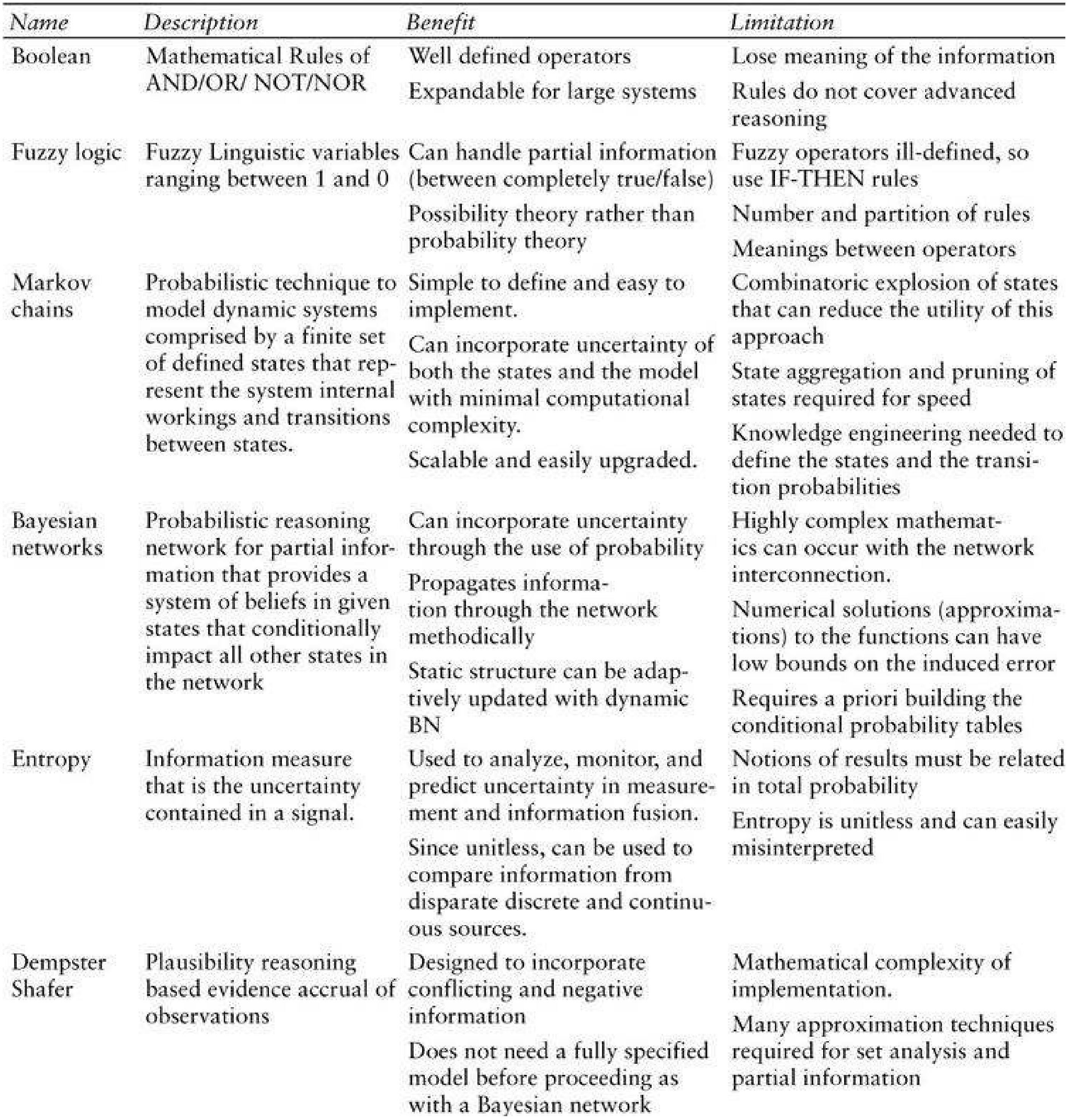
Since the 1990s, many of the techniques in templating, expert systems, and fuzzy logic were deemed difficult to implement for situational analysis in that it was hard to model the entire situation (templating), develop all the situational rules (expert systems), and determine the use of heuristics (fuzzy logic rules) for complex situations. Examples of behavioral tracking [36] and neuro-fuzzy [37] methods were applied in an attempt to model the human's ability to assess situ-ational intent. Likewise, methods for cyber situational awareness (as an emerging domain in SA) used game theory [38], graphical networks [39], Markov models [40], and Dempster-Shafer [41] methods. If we look at the publications in the *Journal of Advances in Information Fusion*, we see the combined use of situation and intent/threat assessment which was detailed in the process models of this chapter. Methods proposed are game theory [42], entropy [43], hypothesis scoring [44], and fuzzy logic [45]. For example, context-aiding tracking [46], which was similar to [37] with extensions, is a method to incorporate the behavioral tracking results as situation assessments of targets with intent.

In *High-Level Data Fusion*, Das [47] develops model-based situation assessment using Bayes nets (BNs), belief networks, and poly trees, as well as information-theoretic approaches, such as entropy, for an information fusion SA gain (see Chapter

16). Also, time elements in SA are processed with hidden Markov models and dynamic BNs with examples and mathematical processing.

Table 2.2 summarizes the techniques applied to SA that seek to reduce the uncertainty in the evaluation of a situation.

Table 2.2 Methods for SA Processing



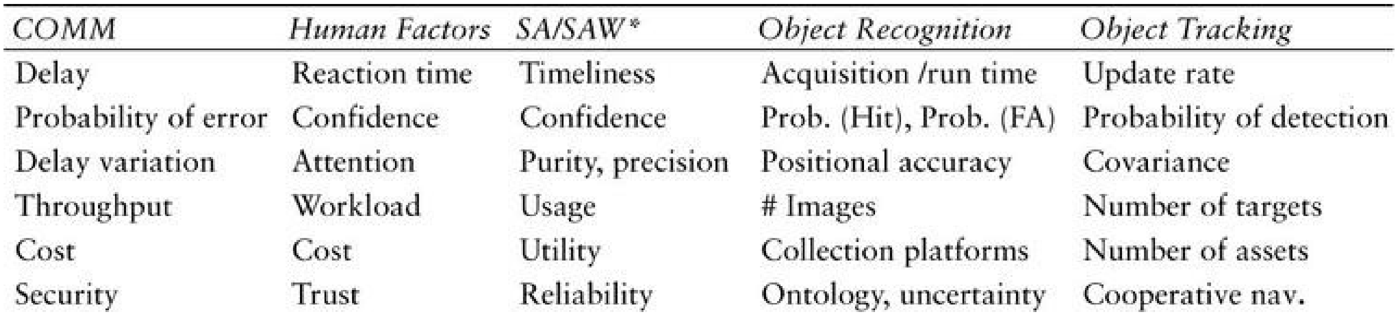
Name	*Description*	*Benefit*	*Limitation*
Boolean	Mathematical Rules of AND/OR/ NOT/NOR	Well defined operators Expandable for large systems	Lose meaning of the information Rules do not cover advanced reasoning
Fuzzy logic	Fuzzy Linguistic variables ranging between 1 and 0	Can handle partial information (between completely true/false) Possibility theory rather than probability theory	Fuzzy operators ill-defined, so use IF-THEN rules Number and partition of rules Meanings between operators
Markov chains	Probabilistic technique to model dynamic systems comprised by a finite set of defined states that represent the system internal workings and transitions between states.	Simple to define and easy to implement. Can incorporate uncertainty of both the states and the model with minimal computational complexity. Scalable and easily upgraded.	Combinatoric explosion of states that can reduce the utility of this approach State aggregation and pruning of states required for speed Knowledge engineering needed to define the states and the transition probabilities
Bayesian networks	Probabilistic reasoning network for partial information that provides a system of beliefs in given states that conditionally impact all other states in the network	Can incorporate uncertainty through the use of probability Propagates information through the network methodically Static structure can be adaptively updated with dynamic BN	Highly complex mathematics can occur with the network interconnection. Numerical solutions (approximations) to the functions can have low bounds on the induced error Requires a priori building the conditional probability tables
Entropy	Information measure that is the uncertainty contained in a signal.	Used to analyze, monitor, and predict uncertainty in measurement and information fusion. Since unitless, can be used to compare information from disparate discrete and continuous sources.	Notions of results must be related in total probability Entropy is unitless and can easily misinterpreted
Dempster Shafer	Plausibility reasoning based evidence accrual of observations	Designed to incorporate conflicting and negative information Does not need a fully specified model before proceeding as with a Bayesian network	Mathematical complexity of implementation. Many approximation techniques required for set analysis and partial information

The ability to utilize these methods for SA/SAW requires a standardized set of metrics.

2.7.2 Situation Assessment Metrics

Metrics [48, 49] are proposed in Table 2.3, as mapped to other relevant disciplines. Traditional methods for information theory communication metrics (e.g., delay variation) and human factors metrics (e.g., workload) are important for SA/SAW. Additionally, the LLIF metrics for object tracking and classification (see Chapter 13) provide inputs to SA. Bringing together various domains, we highlight the key metrics for SA/SAW of timeliness, confidence, precision, usage, utility, and reliability. Other metrics could be constructive, and future research and discussion will elaborate the formal list of standardized metrics. In Chapter 15, we will explore these metrics for SA worthiness evaluation.

Table 2.3 Metrics for Fusion, SA, and SAW



COMM	*Human Factors*	*SA/SAW**	*Object Recognition*	*Object Tracking*
Delay	Reaction time	Timeliness	Acquisition /run time	Update rate
Probability of error	Confidence	Confidence	Prob. (Hit), Prob. (FA)	Probability of detection
Delay variation	Attention	Purity, precision	Positional accuracy	Covariance
Throughput	Workload	Usage	# Images	Number of targets
Cost	Cost	Utility	Collection platforms	Number of assets
Security	Trust	Reliability	Ontology, uncertainty	Cooperative nav.

* Tadda et al. propose some of these for cyber SA: purity for detection, evidence recall, and attack score. [25].

2.7.3 SA/SAW Issues and Challenges

Over the many discussions of the models, we sought to detail the current challenges and issues for SA/SAW for future analysis as much work is left to be completed. For example, there is the need to coordinate SA/SAW with resource management (RM) [2], work over multiple player cultural issues for knowledge representation (KR) [13], as well as refine a set of useful metrics. Table 2.4 details current issues and challenges for SA/SAW.

Table 2.4 Issues and Challenges for SA and SAW

Issues for SA/SAW	*Challenges for SA/SAW*
Standard set of metrics for situational theory and knowledge representation (KR)	Scoping a common terminology and metrics
	Synthesizing timely information for reasoning
	Rigorous testing in experimental designs to define SA
User (individual differences) models for development of formal reasoning methods	Affording control strategies for different users
	Reducing analyst stress, fatigue, and workload
	Providing Semantic synonyms (different meaning between ideas)
Dynamic updating of knowledge delivery processes for planning	Learning associations form what is presented
	Aiding users in discovering new situations
	Projection to future situations
Tailored SA displays for situation representation	Detailing flexible interface designs to different users
	Aiding analysts in accessing multitudes of data
Information needs theory for SA/ SAW resource management	Determining what data to publish and subscribe
	Gathering data with various security and meta-data protocols
	Normalizing and transforming between syntactic algorithms

Conclusions

In this chapter, we highlighted common themes between SA and SAW models, and discussed issues in the variations. Future challenges exist in realizing the use of these models. The common issues for SA/SAW are (1) user-focused models (e.g., perceptual, interactive, control), (2) process analysis for situational modeling and model methods (e.g., Bayes nets, procedural/logical, perceptual, learning), (3) contextual analysis—operational situation (i.e., domain-dependent), (4) meaning instantiation (i.e., semantics and syntactic relations), and (4) SA metrics standardization (e.g., trust, bounds, uncertainty). The common challenges from the multiple SA/SAW tools include (1) explanation of SA processes that addresses evidence accumulation and constraints for situation theoretical reasoning, (2) graphical displays to facilitate inferential chains, collaborative interaction, and knowledge for situation representation, and (3) interactive control for corrections and utility assessment for situation management. The future will require engineers, practitioners, and pioneers to look at the SA/SAW models and improve the process models, formal theory (Chapter 4), and information representations relative to the changing environments (Chapter 4), technologies (Chapter 9), systems designs (Chapter 11), information management strategies (Chapter 5), and users.

rences

[1] Hall, D. L., and Llinas, J., “Introduction to Multisensor Data Fusion,” *Proc. of IEEE*, Vol. 85, No. 1, pp. 6–23, January 1997.

[2] Blasch, E., Kadar, I., Salerno, J., Kokar, M. M., Das, S., Powell, G. M., Corkill, D. D., and Ruspini, E. H., “Issues and Challenges in Situation Assessment (Level 2 Fusion),” *J. of Adv. in Info. Fusion*, Vol. 1, No. 2, pp. 122–139, December 2006.

[3] Blasch, E., Llinas, J., Lambert, D., Valin, P., Das, S., Chong, C-Y., Kokar, M. M., and Shah-bazian, E., “High Level Information Fusion Developments, Issues, and Grand Challenges— Fusion10 Panel Discussion,” *Intl. Conf on Info. Fusion*, 2010.

[4] Salerno, J. J., Sudit, M., Yang, S. J., Tadda, G. P., Kadar, I., and Holsopple, J., “Issues and Challenges in Higher Level Fusion: Threat/Impact Assessment and Intent Modeling (A Panel Summary),” *Intl. Conf on Info. Fusion*, 2010.

[5] Bosse, E., Roy, J., and Wark, S., *Concepts, Models, and Tools for Information Fusion*, Norwood, MA: Artech House, 2007.

[6] Waltz, E., and Llinas, J., *Multisensor and Data Fusion*, Norwood, MA: Artech House, 1990.

[7] *HQ USAF AFISC/SE Safety Investigation Workbook*, AFP-1271-Vol. 3, 1987 (available at *library.ndmctsgh.edu.tw/milmed/avitation/file-air/AFP-127-1-Vol-3.pdf*)

[8] Endsley, M. R., “Toward a Theory of Situation Awareness in Dynamic Systems,” *Human Factors Journal*, Vol. 37(1), pp. 32–64, March 1995.

[9] Fadok, D. S., Boyd, J., and Warden, J., *Air Power’s Quest for Strategic Paralysis*, Maxwell Air Force Base AL: Air University Press, (AD–A291621), 1995.

10] Shahbazian, E., Blodgett, D. E., and Labbé, P., “The Extended OODA Model for Data Fusion Systems,” *Intl. Conf on Info. Fusion*, 2001.

11] Blasch, E., “Situation Impact and User Refinement,” *Proceedings of SPIE*, Vol. 5096, April 2003.

12] Blasch, E., “Proactive Decision Fusion for Site Security,” *International Conference on Information Fusion*, July 2005.

13] Blasch, E., P. Valin, E. Bosse, M. Nilsson, J. Van Laere, and E. Shahbazian, “Implication of Culture: User Roles in Information Fusion for Enhanced Situational Understanding,” *International Conference on Information Fusion*, 2009.

14] Nilsson, M., *Capturing Semi-Automated Decision Making: The Methodology of CASA-DEMA*. Doctorial thesis, Orebro University, Sweden, 2010.

15] Rousseau, R., and R. Breton, “The M-OODA: A Model Incorporating Control Functions and Teamwork in the OODA Loop,” *Proceedings of the Command and Control Res, & Technology Symposium*, 2004.

16] Breton, R., “The Modelling of Three Levels of Cognitive Controls with the Cognitive-OODA Loop Framework,” Def. Res. & Dev. CA-Valcartier, *DRDC TR 2008-111*, September 2008.

17] Klein, G. A., “Recognition-Primed Decisions,” In W. B. Rouse (Ed.), Advances In *Man-machine Systems Res.*, Vol.5. JAI Press, 1989.

18] Blasch, E., and S. Plano, “DFIG Level 5 (User Refinement) issues supporting Situational Assessment Reasoning,” *International Conference on Information Fusion*, 2005.

19] Llinas, J., C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, “Revisions and extensions to the JDL data fusion model II,” *International Conference on Information Fusion*, 2004.

20] Blasch, E. P., and P. Hanselman, “Information Fusion for Information Superiority,” *IEEE National Aerospace* *and Electronics Conference*, 2000.

21] Blasch, E. P., “Ontological Issues in Higher Levels of Information Fusion: User Refnement of the Fusion Process,” *International Conference on Information Fusion*, 2003.

22] Blasch, E. P., “Assembling a distributed fused Information-based Human-Computer Cognitive Decision Making Tool,” *IEEE Aerospace and Electronic Systems Magazine*, Vol. 15, No. 5, pp. 11–17, May 2000.

23] Kadar, I., “Perceptual Reasoning Managed Situation Assessment and Adaptive Fusion Processing,” *Proceedings of SPIE*, Vol. 4380, 2001.

24] Roy, J., “From Data Fusion to Situation Analysis,” *International Conference on Information Fusion*, 2001.

25] Matheus, C. J., M. M. Kokar, and K. Baclawski, “A Core Ontology for Situational Awareness,” *International Conference on Information Fusion*, 2003.

26] Matheus, C. J., M. M. Kokar, K. Baclawski, J. A. Letowski, C. Call, M. Hinman, J. Salerno, and D. Boulware, “SAWA: An assistant for Higher-level Fusion and Situation Awareness,” *Proceedings of SPIE*, Vol. 5813, 2005.

27] Salerno, J. J., M. Hinman, and D. Boulware, “Building A Framework for Situation Awareness,” *International Conference on Information Fusion*, 2004.

28] Tadda, G., J. J. Salerno, D. Boulware, M. Hinman, and S. Gorton, “Realizing situation awareness within a cyber environment,” *Proceedings of SPIE*, Vol. 6242, 2006.

29] Lambert, D. A., “STDF Model based Maritime Situation Assessments,” *International Conference on Information Fusion*, 2007.

30] http://en.wikipedia.org/wiki/Entity

31] http://en.wikipedia.org/wiki/Object_(philosophy)

32] http://en.wikipedia.org/wiki/Action_(philosophy)

33] Salerno, J. J., and G. Tadda, “Ranking Activities based on their Impact and Threat,” *International Conference on Information Fusion*, 2009.

34] Waltz, E., *Knowledge Management in the Intelligence Enterprise*, Artech House, 2003.

35] Blackman, S., and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, MA, 1999.

36] Blasch, E., “Modeling Intent for a target tracking and identification Scenario,” *Proceedings of SPIE*, Vol. 5428, April 2004.

37] Blasch, E. P., and S. Huang, “Multilevel Feature-based fuzzy fusion for target recognition,” *Proceedings of SPIE*, Vol. 4051, 2000.

38] Wei, M., E. Blasch, G. Chen, J. B. Cruz, L. Haynes, M. Kruger, and I. Sityar, “Game Theoretic Behavior Features change prediction in Hostile Environments,” *Proceedings of SPIE*, Vol. 6567, 2007.

39] Chen, H., G. Chen, E. Blasch, J. B. Cruz, L. Haynes, and M. Kruger, “Information Fusion and Visualization of Large Complex Attack Graphs for Networks Security,” *Proceedings of SPIE*, Vol. 6567, 2007.

40] Shen, D., G. Chen, J. B. Cruz, L. Haynes, M. Kruger, and E. Blasch, “A Markov Game Theoretic Data Fusion Approach for Cyber Situational Awareness,” *Proceedings of SPIE*, Vol. 6571, 2007.

41] Holsopple, J. and S. J. Yang, “FuSIA: Future Situation and Impact Awareness,” *International Conference on Information Fusion*, 2008.

42] Chen, G., D. Shen, J. B. Cruz, M. Kruger, and E. Blasch, “Game Theoretic Approach to Threat Prediction and Situation Awareness,” *Journal of Advances in Information Fusion*, Vol. 2, No. 1, June 2007.

43] Suidt, M., M. Holender, A. Stotz, T. Rickard, and R. Yager, “INFERD and Entropy for Situational Awareness,” *Journal of Advances in Information Fusion*, Vol. 2, No. 1, June 2007.

44] Schrag, R. C., M. Takikawa, P. Goger, and J. Eilbert, "Performance Evaluation for Automated Threat Detection," *Journal of Advances in Information Fusion*, Vol. 2, No. 2, Dec. 2007.

45] Foo, P. H., G. W. Ng, K. H. Hg, and R. Yang, "Application of Intent Inference for Air Defense and Conformance Monitoring," *Journal of Advances in Information Fusion*, Vol. 4, No. 1, June 2009.

46] George, J., J. L. Crassidis, T. Signh, and A. M. Fosbury, "Anomaly Detection Using Context-Aided target Tracking," *Journal of Advances in Information Fusion*, Vol. 6, No. 1, June 2011.

47] Das, S., *High-Level Data Fusion*, Artech House, Norwood, MA, 2008.

48] Blasch, E., M. Pribilski, B. Daughtery, B. Roscoe, and J. Gunsett, "Fusion Metrics for Dynamic Situation Analysis," *Proceedings of SPIE*, Vol. 5429, 2004.

49] Salerno, J. J., E. Blasch, M. Hinman, and D. Boulware, "Evaluating Algorithmic Techniques in Supporting Situation Awareness," *Proceedings of SPIE*, Vol. 5813, 2005.

CHAPTER 3

The State Transition Data Fusion Model

Dale A. Lambert (Australia)

Information Revolution

We are in the midst of an information revolution. Adaptations in both transportation and telecommunications have altered the extent of information access. Transportation has provided increased information access through greater physical presence, while telecommunications have provided increased information access through greater virtual presence. Recent advances in satellite communications, mobile telephones, text messaging, video conferencing, and the Internet have all contributed to a substantially richer capacity for information access through virtual presence. The outcome has been the emergence of an information rich economy and a shift in the sphere of influence from localisation to globalisation. These advancements in information access engender an information revolution. The information revolution brings with it an expectation that we should have *awareness* of a greater *quantity* and *quality* of information, and acquire that awareness more *rapidly*.

3.1.1 Situation Awareness

Our awareness of the world is mediated by our mental processes. Endsley defines situation awareness through the mental processes of perception, comprehension and projection.

> Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future [1].

Adding sensation as a precursor to perception yields the coarse model for situation awareness presented in Figure 3.1.

Human situation awareness has well-researched strengths and weaknesses when applied to certain tasks. Processing large quantities of data with precision is not a particular strength of human processing. Miller [2] demonstrated that human short-term memory has a storage capacity of 7 ± 2 items, and so it comes as no surprise that Giompapa et al. [3] report that human operators tracking an aircraft can competently cope with 6.8 tracks at any given time, despite at times being asked to process hundreds of tracks simultaneously. Similarly, an analysis of aviation accidents found that 80 to 85 percent of all aviation accidents are attributable to human error [4], highlighting Reason's [5] observation that while humans can be heroes, they can also be hazards. Moreover, in studies related to human error, poor situation awareness has been identified as the leading contributor:

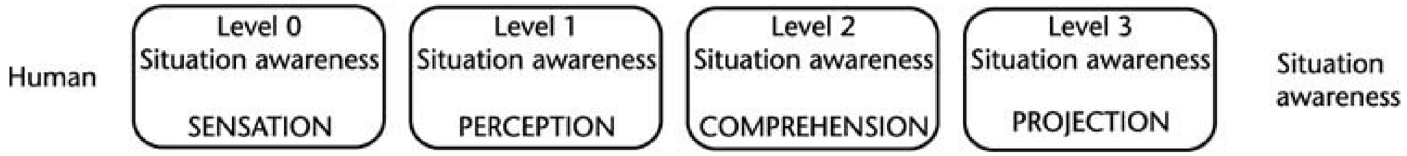


Figure 3.1 Components of situation awareness.

> Problems with SA [Situation Awareness] were found to be the leading causal factor in a review of military mishaps. [and] In a study of accidents among major airlines, 88% of those involving human error could be attributed to problems with situation awareness as opposed to problems with decision making or flights skills [4].

By contrast, processing large quantities of data with precision is a particular strength of machine processing, while in other aspects machines exhibit deficiencies. A natural consideration then, is to combine the complementary strengths of human and machine processing to overcome their individual weaknesses, as a basis for delivering improved human situation awareness.

3.1.2 Data Fusion

Machine assisted human situation awareness comes in the form of data fusion. The 1987 JDL Data Fusion Subgroup's Data Fusion Lexicon [6] defines data fusion as:

> … a process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete and timely assessments of situations and threats, and their significance.

In [7] and [8], data fusion is more broadly defined as:

> … the process of utilising one or more data sources over time to assemble a representation of aspects of interest in an environment.

The traditional roots of the data fusion community are in sensor fusion, where the data sources are established sensors like radars, the aspects of interest in the environment are moving objects, and a set of state vectors serve as the representation of each moving object. The author's broader definition reflects the ambition to generalize sensor fusion, or lower-level fusion, into so-called higher-level fusion, in which the aspects of interest in the environment are not restricted to objects, the data sources need not resemble radars, and the representation of the aspects of interest in the environment can be considerably more complex than state vectors. Higher-level fusion moves beyond sensor fusion to cater for the information revolution, but it requires a renaissance within the data fusion community to address the new situation awareness problems spawned by the information revolution.

The JDL model has emerged as the dominant model within the data fusion community. The Joint Directors of Laboratories (JDL) model was proposed in the late 1980s [9], with various revisions of it [10–12] serving as the dominant model for data fusion. Figure 3.2 illustrates a variant of its revised form [10].

From the standpoint of higher-level fusion systems, the revised JDL model of Figure 3.2 has the benefit of incorporating both lower-level and higher-level fusion, with the product of lower-level fusion taken to include sub-object assessments (Level 0), object assessments (Level 1), and their refinement (Level 4); while the product of higher-level fusion is taken to comprise situation assessments (Level 2), impact assessments (Level 3), and their refinement (Level 4).

Although the JDL model was conceived independently of Endsley's account of human situation awareness, in [7] the author first noted that the machine components of the JDL model represent machine counterparts of the components of Endsley's human situation awareness process. If sensation is added as a Level 0 to Endsley's definition of situation awareness, then there is a fairly direct correspondence between Levels 0 to 3 of situation awareness and Levels 0 to 3 of machine fusion respectively, while the adaptive Level 4 of the JDL model can be partitioned across JDL Levels 0 to 3. Figure 3.3 illustrates this association between situation awareness and data fusion.

Thus, situation awareness can be understood as the human counterpart to machine data fusion, while data fusion can be conceived as the machine counterpart to human situation awareness.

3.1.3 Renaissance

On its own, the closing observation of Section 3.1.2 is not a sufficient precursor for renaissance within the fusion community. Figure 3.4 illustrates the Data Fusion Information Group (DFIG) model [13].

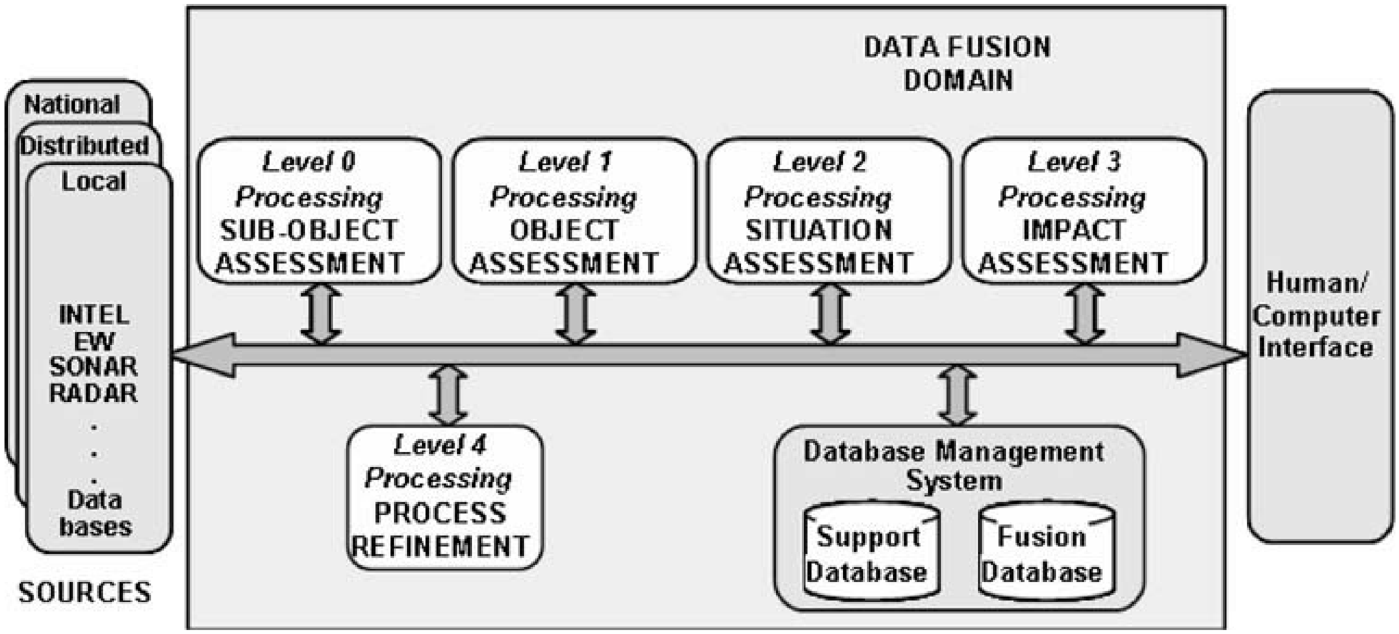


Figure 3.2 A revised JDL model of data fusion.

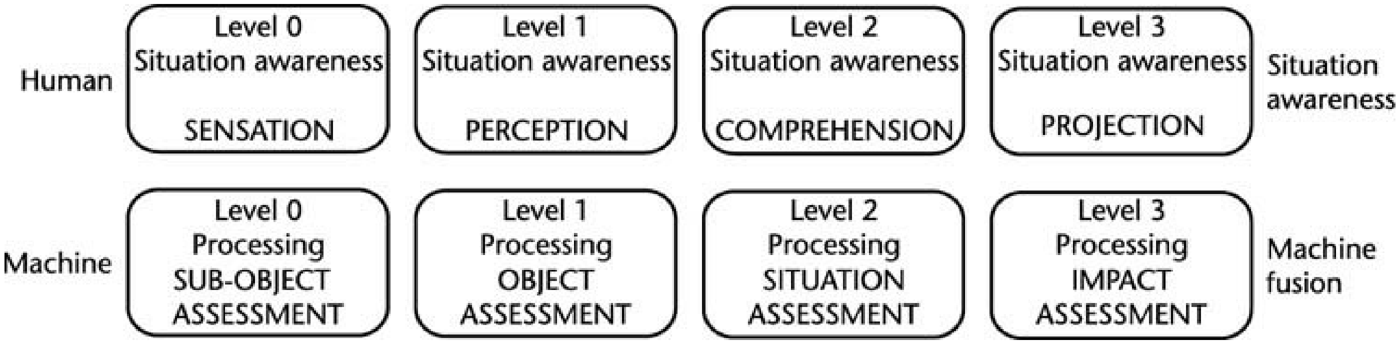


Figure 3.3 Situation awareness and data fusion.

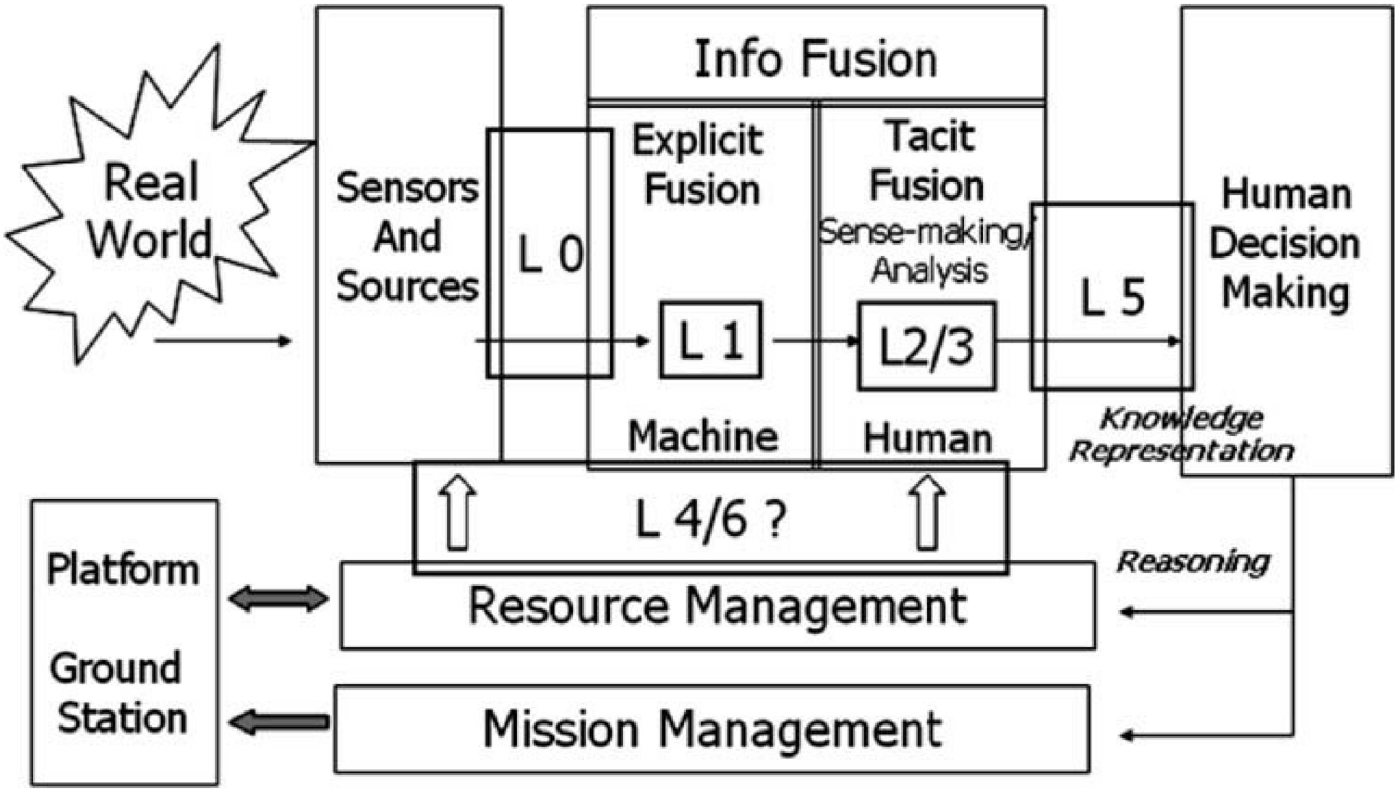


Figure 3.4 DFIG model.

The DFIG model introduces a Level 5 [14] for human decision-making and a Level 6 [11] for resource management, while aligning lower-level fusion with machines and aligning higher-level fusion with people. This division of human and machine labor means that the issue of interfacing machines with people is only a consideration for lower-level fusion, with "dots on maps" and "lines on maps" displays, like those featured in Figure 3.5, typically presenting the machine-based feature vector and state vector representations of moving objects to people.

The DFIG demarcation of human and machine labor across higher and lower level fusion does not progress the required renaissance because it does not promote machine-based comprehension and projection, as indeed it must if the issues of the information revolution are to be genuinely addressed.

The state transition data fusion model (STDF) [15-17] was motivated by the need for a renaissance in the data fusion community, and toward that end, offers the following first tenet:

> Tenet 1: Situation awareness is the function of fusion performed by people, while machine fusion is the function of situation awareness performed by machines.

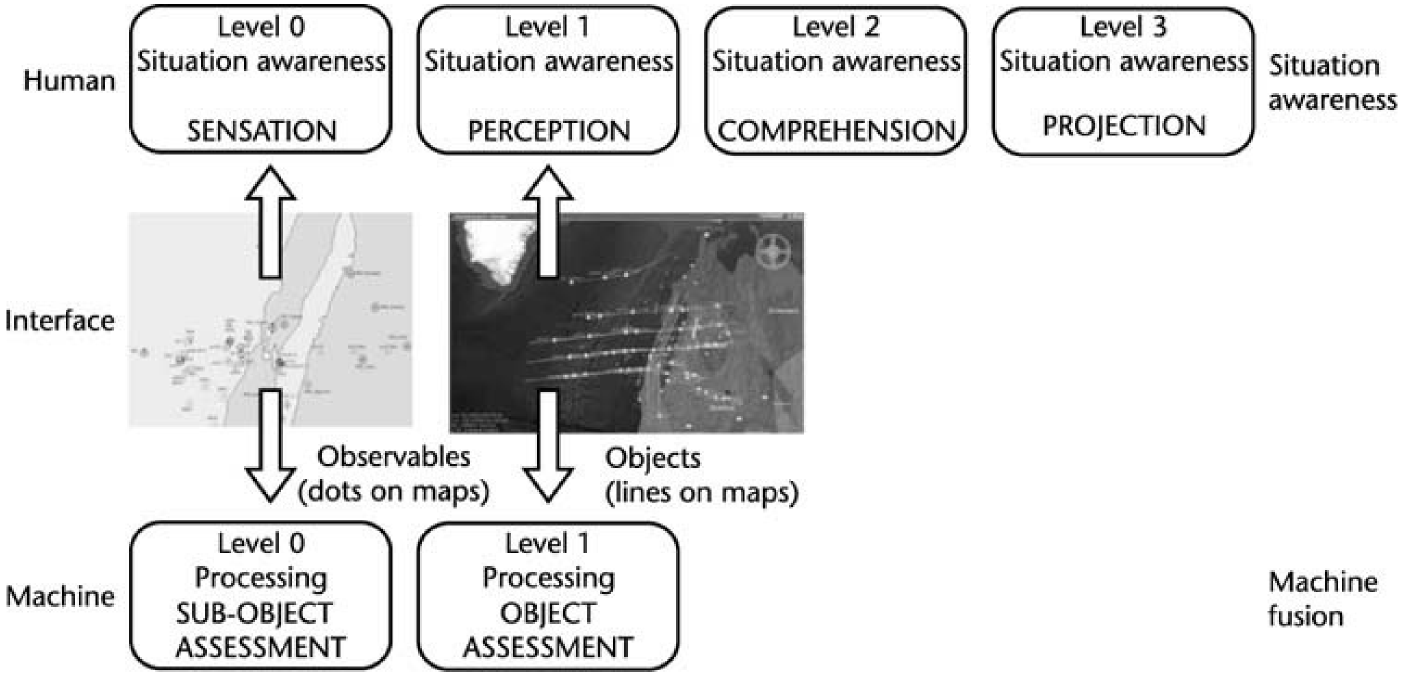


Figure 3.5 DFIG situation awareness and data fusion.

Tenet 1 implicitly rejects a division of human and machine labor based on JDL levels. Under the STDF model, the fusion function is characterized by JDL levels 0 to 3, which can be performed by people, machines, or some combination of the two. When the fusion function is performed by a human, it is termed "situation awareness." When the fusion function is performed by a machine, it is termed "machine fusion," and in contrast to the DFIG model, machine fusion actively includes JDL Levels 2 and 3. The human interpretation of the STDF model subsumes the need for a JDL Level 5. JDL Level 4 is likewise superfluous, as it can be absorbed within each of JDL Levels 0 to 3. This is because the general form of the fusion process outlined in Section 3.3.2 is recursive and adaptive for each of these levels.

Renaissance derives from the capacity for a mixed initiative, in which the weaknesses of humans and machines can be complemented by the strengths of machines and humans respectively. The appropriate level of automation for each of the JDL Levels 0 to 3 then becomes the central question, and this requires empirical resolution. Endsley and Kaber [18] identify 10 levels of automation. As there is an expectation of at least some level of automation for each fusion level, it means the issue of human-machine interface needs to be considered for each of the JDL fusion Levels 0 to 3, as Figure 3.6 suggests. "Dots on maps" displays will not prove sufficient for the information revolution.

The JDL Levels provide an extremely valuable framework for understanding fusion, but the JDL model treats each of the fusion levels as a black box, by not describing how processing within each of the levels is to be undertaken. As a consequence, the JDL model highlights the differences between the fusion levels without celebrating the uniformities across those fusion levels. This has effectively led to two communities of interest: a traditional lower-level fusion community and a fragmented higher-level fusion community, each having limited interaction with the other. This author believes the fusion community would be better served as a whole through a common understanding. Thus, unification served as a second motivation for the STDF model. The aim was to venture inside the JDL's black boxes to expose commonalities and differences across the JDL levels.

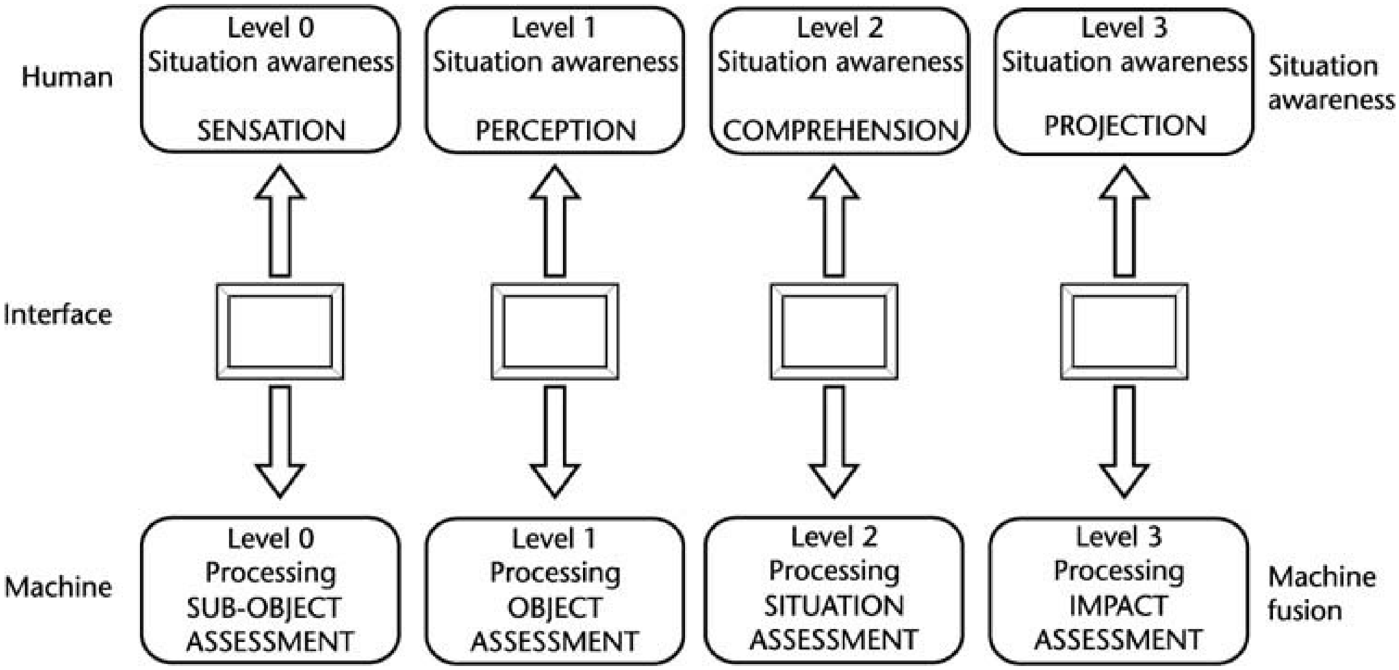


Figure 3.6 STDF situation awareness and data fusion.

State Transitions

State transitions serve as the key concept of the STDF model.

3.2.1 Classifcation

For fusion to occur, an agent (person or machine) needs to be able to classify aspects of interest in its environment. Classification occurs whenever z classifies x as x_q, where z is the classifier process, x is the classified process, and x_q is the qualitative classification. z is the agent, the classifier process performing the classification. x is some fragment of reality being classified by the classifier process. x_q can be understood from two perspectives: the theorist perspective and the agent perspective.

> The theorist perspective is the external perspective through which we assess the workings of our agent. This is the perspective of the scientist examining how the agent operates; it is the "outside looking in" perspective. The theorist perspective is rather like a doctor looking at the patient's electroencephalogram while the patient is looking at a picture of a red rose. From this perspective, x_{red} is a fragment of z, the mental representation of x within z.
> The agent perspective is the internal perspective of the agent assessing the world. This is the perspective of being the embedded operating agent; it is the "inside looking out" perspective. The agent perspective is rather like being the patient seeing a picture of a red rose while a doctor is looking at your electroencephalogram. From this perspective, x_{red} is z's experience of x as an instance of type red. x_{red} is x as a red object.

In classifying the world, the classifier relies upon there being uniformities in the world. This author contends that some patterns of change repeat, and that a classifier's ability to negotiate the dynamic world rests with that classifier's attention to these repeating patterns of change.

> Tenet 2: Repeated patterns of change defne uniformities in the world.

Classifiers are often participants in repeated patterns of change, and because of this, are able to classify uniformities in the world. A repeated pattern of change could be that whenever presented with the picture x of a red rose, classifier z responds with a representation x_{red}. The human retina coarsely possesses three kinds of wavelength receptors, each distinguished by its pigment's ability to absorb different wavelengths of light, and each of which roughly samples the primary colours, while a machine might engage a spectrometer to register primary colours as a bit pattern. From the theorist's perspective, x_{red} is the wavelength receptor response within z (human or machine), and the repeated pattern of change between x and response $x$red reflects the ability of classifier z to classify a uniformity in the world. From the agent's perspective, the experienced uniformity is redness in the world.

A classifier may classify more sophisticated uniformities in the world. The classification of x as a switch, x_{switch}, and y as an appliance, $y_{\text{appliance}}$, could result from a repeated pattern of interaction with fragments x and y in the world where z classifies x as x_{down} accompanies z classifies y as y_{on}, and z classifies x as x_{up} accompanies z classifies y as y_{off}. In that instance, z is classifying a repeated pattern of change between x and y based on z's repeated patterns of change with x and with y. Moreover, having established such a repeated pattern of change between any two fragments of the world x and y, this repeated pattern of change can be used to abstractly classify switches as a uniformity anywhere in the universe. Switches on Mars are still switches even when they are not being observed, because they still adhere to the prescribed repeated pattern of change. Switches, of various forms, are uniformities in the universe.

3.2.2 States

If uniformities are understood through repeated patterns of change in the world, then state transitions can be used to characterize uniformities. State transitions characterize the world in terms of persistence and change. States classify persistence. Transitions classify changes between persistent states. In the STDF model, state transitions characterize the world at each fusion level. This is Tenet 3:

> Tenet 3: At each of the JDL Levels 0 to 3, the world can be understood in terms of transitions between states.

Figure 3.7 shows the abstract notion of a state transition.

Figure 3.7 State transitions.

State classifications are generally constrained in a number of ways.

First, state classifications are usually spatiotemporally bounded. A classified state of the world often refers to a classified region of space over some period of time. The state of the room in which I am writing does not include commentary on matters concurrently occurring in Antarctica, nor is it commenting on what the region of space that is now the room was like in the year 1462. *Part of the skill of the classifier is to be able to select the regions of space over periods of time that they need to consider to satisfy their goals.*

Second, state classifications are always partial in that they never say everything that could be said about the referenced spatiotemporal region-period. To do otherwise is impractical for both affirmations and denials. With respect to denials, a complete account of the state of the room in which I am writing would report that it contains no sea monsters, that Plato is not sitting in the room, ad infinitum. Equally, a state classification of the state of the room in which I am writing will not include all true affirmations, which could extend to include the wavelength of the chairs measured in angstroms, the name of the manufacturer of the glue used in the chairs, and so forth. *Part of the skill of the classifier is to be able to select the affirmations and denials that they need to consider in order to satisfy their goals.*

Third, state classifications are always conceptual classifications of the world. Understanding the state of the room in which I am writing involves me classifying sensations as referring to chairs, tables, lights, and so on. I can arrive at *different state classifications* of the same spatiotemporal region-period *depending upon which concepts* are included or precluded in my classification. I could classify the state of the room in which I am writing in a way that includes the concept of lights. I could equally classify the state of the room in which I am writing in a way that does not consider the concept of lights. So state classifications are not only partial with respect to which affirmations and denials they consider, they are also partial with respect to the very concepts through which those affirmations and denials are formed. The nature of conceptualization varies across the JDL levels. A JDL level $n+1$ state classification can offer a richer account of the same region-period than a JDL level n classification, for all $n \in \{0, 1, 2\}$. *Part of the skill of the classifier is to be able to select the conceptualisations at each STDF level that they need to consider to satisfy their goals at that STDF level.*

Fourthly, state classifications identify *persistence* in the world. The affirmations and denials of a state classification must remain true throughout the nominated spatiotemporal region-period. This could include assertions of *static persistence*, such as there being a chesterfield in the corner of the room in which I am writing. A state classification can also include assertions of *persistent change*, such as a ship traversing a sea lane. Thus, the application of persistence is relative to the conceptualisations used to construct each state, and the application of change as transitions between states, is likewise relative to the conceptualisation used to formulate the states. *Part of the skill of the classifier is to be able to select the instances of persistence at each STDF level that they need to consider to satisfy their goals at that STDF level.*

3.2.3 Transitions

Persistence and change are in part the outcomes of the imposition of conceptualisations of *identity* and *difference* with respect to time. To illustrate, a curtain is blowing back and forth with the wind in the room in which I am writing.

- If I omit the curtain concept from my state classification, then I might have a *static persistent state* classification.
- If I instead admit both the curtain concept and the affirmation that the curtain is blowing back and forth, then I might classify a *persistent change state*.
- Finally, if I admit the curtain concept, but define two states in which the curtain is moving "back" and is moving "forth" respectively, then the motion of the curtain surfaces as change reported by a *state transition*.

State transition conceptualisations load identity content into state classifications and engage transitions merely to report changes between different state classifications. State transitions may also only provide partial classifications of time, and so the author will generally refer to time steps when referring to a period of time associated with a persistent state. A transition from state classification $s_i(k)$ at time step k to state classification $s_i(k+1)$ at time step $k+1$ requires the period of time associated with $s_i(k)$ to complete before $s_i(k+1)$ commences, but it does not necessarily require that $s_i(k+1)$ commence immediately after $s_i(k)$ completes. Some time might be unaccounted for. The index, i, on the state descriptor appears because there may be several different partial states $s_1(k)$, $s_2(k)$, …, $s_n(k)$ associated with time step k. *Part of the skill of the classifier is to be able to select partial states and the transitions between them that they need to consider to satisfy their goals.*

3.2.4 JDL States in the World

The JDL fusion model imposes some additional presumptions about the nature of states in the world. The JDL model is usually presented as a functional model, not an architectural paradigm.

> It should be emphasized that the model was conceived as a functional model, not as a process model or as an architectural paradigm [11].

This is only partly true. The nature of functionalism [20] is to specify how to do something, but only at a level of abstraction. As Steinberg and Bowman [11] remarked,

> Nonetheless, there is often a natural progression from raw measurements to entity states, to situation relationships, to utility prediction, and to system performance assessment, in which data are combined as suggested by the level ordering. Such composition of estimated signals (or features), entities, and aggregates per Levels 0-2 is quite natural; with a corresponding reverse flow of contexts, … Utility is generally assessed as a function of an estimated or predicted situational state, rather than of the state of a lone entity within a situation. That is the reason that Level 3 fits naturally “above” Level 2 [11].

Under the STDF interpretation, there is a natural hierarchy across the JDL levels, with JDL level n states in the world figuring in JDL level $n + 1$ states in the world, for all $n \in \{0, 1, 2\}$.

Level 0 observable assessments presume there are *observables* in the world to assess. Each observable $\{si(t)|t \in$ Time_Step & $t \leq k\}$ is a state transition of states $s_i(t)$ in the world, where each state $si(t)$ is understood as a feature vector $f_i(t)$ comprising measurable feature samples in the world. Thus each observable at time step k is understood as a set of transitioning feature vectors $\underset{\rightarrow}{f_i}(k) = \{f_i(t)|t \in \text{Time\_Step}$ & $t \leq k\}$, as Figure 3.8 illustrates.

Level 1 object assessments presume there are *objects* in the world to assess. Each object $\{s_i(t)|t \in$ Time_Step & $t \leq k\}$ is a state transition of states $s_i(t)$ in the world, where each state $s_i(t)$ is understood as a state vector $u_i(t)$ comprising measurable properties of observables in the world, with the properties based on features. Thus, each object at time step k is understood as a set of transitioning state vectors $\underset{\rightarrow}{u_i}(k) = \{u_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$, as Figure 3.9 illustrates.

Level 2 situation assessments presume there are *situations* in the world to assess. Each situation $\{s_i(t)|t \in$ Time_Step & $t \leq k\}$ is a state transition of states $s_i(t)$ in the world, where each state $s_i(t)$ is understood as a state of affairs $\Sigma_i(t)$ comprising relations between objects in the world. Thus, each situation $\underset{\rightarrow}{\Sigma_i}(k)$ at time step k is understood as a set of transitioning states of affairs $\underset{\rightarrow}{\Sigma_i}(k) = \{\Sigma_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$, as Figure 3.10 illustrates.

Level 3 impact assessments presume there are *scenarios* in the world to assess. Each scenario $\{s_i(t)|t \in$ Time_Step & $t \leq k\}$ is a state transition of states $s_i(t)$ in the world, where each state $s_i(t)$ is understood as a scenario state $S_i(k) = \underset{\rightarrow}{\Sigma_i}(\partial(k))$, being a situation into the future for monotonic look ahead time $\partial(k) > k$, with monotonicity requiring that $\partial(k_2) \geq \partial(k_1)$ whenever $k_2 > k_1$. Thus, $S_i(k) = \{\Sigma_i(n)|n \in$ Time_Step & $n \leq$

$\partial(k)\}$ and so $\underline{\Sigma}_i(k) = (\{\Sigma_i(n) | n \in \text{Time\_Step} \;\&\; n \leq k\} \cup \{\Sigma_i(n) | n \in \text{Time\_Step} \;\&\; k < n \leq \partial(k)\})$ Time_Step $\& \; k < n \leq \partial(k)\})$ is composed of situation $\underline{\Sigma}_i(k) = \{\Sigma_i(n) | n \in \text{Time\_Step} \;\&\; n \leq k\}$ and partial future $\{\Sigma_i(n) | n \in$ Time_Step $\& \; k < n < \partial(k)\}$. Each scenario $\underline{S}_i(k)$ at time step k is understood as a set of transitioning scenario states $\underline{S}_i(\mathrm{k}) = \{S_i(\mathrm{t}) | t \in \text{Time\_Step} \;\&\; t \leq k\} = \{\{\Sigma_i(n) | n \in \text{Time\_Step} \;\&\; n \leq \partial(t)\} | t \in \text{Time\_Step} \;\&\; t \leq k\}$ A scenario $\underline{S}_i(k)$ at time step k is a monotonically increasing set of scenario states (situations) in the sense that if t_1, t_2 $\in$ Time_Step and $t_1 < t_2 \leq k$, then $S_i(t_1) \in \underline{S}_i(\mathrm{k})$, $S_i(t_2) \in \underline{S}_i(k)$ and $S_i(t_1) \subseteq S_i(t_2)$ Figure 3.11 illustrates a scenario.

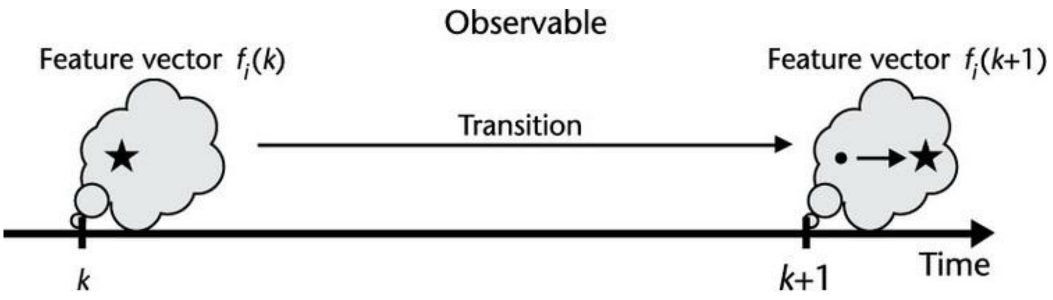


Figure 3.8 Observables.

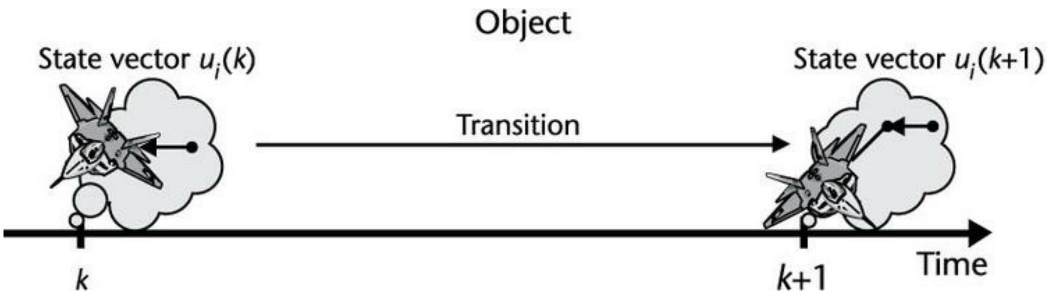


Figure 3.9 Objects.

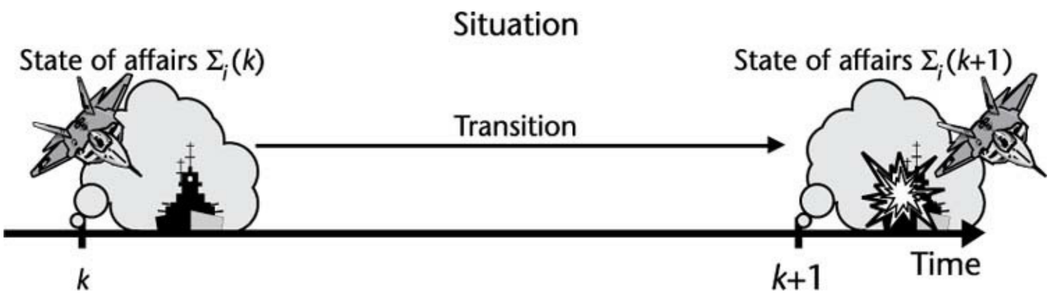


Figure 3.10 Situations.

As we progress through the JDL levels, the nature of the uniformities of interest in the world becomes increasingly more sophisticated. At Level 0, the world is understood in terms of observables with different kinds of measurable features. At Level 1, the world is understood in terms of objects with properties formed from measurable features of observables. At Level 2, the world is understood in terms of situations formed from relations between objects. At Level 3, the world is understood in terms of scenarios formed from situation transitions.

The STDF Fusion Process

Under the STDF model, the world at each fusion level is understood through a common fusion process. This is Tenet 4:

> Tenet 4: At each of the JDL Levels 0 to 3, a common fusion process applies that aims to explain the world through prediction and observation.

3.3.1 Prediction, Observation, and Explanation

At any time step k, the world of interest is composed of a number of individual states $\{s_i(k) | k \in \text{Time\_Step} \; \& \; i \in N^+(p)\}$ of interest where $N^+(p) =_{\text{df}} \{0, 1, \ldots, p\}$ for natural number $p \in N$, and up to Time Step k the world is understood in terms of a set of transited states $\{\{s_i(t) | t \in \text{Time\_Step} \; \& \; t \leq k\} | i \in N+(p)\}$. At time step k+1 the agent, be it human or machine, senses new states in the world through its sensors and transfers the sensed data to an observation process. The observation process draws upon a prediction process, which accesses previous representations of states in the world to identify predicted observations under previous explanations. On the basis of comparisons between actual and predicted observations, control transfers to an explanation process that establishes new representations about states in the world. Under this framework, the fusion process concerns the prediction, observation and explanation of state transitions in the world. The process recursively: observes to explain; explains to predict; and predicts to observe. If I am aware of a falling sheet of glass then I may draw on previous explanations to predict (explain to predict) that I will observe (predict to observe) it shatter, and upon observing the resultant glass fragments, explain that the sheet of glass has shattered (observe to explain). As science is often cast in terms of prediction and explanation through observation of the world, the STDF framework underscores the fundamental nature, and hence broad applicability, of the fusion process. Figure 3.12 presents the STDF model at this abstract level.

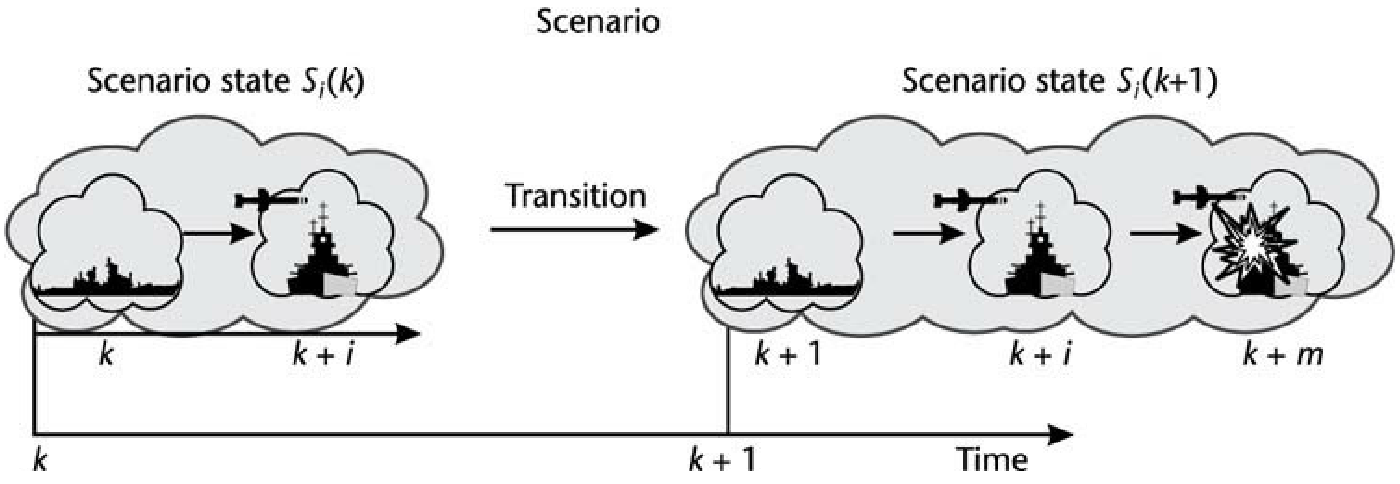


Figure 3.11 Scenario.

The recursive observes-to-explain, explains-to-predict and predicts-to-observe process is a natural strategy for any classifier, z, to adopt. If z is classifying x as x_q, then z is classifying x as an instance of type q, which effectively requires x to conform to some repeated pattern of change. z will classify that repeated pattern of change in terms of a pattern of transitions between certain states, states which z will experience as x_{q1}, …, x_{qn}. If the repeated pattern of change for x being an instance of type q requires state x_{qj} to temporally follow state x_{qi}, then z observing x as an instance of type q_i would result in z explaining x as x_{qi}; then using the explanation x_{qi} to predict state x_{qj}; and then using the predicted state x_{qj}: to try to identify observing x as an instance of type q_j.

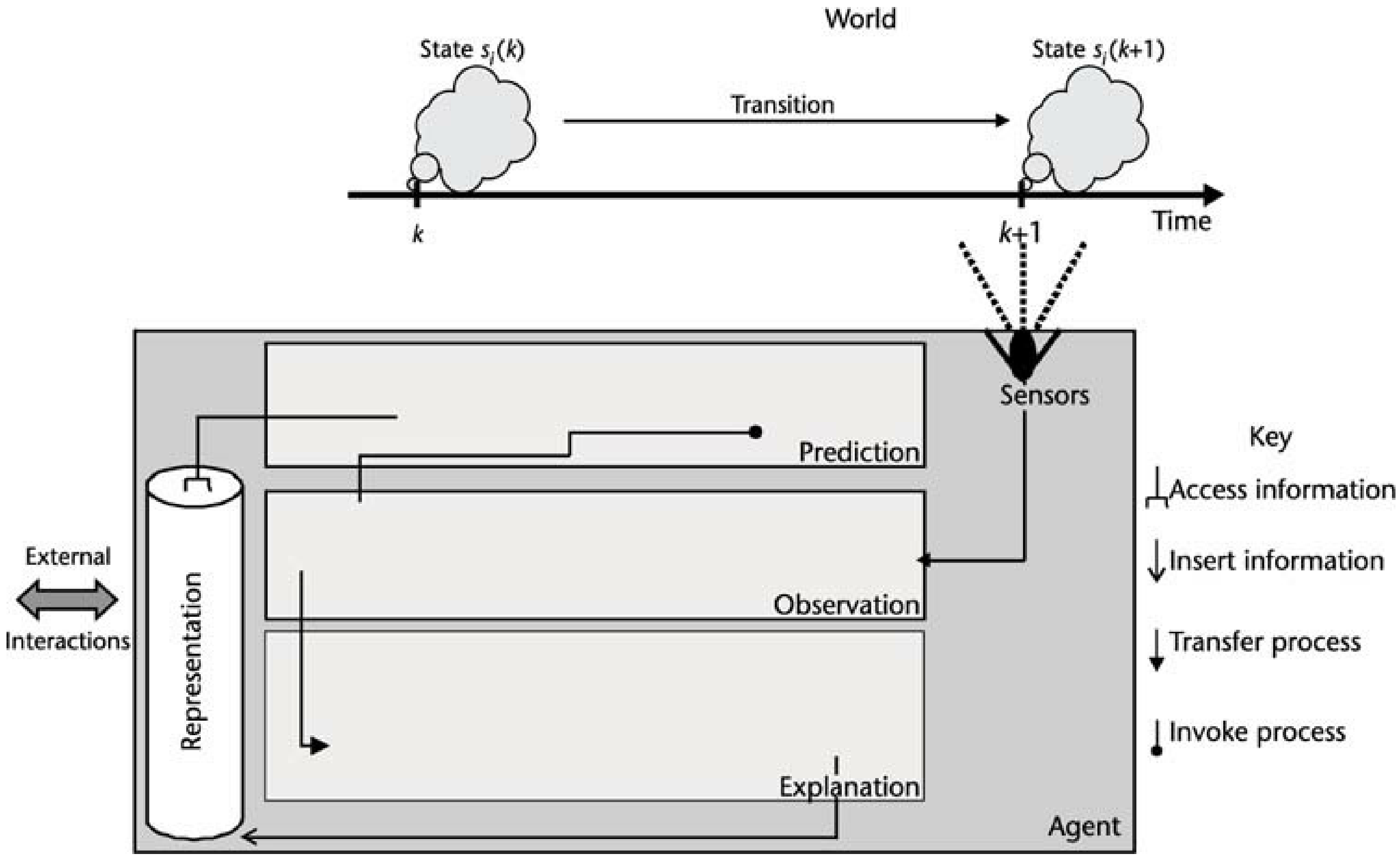


Figure 3.12 Abstract STDF model.

3.3.2 The General Form of a Fusion Process

The abstract prediction, observation and explanation fusion process of Tenet 4 and Figure 3.12, can be further refined as Figure 3.13 indicates. This more refined account of the STDF fusion process equally applies generically to Levels 0 to 3 of the JDL model.

At any time step k, the world of interest is composed of a number of individual states $\{s_i(k) | i \in N^+(p)\}$ for some number, p, and up to time step k the world is understood in terms of a set of transited states $\{\{s_i(t) | i \in N^+(p)\} | t \in \text{Time\_Step} \ \& \ t \leq k\}$ for each of these individual states.

- At time step k +1, the agent senses a number of new states in the world $\{s_i(k+1) \, | i \in N^+(p_1)\}$ through its sensors and transfers the corresponding sensations $\{e_j(k+1) | j \in N^+(q)\}$ to an observation process.
- The observation process involves a detection process to potentially identify a detection, $d_j(k+1)$, from each sense datum, $e_j(k+1)$; a registration process that yields an observation, $o_j(k+1)$, by normalizing the detection, $d_j(k+1)$, relative to a frame of reference; and then an association process.

- The association process first draws upon a prediction process. The prediction process accesses previous representations, $s_i(k|k)$, of states, $s_i(k)$, in the world at time step k; applies a state prediction process to representation, $\hat{s}_i(k|k)$, to posit a predicted state representation(s), $\hat{s}_i(k+1|k)$, of predicted state, $s_i(k+1)$, in the world at time step $k+1$; and then applies an observation prediction process to predicted state representation, $\hat{s}_i(\mathrm{k+1|k})$, to posit predicted observation(s), $\hat{o}_i(k+1|k)$, at time step $k+1$. Where multihypothesis state and observation predictions occur for state $s_i(k+1)$ from $\hat{s}_i(k|k)$, these can be labeled $\hat{s}_{i,1}(k+1k)$, ..., $\hat{s}_{i,w}$ $(k+l|k)$ and $\hat{o}_{i,1}(k+1|k)$, ..., $\hat{o}_{i,w}(k+1|k)$ respectively, for $w \in N^+$.
- The association process then matches the observations $\{o_j(k+1)|k \in \text{Time\_ Step} \;\&\; j \in N^+(\mathrm{q})\}$ at time step $k+1$ to (one or more) predicted observations $\{\hat{o}_i(k+1|k)|k \in \text{Time\_Step} \;\&\; i \in N^+(p)\}$ for time step $k+1$ and then transfers control to an explanation process.
- The explanation process must contend with three possible outcomes from the comparison of observations, $\{o_j(k+1)|k \in \text{Time\_Step} \;\&\; j \in N^+(q)\}$, at time step $k+1$ with the predicted observations, $\{\hat{o}_i(k+1|k)|k \in \text{Time\_Step} \;\&\; i \in N^+(p)\}$, for Time Step $k+1$:
  - If the observation, $o_j(k+1)$, successfully matches with a predicted observation, $\hat{o}_i(k+1|k)$, then an update success process is invoked to produce the explained representation, $\hat{s}_i(k+1|k+1)$, of state $s_i(k+1)$ at time step $k+1$.
  - If predicted observation, $\hat{o}_i(k+1|k)$, fails to match with any observation, $\hat{o}_l(k+1)$, then an update failure process is invoked to produce the explained representation, $\hat{s}_i(k+1|k+1)$, for state $\hat{s}_i(k+1)$ at time step $k+1$.
  - If the observation, $\hat{o}_j(k+1)$, fails to match with any predicted observation, $\hat{o}_i(k+1|k)$, then an initiation process is invoked to produce the explained representation, $\hat{s}_r(k+1|k+1,)$ for new state $s_r(k+1)$ at time step $k+1$.

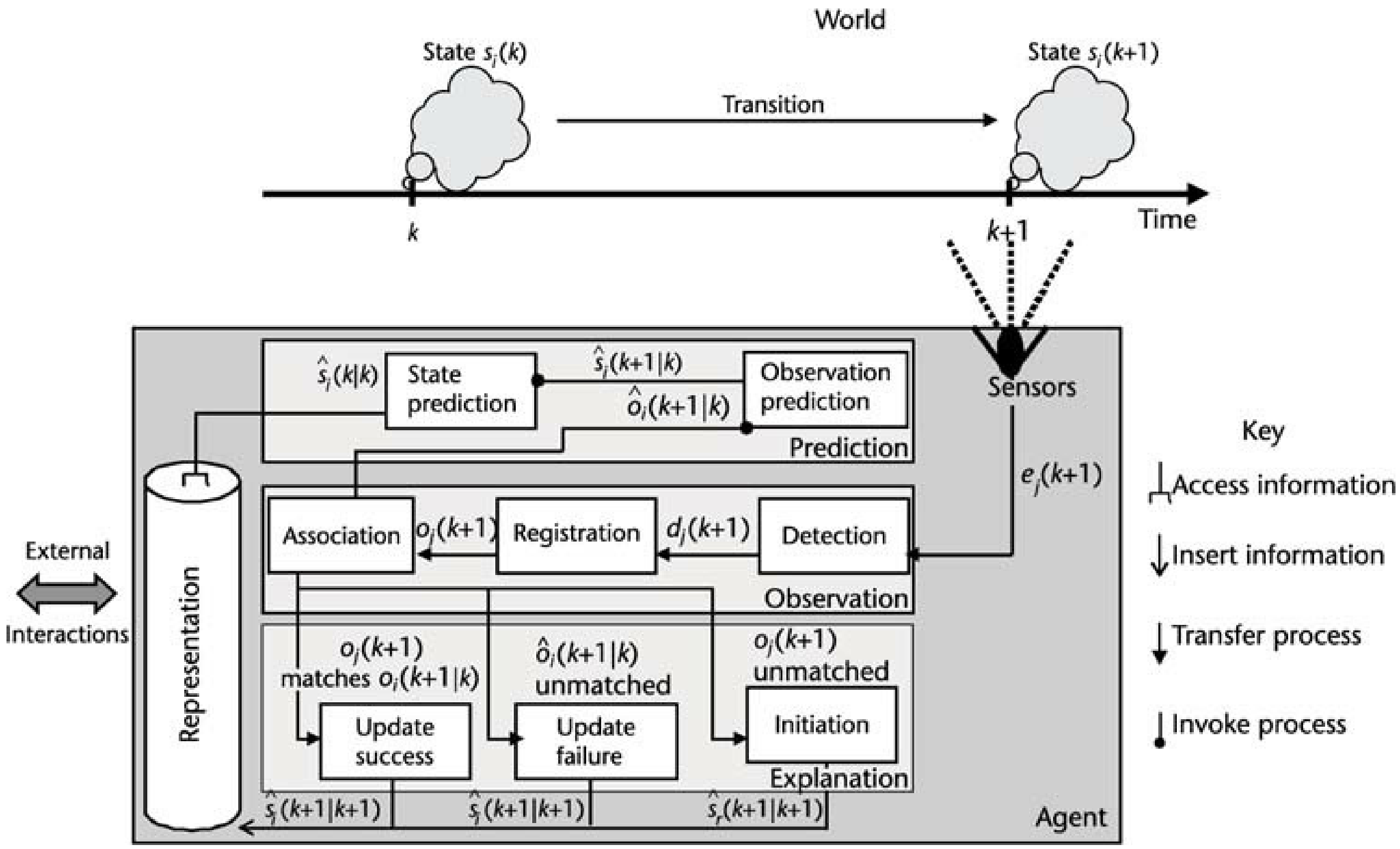


Figure 3.13 General STDF model.

Through a generic fusion process involving prediction, observation and explanation, related states in the world $\{s_j(t)|t \in \text{Time\_Step} \ \& \ t \leq k \ \& \ i \in N^+(p)\}$ come to be represented in the agent as a set of explained states $\{\hat{s}_i(t|t)|t \in \text{Time Step} \ \& \ t \leq k \ \& \ i \in N^+(p)\}$.

3.3.3 JDL Assessments

The JDL fusion model's imposed presumptions about the nature of states in the world carries over to the nature of assessments of states in the world. STDF explanations at each JDL Level 0 to 3 are an assessment at that level, and these are representations *of* states in the world at that level.

Level 0 explanations are observable assessments of observables in the world. The feature vector, $s_i(k)$ as $f_i(k)$, comes to be represented as the estimate, $\hat{f}_i(k|k)$ while observable $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ as $\underset{\rightarrow}{f_i}(\text{k}) = \{f_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ at time step k comes to be represented as a set of transitioning state estimates, $\underset{\rightarrow}{\hat{f}_i}(k) = \{\underset{\rightarrow}{\hat{f}_i}(t|t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$. $\underset{\rightarrow}{\hat{f}_i}(\text{k})$ is an observable assessment composed of observable instance assessments, $\hat{f}_i(t|t)$ for each $t \leq k$. Observable assessments are termed subobject assessments in the revised JDL model.

Level 1 explanations are object assessments of objects in the world. The state vector $s_i(k)$ as $u_i(k)$ comes to be represented as the estimate $\hat{u}_i(k|k)$, while object $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ as $\underline{u}_i(k) = \{u_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ at time step k comes to be represented as a set of transitioning state estimates $\underline{\hat{u}}_i(k) = \{\hat{u}_i(t|t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$. $\hat{u}_i(\mathrm{k})$ is an object assessment composed of object instance assessments $\hat{u}_i(t|t)$ for each $t \leq k$.

Level 2 explanations are situation assessments of situations in the world. The state of affairs $s_j(k)$ as $\Sigma_i(k)$ comes to be represented as the explained set $\hat{\Sigma}_i(k|k)$ while situation $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ as $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ as $\underline{\Sigma}_i(\mathrm{k}) = \{\Sigma_i(t)|t \in \text{Time_Step} \ \& \ t \leq k\}$comes to be represented as a set of transitioning states of affairs $\underline{\hat{\Sigma}}_i(\mathrm{k}) = \{\hat{\Sigma}_i(t|t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$. $\underline{\hat{\Sigma}}_i(k)$ is a situation assessment composed of situation instance assessments $\hat{\Sigma}_i(t|t)$ for each $t \leq k$.

Level 3 explanations provide scenario assessments of scenarios in the world. The scenario state $s_i(k)$ as $S_i(k)$ comes to be represented as the transitioning set $\hat{S}_i(k) = \{\hat{\Sigma}_i(t|k)|t \in \text{Time\_Step} \ \& \ t \leq \partial(k)\}$, while the scenario $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ as $\underline{S}_i(k) = \{S_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\} = \{\{\sum_{\mathrm{i}}(n)|n \leq \text{Time\_Step} \ \& \ n \leq \&(t)\}|t \in \text{Time\_Step} \ \& \ t \leq k\}$ comes to be represented as a set of explained states of affairs $\underline{\hat{S}}_i(k) = \{\{\hat{\Sigma}_i(n|t)|n \in \text{Time\_Step} \ \& \ n \leq \partial(t)\}|t \in \text{Time} \ \& \ t \leq k\}$. $\underline{\hat{S}}_i(k)$ is a scenario assessment composed of scenario instance assessments $\hat{S}_i(k) = \{\hat{\Sigma}_i(n|t)|n \in \text{Time} \ \& \ n \leq \partial(t)\}$ for each $t \leq k$. Scenario assessments are termed impact assessments in the revised JDL model.

The following sections explore the flavour of each JDL Level 0 to 3 from the perspective of the STDF model by very superficially outlining examples of fusion at each of those levels.

Level 0 Fusion

At Level 0, the world is a stark world composed of observables, being fragments of reality with features that can be discriminated by a sensor. The nature of observables varies with the nature of the sensor through which the observation process takes place. Broadly speaking, fusion in the Information Age must contend with signal processing and analysis; image processing and analysis; and textual processing and analysis. To celebrate the similarities and differences of STDF processing and analysis, the following two sections outline the principles behind simplified examples of level 0 signal fusion and textual fusion. For conciseness, Level 0 image processing is not discussed.

3.4.1 Level 0 Signal Fusion

Figure 3.14 illustrates the simple example of a rotating, air search ground based radar. Level 0 signal fusion for this radar involves signal processing.

At any time step k, the world of interest for Level 0 ground based radar signal fusion is composed of a number of individual observable states $\{s_i(k)|i \in N^+(p)\}$ for some number, p, where each observable state $s_i(k)$ is a kinematic feature vector sample $f_i(k) = <R_i, \theta_i, v_{Ri}>^T$, with superscript T as the matrix transpose operator to indicate that the vector is a column vector. The vector component features of $f_i(k)$ are taken to be objective measurable features of observable, i, in the world at time step k relative to the radar, features that the observable has independently of whether the radar actually senses that observable. The range (distance) from the radar sensor to the observable labelled i is R_i meters; the azimuth angle from the radar sensor to the observable labelled i is θ_i, measured clockwise from true north and reported between 0 and 2π radians; and the range rate, or rate at which the observable labelled i is moving away from the radar sensor, is v_{Ri} meters per second.

The Level 0 signal fusion goal is to identify observables through these measurable features by performing signal processing on the radar's sensations to discriminate whether a signal of interest is present within the background noise and clutter radiation. This can be conceptualized through the STDF process of Figure 3.15.

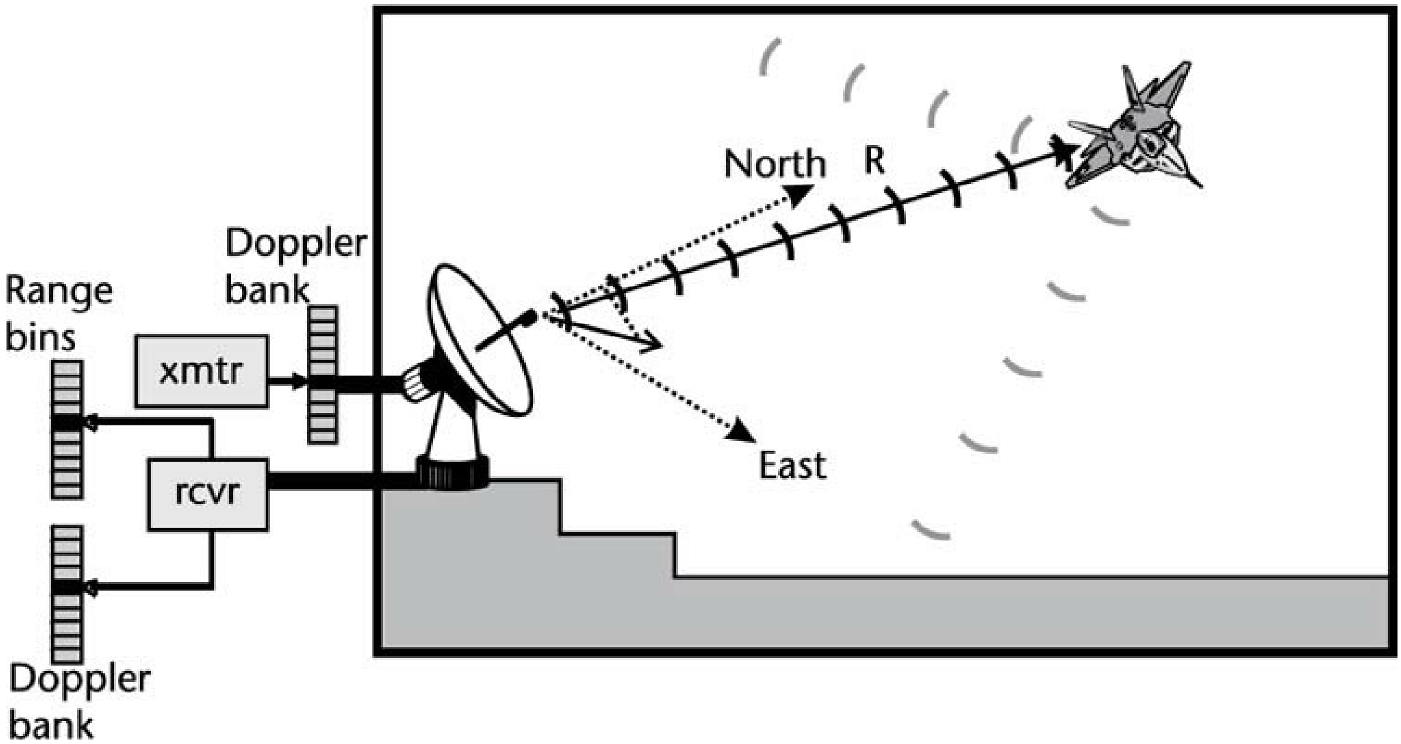


Figure 3.14 Air search ground-based Doppler radar.

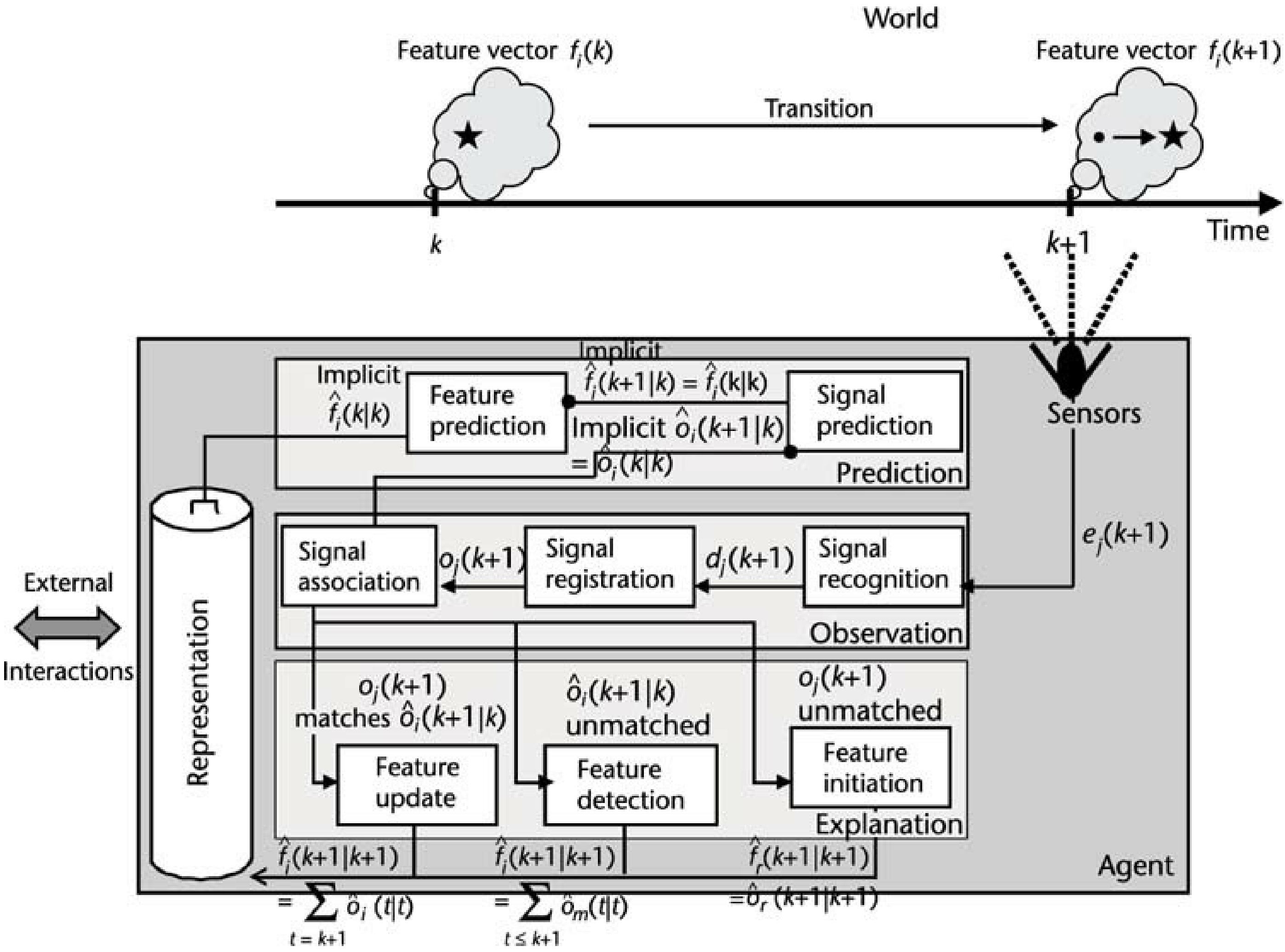


Figure 3.15 Level 0 STDF signal fusion

At time step $k+1$ the radar senses a number of new states in the world $\{f_i(k+1)\ |i \in N^+(p_1)\}$. To sense states in the world, the radar transmits radio waves and detects radio wave echoes reflected back from observables in the world, as Figure 3.14 suggests. We can assume that pulses are transmitted with the receiver off, and that the receiver is receiving pulses when the transmitter is off. Radio waves are electromagnetic energy radiated whenever an electric charge is accelerated, with part of the energy in an electric field and part in a magnetic field. When the reflected waves meet the radar's reception antenna, an alternating current voltage is obtained proportional to the electric field across its terminals. These voltages constitute the sensations $\{e_j(k+1)|j \in N^+(q)\}$ presented to the observation process.

The observation process conceptually begins with a signal recognition process to potentially identify a detection, $d_j(k+1)$, for each sensation, $e_j(k+1)$. This occurs when the radar can detect a sufficiently greater signal return from the observable than from the thermal noise generated from both background radiation and the sensor itself. So, the signal to noise ratio (SNR) associated with a fragment of reality determines whether that fragment qualifies as an observable relative to a particular radar sensor at time step k+1. The transmitted signal, S_{xmtr}, that reaches the observable is the proportion of the average transmitted signal power that is in the direction of the observable compared with being transmitted uniformly (spherically). Thus, $S_{xtmr} = P_{av}\frac{G}{4\pi R^2}$ where P_{av} is the average transmitted power from the radar; G is the gain of the transmitting antenna, indicating the degree to which the transmitted energy is concentrated in the transmission direction; and $4\pi R^2$ is the surface area of a sphere of radius range, R. As the range to the observable increases, the proportion of transmitted signal that reaches the observable decreases. The reflected signal, S_{refl}, back from the observable is characterized by $S_{refl} = \frac{\sigma}{4\pi R^2}$, where σ is the observable's radar cross section. The radar cross section indicates how much energy the observable reflects back in the direction of the radar. The effective area of the radar antenna, A_e, further constrains how much of the reflected signal is intercepted by the radar, and the signal is further reduced by other propagation losses, L. The received signal, S, from a single pulse is given by:

$$S = S_{xtmr} S_{refl} \frac{A_e}{L} = P_{av}\frac{G}{4\pi R^2}\frac{\sigma}{4\pi R^2}\frac{A_e}{L} = \frac{P_{av} G A_e \sigma}{(4\pi)^2 R^4 L}$$

The noise, N, is defined by $N = kT_0BF$, where k = 1.38x10^{-23} is Boltzmann's constant; T_0 = 290 is an assumed receiver effective input noise temperature; B is the receiver noise bandwidth; and F is the receiver noise figure, indicating any additional noise the receiver adds over the assumed T_0. Thus, the signal to noise ratio for a single pulse varies with the observable's range, R, and is given by:

$$SNR = \frac{P_{av} G A_e \sigma}{(4\pi)^2 R^4 k T_0 B F L}$$

Detection of a signal, $d_j(k+1)$, can be assigned when the signal to noise ratio of voltage, $e_j(k+1)$, exceeds a nominated threshold value. If the threshold is too low, false alarms will occur from noise mistakenly classified as a signal. If the threshold is too high, missed detections will occur from signals mistakenly classified as noise.

A signal registration process follows for each signal detection, $d_j(k+1)$, to determine a registration of the range, azimuth and range rate associated with the detection. To discriminate the range to a detected observable state from its received signal, the signals can be collected across n range bins that are cycled through with time step durations of δ seconds. The later the signal is received, the higher the range bin number into which it is deposited. The bins are termed "range bins" because the longer it takes for the signal to return, the further away the observable is, and so the range bin provides a discrete measure of range. For a ground based radar with an antenna rotating clockwise, it can discriminate the azimuth sector, θ_{xtmr}, of its beam at the time of transmission and so can estimate the azimuth of the observable if it is known that the radar rotates with a constant angular velocity of ω radians per second. In the case of a Doppler radar, the Doppler Effect can be exploited to discriminate range rate. The Doppler Effect involves a shift in frequency when a wave is reflected from a moving object. A Doppler radar includes a bank of digital filters, each of which passes a low band of frequencies. A detected signal's frequency can then be discriminated by a low band filter. Thus, a detected signal can be registered through its assignment to a range-Doppler-azimuth cell. The signal registration process registers a detected signal, $d_j(k+1)$, as a signal observation, $o_j(k+1) = <r, d, a>$, where r is the range bin number, a is the azimuth sector number and d is the Doppler filter number. Figure 3.16 illustrates.

Each time the radar beam sweeps over an observable, a stream of pulses, rather than a single pulse is usually received by the radar receiver. By applying the combined signal detection and signal registration observation process over the sequence of pulses, the incidence of false alarms can be reduced when there is a fairly stable signal and statistically variable noise. Consequently, there can be multiple time steps over which received pulses are integrated to obtain a more refined estimation of each observable's existence and measurable features. Integration can generally be of two forms: predetection integration and postdetection integration. Predetection integration involves integration in which observations, $o_j(t+1)$, over a number of time steps t are recorded in the same range-Doppler-azimuth bin. The integration time, t_i, refers to the duration of time steps for pre-detection integration. The signal to noise ratio across multiple pulses for integration time, ti, is then:

$$SNR = \frac{P_{av} G A_e \sigma t_i}{(4\pi)^2 R^4 k T_0 B F L}$$

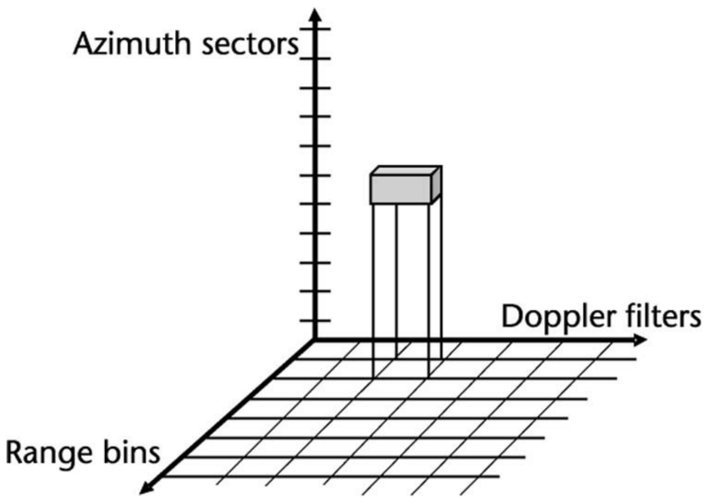


Figure 3.16 Range-Doppler-azimuth cells.

In practice, the application of detection thresholds described in the previous paragraph is typically done for each individual cell over an integration time and is set according to the mean level of noise. Post-detection integration occurs when the time the radar is on the observable involves more than one range-Doppler-azimuth cell, and hence, applies over multiple integration times. An observable is deemed to have been detected if the signal is above the detection thresholds in a predefined sufficient number of the cells in which it features.

There is a fairly direct mapping between an integrated signal in a range-Doppler-azimuth cell, <*r*, *d*, *a*>, and a feature vector state estimate, $\hat{f}_i(k|k) = < \hat{R}_i, \hat{\theta}_i, \hat{\upsilon}_{Ri} >^T$. If each range bin waits to receive a signal for δ seconds duration and the signal is stored in range bin r, then the time between transmitting and receiving that signal is $r\delta$. This is the time that the pulse has taken to reach the observable and return, and so the time to reach the observable is $\left(\frac{r\delta}{2}\right)$. If the observable is an airborne object then the speed of that pulse is $\left(\frac{c}{\sqrt{\kappa_c}}\right)$ meters per second, where κ_c is the dielectric constant for a medium, being 1.000536 for air, and where $c = 299792458$ meters per second is the speed of light in a vacuum. Consequently, as the distance traveled at that speed for that time, the range is $\hat{R}_i = \left(\frac{rc\delta}{2\sqrt{\kappa_c}}\right)$ meters. If the azimuth angle of a beam at the time of transmission to the observable is θ_{xtmr}, and the radar rotates with a constant angular velocity of ω radians per second, then the azimuth angle of the observable can be discriminated as $\hat{\theta}_i = \left(\theta_{xtmr} + \frac{\omega\delta}{2}\right)$ Discrimination of azimuth finer than the beam width of the transmission is also possible by considering the main lobe and sidelobes of the interference pattern of the wave. In the case of a Doppler radar, by noting the filter of the transmission frequency and recording the filter of the received frequency, the Doppler frequency shift

$f_d = f_{rcvr} - f_{xmtr}$ can be identified. The range rate is then determined $\hat{v}_{Ri} = \frac{dR}{dt} = -\frac{\lambda f_d}{2}$ for signal wavelength λ.

The direct mapping between a range-Doppler-azimuth cell, $<r, d, a>$, and a feature vector state estimate, $\hat{f}_i(k|k) =< \hat{R}_i, \hat{\theta}_i, \hat{v}_{Ri} >^T$ allows the remainder of the signal processing story to be conceptualized in terms of the STDF model of Figure 3.15. The prediction process is implicit. The signal prediction process can be thought of as always expecting a signal, $\hat{o}_i(k+1|k)$, in each $<r, d, a>$ cell i that recently received a signal, $\hat{o}_i(k|k)$. The feature prediction process is the feature state counterpart that expects a feature state, $\hat{f}_i(k+1|k)$ that corresponds to the feature state, $\hat{f}_i(k|k)$ just received in relation to cell i. In practice, neither $\hat{f}_i(k+1|k)$ nor $\hat{o}_i(k+1|k)$ are actually calculated. However, with this conceptualization, the signal association process can be understood as trying to match each new observation, $o_j(k+1)$, against the predicted observations, $\hat{o}_i(k+1|k)$. There are three possible explanation outcomes from this:

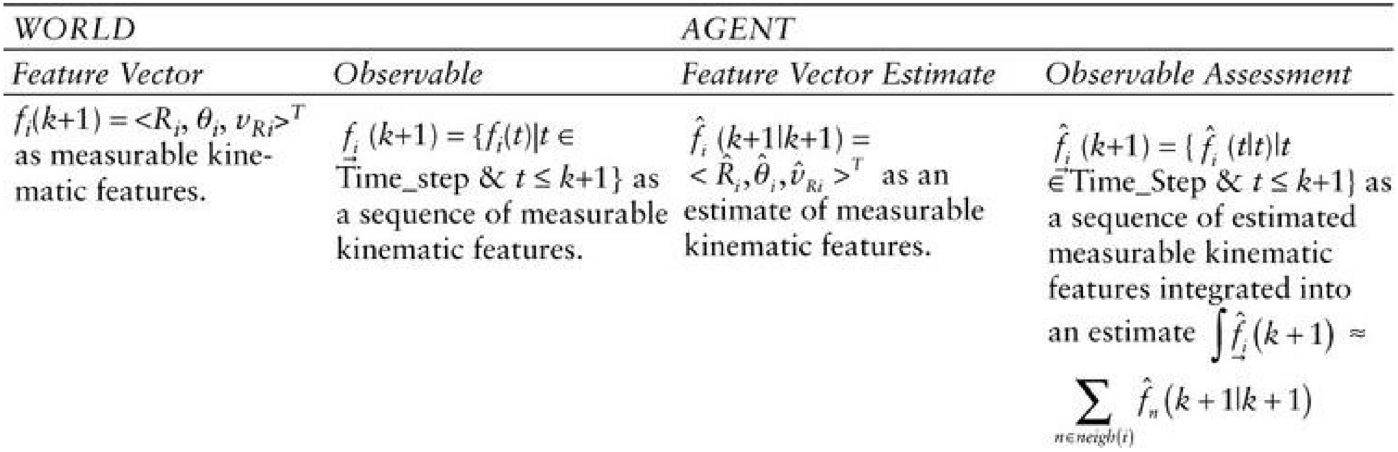
- If $o_j(k+1)$ is matched to $\hat{o}_i(k+1|k)$, then $o_j(k+1)$ has been assigned to cell i and becomes $\hat{o}_i(k+1|k+1)$. In this case, a repeated signal detection has occurred incell i, and so the feature update process integrates $o_j(k+1)$ with the other signals assigned to that cell. The feature update process thus performs predetection integration. A feature vector $\hat{f}_i(k+1|k+1) =< \hat{R}_i, \hat{\theta}_i, \hat{v}_{Ri} >^T$ representative of the integrated signals $\sum_{t \le k+1} \hat{o}_i(t|t)$ can be determined, but in practice it is not required at that time step.
- If $o_j(k+1)$ is not matched to any $\hat{o}_i(k+1|k)$, then $o_j(k+1)$ has unexpectedly been assigned to a cell r not recently assigned to, and becomes $\hat{o}_r\ (k+1|k+1)$. In this case, the feature initiation process assigns new signal detection $\hat{o}_r\ (k+1|k+1)$ ready for a new predicted $\hat{o}_r\ (k+2|k+1)$ in the next time step. A feature vector $\hat{f}_r(k+1|k+1) =< \hat{R}_r, \hat{\theta}_r, \hat{v}_{Ri} >^T$ representative of the signal $\hat{o}(k+1|k+1)$ could be determined, but in practice it is not required at that time step.
- If $\hat{o}_i(k+1|k)$ is not matched to any $o_j(k+1)$, and this occurs for a designated number of time steps, then the radar beam will be judged to have completed its sweep over the observable. At that time step, the feature detection process will consider the integrated signals $\sum_{t \le k+1} \hat{o}_i(t|t)$ and the integrated signals $\sum_{t \le k+1} \hat{o}_n(t|t)$ for certain cells n neighboring cell i. If the integrated signal in a prescribed number of these cells exceeds the signal to noise threshold for those cells, then a feature state detection is deemed to have occurred, and a feature state vector $\hat{f}_i(k+1|k+1) =< \hat{R}_i, \hat{\theta}_i, \hat{v}_{Ri} >^T$ representative of these integrated signals is calculated. The feature detection process thus performs post-detection integration.

In summary, the signal processor predicts and explains kinematic observables as patterns of change in observations of kinematic feature vectors that are detected by the radar. Table 3.1 summarizes the Level 0 signal fusion observable assessment.

3.4.2 Level 0 Textual Fusion

The proliferation of information represented as text expressed in a digital form is a revolutionary consequence of the Information Age. Level 0 textual fusion relates to the initial processing by a tokeniser of text expressed in a digital form. The observables of interest in the world at level 0 are tokens, being a sequence of characters. Section 3.4 began by noting that fusion in the Information Age must contend with signal processing and analysis; image processing and analysis; and textual processing and analysis. Digital tokens arise through all three. Speech to text processing applies signal processing to audio signals to obtain digital tokens. Optical character recognition applies image processing to images to obtain digital tokens. This section is instead about direct digital text processing to obtain digital tokens from digital text. The sensors are sensors for communication protocols, such as the Simple Mail Transfer Protocol (SMTP) and the Hypertext Transfer Protocol (HTTP); storage retrieval sensors for relational databases; or sensors associated with direct input devices, such as keyboards and keypads. In the interest of simplicity, this section focuses on direct input devices, such as keyboards and keypads.

Table 3.1 Level 0 Signal Fusion



WORLD		*AGENT*	
Feature Vector	*Observable*	*Feature Vector Estimate*	*Observable Assessment*
$f_i(k+1) = <R_i, \theta_i, \nu_{Ri}>^T$ as measurable kinematic features.	$\underset{\rightarrow}{f_i}(k+1) = \{f_i(t) \mid t \in$ Time_step & $t \leq k+1\}$ as a sequence of measurable kinematic features.	$\hat{f}_i(k+1 \mid k+1) = < \hat{R}_i, \hat{\theta}_i, \hat{\upsilon}_{Ri} >^T$ as an estimate of measurable kinematic features.	$\underset{\rightarrow}{\hat{f}_i}(k+1) = \{ \hat{f}_i(t \mid t) \mid t \in$ Time_Step & $t \leq k+1\}$ as a sequence of estimated measurable kinematic features integrated into an estimate $\int \underset{\rightarrow}{\hat{f}_i}(k+1) \approx \sum_{n \in neigh(t)} \hat{f}_n(k+1 \mid k+1)$

At any time step *k*, the world of interest for Level 0 textual fusion is composed of a number of individual observable states $\{s_i(k) \mid i \in N^+(p)\}$ for some number, p, where each observable state, $s_i(k)$, is a feature vector, $f_i(k)$, being a character from an observable token. Each observable token at time step k is understood through a sequence of characters $\underset{\rightarrow}{f_i}(k) = \{f_i(t) \mid t \in \text{Time\_Step} \ \& \ t \leq k\}$. The component characters, $f_i(t)$, are taken to be objective features of the observable token in the world at time step t relative to a language, features that the observable has independently of whether that observable is sensed.

The Level 0 text fusion goal of a tokeniser is to identify token observables through a token assessment of digital text sensations. This can be conceptualized through the STDF process in Figure 3.17.

At time step $k+1$, the sensor associated with keyboards and keypads senses a number of new states in the world $\{f_i(k+1)|i \in N^+(p_1)\}$. Typically, a processor of multiple keyboards or keypads will manage a separate digital stream for each device; so each sensor can be treated as effectively only having to deal with one character feature $f_i(k+1)$ during time step $k+1$, and a new time step is associated with the processing of each subsequent character. The sensor associated with a keyboard or keypad receives signals in response to the user pressing the keys. Suppose the user presses the key sequence [m] [i] [s] [s] [i] [l] [e] [ ], where [ ] is the space key and [m] is the key pressed at time step k. This results in a character sequence with one character per time step from time step k to time step $k+7$. The keyboard or keypad will convert this pressed key sequence into a sequence of input voltages, and this sequence of voltages constitutes the sequence of sensations $\{e_j(t)|k \leq t \leq k+7\}$ presented to the observation process.

The observation process begins with a character recognition process to identify a detection $d_j(k+1)$ from sensation $e_j(k+1)$. The voltages associated with each key pressed form a bit vector within the sensor, with a 1 representing voltage at or above the nominated threshold and 0 representing a voltage below the nominated threshold. Each keyboard or keypad character corresponds to a particular bit vector. For a sensor representing 7 bit American Standard Code for Information Interchange (ASCII) codes, the key sequence [m] [i] [s] [s] [i] [l] [e] [ ] produces the following bit sequences:

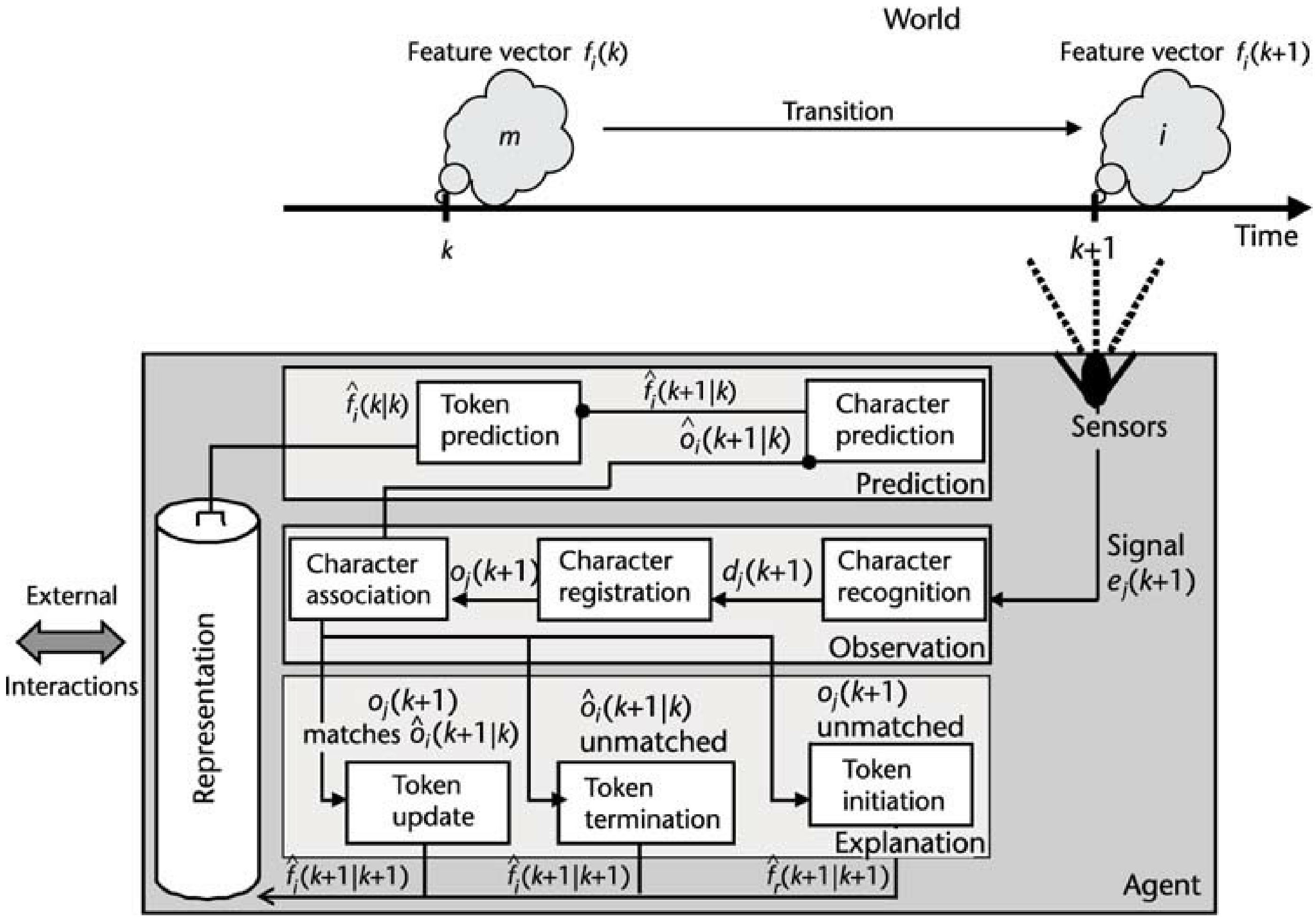


Figure 3.17 Level 0 STDF textual fusion.

$$f_i(k) = 1101101; f_i(k+1) = 1101001; f_i(k+2) = 1110011; f_i(k+3) = 1110011;$$
$$f_i(k+4) = 1101001; f_i(k+5) = 1101100; f_i(k+6) = 1100101; f_i(k+7) = 01000000$$

At time step k+1, the feature vector is the bit vector 1101001, which the character recognition process identifies as the character with ASCII code 105, being a lowercase ‘i’ character. This detection $d_j(k+1)$ of an ‘i’ character is then forwarded to a character registration process. Character registration performs a normalizing operation on characters to ensure that the resulting character sequence is conformable to the token representations used in the system’s lexicon, which stores the allowable tokens of the language of interest. Character registration can include a cumulative refinement based on previous characters, and may involve simplifications and alterations to punctuation and capitalization. The observation process completes with a token association process that attempts to match the output, $o_j(k+1)$, of character registration with the predicted next character of a token in the lexicon.

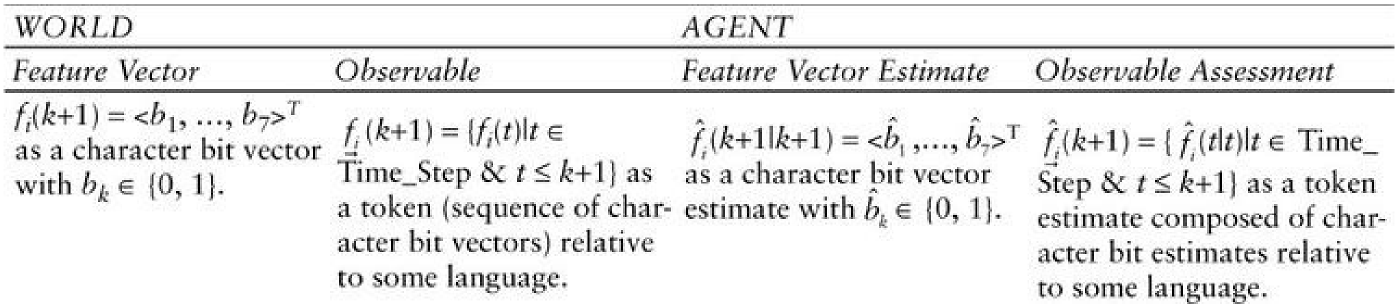
In practice, token observation can occur in one of two ways. With some devices, there is no advantage to predicting subsequent characters while sensing characters. The character association process instead accumulates the character sequence, in this case, the ordered set $\{o_j(t)|k \leq t \leq k+7\}$ = <1101101, 1101001, 1110011, 1110011, 1101001, 1101100, 1100101, 0100000> = <‘m’, ‘i’, ‘s’, ‘s’, ‘i’, ‘l’, ‘e’, ‘ ’>, and collectively attempts to pattern match the accumulated sensed character sequence $\{o_j(t)\ |k \leq t \leq k+6\}$ = <‘m’, ‘i’, ‘s’, ‘s’, ‘i’, ‘l’, ‘e’> with a known token character sequence $\{\hat{o}_i(t+1|t)|k \leq t \leq k+6\}$ = <‘m’, ‘i’, ‘s’, ‘s’, ‘i’, ‘l’, ‘e’> for the *i*th token in the lexicon. However, for devices like mobile telephones, in which several characters are overlayed onto the one key, there is an efficiency gain in having the sensor prompt the user with expected next characters, so that it can be selected with fewer keystrokes. In these cases, the prediction process is utilized more interactively. The prediction process engages a token prediction process to monitor which tokens remain candidate completions of the character sequence to that point and then engages a character prediction process to suggest the next character $\hat{o}_i\ (t+1|t)$ at time step t of the candidate token of the i^{th} token in the lexicon. For example, given the ordered character set $\{o_j(t)|k \leq t \leq k+5\} = \{o_j(k)\} \cup \{\hat{o}_i(t+1|t)|k \leq t \leq k+4\}$ = <‘m’, ‘i’, ‘s’, ‘s’, ‘i’, ‘l’>, the prediction process might nominate $\hat{o}_i(k+6|k+5)$ = ‘e’ as the predicted next character at time step k+5, which the user can then either accept or override.

There are three in principle explanation process outcomes from token association at any time step $k+1$.

- If sensed character $o_j(k+1)$ is matched to predicted character $\hat{o}_i(k+1|k)$, then the token update process notes that the sensed character sequence still conforms to the token of the ith token in the lexicon. If $o_j(k+1)$ is either the space character (ASCII code 32), the end of line character (ASCII code 10), or a punctuation delimiter, then the token has completed and the token estimate $\hat{f}_i(k) = \hat{f}_i(t|t)|t \in \text{Time\_Step}\ \&\ t \leq k\}$ is the explained character sequence.
- If sensed character $o_j(k+1)$ arises when there are no predicted characters $\hat{o}_i(k+1|k)$, either because it is the start of the sensing session or because a token character sequence has just completed, then the token initiation process considers each *i*th token in the lexicon that begins with character $o_j(k+1)$. The lexicon would need discrimination trees [21], or a similar mechanism, to support rapid access to its stored token content.
- If the next predicted character $\hat{o}_i\ (k+1|k)$ of the *i*th token in the lexicon does not match sensed character $o_j(k+1)$, then the token termination process no longer considers the *i*th token in the lexicon as a possible token for the sensed character sequence.

In summary, the lexical analyser predicts and explains token observables as patterns of change in observations of character feature vectors that are detected by the digital text sensor. Table 3.2 summarizes the Level 0 textual fusion observable assessment.

Table 3.2 Level 0 Textual Fusion



WORLD		*AGENT*	
Feature Vector	*Observable*	*Feature Vector Estimate*	*Observable Assessment*
$f_i(k+1) = <b_1, \ldots, b_7>^T$ as a character bit vector with $b_k \in \{0, 1\}$.	$\vec{f_i}(k+1) = \{f_i(t) \mid t \in$ Time_Step & $t \le k+1\}$ as a token (sequence of character bit vectors) relative to some language.	$\hat{f}_i(k+1 \mid k+1) = <\hat{b}_1, \ldots, \hat{b}_7>^T$ as a character bit vector estimate with $\hat{b}_k \in \{0, 1\}$.	$\vec{\hat{f}}_i(k+1) = \{\hat{f}_i(t \mid t) \mid t \in$ Time_Step & $t \le k+1\}$ as a token estimate composed of character bit estimates relative to some language.

Level 1 Fusion

At Level 1, the world is composed of objects, which are persistent fragments of reality identified through their observable properties. The representation of objects differs across signal processing, image processing, and textual processing because the properties of interest differ. In Sections 3.5.1 and 3.5.2, the Level 1 signal and textual counterparts to the Level 0 signal and textual descriptions of Sections 3.4.1 and 3.4.2 are outlined respectively.

3.5.1 Level 1 Signal Fusion

The world of interest to a rotating, air search ground based radar tracker performing Level 1 signal fusion is a collection of airborne objects with kinematic properties. At any time step k, the world of interest is composed of a number of individual airborne object instances $\{s_i(k)|i \in N^+(p)\}$ for some number p, where each airborne object instance $s_i(k)$ is a kinematic state vector $u_i(k) = <x, y, v_x, u_y>^T$. Several Level 0 echo return time steps contribute to a single Level 1 state vector time step, and several Level 1 state vector time steps are required to track an airborne object. Here, x and y in the kinematic state vector, $u_i(k)$, identify the Cartesian coordinate position in meters of the object instance at time step k relative to the sensing radar, while v_x and v_y identify the Cartesian coordinate velocity components in meters per second of the object relative to the radar. These are taken to be the objective qualities of that object instance in the world, qualities that the object has independently of whether the radar senses the object; the object has those x and y distances from the radar and has those v_x and v_y velocities relative to the radar.

The challenge for the tracker system is to track objects through these measurable properties. This can be conceptualized through the STDF process of Figure 3.18.

At time step k+1, the radar receives a number of new sensations $\{e_j(k+1)|j\ N^+(p_1)\}$, which are presented to an observation process. The observation process begins with the signal processing process; this is the entire process outlined in Section 3.4.1. This produces a set of detections $\{d_j(k+1)|j \in N^+(p_2)\}$, where each detection $d_j(k+1) =< \hat{R}_j, \hat{\theta}_j, \hat{v}_{Rj} >^T$ is a feature vector estimate $\hat{f}_j(k+1|k+1) =< \hat{R}_j, \hat{\theta}_j, \hat{v}_{Rj} >^T$. A coordinate registration process then normalizes each detection $d_j(k+1)$ relative to a spatiotemporal frame of reference to yield a measurement vector $o_j(k+1)$. In practice, several different coordinate systems may be engaged, but for descriptive purposes of the simple radar example, the normalization process can be taken to produce an observation vector $o_j(k+1)=<x, y, v_x, v_y>^T$ consisting of a state vector <x, y, vx, vy>^{T}, where $x = R_j\cos(\theta_j)$,

$y = R_j\sin(\theta_j)$, v_x, $= v_{Rj}\cos(\theta_j)$ and $v_y = v_{Rj}\sin(\theta_j)$ are obtained from $\hat{f}_i\,(k+1|k+1) =< \hat{R}_i, \hat{\theta}_i, \hat{v}_{R_i} >^T$. The observation process then completes with a data association process that attempts to match each observation vector $o_j(k+1)$ with each predicted observation $\hat{o}_i\,(k+1|k)$ of the object labeled i, to determine which of the known objects (if any) that observation vector $o_j(k+1)$ is providing updated information on. Each predicted observation $\hat{o}_i\,(k+1|k)$ is determined from a predicted state vector update $\hat{u}_i(k+1|k)$ of state vector estimate $\hat{u}_i(k|k)$ of state vector $u_i(k)$ in the world at time step k.

In practice, there is always some uncertainty about whether $\hat{u}_i\,(k|k)$ genuinely reports $u_i(k)$; it is customary to accompany the state vector with a measure of the uncertainty. Uncertainty for the state vector estimate $< \hat{x}, \hat{y}, \hat{v}_x, \hat{v}_y >^T$ is expressed by a 4x4 covariance matrix $P_i(k|k)$ under the assumption that the error estimate conforms to a multivariate Gaussian distribution. $\hat{u}_i(k|k)$ is the (highest probability) expected value $\mu = <x_\mu, y_\mu, v_{x\mu}, v_{y\mu}>^T$ of the distribution, but the probability $p(<x, y, v_x, v_y>^T)$ of any other column vector $x = <x, y, v_x, v_y>^T$ can also be obtained from the Gaussian probability density function

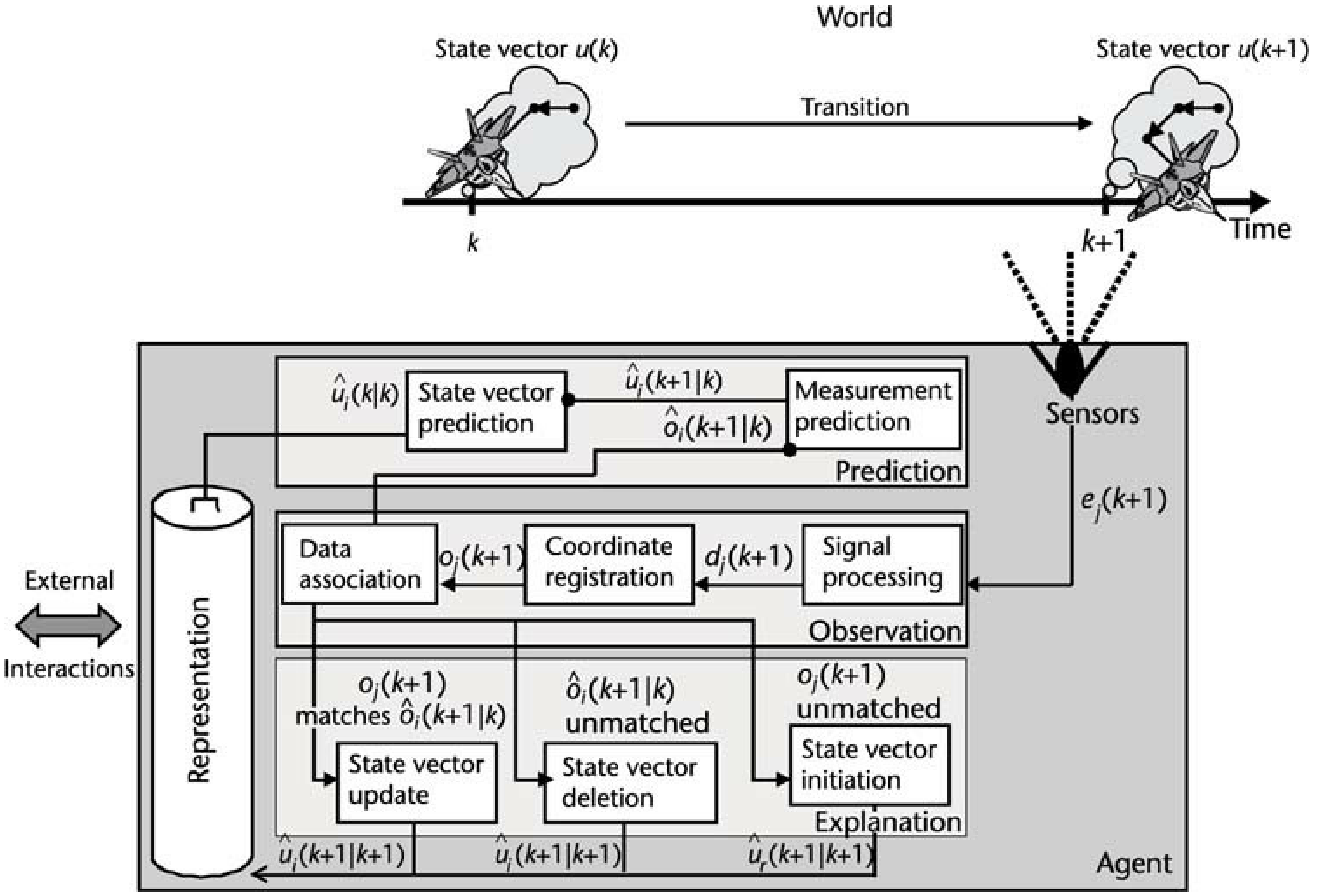
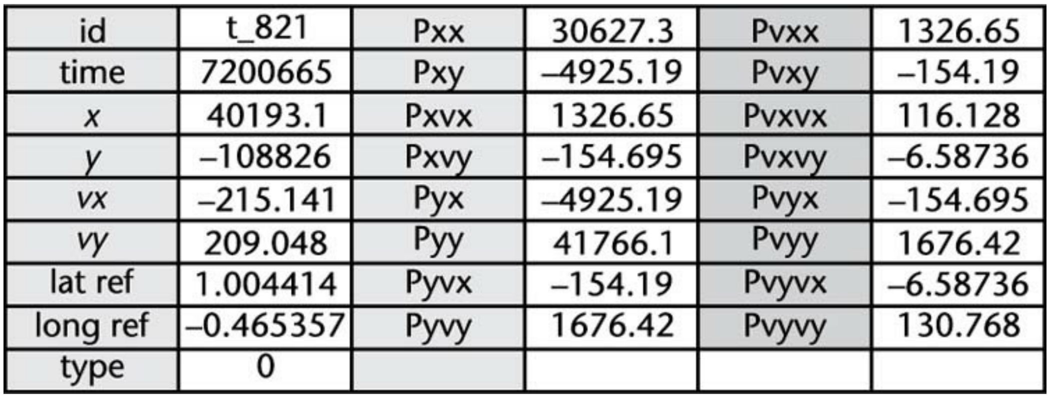


Figure 3.18 Level 1 STDF signal fusion.

$$f(x) = \frac{1}{(2\pi)^{(\#x/2)}|P|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T P^{-1}(x-\mu)\right)$$

The output of an object assessment of the object labelled i at time step k is a state estimate $\hat{u}_i(k|k)$ together with uncertainty information in the form of a co-variance matrix $P_i(k|k)$. In practice this information is usually accessible as a data structure. Figure 3.19 offers an illustrative data structure. t_821 is the generated track identifier number and 7200625 is a relative time stamp. These are indicative of i and k respectively in $\hat{u}_i(k|k)$. The state estimate values in $< \hat{x}, \hat{y}, \hat{v}_x, \hat{v}_y >^T$ are provided by the "x", "y", "vx" and "vy" entries. The "lat ref" and "long ref" values report the latitude and longitude values of the radar at that time, the "type" value of 0 indicates that the object was airborne, and the remaining "P*mn*" entries specify the values of the 4x4 covariance matrix $P_i(k|k)$.

The Kalman filter [19] has served as the basic work horse for updating state estimates. It assumes that the features of an object can be modelled by a linear Markovian relationship so that the next state vector of interest of an object follows from the previous state vector of interest of that object in accordance with a known behavioral model, together with a known input disturbance and some estimate of the process error in the model. A typical Kalman filter object state equation of the environment is thus given by



id	t_821	Pxx	30627.3	Pvxx	1326.65
time	7200665	Pxy	–4925.19	Pvxy	–154.19
x	40193.1	Pxvx	1326.65	Pvxvx	116.128
y	–108826	Pxvy	–154.695	Pvxvy	–6.58736
vx	–215.141	Pyx	–4925.19	Pvyx	–154.695
vy	209.048	Pyy	41766.1	Pvyy	1676.42
lat ref	1.004414	Pyvx	–154.19	Pvyvx	–6.58736
long ref	–0.465357	Pyvy	1676.42	Pvyvy	130.768
type	0				

Figure 3.19 Illustrative state vector estimate data structure.

$$u_i(k+1) = F(k+1)u_i(k) + \omega(k+1)$$

for the object indexed by i, where:

$u_i(k)$ and $u_i(k+1)$ are the unknown n-dimensional state vectors of object i in the world at time steps k and $k+1$, respectively;

$F(k+1)$ is a given object change model of the object at time step $k+1$ expressed as an

n×n state transition matrix;
$\omega(k+1)$ is the process error expressed as an n-dimensional zero mean, white, Gaussian distributed vector with known covariance $Q(k+1)$, the term “white” meaning that knowing the value of the error at time $k+1$ provides no knowledge about the value of the noise at any later time;
$q(k+1)$ is the known deterministic n-dimensional input change between steps k and $k+1$.

It also assumes that observation of the object’s features is in accordance with a known measurement matrix modelling the physics of the measurement process in addition to some estimate of the measurement error. The observation of object i at time step $k+1$ is modelled by the observation equation

$$o_i(k+1) = H(k+1)u_i(k+1) + v(k+1)$$

where:

$H(k+1)$ is a 1×n measurement matrix;
$v(k+1)$ is the n-dimensional zero mean, white, Gaussian distributed noise vector with known n×n measurement noise covariance matrix $R(k+1)$.

The state vector prediction process of Figure 3.18 can use the linear Markovian model to generate a predicted state estimate $\hat{u}_i(k+1|k)$ for each object i at time step $k+1$ from the state vector estimate $\hat{u}_i(k|k)$ for object i at time step k by

$$\hat{u}_i(k+1|k) = F(k+1)\hat{u}_i(k|k) + q(k+1)$$

while the predicted covariance estimate for object i at time step $k+1$ is given by

$$P_i(k+1|k) = F(k+1)P_i(k+1|k)F(k+1)^T + Q(k+1)$$

The measurement prediction process of Figure 3.18 then uses the state vector prediction $\hat{u}_i(k+1|k)$ to predict the observation update $\hat{o}(k+1k)$ for object i by

$$\hat{o}_i(k+1|k) = H(k+1)\hat{u}_i(k+1|k) + v(k+1)$$

Given the set of new observations $\{o_j(k+1)|j \in N^+(p_2)\}$ at time step $k+1$ and the set of predicted observations $\{\hat{o}_i(k+1|k)|i \in N^+(p_3)\}$ for known objects at time step $k+1$, the data association process evaluates how well each observation $o_j(k+1)$ matches with each predicted observation $\hat{o}_i(k+1|k)$, and on that basis, performs an assignment of observations to predicted observations. There are various ways of assessing the degree of match between $o_j(k+1)$ and $\hat{o}_i(k+1|k)$. Multihypothesis data association approaches allow multiple observations to be considered as candidate matches to the same predicted observation, but they are not discussed here. The simplest data association technique is to use maximum likelihood estimation.

The explanation process must deal with three possible outcomes from the data association process:

- If $o_j(k+1)$ is matched to $\hat{o}_i(k+1|k)$ then the Kalman state vector update process is invoked for this pair. The measurement residual $\Delta_i(k+1) = o_j(k+1) - (H(k+1)\ u_j(k+1))$ captures the disparity between $o_j(k+1)$ and $\hat{o}_i(k+1|k)$, while $S_i(k+1) = \mathrm{cov}(\Delta_i(k+1)) = H(k+1)\ P_i(k+1|k)\ H(k+1)^T + R(k+1)$ identifies the residual covariance. The optimal gain is given by $K(k+1) = P_i(k+1|k)\ H(k+1)^T\ S(k+1)^{-1}$. The updated estimate of state vector u_i(k+1) for object i at time step k+1 is then given by $\hat{u}_i(k+1|k+1) = \hat{o}_i(k+1|k) + (K(k+1)\ \Delta_i(k+1))$, and the updated covariance is $P(k+1|k+1) = (I - (K(k+1)\ H(k+1))\ P(k+1|k)$. The optimal gain is optimal by ensuring the expected values $\mathrm{E}(\underline{\Delta}_i(\mathrm{k}+1)) = \mathrm{E}(\underline{u}_i(\mathrm{k}+1) - \hat{u}_i(k+1|k+1)) = E(u_i(k+1) - \hat{u}_i(k+1|k)) = 0$.
- If o_j(k+1) is not matched to any $\hat{o}_i$(k+1|k) then $o_j(k+1)$ can be taken to represent the observation of a new object. In that event the state vector initiation process is invoked. It introduces a new object index, r, and assigns $\hat{u}_r\ (k+1|k+1) = o_j(k+1)$ for the simple radar example and sets the covariance $P_r(k+1|k+1) = \alpha I$ for large scalar α and identity matrix I, to bias toward the data in subsequent measurements.
- If $\hat{o}_i(k+1|k)$ is not matched to any $o_j(k+1)$, and this occurs for a designated number of time steps or through a more sophisticated deletion logic, then the object, i, will be deemed to no longer be observable and attempts to match against it will no longer be made. This is monitored by the state vector deletion process.

In summary, the tracker predicts and explains kinematic objects as patterns of change in observations of kinematic state vectors with measurable properties that are derived from the signal processor. Table 3.3 summarizes the Level 1 signal fusion object assessment.

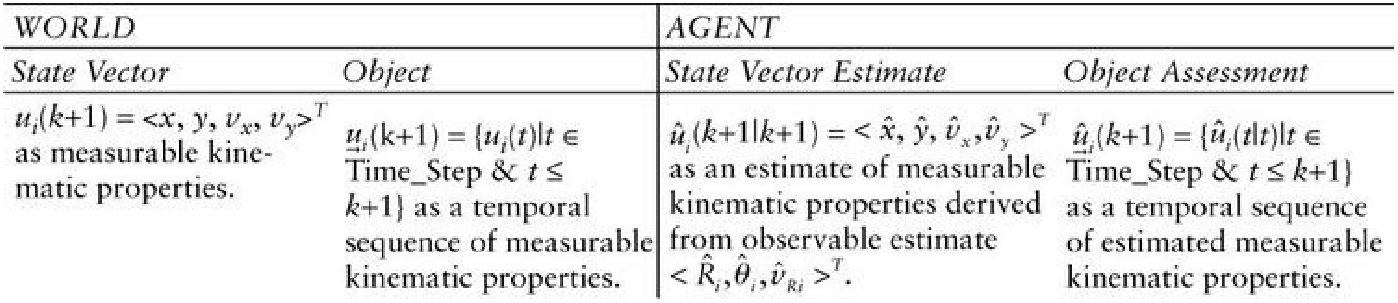
3.5.2 Level 1 Textual Fusion

For a rotating, air search ground based radar tracker, the objects of interest are observable aircraft tracks in the world and the object instances are the temporal sequences of observable state vector instances of those objects. For a Level 1 textual fusion parser, the objects of interest are sentences in the world as sequences of object instances as words associated with observable tokens. Several Level 0 character recognition time steps contribute to a single Level 1 word recognition time step, and several Level 1 word recognition time steps are required to identify a sentence object. At any time step k, the world of interest to a parser is composed of a number of individual object instances $\{s_i(k)|i \in N^+(p)\}$ for some number, p, where each object instance $s_i(k)$ is a word state vector $u_i(k) = <t_i(k), a_i(k)>^T$, where $t_i(k)$ is the token associated with the word, and $a_i(k)$ is a list of lexical attributes associated with the word. Each sentence object at time step k is understood as a set of transitioning word state vectors $\underset{\rightarrow}{u}_i(k) = \{u_i(t)|t \in \text{Time\_Step} \ \& \ t \le k\}$ arranged as a parse tree.

Each word state vector $u_i(k) = <t_i(k), a_i(k)>^T$ is taken to be an objective quality of a sentence in the world at time step t relative to a language, qualities that the sentence has independently of whether it is sensed. For example, in English, the token ‘missile’ is objectively associated with a word that has a defined list of attributes; the relevant attribute list will vary with the nature of the language lexicon that records word state vectors. For example, Norvig’s lexicon style [21] would associate the word state vector <‘missile’, [noun, (n,n,p,n), [], missile)]> with the word missile. ‘missile’ is the token for the word missile. The lexical attribute list [noun, (n,n,p,n), [], missile)] identifies lexical features of the word missile: noun identifies the part of speech as noun; (n,n,p,n) identifies person-number agreement for subject-predicate matching as third person singular; [] identifies no complement modifiers; and missile is the token returned for semantic definition.

The goal of Level 1 textual fusion is to parse sentence objects by tracking their processed component word state vectors over time. This can be conceptualized through the STDF process of Figure 3.20.

Table 3.3 Level 1 Signal Fusion



WORLD		*AGENT*	
State Vector	*Object*	*State Vector Estimate*	*Object Assessment*
$u_i(k+1) = <x, y, v_x, v_y>^T$ as measurable kinematic properties.	$\underset{\rightarrow}{u}_i(\text{k}+1) = \{u_i(t)\|t \in \text{Time\_Step} \ \& \ t \le k+1\}$ as a temporal sequence of measurable kinematic properties.	$\hat{u}_i(k+1\|k+1) = <\hat{x}, \hat{y}, \hat{v}_x, \hat{v}_y>^T$ as an estimate of measurable kinematic properties derived from observable estimate $<\hat{R}_i, \hat{\theta}_i, \hat{v}_{Ri}>^T$.	$\underset{\rightarrow}{\hat{u}}_i(k+1) = \{\hat{u}_i(t\|t)\|t \in \text{Time\_Step} \ \& \ t \le k+1\}$ as a temporal sequence of estimated measurable kinematic properties.

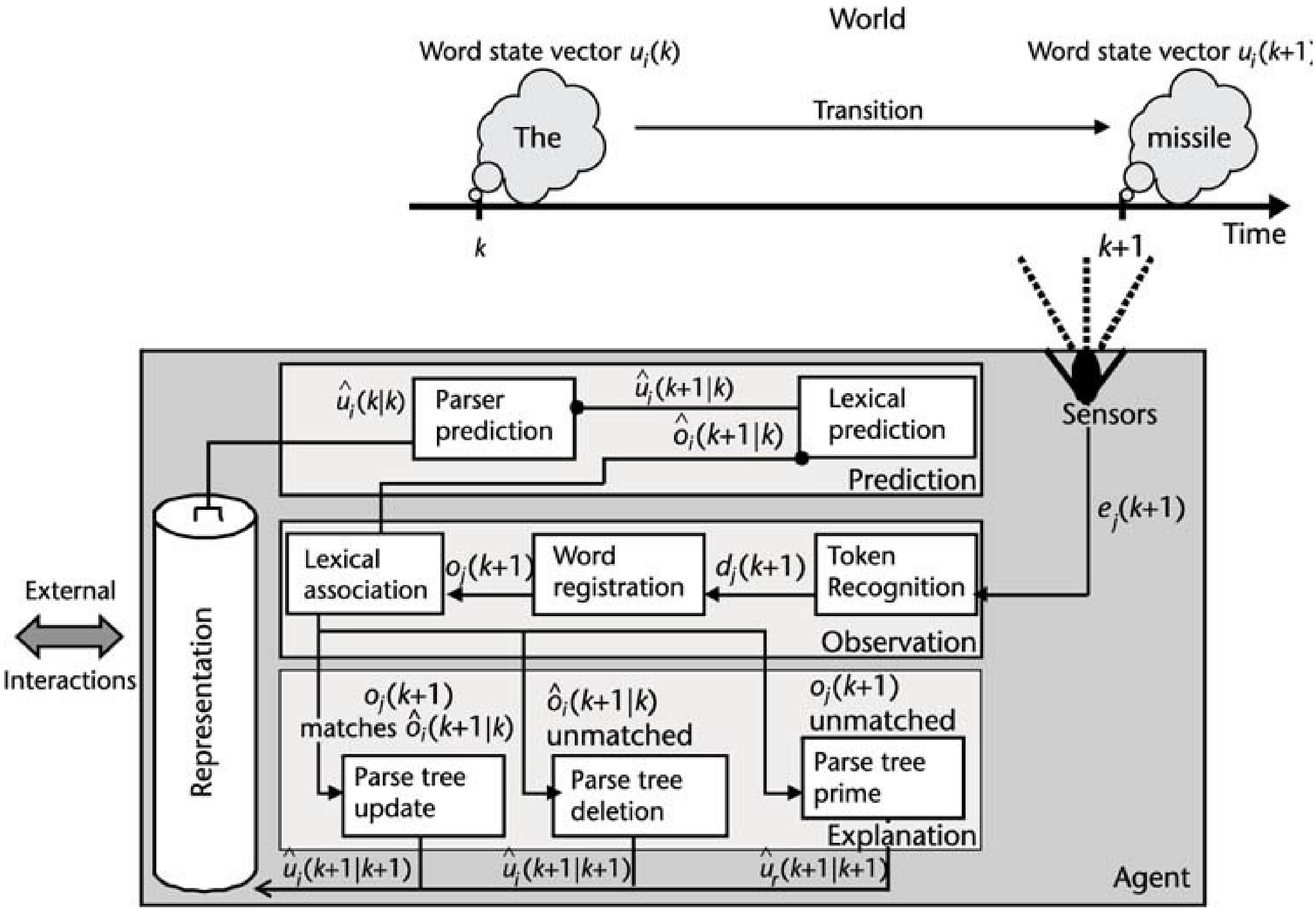


Figure 3.20 Level 1 STDF textual fusion.

At time step k+1, the parser receives the next in a sequence of sensations $\{e_j(t)|t \leq k+1\}$ which are presented to an observation process. The observation process begins with the token recognition process, being the entire process outlined in Section 3.4.2. The token recognition process produces a token detection $d_j(k+1) = \hat{f}_j(k+1)$. A lexical analyser performs word registration on token $d_j(k+1)$ to produce a word state vector observation $o_j(k+1) = <\hat{f}_j(k+1), a_j>^T$ by associating a list of lexical attributes a_j with the token $d(k+1)$ based on the lexicon. The observation process completes when the lexical association process attempts to match the word state vector observation $o_j(k+1)$ with a predicted observation $\hat{o}_i(k+1|k)$ generated by the parser.

Suppose the user presses the key sequence that results in the observation process producing the following sequence of tokens: ‘The’ ‘missile’ ‘hit’ ‘the’ ‘frigate’ and suppose that at time step k+1, the observed word is:

$$o_j(k+1) = \text{<w\_2205, 'missile', [noun, (n,n,p,n), [], missile)]>}$$

The prediction process includes a parser prediction process that implements a grammatical search process for a nominated grammar. The parser accepts any sentence (sequence of words) that conforms to the grammar expressed in terms of noun phrases (NP), verb phrases (VP), verbs (V), nouns (N), determiners (Det), and so on. The sentence ‘The missile hit the frigate’ conforms to a conventional English grammar, with a simple token parse tree presented in Figure 3.21. As the parser searches, it undertakes a lexical prediction process that nominates candidate word types that conform to the grammar for the next time step. Consequently at each time step k+1, there is a collection of candidate words $\{\hat{o}_i(k+1|k)|i \in N^+(p_1)\}$ that can match with the next word $o_j(k+1)$ in accordance with the grammar.

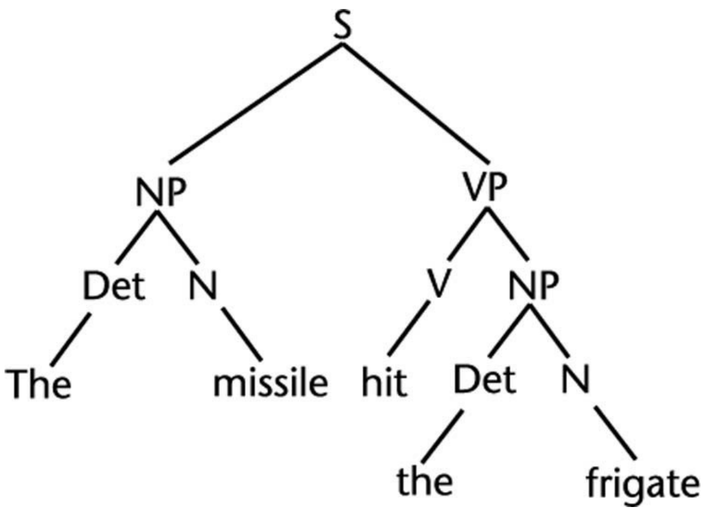


Figure 3.21 Token parse tree.

The parser can require strict adherence to the grammar or it can be designed to accommodate a degree of uncertainty in its lexical association pattern matching. For example, a user might mistakenly type the token ‘misslw’ instead of ‘missile’. The prior probability that ‘misslw’ is meant to be ‘missile’, denoted p(‘misslw’≈‘missile’) can be defined by $p(\text{‘misslw’} \approx \text{‘missile’}) = 1 - \frac{\text{sed}(\text{'misslw'}, \text{'missile'})}{\max(\{\text{len}(\text{'misslw'}), \text{len}(\text{'missile'})\})}$ where sed(‘misslw’, ‘missile’) is the string edit distance between ‘misslw’ and ‘missile’ and len(x) is the length of string x, and so in this case, $p(\text{‘misslw’} \approx \text{‘missile’}) = \frac{5}{7}$ Grammatical uncertainty can also be accommodated. To illustrate, English singular noun phrases require a determiner to accompany the noun, and so the token sequence: ‘The’ ‘missile’ ‘hit’ ‘frigate,’ is not grammatically a sentence. However, strict grammatical compliance can be relaxed by treating the parse tree of Figure 3.21 as a Bayesian tree. For token w and grammatical category G, the probability that w belongs to G, denoted by $p(w \in G)$, is defined by the best word match $p(w \in G) = \max(\{p(w, v)|v \in G\}$ with a word v in category G. Conditional probabilities are then required to weight the relative importance of grammatical subtrees. For example, by nominating values for the conditional probabilities, $p(w_1w_2 \in \mathrm{NP}|w_1 \in \mathrm{Det}\ \&\ w_2 \in N)$, $p(w_1w_2 \in \mathrm{NP}|w_1 \in \mathrm{Det}\ \&\ w_2 \notin N)$, $p(w_1w_2 \in \mathrm{NP}|w_1 \notin \mathrm{Det}\ \&\ w_2\ N)$ and $p(w_1w_2 \in \mathrm{NP}|w_1 \notin \mathrm{Det}\ \&\ w_2 \notin N)$, the relative importance of determiner and noun compliance in noun phrases can be factored into the probability that the observed token sequence is meant to be the best word match sentence, with $p(w_1w_2 \in \mathrm{NP}) =$

$$\begin{aligned}
&p(w_1w_2 \in NP|w_1 \in Det\ \&\ w_2 \in N)\cdot p(w_1 \in Det)\cdot p(w_2 \in N)\\
&+p(w_1w_2 \in NP|w_1 \in Det\ \&\ w_2 \notin N)\cdot p(w_1 \in Det)\cdot(1-p(w_2 \in N))\\
&+p(w_1w_2 \in NP|w_1 \notin Det\ \&\ w_2 \in N)\cdot(1-p(w_1 \in Det))\cdot p(w_2 \in N)\\
&+p(w_1w_2 \in NP|w_1 \notin Det\ \&\ w_2 \notin N)\cdot(1-p(w_1 \in Det))\cdot(1-p(w_2 \in N))
\end{aligned}$$

This approach allows a token sequence like ‘The’ ‘misslw’ ‘hit’ ‘frrigate’ to be matched to a sentence like ‘The missile hit the frigate’ with some probability, the acceptability of which can then be determined by a probability threshold.

The explanation process must deal with three possible outcomes from the lexical association process:

- If word $o_j(k+1)$ = <w_2205, ’missile’, [noun, (n,n,p,n), [], missile)]> is matched to candidate word $\hat{o}_i(k+1|k)$, both in terms of token compliance and lexical attribute compliance, then the parse tree update process updates the parse tree with the assignment of $o_j(k+1)$ as $\hat{u}_i(k+1|k+1)$.
- If word $o_j(k+1)$ = <w_2205, ’missile’, [noun, (n,n,p,n), [], missile)]< is not matched to the current parse tree branch $\hat{o}_i(k+1|k)$, then the parse tree prune process prunes the parse tree by backtracking to attempt to match through a different parsing branch candidate $\hat{o}_r(k+1|k)$. If this succeeds, the fusion process continues the assignment of $o_j(k+1)$ as $\hat{o}_r(k+1|k+1)$.
- If word $o_j(k+1)$ = >w_2205, ’missile’, [noun, (n,n,p,n), [], missile)]< does not match $\hat{o}_i(k+1|k)$ for any i, then $o_j(k+1)$ cannot appear as a word in that sentence at that point and be conformable with the grammar, so the parse fails and the parse tree is deleted.

In summary, the parser predicts and explains sentence objects as patterns of change in observations of word state vectors that are generated by the lexical analysis of tokeniser output. Table 3.4 summarizes the Level 1 textual fusion object assessment.

Level 2 Fusion

At level 2, the world is a world composed of situations, being sets of sets of relations between objects in the world. Level 2 is a significant departure from Level 1 fusion for three reasons.

First, situations represent a revolutionary shift in conceptualisation from the former Aristotelian view of the world as objects with features or properties, as evident at Levels 0 and 1, to the more recent Wittgensteinian view of the world as relations between objects, with the accompanying gains in expressivity. This major shift in Western philosophy represented a change in human understanding from understanding grounded in metaphysics to understanding grounded in language. Second, it is this shift in understanding that allows for a common linguistic interpretation of Level 1 signal processing; image processing; and textual processing outputs, and thus allows Level 2 to provide the basis for multi-source fusion. Figure 3.22 illustrates a multisource fusion framework of interacting STDF models at each level. In general, the detection process within the observation process (see Figure 3.13) at level n+1 *is* the STDF process at Level n, while the state prediction process within the prediction process (see Figure 3.13) at Level n is at the very least influenced by the explanations derived at Level n+1. Third, the common linguistic interpretation allows Level 2 fusion to be understood as providing the meaning of Level 1 fusion. Level 2 assessments address some of the following questions: what is the meaning of the kinematic behavior of a nominated object? What is the meaning of a nominated video sequence? What is the meaning of a nominated sentence?

Figure 3.23 exposes the STDF process at Level 2. At any time step k, the world of interest is composed of a number of individual situation instances $\{s_i(k)|i \in N^+(p)\}$ for some number, p, where each situation instance $s_i(k)$ is a state of affairs $\sum_i(k)$ represented as a set of sentences about the world expressed in a formal language. $\sum_i(k)$ = {missile(x56), targeting(x56,x7), frigate(x7)} is illustrative of a state of affairs. A situation $\{s_i(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$ at time step k is understood as a set of transitioning situation instances and so each situation $\underset{\rightarrow}{\Sigma}(k)$ at time step k is understood as a set of states of affairs $\underset{\rightarrow}{\Sigma}(k) = \{\Sigma(t)|t \in \text{Time\_Step} \ \& \ t \leq k\}$. Thus, situations are sets of sets of statements about the world and are primarily symbolically based, with changes in the statements over time taken to reflect situations as transitions in situation instances over time. States of affairs are taken to be objective statements about the world—the referenced objects have the referenced relations independently of an observer observing them.

Table 3.4 Level 1 Textual Fusion

WORLD		*AGENT*	
State Vector	*Object*	*State Vector Estimate*	*Object Assessment*
$u_i(k+1) = <t_i, a_i>^T$ as a word with token t_i and lexical attributes a_i.	$\underrightarrow{u}_i(\mathrm{k}+1) = \{u_i(t) \mid t \in$ Time_Step & $t \leq k+1\}$ as a sentence composed of a sequence of words relative to some language.	$\hat{u}_i(k+1 \mid k+1) = < \hat{t}_i, a_i>^T$ as a word estimate with token estimate $\hat{t}_i$ and lexical attributes a_i derived from token estimate.	$\underrightarrow{\hat{u}}_i(k+1) = \{\hat{u}_i(t \mid t) \mid t \in$ Time_Step & $t \leq k+1\}$ as a sentence estimate composed as a sequence of word estimates arranged in a parse tree.

Figure 3.22 Multisource fusion.

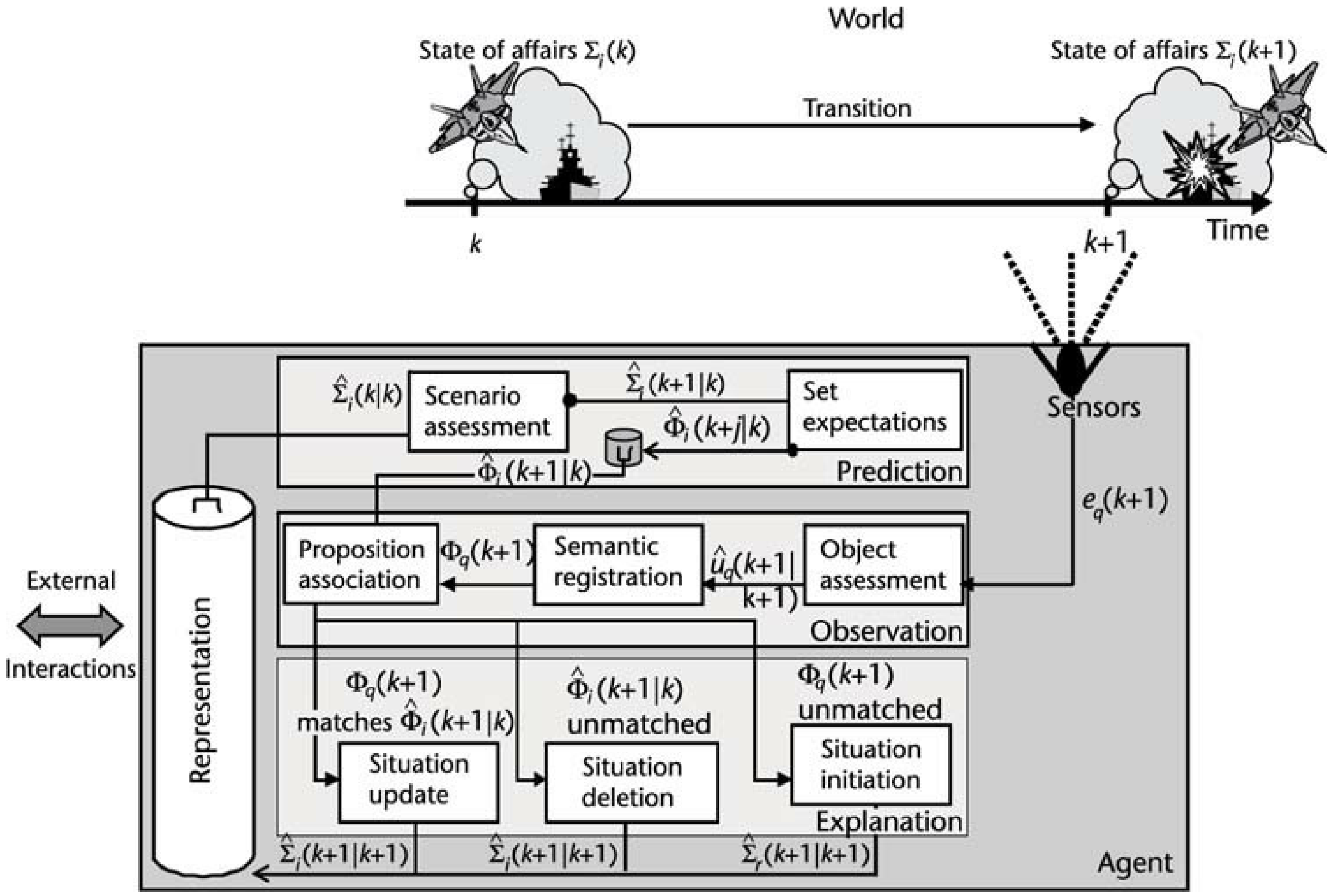
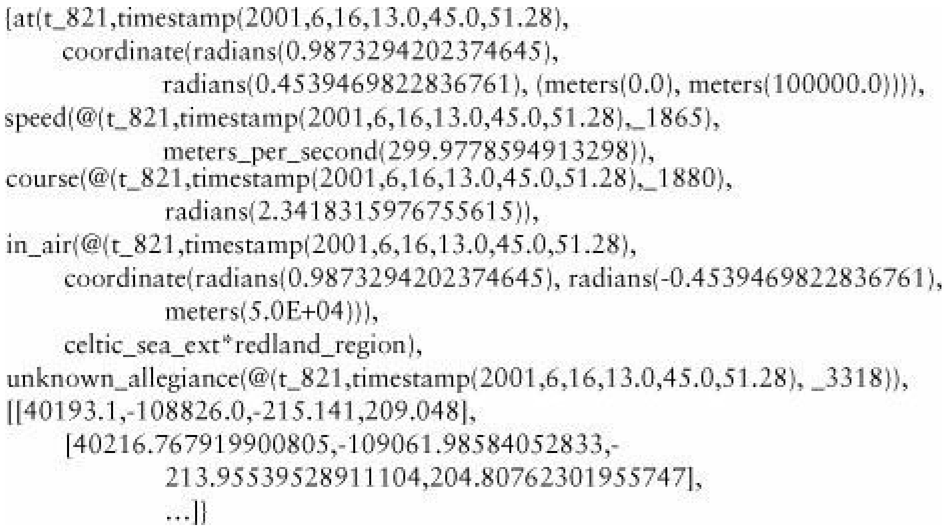


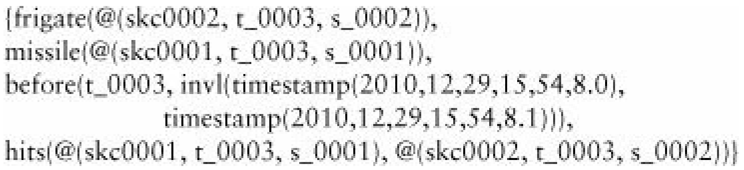
Figure 3.23 Level 2 STDF fusion.

The challenge for the situation assessment system is to identify situations in the world based on object assessments from Level 1 fusion, be they signal, image, or textually based. At time step k+1, the situation assessment system receives a number of new sensations $\{e_q(k+1)|q \in N^+(p_1)\}$ which are presented to an observation process. The observation process begins with a signal, image, or textual object assessment process, being the entire respective process of Section 3.5. This produces a set of state vector detections $\{d_q(k+1)|q \in N^+(p_2)\} = \{\hat{u}_i\ (k+1|k+1)|i \in N^+(p_2)\}$, akin to those outlined in Sections 3.5.1 and 3.5.2. A semantic registration process then normalizes each state vector detection $d_q(k+1)$ relative to a semantic frame of reference, to yield a propositional observation $\Phi_q(k+1)$. This author's Mephisto semantic framework is designed for semantic registration. When expressed in Mephisto in relation to the North Atlantis scenario [23], the kinematic state vector information of Figure 3.19 in Section 3.5.1 becomes $\Phi_q(k+1) =$



```
{at(t_821,timestamp(2001,6,16,13.0,45.0,51.28),
     coordinate(radians(0.9873294202374645),
              radians(0.4539469822836761), (meters(0.0), meters(100000.0)))),
speed(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),_1865),
              meters_per_second(299.9778594913298)),
course(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),_1880),
              radians(2.3418315976755615)),
in_air(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),
     coordinate(radians(0.9873294202374645), radians(-0.4539469822836761),
              meters(5.0E+04))),
     celtic_sea_ext*redland_region),
unknown_allegiance(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28), _3318)),
[[40193.1,-108826.0,-215.141,209.048],
     [40216.767919900805,-109061.98584052833,-
              213.95539528911104,204.80762301955747],
              ...]}
```

This records the position, speed, course, environmental region, allegiance, and a sample distribution from the state vector information for object t_821. When expressed in Mephisto at 15:54:8.1 on December 29, 2010, the sentence object assessment obtained from the token sequence ‘The’ ‘missile’ ‘hit’ ‘the’ ‘frigate’ of section 3.5.2 becomes $\Phi_q(k+1)$ =



```
{frigate(@(skc0002, t_0003, s_0002)),
missile(@(skc0001, t_0003, s_0001)),
before(t_0003, invl(timestamp(2010,12,29,15,54,8.0),
              timestamp(2010,12,29,15,54,8.1))),
hits(@(skc0001, t_0003, s_0001), @(skc0002, t_0003, s_0002))}
```

This records that missile skc0001 at time t_0004 at location s_0001 hit frigate skc0003 at time t_0004 at location s_0003, where t_0004 is before 15:54:8.1 on December 29, 2010. Semantically registering signal, image and textual information through the same canonical formal language allows inferences to be made seamlessly regardless of the different sources and source types of that information. This provides the basis for multi-source fusion. The propositional association process is tasked with matching semantically registered propositional observations $\Phi_q(k+1)$ from whatever source, with expected observations $\hat{\Phi}_i(k+1|k)$ generated by the pre-diction process.

The prediction process includes a scenario assessment process and a set expectations process. For each explained state of affairs $\hat{\Sigma}_i(k|k)$ from time step k, the scenario assessment process nominates a (or more) predicted state of affairs $\hat{\Sigma}_i(k+1|k)$ for time step k. The set expectations process then nominates an expected observation(s) $\hat{\Phi}_i(k+1|k)$ for that predicted state of affairs. Section 3.5.1 introduced the object assessment state equation $u_i(k+1) = F(k+1)\, u_i(k) + \omega(k+1) + q(k+1)$, which assumes a linear model in which uncertainty is presumed to accord with a multivariate Gaussian distribution determined from expected value (mean) and covariance parameters. The reconceptualization from states as sets of measurable state vectors for object assessments to states as sets of symbolically based states of affairs for situation assessments, means that the state equation for object assessments is reconceptualized to state inference

$$\Sigma_i(k+1) \subseteq \left\{ \sigma \middle| \left(\mathbb{F}(k+1) \cup \Sigma_i(k) \cup \Pi(k+1) \cup \Delta(k+1) \right) \models_M \sigma \right\}$$

for situation assessments, where $\Sigma_i(k)$ is the unknown state of affairs at time step k, expressed as a formal theory; $\mathbb{F}(k+1)$ is a given situation change model at time step k+1 expressed as a formal theory; $\Pi(k+1)$ is uncertainty expressed as a probabilistic assignment of truth to basic formulae; $\Delta(k+1)$ is the known input change between time steps k and k+1, expressed as a formal theory; $\Sigma_i(k+1)$ is the unknown state of affairs at time step k+1, expressed as a formal theory; M is a formal theory of meaning; for any formal theory X, $X \models_M \sigma$ if and only if $(X \cup M) \models \sigma$; and $A \models \alpha$ means formal sentence α can be inferred from formal theory A, for inference relation $\models$. Here, as in [24], the term "formal theory" means a set of formal language sentences and does not assume a logically closed set of formal language sentences. The observation inference counterpart to the observation equation $o_i(k+1) = H(k+1)u_i(k+1) + v(k+1)$ is $\Phi_i(k+1) \subseteq \{\sigma | (\mathbb{H}(k+1) \cup \Sigma_i(k) \cup \Lambda(k+1)) \models_M \sigma\}$ where $\mathbb{H}(k+1)$ is a given expectation model at time step k+1 expressed as a formal theory, and $\Lambda(k+1)$ is uncertainty expressed as a probabilistic assignment of truth to basic formulae. Cognitive routines from the author's ATTITUDE and ATTITUDE TOO cognitive models are introduced to implement this functionality [17].

In general, the formal theory of uncertainty $\Pi(k+1)$ cannot be assumed to be a multivariate Gaussian distribution, and so situation assessment prediction requires an alternative to the traditional expected state vector with covariance matrix approach often used for object assessments. States of affairs represent the world propositionally, but they also usually represent the world partially. The two states of affairs

$\Sigma_1(k)$ = {frigate(@(skc0002,t_0003,s_0002)), missile(@(skc0001,t_0003,s_0001)),
hits(@(skc0001,t_0003,s_0001),@(skc0002,t_0003,s_0002))} and
$\Sigma_2(k)$ = {missile(@(skc0001,t_0003,s_0001)),
hits(@(skc0001,t_0003,s_0001),@(skc0002,t_0003,s_0002))}

could both be used to correctly report the same piece of the world, but $\Sigma_2(k)$ is a more partial description than $\Sigma_1(k)$. Possible worlds are complete states of affairs relative to a formal language of discourse. To illustrate, if P denotes the proposition in_port(skc0002); T denotes the proposition targeting(skc0001, skc0003) and S denotes the proposition stationary(skc0003), then there are $2^3 = 8$ (complete) possible worlds using this small formal language, as Figure 3.24 demonstrates.

Possible worlds describe the different possible mutually exclusive ways the world can be relative to the language of discourse and we can understand states of affairs in terms of the possible worlds that they partially describe. Figure 3.25 itemizes a few partial states of affairs based upon the set of possible worlds from Figure 3.24.

The key point about sets of possible worlds from the standpoint of uncertainty is that a probability distribution can be assigned to any set of possible worlds, and therefore the probability of a state of affairs can be readily determined as the sum of the probabilities of the possible worlds that the state of affairs partially defines [16]. Figure 3.26 captures the sentiment.

The language of discourse for a situation about merchant ships and the language of discourse about a situation about missile strikes will usually vary considerably. To reduce the cardinality of possible worlds to consider, it is prudent to develop different possible world models for different situation types. To formally superimpose a probability distribution on the situation labelled by i, let $<\mathcal{W}^i, E^i, p_{\mathcal{W}}>$ be a probability space in which: $\mathcal{W}^i$ is the sample space of all possible worlds for the situation labelled by i; $\mathcal{E}^i = P(W^i)$ is the set of possible events for powerset $P(x) = \{u \mid u \subseteq x\}$; and $p_{\mathcal{W}} : \mathcal{E}^i \rightarrow [0, 1]$ is a probability measure. For each time step k, let random variable $W^i(k): \mathcal{W}^i \rightarrow \{0, 1\}$ identify the truth of possible worlds at time step k. Situations can be understood in terms of partial worlds. Each partial world $\omega^i(k)$ at pending time step k then corresponds to an event in E^i and $P_w(\omega^i(k)) = \sum_{W^i(k) \in \omega^i(k)} p_w(W^i(k))$ A state of affairs $\Sigma_i(k)$ at time step k includes the meaningful consequences of a partial world, i.e. $\Sigma_i(k) \subseteq \{\sigma \mid \omega^i(k) \models_M \sigma\}$ relative to semantic inference relation $\models_M$. $\Phi_q(k+1)$ is the k^{th} propositional observation at time step $k+1$. $\Phi^i(k)$, on the other hand, denotes the observation of the situation labelled by i at time step k, which comes to be understood through the explained observation $\hat{\Phi}_i(k|k)$ of the situation labelled by i. At time step k, the conditional distribution $p(W^i_m(k|k) = p(W^i_m(k) \mid \{\Phi^i(1), \ldots, \Phi^i(k)\})$ over all possible worlds for the situation labelled by i given observations of that situation up to time step k, is stored and so can be used to compute $p(\omega^i(k))$ for each partial world $\omega^i(k)$, and therefore $p(\Sigma_i(k))$ for any state of affairs $\Sigma_i(k)$.

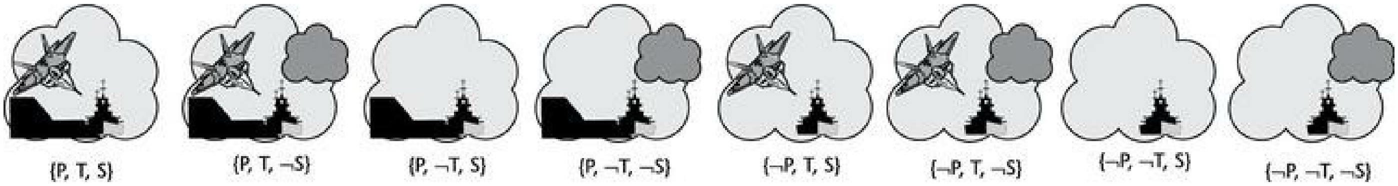


Figure 3.24 Possible Worlds.

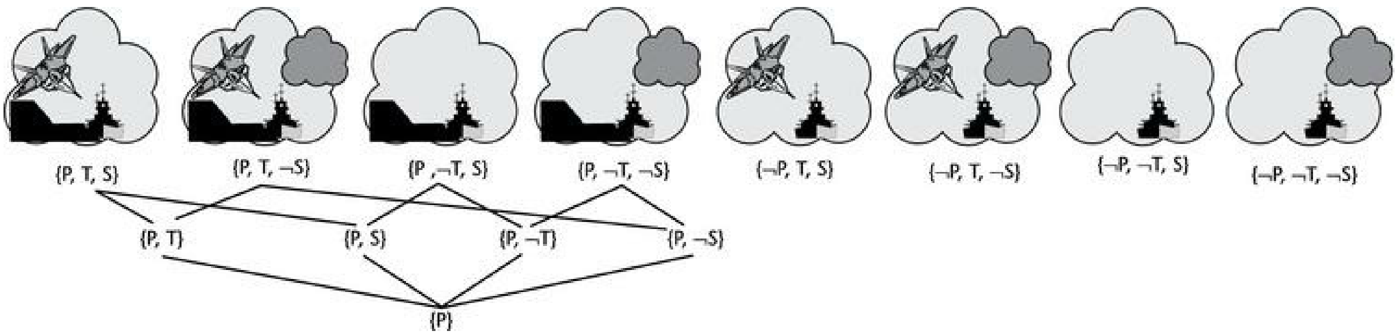


Figure 3.25 Partial states of affairs.

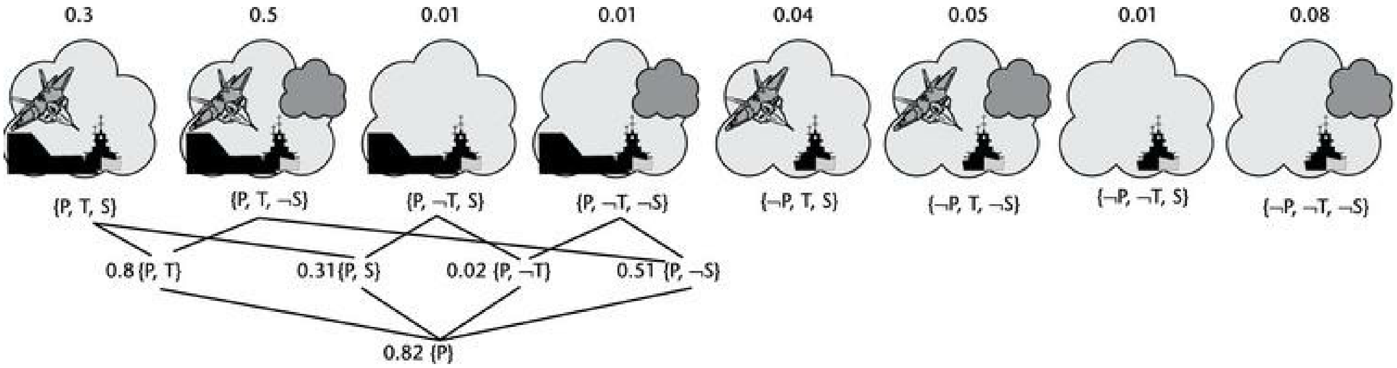


Figure 3.26 States of affairs probabilities.

As the conditional probability of each state of affairs of interest can be derived from the conditional distribution across possible worlds, it is sufficient to concentrate on the latter for predicting state of affairs probabilities. The predicted state of affairs $\Sigma_i(k+1|k)$ at time step k for the situation labelled i can be determined from the maximum posterior probability $p(W^i_m(k)|\{\Phi^i(1), \ldots, \Phi^i(k)\})$. With a conditional independence assumption and $p(W^i_m(k|k))$ denoting $p\left(W^i_m(k)\middle|\{\Phi^i(1),\ldots,\Phi^i(k)\}\right)$,

$$
\begin{aligned}
p\left(W^i_j(k+1|k)\right) &= p\left(W^i_j(k+1)\middle|\{\Phi^i(1),\ldots,\Phi^i(k)\}\right) \\
&= \sum_{W^i_m(k)\in\mathbb{W}^i} \left(p\left(W^i_j(k+1)\middle|\{W^i_m(k),\Phi^i(1),\ldots,\Phi^i(k)\}\right)\cdot p\left(W^i_m(k|k)\right)\right) \\
&= \sum_{W^i_m(k)\in\mathbb{W}^i} \left(p\left(W^i_j(k+1)\middle|W^i_m(k)\right)\cdot p\left(W^i_m(k|k)\right)\right)
\end{aligned}
$$

for each j. $p(W^i_j(k+1)|W^i_m(k))$ is obtained by probabilistic inference from the predictive model $\mathbb{F}(k+1)$ for the situation labelled by i in accordance with state inference, while the value $p(W^i_m(k\,|\,k))$ is obtained inductively from past explanations. The expected observation $\hat{\Phi}_i(k+1|k)$ is then inferred from $\hat{\Sigma}_i(k+1|k)$ combined with the observation theory $\mathbb{H}(k+1)$ and the current probabilistic assignments.

The explanation process must deal with three possible outcomes from the propositional association process:

- If propositional observation Φq(k+1) is matched with expected observation $\hat{\Phi}_j(k+1|k)$ by the propositional association process, then the situation update process uses this match to formulate a state of affairs update $\hat{\Sigma}_j(k+1|k+1)$ for time step k+1, and generally continues to employ the predictive theory $\mathbb{F}(k+1)$ responsible for that update. A Bayesian approach to updating uncertainty gives

$$p\left(W_j^i\left(k+1\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k+1)\right\}\right)\right)=\frac{p\left(\left\{W_j^i(k+1),\Phi^i(k+1)\right\}\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)}{p\left(\Phi^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)}$$

  But

$$\begin{aligned}&p\left(W_j^i(k+1),\Phi^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\\&=p\left(\Phi^i(k+1)\middle|\left\{W_j^i(k+1),\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\cdot p\left(W_j^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\\&=p\left(\Phi^i(k+1)\middle|W_j^i(k+1)\right)\cdot p\left(W_j^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\end{aligned}$$

  under an assumption of conditional independence, while

$$\begin{aligned}&p\left(\Phi^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\\&=\sum_{W_m^i(k)\in\mathcal{W}}p\left(\Phi^i(k+1)\middle|\left\{W_m^i(k),\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\cdot p\left(W_m^i(k)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\\&=\sum_{W_m^i(k)\in\mathcal{W}}\left(p\left(\Phi^i(k+1)\middle|W_m^i(k)\right)\cdot p\left(W_m^i(k)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\right)\end{aligned}$$

  under an assumption of conditional independence. Thus,

$$\begin{aligned}&p\left(W_j^i(k+1|k+1)\right)=p\left(W_j^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k+1)\right\}\right)\\&=\frac{p\left(\Phi^i(k+1)\middle|W_j^i(k+1)\right)\cdot p\left(W_j^i(k+1)\middle|\Phi^i(1),\ldots,\Phi^i(k)\right)}{\sum_{W_m^i(k+1)\in\mathcal{W}}\left(p\left(\Phi^i(k+1)\middle|W_m^i(k+1)\right)\cdot p\left(W_m^i(k+1)\middle|\left\{\Phi^i(1),\ldots,\Phi^i(k)\right\}\right)\right)}\end{aligned}$$

  Each $p(W_m^i(k+1)|\{\Phi^i(1),\ldots,\Phi^i(k)\})$ term is obtained from the prediction step while $p(\Phi^i(k+1)|W_m^i(k+1))$ is obtained from the expected observation step.
- If propositional observation Φ_q(k+1) is not matched with any expected observation $\hat{\Phi}_j(k+1|k)$ by the propositional association process, but is a propositional pattern anticipated by some prediction model $\mathbb{F}(k+1)$ in the system, then Φq(k+1) can be taken to represent the observation of a new state of affairs. In that event the situation initiation process is invoked. It introduces a new situation index, *r*, assigns $\hat{\Sigma}'_j(k+1|k+1)$ on the basis of Φ_q(k+1), and initiates execution of the predictive model *F*(k+1) that anticipated Φ_q(k+1) so that an expected state of affairs

$(\hat{\Sigma}_i^r(k+2|k+1)$ will be predicted by that model for the next time step k+2. A Bayesian approach gives

$$p\left(W_i^r(k+1|k+1)\right) = p\left(W_i^r(k+1)\middle|\left\{\Phi^r(k+1)\right\}\right)$$
$$= \frac{p\left(\Phi^r(k+1)\middle|W_i^r(k+1)\right)\cdot p\left(W_i^r(k+1)\right)}{p\left(\Phi^r(k+1)\right)}$$

with

$$p\left(\Phi^r(k+1)\right) = \sum_{W_m^r(k+1)\in\mathcal{W}^r}\left(p\left(\Phi^r(k+1)\middle|W_m^r(k+1)\right)\cdot p\left(W_m^r(k+1)\right)\right)$$

Each $p(\Phi^r(k+1|\,W_m^r(k+1))$ term is computed by integrating across the probability density function associated with $\Phi^r(k+1)$ using the sample distribution identified during semantic registration. Each $p(\,W_m^r(k+1))$ term relates to the a priori probability of possible world $W_m^r(k+1)$.

- If $\hat{\Phi}_i(k+1|k)$ is not matched with any expected observation $\Phi_q(k+1)$ by the propositional association process, and this occurs for a designated number of time steps or through a more sophisticated deletion logic, then the situation labeled by i will be deemed to no longer be observable and attempts to match states of affairs against the situation labeled by i will no longer be made. This is monitored by the situation deletion process.

In summary, the situation assessment system predicts and explains situations as patterns of change in observations of states of affairs that are generated by the signal, image and textual object assessment processes. Table 3.5 summarizes the Level 2 STDF fusion situation assessment.

Level 3 Fusion

At Level 3, the world is a world composed of scenarios, being the consequences of relations between objects in the world. Level 3 is an extension of Level 2 in which the prediction process extends forward beyond the next time step and the explanation process extends backward beyond the previous time step. At any time step k, the world of interest is composed of a number of individual scenario instances $\{s_i(k) \mid i \in N^+(p)\}$ for some number p, where each scenario instance $s_i(k)$ is a scenario state $S_i(k) = \underrightarrow{\Sigma}_i(\partial(k))$, being a situation into the future for $k < \partial(k)$. Thus $S_i(k) = (\{\Sigma_i(n) \mid n \in \text{Time\_Step} \ \& \ n \leq k\} \cup \{\Sigma_i(n) \mid n \in \text{Time\_Step} \ \& \ k < n \leq \partial(k)\})$ is composed of past (and present) situation $\underrightarrow{\Sigma}_i(k) = \{\Sigma_i(n) \mid n \in \text{Time\_Step} \ \& \ n \leq k\}$ and partial future $\{\Sigma_i(n) \mid n \in \text{Time\_Step} \ \& \ k < n \leq \partial(k)\}$, for monotonic look ahead time $\partial(\mathrm{k}) > \mathrm{k}$. Each scenario $\underrightarrow{S}_i(k)$ at time step k is understood as a set of monotonic transitioning scenario states $\underrightarrow{S}_i(k) = \{S_i(t) \mid t \in \text{Time\_Step} \ \& \ t \leq k\} = \{\{\Sigma_i(n) \mid n \in \text{Time\_Step} \ \& \ n \leq \partial(t)\} \mid t \in \text{Time\_Step} \ \& \ t \leq k\}$ such that $\underrightarrow{S}_i(t) \subseteq \underrightarrow{S}_i(t+1)$ for all $t \leq k$. Thus, scenarios are situations that include the forecast of future states of affairs. Scenarios are taken to be objective statements about the world in that each situation in the world has some scenario extension that is a situation in the (possibly future) world, independently of an observer observing it. For example, if I eventually spend my 60th birthday in Australia, then me spending my 60th birthday in Australia *is* a factual situation in the world, even though I am not yet aware of it, and even though the reader may never observe it.

Table 3.5 Level 2 STDF Fusion

World		*Agent*	
State of Affairs	*Situation*	*State of Affairs Estimate*	*Situation Assessment*
$\Sigma_i(k+1) \subseteq L$ as a set of formal language sentences from formal language L.	$\underrightarrow{\Sigma}_i(k+1) = \{\Sigma_i(t) \mid t \in \text{Time\_Step} \ \& \ t \leq k+1\}$ as a temporal sequence of states of affairs.	$\hat{\Sigma}_i(k+1 \mid k+1) \subseteq L$ as an estimate set of formal language sentences from formal language L.	$\underrightarrow{\hat{\Sigma}}_i(k+1) = \{\hat{\Sigma}_i(t \mid t) \mid t \in \text{Time\_Step} \ \& \ t \leq k+1\}$ as a situation estimate composed of a sequence of states of affairs estimates.

The challenge for the impact assessment system is to identify scenarios in the world based on situation assessments from Level 2 fusion. In essence the challenge is to take assessed situations, forecast the situations that they will become, and potentially revise the explanations of the situations that they have been. Figure 3.27 outlines the process. At time step $k+1$, the situation assessment system receives a number of new sensations $\{e_q(k+1) \mid q \in N^+(p_1)\}$ which are presented to an observation process. The observation process begins with a signal, image or textual object assessment process from Section 3.5. This produces a set of state vector detections $\{d_q(k+1)|q \in N^+(p_2)\} = \{\hat{u}_q(k+1|k+1)|q \in N^+(p_2)\}$, akin to those outlined in Sections 3.5.1 and 3.5.2. A registration process then follows in the form of the situation assessment process described in Section 3.6. This provides a situation update $\hat{\Sigma}_q(k+1|k+1)$ for the situation indexed by q as a state of affairs in a canonical form at time step $k+1$. The situation association process then needs to match this against certain expected states of affairs extensions $\hat{\Sigma}_i(k+1|v)$ of explained situations $\underset{\rightarrow}{\hat{\Sigma}}_i(k)$ labeled by i.

The prediction process begins with a predictive assessment, which is responsible for the scenario assessment state prediction process at Level 2 and which involves assessing intent, capability and awareness. Concerning intent, at time step k each agent A in the environment will harbor a number of intended effects. Each intended effect $\Psi A(t|k)$ for $t > k$ is a future state of affairs intended for the environment at time future t by agent A at time step k. As a state of affairs, it is represented by a set of statements about the world. Figure 3.28 depicts two intended effects by agent A, $\Psi_A(k+u|k)$ and $\Psi_A(k+v|k)$ for some $v > u$. A set of intended effects for an agent, such as $\Psi_A(k+u|k)$, $\Psi_A(k+v|k)$, reflects the intended transitions in intended effects over time. The set set $\mathcal{I}_A(\mathrm{k}) = \{\Psi_A(t|k)|t \in \mathrm{Time\_Step}\}$ identifies agent A’s course of intent (COI) at time k, with intended future effects $\mathcal{F}_A(k) = \{\Psi_A(t|k)|t \in \mathrm{Time\_Step}\ \&\ t > k\}$ and past intentions $\mathcal{P}_A(k) = \{\Psi_A(t|k)|t \in \mathrm{Time\_Step}\ \&\ t \le k\}$. Past intentions are monotonic in that $\mathcal{P}_A(k) \subseteq \mathcal{P}_A(t)$ whenever $k < t$.

Concerning capability and awareness, at time step k agent A’s awareness of some particular aspect of the world is represented by its current situation assessment $\hat{\Sigma}_i(k|k)$ of that aspect of the world. Agent A will also typically retain an updated awareness of the past at time step k, which comprises the states of affairs of the form $\hat{\Sigma}_i(t|k)$ for $t < k$. A capability option is hereafter defined to be any activity that has the ability to transition the state of the world. Capability options can be modelled as functions by bundling together prospective concurrent actions at a given time. At time step k, agent A will usually have a number of capability options available to it. Performing capability option $c_{A,1}$ alone at time step k transitions the state of the world from state of affairs $\Sigma(k)$ to state of affairs $\Sigma(k+u) = c_{A,1}(\Sigma(\mathrm{k}))$ for some u. To agent A, performing capability option $c_{A,1}$ at time step k transitions its awareness $\hat{\Sigma}_i(k|k)$ of state of affairs $\Sigma_i(k)$ to its projected awareness $\hat{\Sigma}_{i,<1>}(k+u|k)$ of the state of affairs $\Sigma_i(k+u)$. So to agent A, $c_{A,1}(\hat{\Sigma}_i(k|k)) = \hat{\Sigma}_{i,<1>}(k+u|k)$. In Figure 3.28, agent A also considers capability option $c_{A,2}$ at time step k, with $c_{A,2}(\hat{\Sigma}_i(k|k)) = \hat{\Sigma}_{i,<2>}(k+u|k)$.

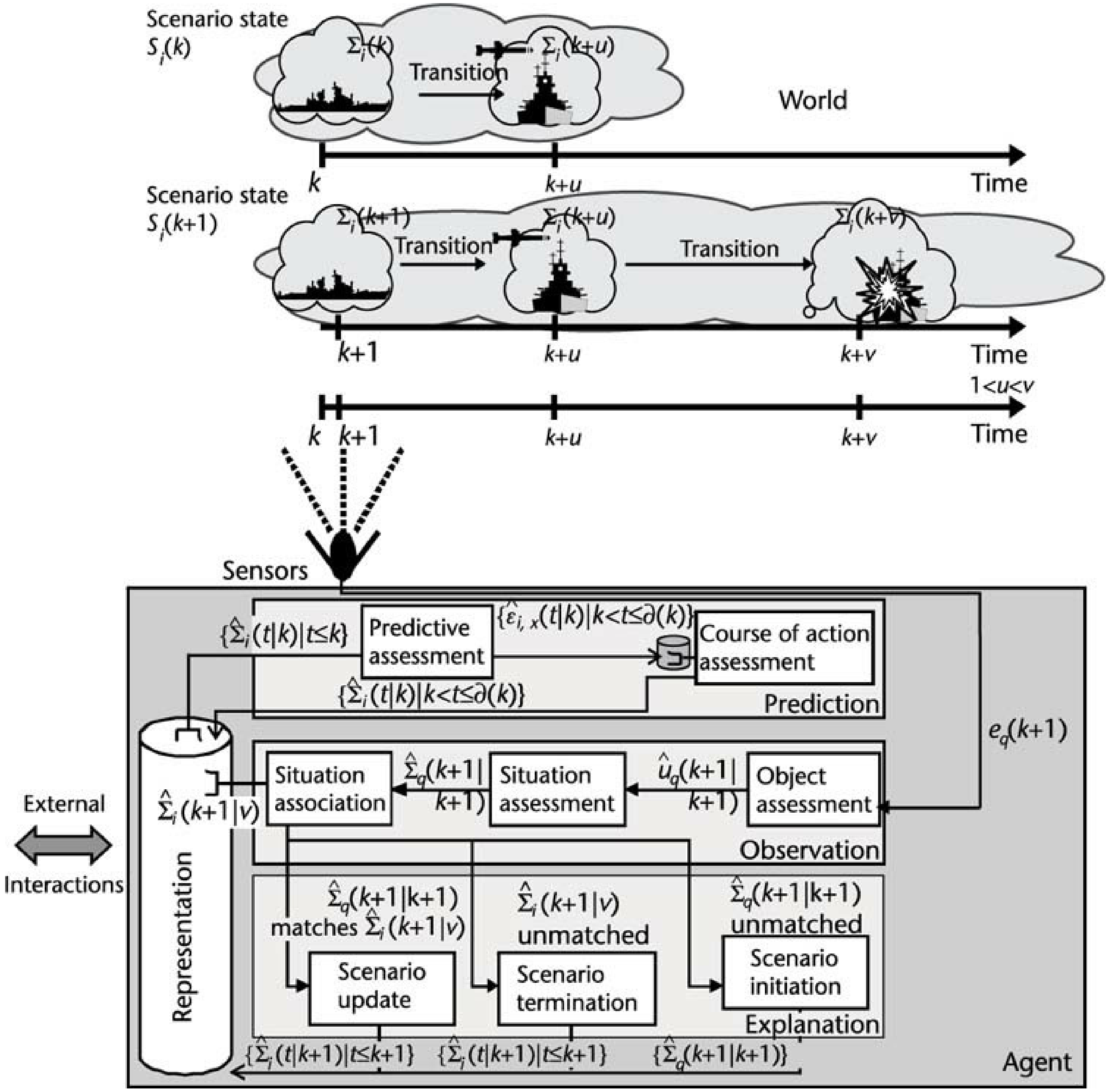


Figure 3.27 Level 3 STDF fusion.

The projection process can be recursively repeated on the projected states of affairs. Capability options $c_{A,3}$ and $c_{A,4}$ might be considered for projected state of affairs $\hat{\Sigma}_{i,<1>}(k+u|k)$ resulting in projected states of affairs $\hat{\Sigma}_{i,<1,3>}(k+j|k) = c_{A,3}(\hat{\Sigma}_{i,<1>}(k+u|k))$ and $\hat{\Sigma}_{i,<1,4>}(k+j|k) = c_{A,4}(\hat{\Sigma}_{i,<1>}(k+u|k))$ respectively. In general, $\hat{\Sigma}_{i,<a_1,\ldots a_w,a_{w+1}>}(t_{w+1}|k) = c_{A,a_{w+1}}(\hat{\Sigma}_{i,<a_1,\ldots a_w>}(t_w|k))$ $t_w < t_{w+1}$ with each sequence of capability options $<c_{A,a_1}, \ldots, c_{A,a_{w+1}}>$ termed a course of action (COA) and the associated set $\{\hat{\Sigma}_{i,<a_1>}(t_1|k), \ldots, \hat{\Sigma}_{i,<a_1,\ldots,a_{w+1}>}(t_{w+1}|k)\}$ of projected states of affairs termed a course of events (COE). The predictive assessment process is responsible for generating these candidate COEs.

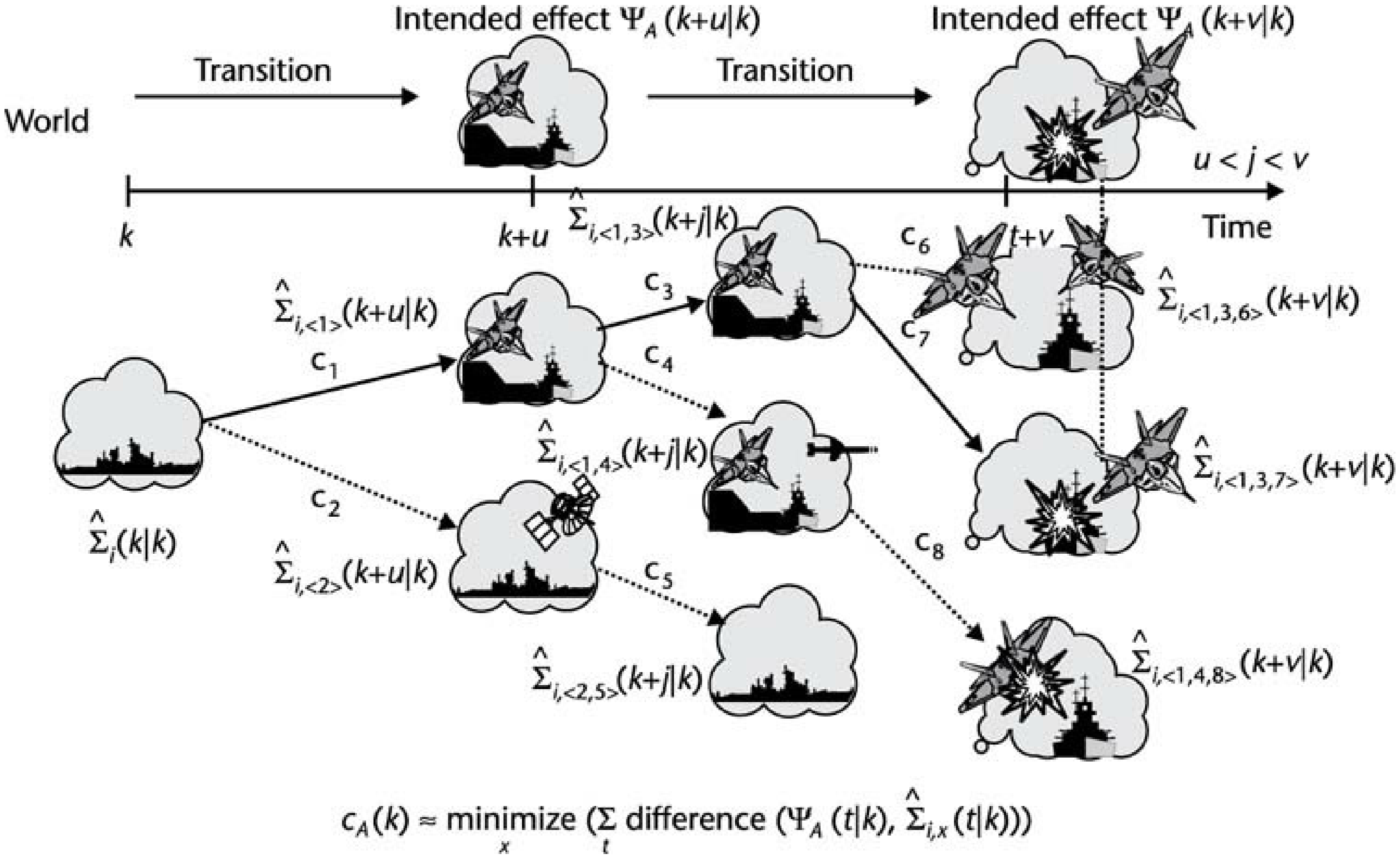


Figure 3.28 A scenario assessment.

The course of action process is responsible for determining the intended course of action for agent A, shown as the solid arrow line in Figure 3.28. Of course when contemplating the effect $\hat{\Sigma}_{i,<a_1,\ldots,a_{w+1}>}(t_{w+1}|k)$ obtained from performing capability option $c_{A,a_{w+1}}$ on the projected state of affairs $\hat{\Sigma}_{i,<a_1,\ldots,a_w>}(t_w|k)$, agent A's predictive assessment process needs to not only consider the "normal effect" associated with $c_{A,a_{w+1}}$, but also the reaction and independent actions of both the environment and other agents, be they people or machines, each of whom will potentially have their own intended effects and be contemplating courses of action to achieve them. To illustrate, in Figure 3.28 $\hat{\Sigma}_{i,<1>}(t|k)$ begins a particular course of action to satisfy the intended course of intent $\{\Psi_A(k+u|k), \Psi_A(k+v|k)\}$ to effect an air strike on a given enemy ship, but $\hat{\Sigma}_{i,<1,4>}(k+j|k)$ considers an enemy course of action in which an enemy missile is launched at the aircraft to prevent that air strike, and $\hat{\Sigma}_{i,<1,4,8>}(k+v|k)$ presents the environmental impact of the missile striking the strike aircraft.

At time step k, agent A performs a predictive assessment by evaluating possible tuples $\{<\{\Psi_A(t|k)|t \in \text{Time\_Step}\ \&\ k < t < \partial(k)\}, x_A(k), \{\hat{\Sigma}_{A,x}(t|k)|t \in \text{Time\_Step}\ \&\ k < t \leq \partial(\text{k})\}>|x_A(k) \in \text{COA}_A(k)\}$, and selects a preferred course of action

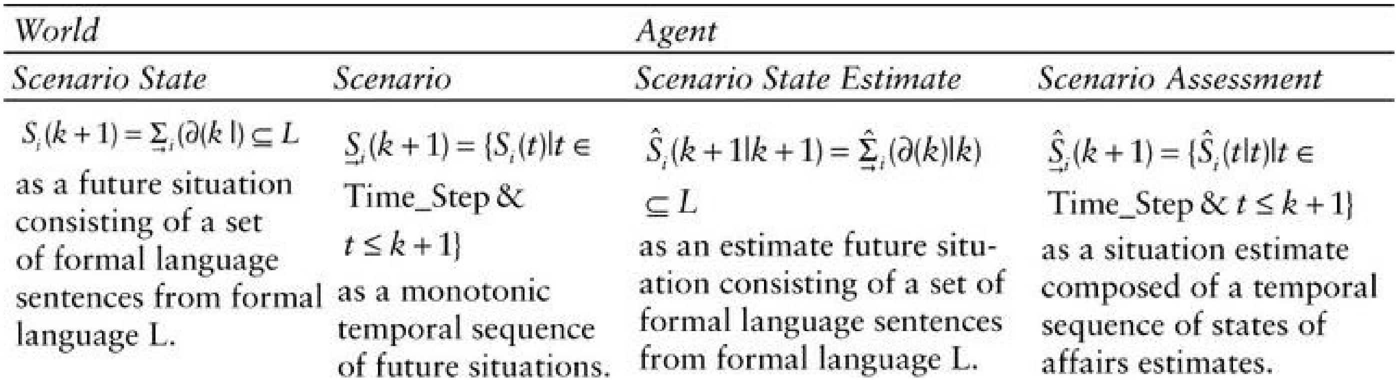
c_A(k) at time step k to identify its predicted tuple $<\mathcal{I}_A(k), c_A(\mathrm{k}), \hat{\varepsilon}_{i,c}(k)>$ at time step k, with estimated COE $\hat{\varepsilon}_{i,c}(k) = \{\hat{\Sigma}_{i,c}(t|k) | t \in \text{Time\_Step} \,\&\, k < t \leq \partial(k)\}$ for scenario state $S_i(k)$. *If* selected optimally for A, $c_A(k)$ will be the course of action that minimizes the difference between the projected states of affairs in the COA and the intended effects in A's COI over time. Formally, this amounts to solving a dynamic programming problem, as suggested in Figure 3.28, or a probabilistic dynamic programming problem if probabilistic transitions are considered. Based on the estimated COE $\hat{\varepsilon}_{i,c}(\mathrm{k})$, the course of action process identifies the expected state of affairs observations at certain times. These become the expected states of affairs observations $\hat{\Sigma}_i(k+1|\mathrm{v})$ that are matched against the actual states of affairs observations $\hat{\Sigma}_q(k+1|k+1)$ by the situation association process at time step k+1.

The explanation process must deal with three possible outcomes from the situation association process:

- If actual state of affairs observation $\hat{\Sigma}_q(k+1|k+1)$ is matched with an expected state of affairs observation $\hat{\Sigma}_i(k+1|v)$ by the situation association process, then the scenario update process uses this match to update the situation estimate $\underline{\underline{\hat{\Sigma}}}_i(k)$ by including $\hat{\Sigma}_q(k+1|k+1)$ and possibly revising other prior states of affairs $\hat{\Sigma}_i(t|k+1)$ in situation estimate $\underline{\underline{\hat{\Sigma}}}_i(\mathrm{k})$ for $t < k$+1.
- If actual state of affairs observation $\hat{\Sigma}_q(k+1|k+1)$ is not matched with any expected state of affairs observation $\hat{\Sigma}_i(k+1|v)$ by the situation association process, then the situation initiation process at Level 2 must have been responsible for $\hat{\Sigma}_q(k+1|k+1)$ and so the scenario initiation process is invoked at Level 3. It introduces a new scenario based on $\hat{\Sigma}_q(k+1|k+1)$ so that scenario prediction can occur in the next time step k+2.
- If expected states of affairs observation $\hat{\Sigma}_i(k+1|v)$ is not matched with anactual state of affairs observation $\hat{\Sigma}_q(k+1|k+1)$ by the situation association process, and this occurs for a designated number of time steps or through a more sophisticated deletion logic, then the scenario indexed by i will be deemed to no longer be observable and attempts to match states of affairs against the scenario indexed by i will no longer be made. This is monitored by the scenario termination process.

In summary, the scenario assessment system predicts and explains scenarios as patterns of change in observations of situations that are generated by the situation assessment process. Table 3.6 summarizes the Level 3 STDF fusion scenario assessment.

Table 3.6 Level 3 STDF Fusion



World		*Agent*	
Scenario State	*Scenario*	*Scenario State Estimate*	*Scenario Assessment*
$S_i(k+1) = \underset{\rightarrow}{\Sigma}_i(\partial(k\,\vert) \subseteq L$ as a future situation consisting of a set of formal language sentences from formal language L.	$\underset{\rightarrow}{S}_i(k+1) = \{S_i(t) \vert t \in$ Time_Step & $t \leq k+1\}$ as a monotonic temporal sequence of future situations.	$\hat{S}_i(k+1 \vert k+1) = \underset{\rightarrow}{\hat{\Sigma}}_i(\partial(k) \vert k) \subseteq L$ as an estimate future situation consisting of a set of formal language sentences from formal language L.	$\underset{\rightarrow}{\hat{S}}_i(k+1) = \{\hat{S}_i(t \vert t) \vert t \in$ Time_Step & $t \leq k+1\}$ as a situation estimate composed of a temporal sequence of states of affairs estimates.

rences

[1] Endsley, M.R., "*Design and Evaluation for Situation Awareness Enhancement*," In Proceedings of the Human Factors Society 32nd Annual Meeting, 1988.

[2] Miller, G. A., "The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information," *The Psychological Review*, Vol. 63, 1956, pp. 81–97.

[3] Giompapa, S., A. Farina, F. Gini, A. Graziano, and R. Di Stefano, "A Model for a Human Decision-Maker in a Command and Control Radar System: Surveillance Tracking of Multiple Targets," *International Conference on Information Fusion*, 2006.

[4] Endsley, M. R., "Situation Awareness and Human Error: Designing to Support Human Performance," *Proceedings of the High Consequence Systems Surety Conference*, 1999.

[5] Reason, J., *Human Error*, Cambridge University Press, 1990.

[6] White, F. E., "Data Fusion Lexicon," Joint Directors of Laboratories, Technical Panel for C3, Data Fusion Sub-Panel, Naval Ocean Systems Center, San Diego, 1987.

[7] Lambert, D. A., "Situations for Situation Awareness," *International Conference on Information Fusion*, 2001.

[8] Lambert, D. A., "Grand Challenges of Information Fusion," *International Conference on Information Fusion*, 2003.

[9] White, F. E., "A Model for Data Fusion," *National Symposium on Sensor and Data Fusion*, 1998.

10] Steinberg, A. N., Bowman, C. L., and White, F. E., "Revisions to the JDL Data Fusion Model," *Proceedings of SPIE*, Vol. 3719, 1999.

11] Steinberg, A. N., and Bowman, C. L., "Rethinking the JDL Data Fusion Levels," *National Symposium on Sensor and Data Fusion*, 2004.

12] Steinberg, A. N., Bowman, C. L., and White, F. E., "Revisions to the JDL Data Fusion Model," *The Joint NATO/IRIS Conference*, 1998.

13] Blasch, E., Kadar, I., Salerno, J., Kokar, M. M., Das, S., Powell, G. M., Corkill, D. D., and Ruspini, E. H., "Issues and challenges in situation assessment (Level 2 Fusion)," *Journal of Advancement in Information Fusion*, Vol. 1, No. 2, December 2006, pp. 122–139.

14] Blasch, E. and Plano, S., "Level 5: User Refinement to aid the Fusion Process," *Proceedings of SPIE*, Vol. 5099, 2003.

15] Lambert, D. A., "A Unification of Sensor and Higher-Level Fusion," *International Conference on Information Fusion*, 2006.

16] Lambert, D. A., "STDF Model Based Maritime Situation Assessments," *International Conference on Information Fusion*, 2007.

17] Lambert, D. A., "A Blueprint for Higher-Level Fusion Systems," *Journal of Information Fusion*, Vol. 10 p. 6–24, Elsevier, 2009.

18] Endsley, M. R., and D. B. Kaber, "Level of Automation effects on performance, situation awareness and workload in a dynamic control task," *Ergonomics*, Vol. 42, No. 3, 1999, pp. 462–492.

19] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME–Journal of Basic Engineering*, Vol. 82, Series D, 1960, pp. 35–45.

20] Block, N. J., "Introduction : What is Functionalism?," in *Readings in Philosophy of Psychology*, Vol. 1, N. J. Block (ed.), Cambridge, MA: Harvard University Press, 1980, pp. 171–184.

21] Norvig, P., *Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp*, San Mateo, CA: Morgan Kaufmann Publishers Inc., 1992.

22] Lambert, D. A., and C. Nowak, “The Mephisto Conceptual Framework,” *DSTO Technical Report TR-2162*, Department of Defence, 2008.

23] Blanchette, M., “Military Strikes in Atlantis—A Baseline Scenario for Coalition Situation Analysis,” The Technical Cooperation Panel, *Technical Report TR-C3I-TP1-1-2005*, 2005.

24] Chang, C. C., and H. J. Keisler, *Model Theory (2nd Ed.)* Amsterdam: North-Holland Publishing Company, 1977.

CHAPTER 4

Formalization of Situation Analysis Through Interpreted Systems Semantics

Patrick Maupin and Anne-Laure Jousselme (Canada)[1]

Introduction

We present the following work initiated in [2], where the interpreted systems semantics was proposed as a general framework for situation analysis. This framework is particularly efficient for representing and reasoning about knowledge and uncertainty when performing situation analysis tasks. Here, we detail and deepen the exposition of the interpreted systems semantics and give mathematical definitions of the concepts that we feel are needed to achieve a formal binding between the notions of situation analysis (SAN) and situation awareness (SAW). Our approach of SAN is to base our analysis on the production of state transition systems consisting in the set of all temporal trajectories possibly obtained upon the execution of a given set of agents' protocols. Thus seen, the SAN task involves the definition of more or less subtle reasoning about graph structures.

This follows a standard intuition about Levels 2 and 3 of the standard DFIG (Chapter 2) and JDL Information Fusion model where, according to Steinberg and Bowman [3], situation assessment (SA) (Level 2) is the estimation and prediction of entity states on the basis of inferred relations among entities, whereas Impact Assessment (Level 3) is usually implemented as a prediction, drawing particular kinds of inferences from Level 2 associations. The following sections briefly review the principal formal models of information fusion as they relate to the SA task (discussed in Section 4.1.1), and the basic problems tackled in this chapter about the notions of situation, situation awareness, and situation assessment.

4.1.1 Formal Models of Higher Levels of Information Fusion

Information fusion is often defined as the process of combining information in order to estimate or predict entity states. In the literature, three kinds of information fusion models which can be distinguished [3]; process, functional, and formal have already been proposed as guides for the design and implementation of information fusion systems. Some examples of process models are John Boyd's Observe-Orient-Decide-Act loop, the Predict-Match-Extract-Search loop, and the UK's Intelligence community Collection, Collation, Evaluation, and Dissemination cycle [4]. The JDL functional model [3, 5] has different levels and corresponding applications, and is one of the most referred to information fusion models. The usual functions considered for information fusion are conveniently separated into so-called levels of processing, even though practice has shown a less clear-cut specialization.

The main advantages of the formal models of information fusion is that they allow for the verification of the systems before they are fielded, they provide specification means that will allow a more rapid and robust software implementation, and they allow characterization of the complexity of representations and tasks. Usually, the formal models are verified using simulation, scenario-based design techniques, theorem-provers, model checking techniques or combinations of these. Finally, formal models allow the confirmation or falsification of theories by making results reproducible.

We have conveniently distinguished the formal approaches to the problem of situation analysis and higher-levels of information fusion into two formal modeling families, as illustrated in Figure 4.1.

It is worth mentioning that members of these families of formal models are sometimes, mathematically and practically speaking, closely related and differences are often only a question of terminology. First, we distinguish algebraic frameworks from the formal methods commonly used in computer science and software engineering. These formal methods can be further distinguished into highly formal ontologies, such as the ones proposed in [6] for situation awareness and specification languages (for example, the well-known UML and DARPA's DAML are also used to model the situation awareness process). Usually, these methods are supported by specialized software and displays that will either guide the engineer through the usual software development cycle or, as in the case of highly formal ontologies, help him verify if the specified ontologies are complete and sound. On the other hand, the specification of situation analysis and high-level information fusion, based on algebraic frameworks, include work on Category Theory by Kokar et al. [7] (where information fusion processes are studied); the use of generalized information theory [8] for the modelization of information processing and uncertainty characterization at all levels of the JDL model; and a new approach based on the interpreted systems semantics used originally for the analysis of distributed systems [9] and proposed in [2] to model the situation analysis process. These three approaches to high-level information fusion typically use different verification or query strategies. While category theory uses the traditional theorem, proving approach and generalized information theory are mainly based on the use of induction, the interpreted system approach is tradition ally based on the very efficient technique of model checking. In practice, matters are however not so clear-cut and cross-fertilization between the various specification and verification techniques are common.

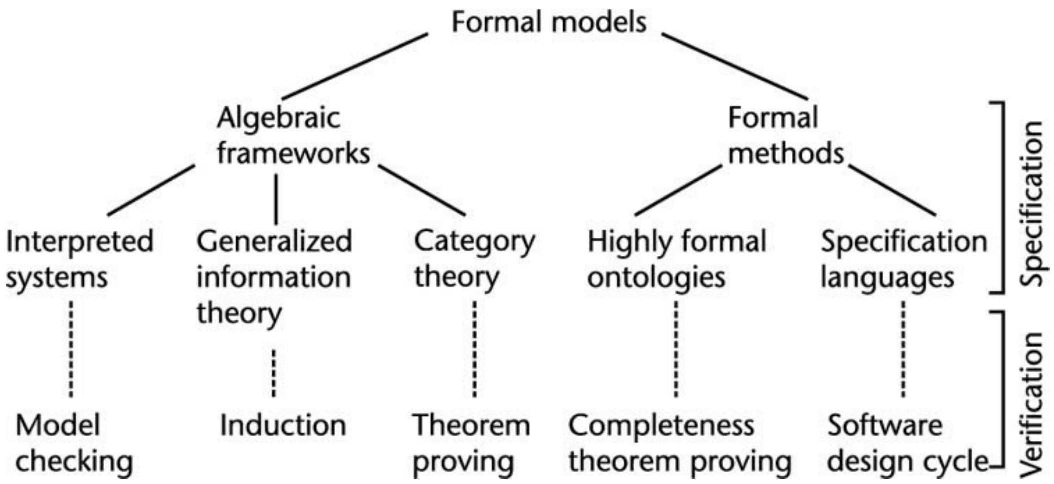


Figure 4.1 Formal models for information fusion.

While lower levels of information fusion L.0 and L.1 (see Chapter 2) already lay on solid formal foundations, at least as far as information, uncertainty and aggregation processes modeling are concerned, higher levels of information fusion (L.2–L.6) still lack a unified formal theory, although many works recently emerged [7, 10]. The mathematical modeling approach is typically based on a single abstract theoretical framework used to model either information or processes. For instance, one will use probability theory and model random variables carrying information about perception or measurements. On the other hand, mathematical theories can be used instead to model explicitly the information fusion process. A good example of this approach is given by the work of Kokar and his collaborators, where model theory [11] and category theory [7] are used to model information fusion processes, and for instance demonstrated in [12], that decision fusion is a subclass data fusion. Because of space limitations we do not review the very interesting theories offered by theoretical computer science, economy and game theory for reasoning about processes. Extending the notion of SAN to the general idea of representing and reasoning about processes one can consider extensive games as process models useful for the representation of mental states such as uncertainty and knowledge. The state space representation of knowledge and common proposed by Aumann [13] in economy and game theory is also an interesting model although it lacks an explicit representation of the passage of time. Computer science proposed *process calculi*, such as Hoare-Dijkstra calculus, process algebra, modal, and dynamic logics.

Interpreted systems semantics (as discussed Section 4.2.1) combine many features of these theories and can be seen as a conjunction of the state space and the extensive game modeling approaches (demonstrated in Chapter 14) while being much more explicit about the agents' sources of information. While the rational behavior of interacting groups of agents can be efficiently modeled using the interpreted systems semantics, the approach also offers the flexibility needed to model the most common departures from rationality, including resource-boundedness. Section 4.2.2.1 deals explicitly with the algorithmic modelization of processing power limitations. This discussion can be easily extended to deal with most kinds of mental limitations. Section 4.2.2.2 deals with a specific type of interpreted system used to model plausibility, a general concept allowing to model and reason about uncertainty.

In practice formal models, such as the Interpreted Systems, semantics have two principal aims to describe the agents' reasoning schemes and to de scribe the way modelers' reason about such distributed systems. This distinction between levels of analysis leads to the distinctions and definitions of Section 4.3 and, most particularly, to our definition of situation, situation awareness, and situation assessment. Practical applications of these concepts are presented in Section 4.4.

4.1.2 Situations in State Spaces

Situation analysis (SAN) in the military domain can be performed either at the tactical, operational or strategic level. At the tactical level one is faced with an adversary hiding its presence or its true identity by using stealth or dual-use technologies. High velocity targets also add a level of complexity to the SAN task at this level. Adversaries also try to confuse one another by acting with audacity, by causing surprises, such as those triggered by using new tactics for which no corresponding course of action exists. The knowledge of terrain characteristics is central at the tactical level. At the operational level, one is confronted with adversaries hiding their intentions or plans, as well as the extent of their resources. At the strategic level, the previous issues also have to be taken into account when performing situation analysis. Here, the adversaries attempt to shatter the enemy's will and cohesion by using political, diplomatic, psychological, and financial means at their disposal.

There have been many attempts to formalize the process of situation analysis for the design of decision support systems and the automation of tasks. From the above sketch, one can abstract the main goals of situation analysis in terms of reasoning about relations holding among entities of interest and also in terms of reasoning about the properties of these relational structures thus revealed. Modeling the interaction between agents' knowledge, awareness, resources, abilities, and plans in a unified framework seems necessary to encompass the above mentioned levels of SAN. In [14], we have shown how to bind plans and abilities to the interpreted systems framework, the latter used to model epistemic changes occurring through time.

In probability theory, Shafer [15] defines a probability-tree structure involving special events designated as situations in an attempt to reconcile the subjective and objective interpretations of probability. In this work, Shafer explicitly links a state space representation and the time flow for the definitions of a situation. In logic, rather than defining an inference process, McAllester [16] defines the notion of a *situation*, which is a more general concept in the semantics theoretical framework, and defines a meaning function such that each proposition is either TRUE or FALSE in each situation. In the terminology of mathematical logic, the situations are also called models, and the meaning of an expression is called the denotation of that expression. A language of propositions, together with a set of models, or situations, and a way of expressing a truth value to a proposition in a model, is called a “logic.” This work by McAllester encouraged us to pursue the research of logic-based approaches for the formalization of situation analysis. In game theory, it is even possible to draw a correspondence between the type of game played and the notion of a situation. In this case, this is a meta system concept, since it involves classifying and comparing systems. Zero-sum games, repeated games, and perfect information games correspond to typical situations. A game can be characterized by a set of rules called the rules of the game, including normative accounts about players, actions, payoffs, and information. Indeed, for Rasmussen [17] the modeler’s objective is to describe a situation in terms of the rules of a game so as to explain what will happen in the situation. The aim of a player in game theory is to plan actions, or strategies, based on the information it has access to in order to maximize its payoffs. The set of strategies used by a group of agents playing the same game is called the equilibrium. This equilibrium gives the modeler a precise idea about what will result from the game. Given the set of possible equilibria, the goal of the modeler will be to calculate the possible outcome or find the best possible equilibrium.

Our general definition of a situation (Def. 4.3.1) is an attempt to bring together the best of these worlds.

Background

This section gives the basic definitions of interpreted systems semantics, including truth sets, as well as the basic modal knowledge and temporal operators. In Section 4.2.2, we will show how the notion of an interpreted system can be tailored to model various notions of resource boundedness, including situation awareness as well as mental states such as uncertainty or preferences.

4.2.1 Interpreted Systems

Let $\mathcal{A} = \{1, 2, 3, \ldots, n, e\}$ be a set of agents where e is a special agent denoting the environment. Each agent i is assumed to be in some local state l_i at a given time, encapsulating all the information the agent has access to. This information can also include past states, actions, and information about the protocols used by the agent. The local state of the environment is denoted by l_e and encodes all the relevant information that is not encoded in the agents' local states. In particular, l_e can encode the objective state of the world, for example, one not attainable by the agents' perception and reasoning means. For SAN applications, apart from the objective states of the world, l_e can contain maps of the environment, network information, or any other information describing the outside world. The agents' local states can encode partial or imperfect views of this outside world.

A global state s is an element of $S \subseteq L1 \times \ldots \times L_n L_e$, where L_i is the set of the possible local states of the agent i. The local state of Agent i corresponds to the ith component of the global state $s = (l_1, \ldots, l_n, l_e)$. A sequence of global states $s^1, s^2, \ldots$ is called a run r over S and is a function from time to global states. A *system* $\mathfrak{R}$ is a set of runs and (r, m) denotes a point in $\mathfrak{R}$, consisting of a run r and a time m. The state of the system at time m in the run r is $r(m)$. If $r(w) = (l_1, \ldots, l_n, l_e)$ is the global state at point (r, m), then $r_e(m) = l_e$ and $r_i(m) = l_i$ for $i = 1, \ldots, n$ are respectively the environment's and the agents' local states at point (r, m). A round m in run r takes place between time $m - 1$ and time m.

Actions are the cause of changes in the system and are performed in rounds by the agents and the environment. Let ACT_i be the set of actions that can be performed by Agent i, and let ACT_e be the set of actions performed by the environment. A joint action is an element of $ACT_e \times ACT_1 \times \ldots \times ACT_n$, for example, a tuple $(a_e, a_1, \ldots, a_n)$ of actions performed by the set of agents and the environment, where a_e is the action performed by the environment, and a, is the action performed by the agent i.

A protocol P_i for the agent i is a mapping from the set L_i of local states of the agent i

to nonempty sets of actions in $ACT_i, i \in \mathcal{A}$. A protocol is a function on local states rather than on global states. A joint protocol P is a tuple $(P_1, P_2,..., P_n)$ consisting of the protocols of each of the agents i, $i = 1,..., n$. This corresponds to the definition of equilibrium in game theory. Note that P_e, the protocol of the environment, is not included in the joint protocol. Rather, the protocol of the environment is supposed to be given or at least estimated, and P and P_e can be viewed as the strategies of opposing players. A context γ is a tuple (P_e, S_0, τ, Ψ) where P_e is a protocol for the environment, S_0 is a nonempty subset of S describing the state of the system at the initiation of the protocol, τ is a transition function, and Ψ is an admissibility condition on runs. The environment's protocol P_e can be used to model the adversary's strategies, or to simply model random errors or events in a given situation. A transition function τ assigns for each global state and each joint action the resulting global state obtained after performing the joint action. The transition function describes which actions can be performed from a given global state. The admissibility condition Ψ on runs tells which ones are legal. In practice, Ψ can be used to shrink down a large system or to model fairness conditions. Formally, Ψ is a set of runs and $r \in \Psi$ if and only if r satisfies the condition Ψ. Note that the description of the behavior of a system is contextual, such as a joint protocol P is always described within a given context γ.

Let Φ be a set of primitive propositions, describing basic facts about the system. Formulas are built using the classical operators of propositional logic. The set of formulas is closed under $\neg$ and $\wedge$ (negation and conjunction). Hence, given two formulas ϕ and ψ, ϕ, $\phi \wedge \psi$ are also formulas. Let $\mathcal{L}(\Phi)$ denote the language of Φ, such as the set of well-formed formulas.

An *interpreted system* $\mathcal{I}$ consists of a pair $(\mathcal{R}, \pi)$ where $\mathcal{R}$ is a system over a set S of global states and π is an interpretation for the propositions in Φ over S, which assigns truth values (either true or false) to the primitive propositions at the global states. Thus, for every $p \in \Phi$ and state $s \in S$, we have $\pi(s)(p) \in \{0; 1\}$. The satisfaction of formulas in $\mathcal{L}(\Phi)$ is given by:

$$
\begin{aligned}
&(\mathcal{I},r,m) \models p \text{ iff } \pi\big(r(m)\big)(p) = 1\\
&(\mathcal{I},r,m) \models \phi \wedge \psi \text{ iff } (\mathcal{I},r,m) = \phi \text{ and } (\mathcal{I},r,m) = \psi\\
&(\mathcal{I},r,m) \models \neg\phi \text{ iff } (\mathcal{I},r,m) = \phi
\end{aligned}
\tag{4.1}
$$

The truth set of ϕ is the set of points satisfying ϕ, such as:

$$
\|\phi\| = \{(r,m) \in \mathcal{I} | (\mathcal{I},r,m) \not\models \phi\} \tag{4.2}
$$

Defining local states for each agent induces a series of equivalence relations $\sim_i$ over the points of the system. Hence, for $i \in \mathcal{A}$, $R_i(r, m) = \{(r', m')|(r, m) \sim_i (r', m')\}$ is the equivalence class containing the points at which i has the same information than at (r, m). The standard model of modal logic S5[2], represented by the operator K_i, $i \in A$, and whose semantics are:

$$\begin{aligned}&(\mathcal{I}, r, m) \models K_i\phi \text{ iff } (\mathcal{I}, r', m') \models \phi \text{ for all } (r', m')\\&\text{such that } (r, m) \sim_i (r', m')\end{aligned} \tag{4.3}$$

The interpreted system $(\mathcal{I}, r, m)$ logically entails that $\mathcal{I}$ knows ϕ, iff, the agent i *knows* that ϕ "if and only if ϕ holds in all the points it cannot distinguish from (r, m)," for example, the points in which i is in the same local state $r_i(m)$.

Besides all compositions of individual knowledge operators, like $K_iK_j\phi$ ("Agent i knows Agent j knows ϕ") or $K_i \neg K_jK_k\psi$ ("Agent i knows Agent j does not know that Agent k knows ψ"), we are interested in more global statements involving a group of agents G. Group knowledge operators are then defined with their semantics in interpreted systems [18]:

$$\begin{aligned}&(\mathcal{I}, r, m) \models S_G\phi \text{ iff } \exists \mathcal{I} \text{ such that } (\mathcal{I}, r, m) \models K_i\phi\\&(\mathcal{I}, r, m) \models E_G\phi \text{ iff } (\mathcal{I}, r, m) \models K_i\phi \text{ for all } i \in G\\&(\mathcal{I}, r, m) \models C_G\phi \text{ iff } (\mathcal{I}, r, m) \models E_i^k \text{ for all } k > 0\\&(\mathcal{I}, r, m) \models D_G\phi \text{ iff } (\mathcal{I}, r', m') \models \phi \text{ for all } (r', m')\\&\quad\text{such that } (r, m) \sim_i (r', m') \text{ for all } i \in \mathcal{A}\end{aligned} \tag{4.4}$$

$S_G\phi$ reads "Someone in group G knows that ϕ", $E_G\phi$ reads "Everyone in group G knows that ϕ," $C_G\phi$ reads "ϕ is common knowledge in group G," $D_G\phi$ reads "Group G has distributed knowledge that ϕ." Distributed knowledge is the combined knowledge of all the members of G [18], and corresponds to the knowledge that would be ascribed to an agent that would have fused the individual knowledge of the members of G.

Finally, to reason about the temporal evolution of the system, temporal operators are defined:

$$\begin{aligned}&(\mathcal{I}, r, m) \models \bigcirc\phi \text{ iff } (\mathcal{I}, r, m+1) \models \phi\\&(\mathcal{I}, r, m) \models \phi U \psi \text{ iff } \exists m \geq m' \text{ such that } (\mathcal{I}, r, m') \models \psi\\&\text{and } \forall m'' \text{ such that } m \leq m'' \leq m', (\mathcal{I}, r, m'') \models \phi\end{aligned} \tag{4.5}$$

$O\phi$ reads "ϕ will be true next time," and $\phi U\psi$ reads "ϕ is true until ψ is true." These basic temporal operators of Linear Time Logic (LTL) can be extended to include some shortcuts such as **F**ϕ (eventually ϕ) or **G**ϕ (always ϕ), but also to include Computational Tree Logic (CTL) operators, such as **A**ϕ (for all sequences ϕ).

4.2.2 Different Kinds of Interpreted Systems

In this section, we detail different versions of interpreted systems, providing a basis for our situation analysis modelization problem. Based on these models of interpreted systems and combining their principal features, we propose in Section 4.3 the notion of interpreted belief change system as a topic for further investigation.

4.2.2.1 Interpreted Algorithmic Systems

The knowledge notion introduced previously through the modal operator K_i is an implicit notion of knowledge; the agents are not assumed to compute this knowledge, implicit knowledge being rather the vision of the analyst of the system. Indeed, this notion of knowledge is based on the standard logic modal system S5 which suffers from the logical omniscience problem; the agents are assumed to know all tautologies as well as all the logical consequences of their knowledge, making them perfect reasoners. A notion of explicit knowledge called "algorithmic knowledge" has been introduced in [19]. Algorithmic knowledge is an internal notion of knowledge that the agent can compute, given an internal algorithm. This explicit notion of knowledge is a general epistemic model of agent resource-boundedness and is at the basis of our definition of situation awareness in Section 4.3. While $K_i\phi$ denotes that fact that i is granted the knowledge of ϕ (without itself necessary knowing this fact), $X_i\phi$ is used in [19] to denote the fact that i can compute that it knows ϕ. Each agent owns a local algorithm A_i allowing it at each point of the system to decide if it knows ϕ Such an algorithm A_i takes as inputs a point (r, m) together with a formula ϕ and outputs "Yes" if i knows that ϕ is true, "No" if i does not know if ϕ is true and "?" if i is unable to compute if it knows ϕ A_i is sound if it never returns wrong answers and it is complete if it never returns "?". It follows that for a sound and complete algorithm, the explicit knowledge equals the implicit knowledge ($X_i\phi \Leftrightarrow K_i\phi$).

Definition 4.2.1 (*Interpreted Algorithmic System [19]*): An Interpreted Algorithmic System (IAS) is an interpreted system $\mathcal{I}$ in which the local state of each agent i at point (r, m) is a pair $(\mathbf{A}_i, l_i)$ where $\mathbf{A}_i$ is i's local algorithm and l_i is i's local data. The algorithmic knowledge denoted by the modal operator X_i is then defined by:

$$\begin{gathered}(\mathcal{I}, r, m) \models X_i\phi \text{ iff } \mathbf{A}_i(\phi, l_i) = \text{"Yes"} \\ \text{for } \mathbf{A}_i = \mathrm{alg}_i(r, m) \text{ and } l_i = \mathrm{data}_i(r, m)\end{gathered} \tag{4.6}$$

4.2.2.2 Interpreted Plausibility Systems

Kripke structures model belief instead of knowledge by simply changing the properties of the binary relation between points. In particular, instead of an equivalence relation, belief has often been represented by either Euclidean and transitive relations (K45 system) or Euclidean, symmetric, and transitive binary relations (KD45 system)[3]. Contrary to systems for knowledge in which only true formulas can be known, systems for belief do not satisfy the truth axiom (T) as false formulas can be believed. An alternative to this model of belief which turns to be more general is proposed in [20]. The agent's beliefs are not determined through binary accessibility relations but rather by plausibility spaces $\mathcal{P}_i(r, m)=(S_{(r, m, i)}$, $\mathrm{PL}_{(r, m, i)})$ for each agent i and each point (r, m) of $\mathcal{I}$, $S_{(r, m, i)}$. PL is a plausibility measure defined in [21] $\mathrm{PL} : 2^S \rightarrow D$ to satisfy the following properties:

$$\begin{aligned} &\mathrm{PL}(\varnothing) = \bot_D \\ &\mathrm{PL}(S) = \top_D \\ &\mathrm{PL}(A) \leq \mathrm{PL}(B) \text{ if } A \subseteq B \end{aligned} \tag{4.7}$$

D is a partially ordered set (poset) by the relation ≤. This notion of plausibility measure generalizes among others probability measures, belief, and plausibility measures of Shafer [22], possibility and necessity measures of Zadeh [23], in which cases $D = [0, 1]$, $\bot_D = 0$ and $\mathrm{T}_D = 1$. PL must not be confused with Shafer's plausibility measure, such as Pl in Section 4.4.3.

Definition 4.2.2 (Interpreted Plausibility System [20]): An interpreted plausibility system (IPS) is a tuple $(\mathcal{R}, \pi, \mathcal{P}_1, \ldots, \mathcal{P}_n)$ where $(\mathcal{R}, \pi)$ is an interpreted system and $\mathcal{P}_i$ is a plausibility assignment mapping each point (r, m) to a plausibility space $\mathcal{P}_i(r, m)=(S_{(r, m, i)}, \mathrm{PL}_{(r, m, i)})$, describing the relative plausibility of events from the point of view of the agent i at (r, m).

In such a system, the semantics for belief, B_i, is:

$$(\mathcal{I}, r, m) = B_i \phi \text{ iff } K_i\left(\mathrm{PL}(\phi) \geq \mathrm{PL}(\neg\phi)\right) \tag{4.8}$$

Hence, an agent i is said to believe ϕ if it knows that ϕ is more plausible than $\neg\phi$ in all the worlds it considers plausible [20].

The plausibility of a formula of $\mathcal{L}(\phi)$ at point (r, m) and according to agent i is then defined as the plausibility of its truth set:

$$\mathrm{PL}_{(r,m,i)}(\phi) = \mathrm{PL}_{(r,m,i)}(\|\phi\|) \tag{4.9}$$

Although Definition 4.2.2 is quite general, some restrictions naturally arise in most of its applications. In particular, we may want that $S_{(r, m, i)} = R_i(r, m)$ that is the agent considers plausible only states that are possible according to its knowledge, such as its local state. Moreover, we may also often require that if $(r, m) \sim_i (r', m')$ then $\mathcal{P}_i(r, m)=\mathcal{P}_i(r', m')$ that is that the plausibility space of the agent depends only on its local state. Other restrictions can also been envisaged, allowing to better fit to a particular application but also to make the modelization much simpler.

4.2.2.3 Interpreted Belief Change Systems

In order to model belief change (revision and update), interpreted plausibility systems have been restricted to satisfy some additional conditions, leading to interpreted belief change systems.

Definition 4.2.3 (Interpreted Belief Change System [24]): An interpreted belief change system is an interpreted plausibility system $(\mathcal{R}, \pi, \mathcal{P}_1, \ldots, \mathcal{P}_n)$ satisfying the five following conditions:

(IBCS1) ϕ can be evaluated with respect to $r_e(m)$ only:

$$l_e \models \phi \text{ if } (\mathcal{I}, r, m) \models \phi \text{ for some } (r, m); r_e(m) = l_e \tag{4.10}$$

(IBCS2) The agent's local state is of the form:

$$r_i(m+1) = \left\langle r_i(m), o_{(r,m)} \right\rangle \tag{4.11}$$

where $r_i(m) = (o_{(r,1)}, \ldots, o_{(r,m)})$ and $o_{(r,m)} \in \mathcal{L}_e$ is the observation at time m in run r.

(IBCS3) The language $\mathcal{L}(\phi)$ includes a set ϕ_{obs} of primitive propositions disjoint from ϕ_e such that $\phi_{obs} = \{\text{learn}(\phi); \phi \in \mathcal{L}_e\}$ with:

$$\pi(r, m)(\text{learn}(\phi)) = 1 \text{ iff } o_{(r,m)} = \phi, \forall r \text{ and } \forall m \tag{4.12}$$

(IBCS4) The agent never observes false:

$$(\mathcal{I}, r, m) \models o_{(r,m)} \text{ for all runs } r \text{ and times } m \tag{4.13}$$

(IBCS5) There is a prior $\mathcal{P}_i = (\mathcal{R}, \text{PL}_i)$ such that PL_i is a plausibility measure.

While in [24], the observations are reliable (ϕ is observed only if ϕ is true), the model has been extended to unreliable observations in [25] to account for Markovian observation models. This kind of interpreted belief change system is then called an observational system.

Formalization of the Situation Analysis Process

We claim that the interpreted systems semantics can be used as a blueprint for situation analysis as it provides the basics for the formalization of situation analysis elements. Moreover, the different kinds of interpreted systems detailed in Section 4.2.2 cover most of the features required in the design of a situation analysis problem. Indeed the limitation of agents' capabilities or awareness, the representation of a wide variety of uncertainty types, information aggregation, updating and revision can all be efficiently modeled within the IS formalism.

Hence, our model of situation analysis is an Interpreted Algorithmic Belief Change System (IABCS) $\mathcal{I}$ generated by the joint protocol of n agents within a context γ, the local states of the agents are tuples of the form:

$$r_i(m) = \langle \mathrm{alg}_i(r,m); \mathcal{P}_i(r,m); \mathrm{obs}_i(r,m) \rangle \tag{4.14}$$

where for each agent i and point (r, m) of the system $\mathcal{R}$, $\mathrm{alg}_i(r, m)$ is an algorithm of truth evaluation, $\mathcal{P}_i(r, m)$ is a plausibility space and $\mathrm{obs}_i(r, m)$ is a sequence of observations.

4.3.1 Situation

Based on the interpreted systems semantics, we define the situation in terms of a transition state system in which the arcs are labeled by joint actions and nodes by global states. Formally, using the IS semantics:

Definition 4.3.1 (Situation): Let $\mathcal{A}$ be a set of $n + 1$ agents including the environment and let $\mathcal{I}$ be the interpreted system representing the joint protocol P of the n agents in the interpreted context (γ, π), where γ is the context and π is the interpertation. A *situation* is the subsystem $\mathcal{I}_{(r,\ m)}$ of $\mathcal{I}$, that is the system representing P in the interpreted context $(\gamma_{(r,\ m)}, \pi)$ where $\gamma_{(r,\ m)} = ((r,\ m), P_e, \ \Psi, \tau)$. For a given agent, the (local) situation is the projection of $\mathcal{I}_{(r,\ m)}$ over $r_i(m)$, the local state of i at (r, m), provided a partial view of it.

A situation can thus be seen as a subset of states and corresponding transitions inside a set of protocols runs. One can distinguish three remarkable cases of situations (Figure 4.2):

1. The first case [Figure 4.2(a)] is simply the global state denoted either simply by s, or by (r, m), which is a point of an arbitrary run r at time m. This case of a situation as a rather static object is the typical sort of situation modeled in information fusion, as proposed by Steinberg and Bowman [3] and followed by Baclawski, et al. [26], although the state of the environment is not often explicitly

encoded.

2. The second remarkable case of a situation [Figure 4.2(b)] is simply given an initial state $(r, 0)$ the sequence of state transitions leading to (r, m), which is for instance the current global state and the fan of all possible state transitions thereafter. Of course, one can restrict the analysis of such a situation to an interval around (r, m). Multisensor tracking and multihypothesis tracking (MHT) rely on this kind of situational modeling and more generally game situations modeling.
3. The third remarkable case of situation of notable interest [Figure 4.2(c)] is the one where the study of the agents' joint protocol is not restricted to a single initial state, but rather the full spectrum of the k possible initial states S_0 is unfolded leading to the full system $\mathcal{R}$. This is typically the kind of situation studied in the analysis of parallel and distributed systems using formal notions of knowledge [27–29].

In Figure 4.2, the global states do not necessary correspond to the grid and illustrates the fact that some approximation may be required to represent the uncertainty, the difference between the measures, and the universe of discourse for instance.

4.3.2 Situation Awareness

Awareness is not simply a special state of knowledge. It also refers to a limited capacity of the agents to reach a perfect state of knowledge, the one that would be reached by perfect logically omniscient reasoners. Indeed, when defining situation awareness, some concepts of attention, vigilance, intelligence, and stress arise [30].

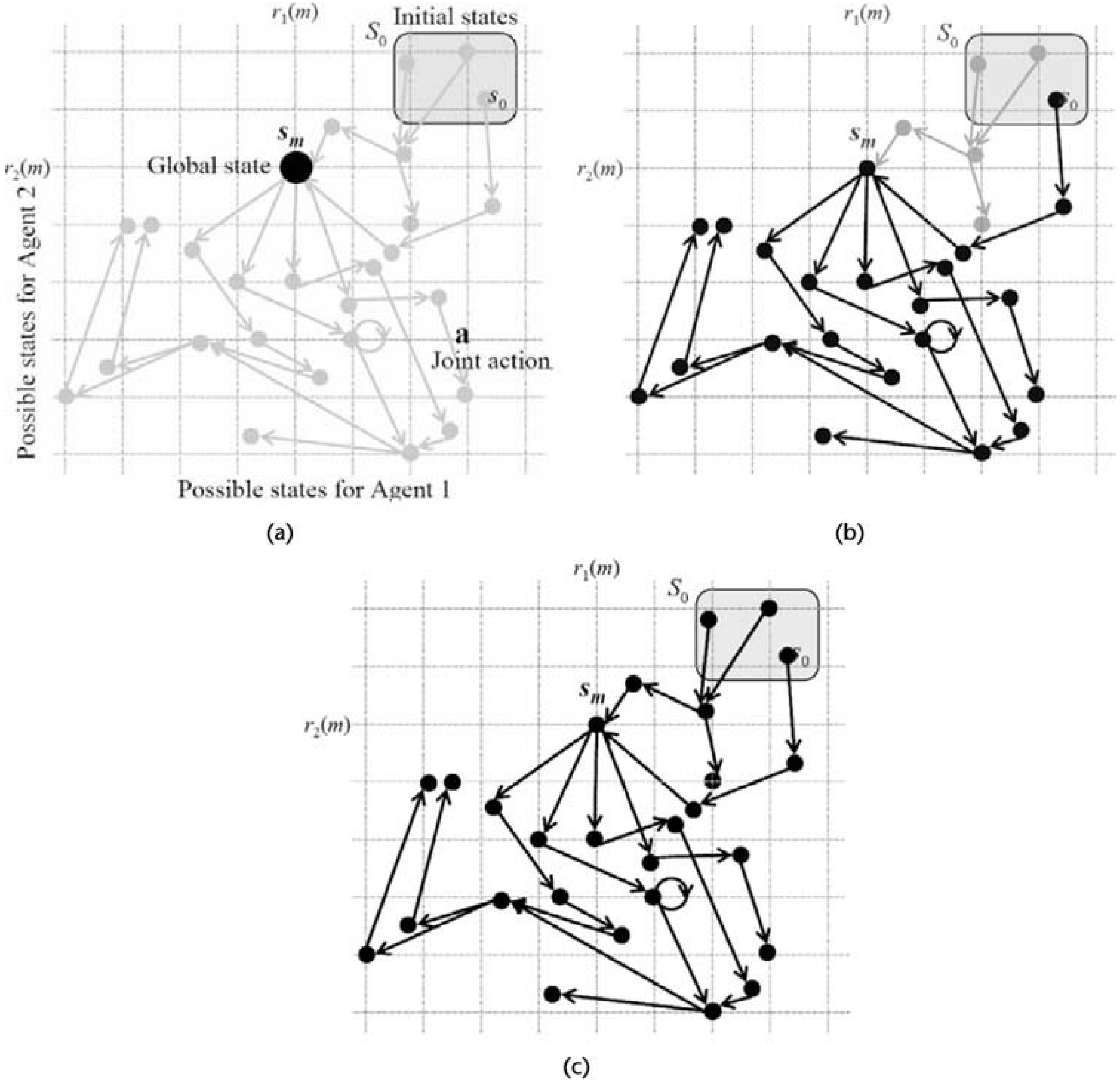


Figure 4.2 Three remarkable cases of situations defined in terms of states transitions.

In order to account for limited resources of agents Fagin and Halpern introduced, a logic of general awareness in [31]. Built upon a standard S5 model, this logic assigns to each point (r, m) of the system and for each agent i an arbitrary subset of formulas of $\mathcal{L}(\phi)$ about which the agent is aware, denoted by $A_i(r, m)$. The semantics of the awareness operator A_i are thensimply:

$$(\mathcal{I}, r, m) \models A_i\phi \text{ iff } \phi \in A_i(r, m) \tag{4 15}$$

that reads “the agent i is aware of ϕ at point (r, m) if and only if (ϕ belongs to $A_i(r, m)$.” The operator A_i acts as a filter on implicit knowledge to define explicit knowledge $X_i\phi \Leftrightarrow K_i\phi \land A_i\phi$. Figure 4.3 illustrates the interaction of awareness, explicit knowledge and implicit knowledge. In Figure 4.3, three kinds of formulas are distinguished: those about which the agent is aware, those about which it has implicit knowledge, and those about which it has explicit knowledge.

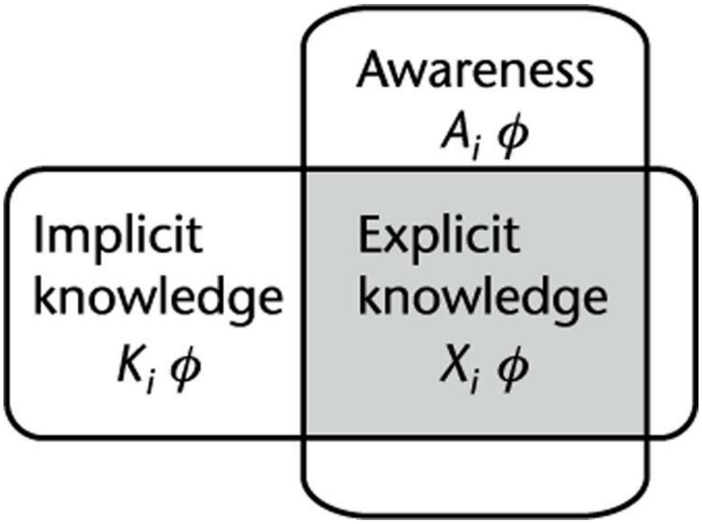


Figure 4.3 Awareness, implicit, and explicit knowledge [32].

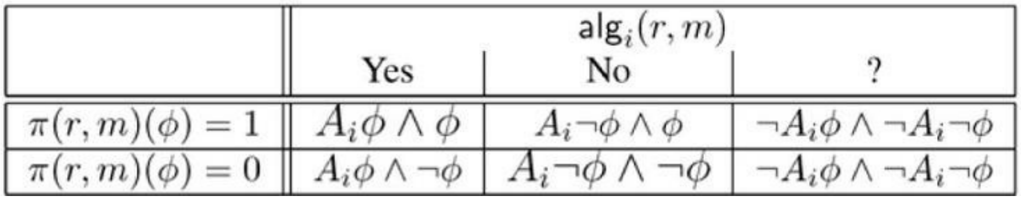
In the logic of general awareness, A_i is a purely syntactic operator as the set of formulas about which the agent is aware at a given point is arbitrary. Our standpoint is quite different, since we are interested in how this set $A_i\phi(r, m)$ is built. Among the different possible interpretations of A_i, we follow Halpern, et al. [19], and say that an agent is aware of a formula ϕ at a given point (r, m) if it is able to compute the truth value of ϕ understood as capturing any constraints like time, memory, and reasoning abilities.

Definition 4.3.2 (Awareness): Let $\mathcal{A}$ be a set agents and let $\mathcal{I}$ be the IABCS representing the joint protocol P of the n agents in the interpreted context (γ, π). The local state of each agent i at point (r, m) includes both a local algorithm $A_i = \mathrm{alg}_i(r, m)$ and local data $l_i = \mathrm{obs}_i(r, m)$. An agent i of $\mathcal{A}$ is aware of ϕ at a given point (r, m) in $\mathcal{I}$ which we denote by $A_i\phi$ iff it is able to compute the truth value of ϕ:

$$(\mathcal{I}, r, m) \models A_i\phi \text{ iff } \phi \in \mathrm{A}_i(\phi, l_i) \neq \text{"?"} \tag{4.16}$$

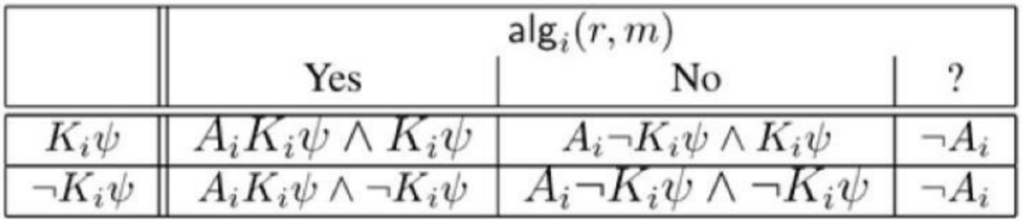
The fact that the algorithm can compute the truth value of ϕ does not mean that this is the correct truth value $\pi(r, m)(\phi)$. The 6 possible configurations are illustrated in Table 4.1: the algorithm answers “Yes” and ϕ holds, “Yes” and $\neg\phi$ holds, and so on. Although to be aware of a formula i's algorithm does not need to answer the correct value, a desirable property for the algorithm is soundness which guarantees that it always gives correct answers. This property is generally easily proved beforehand and encoded in the local state of the agent, so that when its algorithm answers "Yes," ϕ is effectively true at the corresponding point, and then explicitly known by the agent. In Table 4.1, both cases of sound algorithm are highlighted in larger print.

Table 4.1 Awareness and Truth Values in the Set of Points of $\mathcal{R}$



	$\mathrm{alg}_i(r,m)$		
	Yes	No	?
$\pi(r,m)(\phi)=1$	$A_i\phi \wedge \phi$	$A_i\neg\phi \wedge \phi$	$\neg A_i\phi \wedge \neg A_i\neg\phi$
$\pi(r,m)(\phi)=0$	$A_i\phi \wedge \neg\phi$	$A_i\neg\phi \wedge \neg\phi$	$\neg A_i\phi \wedge \neg A_i\neg\phi$

Table 4.2 Awareness, Explicit Knowledge and Implicit Knowledge



	$\mathrm{alg}_i(r,m)$		
	Yes	No	?
$K_i\psi$	$A_iK_i\psi \wedge K_i\psi$	$A_i\neg K_i\psi \wedge K_i\psi$	$\neg A_i$
$\neg K_i\psi$	$A_iK_i\psi \wedge \neg K_i\psi$	$A_i\neg K_i\psi \wedge \neg K_i\psi$	$\neg A_i$

A remarkable case of interest is about the formulas of type $K_i\psi$. Indeed, if we replace ϕ in the discussion above by $K_i\psi$, we obtain Table 4.2. In Table 4.2, $\neg A_i$ is an abbreviation $A_iK_i\psi \wedge K_i\psi$ and $A_i \neg K_i\psi \wedge \neg K_i\psi$ correspond to two cases of explicit knowledge (denoted by $X_i\psi$ in [19] and in this Section above), that are "i explicitly knows that ψ" and "i explicitly does not know that ψ". We can now state our definition of situation awareness.

Definition 4.3.3 (*Situation Awareness*)*:* Let $\mathcal{A}$ be a set of agents and let $\mathcal{I}$ be the IABCS representing the joint protocol $\mathcal{P}$ of the n agents in the interpreted context (γ, π). For an agent i of $\mathcal{A}$, the *situation awareness* at point (r, m) is the set of formulas of $\mathcal{L}(\phi)$ about which the agent i is aware:

$$\mathrm{A}_i(r,m)=\{\phi\in\mathcal{L}(\phi)\,|\,(\mathcal{I},r,m)\vDash A_i\phi\} \tag{4.17}$$

Situation awareness is thus defined in terms of states; for example, in terms of a set of points in the system. For a given agent, the situation awareness is provided by test evaluations on observations about the environment or the objective state of the world. For the analyst, the situation awareness is the set of queries that the situation analysis system is able to answer. Contrary to a particular agent, the analyst is outside the situation and can only make queries about the system it has itself designed. This approach is close to the one followed in [33].

4.3.3 Situation Perception and Comprehension

In this section we step aside from the main discussion and show how the notions of situation perception and situation comprehension, which are popular with psychologists and ergonomists like Endsley [34] could be modeled in the Interpreted Systems framework. A situation is a system or a subsystem, generated by some agents evolving in some context. An interesting question now is, what kind of proto col can be ascribed to the agents? Note that a protocol is a function from local states to actions. What are the kinds of actions that can be performed by the agents? Two different perspectives can be adopted: the planning point of view, where the agents mainly act to reach a certain goal, and the situation analysis point of view, where the agents mainly perform epistemic actions aimed at gathering information, sharing information with other agents, reasoning and consulting their memory. Thus, we conceive situation perception as an epistemic task represented by an epistemic protocol, whose actions involved aimed at changing the information the agent has access to, such as its local state.

Definition 4.3.4 (*Situation Perception*)*:* Let $\mathcal{A}$ be a set of agents and let $\mathcal{I}$ be the IABCS representing the joint protocol $\mathcal{P}$ of the n agents in the interpreted context (γ, π). The situation perception is a mapping between the state of the environment and the local state of the agent Obs: $L_e \rightarrow L_i$ such that $\mathrm{obs}(l_e) = \phi$ with ϕ being a formula of $\mathcal{L}(\phi)$, is Agent i's observation of the environment's local state. Then we have:

$$\mathrm{obs}(r,m) = \langle \mathrm{obs}(r,m-1).\phi \rangle \tag{4.18}$$

where . denotes an append operation.

Situation perception is an external task, as it is a mapping between the environment and the agent.

In return, situation comprehension is an internal task, as it can be seen as a self-mapping on the agent's local state. The situation comprehension is then any action that makes the agent change its set of beliefs. Reasoning, aggregation, deduction, recall, and compression of database are all tasks of situation comprehension.

4.3.4 Situation Analysis

Situation analysis is the process by which the decision maker (or analyst) reaches a state of situation awareness which will then allow him to make decisions.

Definition 4.3.5 (*Situation Analysis*)*:* Let $\mathcal{A}$ be a set of n agents and let $\mathcal{I}$ be the IABCS representing the joint protocol P of the n agents in the interpreted context (γ, π). *Situation analysis* is the process of verifying properties of $\mathcal{I}$. If ϕ_{KT} is a formula of $\mathcal{L}(\phi)$

expressing such a property, including knowledge (K) and Time (T) modalities, then analysis the situation comes to answer the question:

$$(\mathcal{I}, r, m) \models \phi_{KT} \tag{4.19}$$

If ϕ_{KT} is satisfied in $\mathcal{I}$ at (r, m), then the analyst of the system is said to be *aware* of ϕ_{KT}.

The task of deciding if a formula is true in a given model $\mathcal{I}$ is known as model checking. For more details, see [35].

Indeed, the IS semantics allows two views of the systems: the agent's view, which is partial, represented by its local state, and the analyst's view which is total, represented by the global state. The analyst is the person who designed the system; such as assigned protocols to the agents and defined the context, that gave an estimation of the protocol of the environment, defined the working of the world, specified the possible initial states, and introduced some additional constraints. On the other hand, since agents are not perfect reasoners and are thus limited in computation and reasoning abilities, the partial view of the agent is moreover restricted. The analyst, however, is assumed to be a perfect reasoner and is thus ascribed implicit knowledge of the agents. Although it is not a standard view we can further consider the analyst itself as limited in computation and reasoning capabilities, constraint by the algorithm used for its decision procedure, such as model checking (see Table 4.3).

Table 4.3 Different Views Versus Implicit and Explicit Knowledge

	Local view (agent)	Global view (analyst)	God Eye View
Explicit	$X_i\phi$	$\mathbf{X}X_i\phi$	$\mathbf{K}X_i\phi$
Implicit	$K_i\phi$	$\mathbf{X}K_i\phi$	$\mathbf{K}K_i\phi$

If the model checking algorithm is sound and always gives correct answers, then the analyst has explicit knowledge of ϕ_{KT} Note that the local algorithms introduced in Section 4.3.2 could be model checking algorithms, and conversely, the discussion of this latter section holds for the awareness of the analyst.

Illustrations on a Surveillance Scenario

Let us consider the setting of Figure 4.4, in which the behavior and knowledge of two agents 1 and 2 are analyzed according their mutual interaction as well as their interaction with a particular Area of Interest (AOI). Suppose that both agents are able to sense range ρ and bearing θ about the other agent's spatial position that we will denote by $(\hat{\rho}_1, \hat{\theta}_1)$ for the estimated position of Agent 1 (made by Agent 2) and $(\hat{\rho}_2, \hat{\theta}_2)$ for the estimated position of Agent 2 (made by Agent 1).

4.4.1 Situation

Since the kind of interpreted system considered is observational, the local state of each agent is composed by successive observations or measures about the target's position and the state of the environment contains the real targets positions:

$$\begin{aligned}
\mathrm{obs}_1(r,m) &= \left\langle \left(\hat{\rho}_2(0), \hat{\theta}_2(0)\right); \ldots, \left(\hat{\rho}_2(m), \hat{\theta}_2(m)\right) \right\rangle \\
\mathrm{obs}_2(r,m) &= \left\langle \left(\hat{\rho}_1(0), \hat{\theta}_1(0)\right); \ldots, \left(\hat{\rho}_1(m), \hat{\theta}_1(m)\right) \right\rangle \\
r_e(m) &= \left\langle \left(\hat{\rho}_1(0), \hat{\theta}_1(0) \hat{\rho}_2(0), \hat{\theta}_2(0)\right), \ldots \right\rangle
\end{aligned}$$

Thus, a global state will be of the form:

$$s = \left[\left(\hat{\rho}_1, \hat{\theta}_1\right); \left(\hat{\rho}_2, \hat{\theta}_2\right); \left(\rho_1, \theta_1, \rho_2, \theta_2\right)\right] \tag{4.20}$$

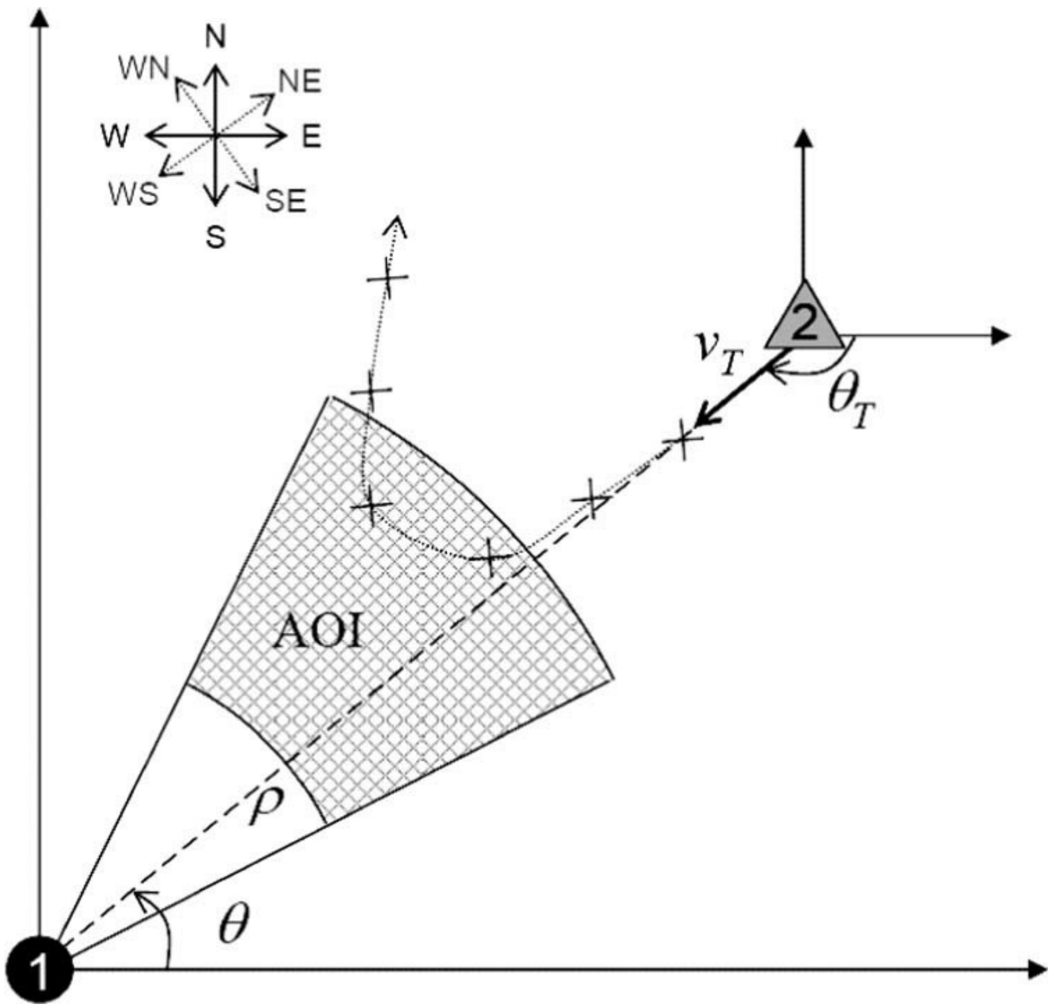


Figure 4.4 Scheme of a surveillance scenario.

Moreover, both agents are able to move in 8 possible directions North (N), North-West (NW), West (W), West-South (WS), South (S), South-East (SE), East (E) and East-North (EN). Here are the possible actions for both agents:

$$\begin{aligned} ACT_1 = ACT_2 &= \{Move_N;\ldots,Move_{EN};\Lambda\} \\ &= \{N,\ldots;EN;\Lambda\} \end{aligned} \tag{4.21}$$

Since the agents are always sensing, no action Sense is included in the sets ACT_i, i =1, 2, and this for simplifying the expressions. A joint action, a, is composed of one action of agent 1 and one action of agent 2, where ACT $\_i$ is the SET of actions of agent i. Thus, the possible joint actions for both agents are of the form:

$$a = (Move_x; Move_y) \tag{4.22}$$

with $(x, y) \in ACT_1 \times ACT_2$. For example, a_1 =(N; S), meaning that in round 1, Agent 1 moves in direction North and Agent 2 moves in direction South. For simplicity, we assume that Agent 1 is static and only performs the null action Λ. The protocol of Agent

1 consists simply in doing nothing except if it perceives that Agent 2 is in the AOI, in which case it will send the information to an agent designated agent of its hierarchy. The protocol of Agent 2 consists in moving toward Agent 1 until it perceives it is too close, in which case it must make a u-turn.

The uncertainty in general can be modeled by environment actions. For instance, the measures of both agents may be affected by sensing errors like $\varepsilon_{\rho}^{1} = \pm 10$ for the range estimated by Agent 1, $\varepsilon_{p}^{2} = \pm 20$ for the range estimated by Agent 2 and $\varepsilon_{\theta} = \pm 1$ for the bearing estimated by both agents. Thus the environment is able to perform one joint action of the form:

$$a_e = \left\{\left(\varepsilon_r^1; \varepsilon_r^2; \varepsilon_\theta; \varepsilon_\theta\right)\right\} \tag{4.23}$$

The protocol of the environment simply selects randomly one action of ACT_e in each round, affecting thus the measures of both agents. For instance, $\varepsilon_e = (0; 0; 0; 0)$ means that no error occurred in the observations of both agents, $\varepsilon_e = (+10; 0; 0; -1)$ means that an error of +10 occurred in the measured range by Agent 1 and -1 in the measured bearing of Agent 2.

The situation is represented by the execution of the joint protocol of 1 and 2 in the given context, as illustrated in Figure 4.5.

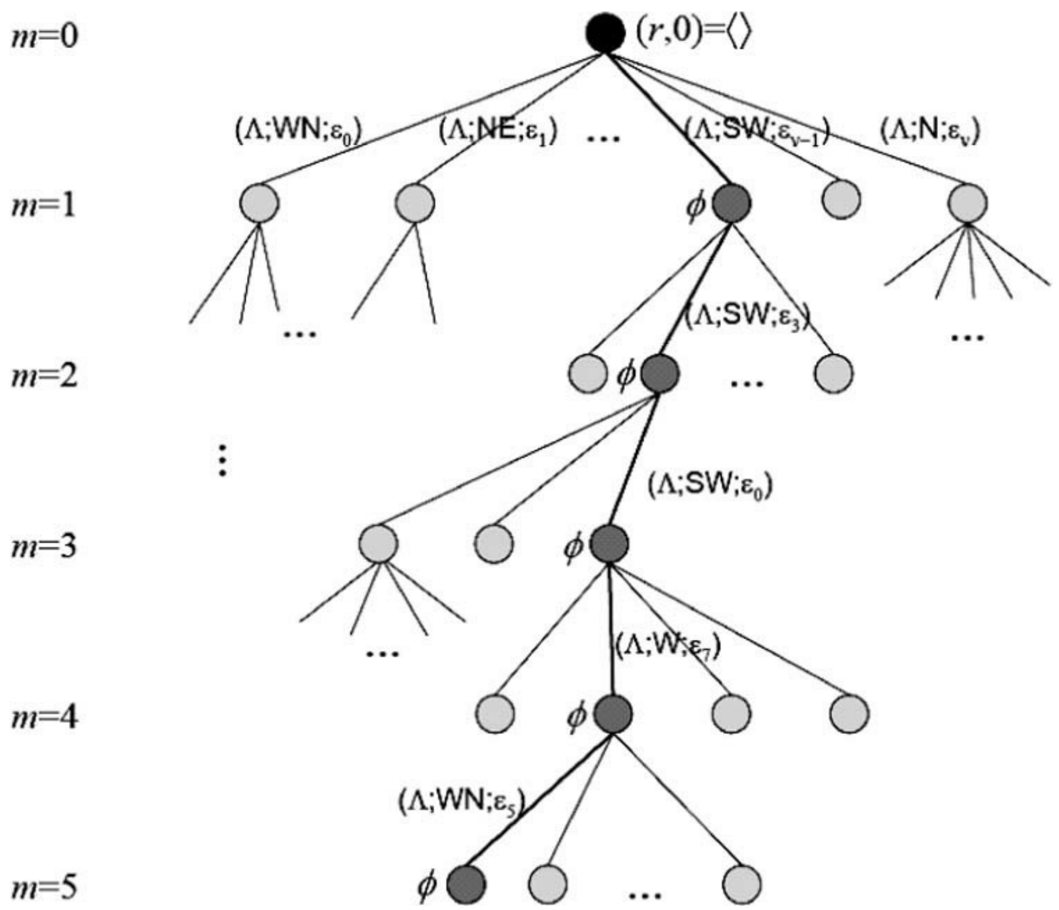


Figure 4.5 System generated by the joint protocol of agents 1 and 2 and the environment which represents the target's motion.

4.4.2 Situation Awareness

We consider the basic set of propositions $\Phi = \{\phi_\rho, \phi_\theta\}$ with ϕ_ρ equal to "the range of Agent 2 crosses the AOI," and ϕ_θ equal to the "bearing of Agent 2 crosses the AOI." The formula $\phi_{\mathrm{AOI}} = \phi_\rho \wedge \phi_\theta$ is then Agent 2 is in the AOI. We have this for example that:

$$\phi_\rho = \{(r,m) | r_e(m) \models \phi_\rho\} \tag{4.24}$$

is the set of points in which ϕ_ρ holds, *i.e.* the set of states of the system in which the target is the AOI.

We assume that each agent has an algorithm $\mathrm{alg}_i(r, m)$ in its local state $r_i(m)$ to decide if a particular formula ϕ is true. For example, Agent 1 may evaluate that ϕ_ρ holds if the range it measured about Agent $2\hat{\rho}_2$ lies in the range interval defining the AOI. Since the evaluation of such a basic proposition could be seen as immediate, according to Definition 4.3.2, Agent 1 is aware of ϕ_ρ, at any point in $\mathcal{I}$. The situation awareness for Agent 1 will be then $A1(r, m) = \{\phi_\rho\}$, for all (r, m) or in term of states, $A_1(r, m) = \|\phi_\rho\|$

which is the set of points in which (ϕ_r holds. However, the task may be not as simpler for formulas like ϕ_{us} is equal to Agent 2 is coming toward Agent 1. Indeed, such an evaluation involves more computation, such as a test between the last and second last observation to be able to say whether or not the measure at $m - 1$ is greater or not than the measure at m. In this case, we may have for some (r, m), $(\mathcal{I}, r, m) \models \neg A_1\phi_r \wedge \neg A_1\phi_r$.

4.4.3 Belief, Revision, and Update

Remember from Section 4.2.2 that if algorithm alg_i returns "Yes" that means that the agent i is aware of ϕ_ρ, and not that ϕ_ρ is true. The algorithm may be wrong. If the algorithm is sound, meaning that it always returns the correct answer, then the agent is fully reliable, and all pieces of information coming from it can be taken as knowledge. However, the soundness property may not be provable in some cases and thus, the notion of awareness we model is rather close to belief.

In order to model belief, we assume that each agent is able to assign a belief space (in the sense of Shafer) to each point of the system, $P\_{i(r, m)} = (S_{(i, r, m)}, \text{Bel}_{(i, r, m)})$, with $\text{Bel}_{(i, r, m)}(A)$ being the belief degree the agent i has at time m in run r about the event (or formula) A. If the considered events are ϕ_ρ and ϕ_θ and if we denote by $m_{(i, r, m)}$, the basic Probability Assignment (BPA) corresponding to $\text{Bel}_{(i, r, m)}$, then we could have for example $m_{(1, r, 3)}(\phi_\rho) = 0.7$ and $m_{(1\ r, 3)}(\top) = 0.3$ for Agent 1 and $m_{(2, r, 3)}(\phi_\theta) = 0.6$ and $m_{(1, r, 3)}(\top) = 0.4$ for Agent 2. Using Dempster's rule, we then obtain $m_{(12, r, 3)}(\phi_{\text{AOI}}) = 0.42$, the joint degree of belief of Agents 1 and 2 that Agent 2 is in the AOI. If the agents are cooperative, then the degree of belief could serve as a criterion for exchanging information.

Another example of use of this plausibility measure is the update of $\text{Bel}_{(i, r, m)}(\phi)$ in light of new pieces of information. In this case, a conditional rule could be used leading for example to $\text{Pl}_{(1, r, m+1)}(\phi_{\text{AOI}}|\phi_r) = \text{Pl}_{(1, r, m)}(\phi_{\text{AOI}} \wedge \phi_r)/\text{Pl}_{(1, r, m)}(\phi_\rho)$, using Dempster's rule of conditioning, where Pl is the plausibility measure in Shafer's sense.

4.4.4 Situation Analysis

Three kinds of queries can be used to analyze the situation:

- Queries about truth: "Does ϕ hold at point (r, m) in $\mathcal{I}$?" formally written $(\mathcal{I}, r, m) \models \phi$?"
- Queries about knowledge: "Does i know ϕ hold at point (r, m) in $\mathcal{I}$?" formally written $(\mathcal{I}, r, m) \models K_i\phi$? but also "Does i know that j knows that ϕ hold at point (r, m) in $\mathcal{I}$?" (formally written $(\mathcal{I}, r, m) \models K_iK_j\phi?$), and queries about group knowledge, such

as, “Does the group G of agents has distributed knowledge of ϕ at point (r, m) in $\mathcal{I}$?” (formally written $(\mathcal{I}, r, m) \models D_G\phi$?), or “Is ϕ common knowledge among members of G at point (r, m) in $\mathcal{I}$?” (formally written $(\mathcal{I}, r, m) \models C_G\phi$?)

- Queries about time: “Does ϕ eventually hold in run r?” formally written $(\mathcal{I}, r) \models \mathbf{F}\phi$? (Linear Temporal Logic), but also “Does ϕ hold for all sequences r from s?” formally written $(\mathcal{I}, s) \models \mathbf{A}\phi$? (Computation Tree Logic).

In general, queries are combinations of these three types, for example “Do both agents 1 and 2 always know that $\neg\phi_{\mathrm{AOI}}$ holds in $\mathcal{I}$?”, formally written $(\mathcal{I}, s) \models \mathbf{A}E_{12}\phi$?, where $\neg\phi_{\mathrm{AOI}}$ means that Agent 1 is not in the AOI. In case of a negative answer, either the surveillance protocol or equipment may have to be modified.

Conclusions

In this chapter, we extended a discussion initiated in [2] where the interpreted semantics (IS) had been presented as a potential blueprint for the situation analysis purpose. Our hypothesis for further research is then:

A formal situation analysis model is an interpreted algorithmic belief change system.

In general, IS models implicit notions of knowledge and the analysis is done by verifying some epistemic and temporal properties by the analyst, through some decision procedure. Although more research needs to be done in the direction, the efficient model checking procedure is particularly well developed for the IS semantics. Our definition of a situation is closely related to the one of a state transition system.

The algorithmic characteristic of an IS allows one to distinguish between implicit and explicit knowledge and thus allows one to compute what an agent can know given limited resources, such as time, computation capability, and expertise. This notion provides a means of formally defining situation awareness.

Introducing plausibility measures in interpreted systems, or interpreted plausibility systems, allows one to introduce the notion of belief in the model. The generality of plausibility measures gives the advantage of using most of quantitative representations of uncertainty, such as probabilities, belief functions, or possibilities. Vagueness can also be represented, but this will be detailed in a further work.

Restricting interpreted plausibility systems to satisfy some constraints leads to interpreted belief change systems and allows one to represent belief changes, updates, and revisions. This modelization, while compatible with the implicit representation of knowledge, makes an explicit link with the observations and the environment. The local process of perceiving the environment, computing and changing beliefs based on new pieces of information, is seen as the situation assessment process.

rences

[1] Maupin, P., and A-L. Jousselme, “Interpreted Systems for Situation Analysis,” *International Conference on Information Fusion*, 2007.

[2] Maupin, P. and A-L. Jousselme, “A General Algebraic Framework for Situation Analysis,” *International Conference on Information Fusion*, 2005.

[3] Steinberg, A. N., and C. L. Bowman, “Revisions to the JDL Data Fusion Model,” in *The Handbook of Multisensor Data Fusion*, The Electrical Engineering and Applied Signal Processing Series, D. L. Hall and J. Llinas (eds.), Boca Raton, FL: CRC Press, 2001.

[4] Bowman, C. L., A. N. Steinberg, “A Systems Engineering Approach for Implementing Data Fusion Systems,” in *The Handbook of Multisensor Data Fusion*, The Electrical Engineering and Applied Signal Processing Series, D. L. Hall and J. Llinas (eds), pp. 16-1–16-39, Boca Raton, FL CRC Press, 2001.

[5] Llinas, J., et al., “Revisions and Extension to the JDL Data Fusion Model II,” *International Conference on Information Fusion*, 2004.

[6] Matheus, C. J., M. M. Kokar, and K. Baclawski, “A Core Ontology for Situation Awareness,” *International Conference on Information Fusion*, 2003.

[7] Kokar, M. M., J. A. Tomasik, and J. Weyman, “Formalizing Classes of Information Fusion Systems,” *Information Fusion*, Vol. 5, No. 3, 2004, pp. 189–202.

[8] Klir, G. J., and M. J. Wierman, *Uncertainty-Based Information*, Vol. 15 of *Studies in Fuzzi-ness and Soft Computing*, *2nd Edition*, Heidelberg, Germany: Physica-Verlag, 1999.

[9] Halpern, J. Y. *Reasoning about Uncertainty*, Cambridge, MA: The MIT Press, 2003.

10] Thorsen, S. N., and M. E. Oxley, “Comparing Fusors within a Category of Fusors,” *International Conference on Information Fusion*, Sweden, 2004.

11] Kokar, M., and J. Tomasik, “Towards a Formal Theory of Sensor/Data Fusion,” Tech. Rep. COE-ECEMMK-1/94, Northeastern University, Boston, 1994

12] Kokar, M. M., J. A. Tomasik, and J. Weyman, “Data vs. Decision Fusion in the Category Theory Framework,” *International Conference on Information Fusion*, 2001.

13] Aumann, R. J., “Agreeing to Disagree,” *The Annals of Statistics*, Vol. 4, 1976, pp. 1236–1239.

14] Jousselme, A.-L., P. Maupin, C. Garion, L. Cholvy, and C. Saurel, “Situation Awareness and Ability in Coalitions,” *International Conference on Information Fusion*, 2007.

15] Shafer, G. *The Art of Causal Conjecture*, MIT Press Artifcial Intelligence Series, 1996.

16] McAllester, D., *Semantics*. Lecture notes, MIT 1993.

17] Rasmussen, E., *Games and Information: An Introduction to Game Theory*, *4th Edition*, Blackwell Publishing, 2006.

18] Halpern, J. Y., and Y. Moses, “Knowledge and Common Knowledge in a Distributed Environment,” *Journal of the Association for Computing Machinery*, Vol. 37, 1990, pp. 549–587.

19] Halpern, J. Y., Y. Moses, and M. Y. Vardi, “Algorithmic Knowledge,” *Proceedings of the 5th Conference on Theoretical Aspects of Reasoning about Knowledge* (*TARK’94*), Morgan Kaufmann, 1994, pp. 255–266.

20] Friedman, N., and J. Y. Halpern, “Modeling Belief in Dynamic Systems, Part I: Foundations,” *Artificial Intelligence*, Vol. 95, 1997, pp. 257–316.

21] Friedman, N., and J. Y. Halpern, “Plausibility Measures: A User’s Guide,” *Conference on Uncertainty in Artificial Intelligence* (*UAI*), 1995, pp. 175–184.

22] Shafer, G. *A Mathematical Theory of Evidence*, Princeton University Press, 1976.

23] Zadeh, L. A., “Fuzzy Sets as a Basis for a Theory of Possibility,” *Fuzzy Sets and Systems*, Vol. 1, 1978, pp. 3–28.

24] Friedman, N., and J. Y. Halpern, “Modeling Belief in Dynamic Systems, Part II: Revision and Update,” *Journal of Artificial Intelligence Research*, Vol. 10, 1999, pp. 117–167.

25] Boutilier, C., N. Friedman, and J. Y. Halpern, “Belief Revision with Unreliable Observations,” *National Conference on Artificial Intelligence*, 127–134, 1998.

26] Baclawski, K., M. K. Kokar, C. J. Matheus, J. Letkowski, and M. Malczewski, “Formal-ization of Situation Awareness,” *OOPSLA Workshop on Behavioral Semantics*, 2002, pp. 1–15.

27] Parikh, R., and R. Ramanujam, “Distributed Processes and the Logic of Knowledge,” pp. 256–268, in *Proceedings of the Conference on Logics of Programs*, R. Parikh (ed), Vol. 193 of *Lecture Notes in Computer Science*, Brooklyn, NY: Springer-Verlag, 1985.

28] Chandy, K. M., and J. Misra, “How Processes Learn,” *Distributed Computing*, Vol. 1, 1986, pp. 40–52.

29] Fagin, R., J. Y. Halpern, Y. Moses, and M. Y. Vardi, *Reasoning About Knowledge*, The MIT Press, Cambridge, MA, 2003.

30] Pew, R. W., “The State of Situation Awareness Measurement; Heading Toward the Next Century,” pp. 33–50, in *Situation Awareness Analysis and Measurement*, M. Endsley and D. Garland (eds), Mahwah, NJ: Lawrence Erlbaum Associates, 2000.

31] Fagin, R., and J. Y. Halpern, “Belief, Awareness, and Limited Reasoning,” *Artificial Intelligence*, Vol. 34, 1988, pp. 39–76.

32] Konolige, K., “What Awareness Isn’t: a Sentential View of Implicit and Explicit Belief,” *Journal of Symbolic Logic*, Vol. 53, 1988, pp. 667–668.

33] Ramanujam, R., “View-Based Explicit Knowledge,” *Annals of Pure and Applied Logic*, Vol. 96, 1999, pp. 343–368.

34] Endsley, M. R., “Theoretical Underpinnings of Situation Awareness: A Critical Review,” *Situation Awareness Analysis and Measurement*, M. R. Endsely and D. J. Garland (eds), Mahwah, NJ: Lawrence Erlbaum Associates, 2000.

35] van der Meyden, R., “Common Knowledge and Update in Finite Environments,” *Information and Computation*, Vol. 140, 1998, pp. 115–157.

---

1. This chapter is based on material previously published in [1].

2. S5 is one of five systems of model logic (defines the axioms for knowledge) proposed by Clarence Irving Lewis and Cooper Harold Langford in 1932.

3. K45 amd KD45 are systems of model logic proposed by C. I. Lewis and C. H. Langford in 1932.

Part II
Distributed Information Fusion and Management

CHAPTER 5

The Role of Information Management to Support High-Level Fusion

Mark H. Linderman (United States), Greg Chase (Austraila), Anne-Claire Boury-Brisset (Canada), Justin Nevitt (United States), Justin Henley (United Kingdom), Robert Read (New Zealand), Stephen Russell (United States), and Paul Hyden (United States)

High-level information fusion (HLIF) processes are critically dependent on quality information sources in order to generate meaningful outcomes. Quality information necessitates the right combination of people, processes, and tools to form an effective information space. This critical task is generally known as Information Management (IM). This chapter overviews IM, discusses IM goals and challenges in a coalition environment, and presents an IM model. The model provides an abstract representation of the important artifacts, actors and services that are required to form an effective IM capability. After a brief exposition of the model, IM best practices are considered. IM issues of particular interest to fusion scenarios are identified, including the information lifecycle that exists between the sensors and information exploitation (IX) systems, interoperability issues, information context, partially structured information, IM as a service, IM workflows, and IM from an agent perspective.

Introduction: What Is Information Management and Why Do We Care?

There are numerous definitions of IM, many of which apply generic management terms to the information domain. For example, the *U.S. Office of Management and Budget (OMB) Circular A-130* [1] defines information management as:

> the planning, budgeting, manipulating, and controlling of information throughout its life cycle.

In this chapter, we will focus on information manipulation and control and other facets that are significant. We define IM as:

> a set of intentional activities to maximize the value of information to support the objectives of the enterprise.

These activities include the promulgation of standards, control of information, administrative activities, and the acquisition, dissemination, manipulation, persistence, and disposal of information.

The goal of IM is to maximize the effectiveness of an enterprise, such as a military operation, by maximizing its ability to act upon information that is produced or consumed within the enterprise. There are several means by which this can be accomplished:

- Reducing barriers to effective information use by providing notification, mediation, access control, and persistence services;
- Providing an information space wherein information is managed directly, rather than delegating information management responsibilities to applications that produce and consume information;
- Focusing on consumer needs rather than producer preferences to ensure that information can be effectively used;
- Providing tools to assess information quality and suitability;
- Exploiting producer-provided characterization of information to support automated management of information.

If these means can be accomplished, it can make edge-user[1] applications less complicated and enable the enterprise to be more agile to adapt to changing requirements and environments. One approach is the efficient brokering of information through common information services so that consumers can obtain the information they need without having to discover its producers. Simple interfaces for “universal” IM services, preferably ones that are operating system and programming language independent, can reduce the burden of managing information within applications through consistent semantics and reuse. Of course, the representation of information is critical to its proper handling. In particular, metadata is crucial to support management processes such as prioritization and filtering. Support for semantic annotations in metadata can also potentially reduce ambiguity and the inclusion of predicates[2] can assist in process automation [2].

Agile control of the information flow throughout an enterprise enables it to make efficient use of available network resources, change access control policies in federated/coalition environments, or simply enable applications access to information from new sources. Accomplishing this requires a manageable number of control points where the behavior of the Command and Control (C2) system of systems can be influenced based upon policy.

There are several best practices that help achieve the goals of information management. Organizations will greatly improve the interoperability and agility of their future net-centric fusion (and C2) systems by:

1. Adopting dedicated information management infrastructures;
2. “Packaging” information for dissemination and management;
3. Creating simple, ubiquitous services that are independent of operating system and programming language;
4. Using a common syntax and semantics for common information attributes such as location, time, and subject.

If appropriately employed, these best practices can reduce the complexity of information fusion and C2 systems, allow for effective control of the information space, and facilitate more effective sharing of information.

Model of Information Management

In order to discuss the role of IM in support of information fusion, we have defined a framework, or model, that describes the concepts, participants, and actions performed in managing information. Previously, we defined IM *as a set of intentional activities to maximize the value of information to support the objectives of the enterprise*.

There are several aspects of this definition that bear upon the structure of the IM model. The first is the word *activities*. Managing is not an end state or a frame of mind, it is something one does. Therefore, the model enumerates and describes the activities that comprise the verbs of management. Implicit in the definition are actors that interact with the managed information environment by either managing information or by sharing managed information. Finally, there is the information itself that, together with the actors, comprise the nouns of the IM model.

The next word that bears scrutiny is *intentional*. Arguably, one might consider unintentional (de facto) activities, but we emphasize intention precisely because there appears to be a prevalent attitude that IM is “everywhere and nowhere.” Our premise is that enterprises are not maximizing the value of the information they possess because IM is either viewed as “someone else’s responsibility,” or is delegated down to individuals or organizations without sufficient authority or resources to perform it effectively.

Information management is a *set* of intentional activities that act in concert with one another to achieve a desired result. If these activities operate independently of one another, the results may be minimally acceptable but fall short of optimal. For example, access control and information transformation may act together to sanitize information in accordance with disclosure policies prior to sharing it with a given consumer. Acting independently, the access control activity may be forced to decide between passing all or none of the information. Therefore, while passing nothing might adhere to the prevailing security policy, operational effectiveness may suffer because finer-grained enforcement was not an option.

The definition explicitly states that the role of information is to *support the enterprise*. Information seldom has a value in itself; rather, it is exploited to achieve a number of things, including situation understanding and decision making. The value of information is not an intrinsic quality, but is dependent upon how it is to be used. To effectively perform value-based management, the managers and ideally the infrastructure must be able to estimate the potential utility of information. To do this, it is important to understand the business processes supported by that information and the relative importance of those processes. It is desirable to prioritize resource allocation at a high level, such as the process level, and to have the priorities of the inputs to that process (i.e., the supporting information) be automatically derived from it.

The IM model presented in Figure 5.1 is an abstract representation of the essential activities of IM, and the actors that interact within a managed information environment to achieve their objectives. Another model of IM [3] served as a departure point for the model we present. Our model differs in that it:

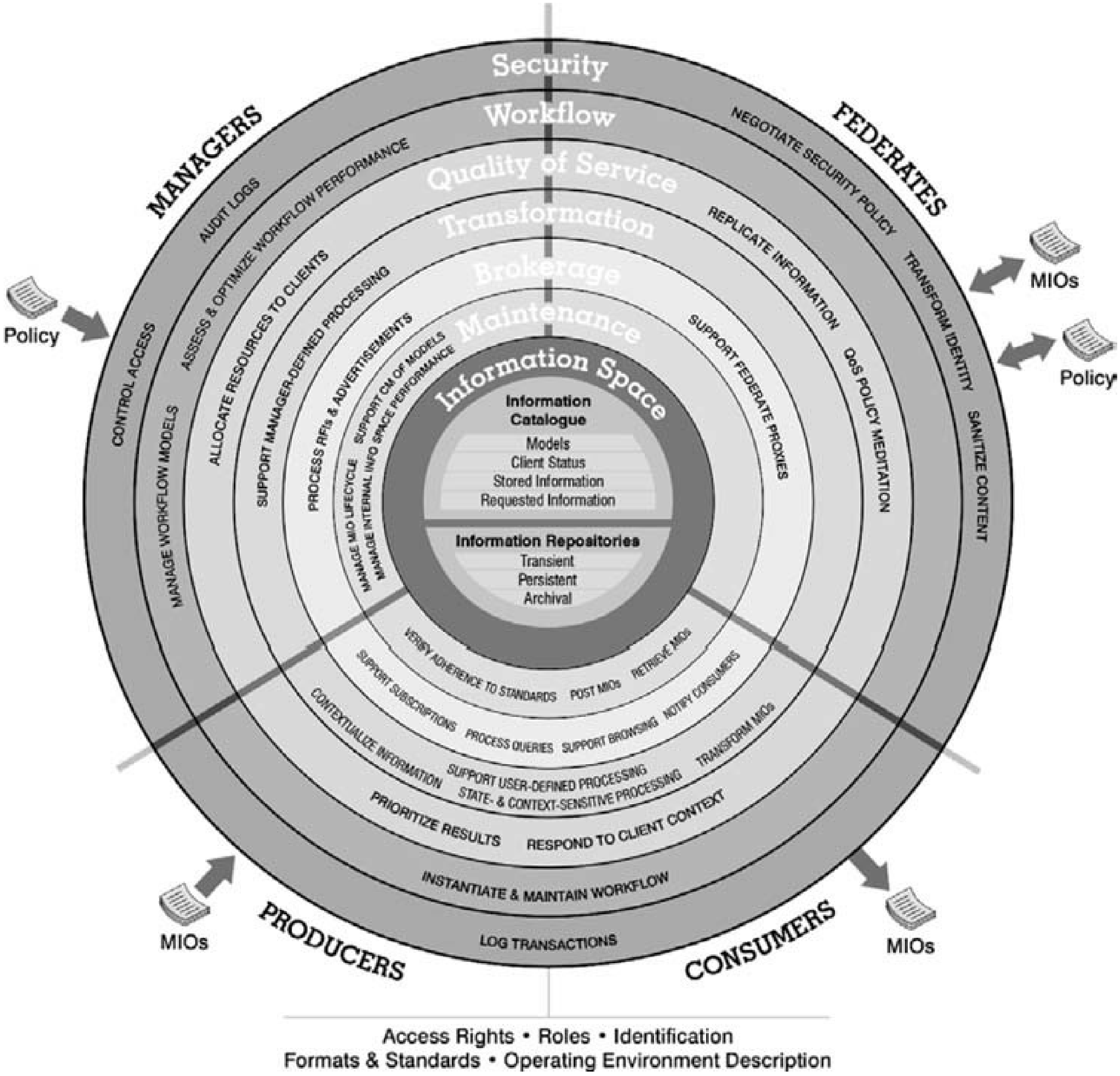


Figure 5.1 Information management model.

- Makes a clear distinction between actors and the services they employ;
- Addresses federated information spaces to support diverse communities (such as

coalitions);

- Organizes activities according to type of service (i.e., service layer) rather than by process steps (e.g., task, submit, consume);
- Is more explicit on maintenance and security activities, and significantly enhances transformation services; and
- Supports flexible workflow rather than implying a specific information process through which all information must pass from production to consumption.

The model is described in detail elsewhere [4]; here we just outline the four major components that comprise the model: managed information objects (MIOs), actors, service layers, and the information space. We then consider the utility of the model itself.

5.2.1 Managed Information Objects

Informal words, such as "document" and "object," are often used to describe a quantum of information. The managers of information and IM infrastructures only want to know enough about a quantum of information to do their jobs; as they do not need to know everything about it. They need a characterization of the information. Furthermore, given the volume of information a typical enterprise uses, it is important that the characterization be readily available.

In our model of IM, a quantum of managed information is called a *managed information object* (MIO). A MIO comprises a payload and metadata that characterize the object such as topic, time, and location. It is desirable that all of the information needed for making management decisions, such as content-based dissemination, be present in or referenced within the metadata in a form that permits efficient processing.

An important element of characterization is the concept of *type*. The type of an object, such as satellite imagery, is useful for determining how the information should be characterized and for setting policy on its appropriate use. Type is dis-tinct[3] from format because type relates to the purpose of the information, whereas the format relates to its encoding. While format is essential for processing or presenting the information, type is more important for determining management of the information.

The model does not address how metadata standards evolve or the extent of their universality, but metadata standards to characterize information are essential to effective information sharing. The Dublin Core [5] is an example of a high-level universal standard. This has been adapted to military applications by efforts like the Department of Defense Discovery Metadata Specification [6]. Lower-level standards may be developed with organizations or communities of interest.

5.2.2 Actors

People, or autonomous agents, interact with the managed information environment by producing and consuming information or by managing it. Federated information spaces that interact with a given information space are also considered actors from the perspective of that information space. This is shown in Figure 5.2, with a summary of the elements in Table 5.1.

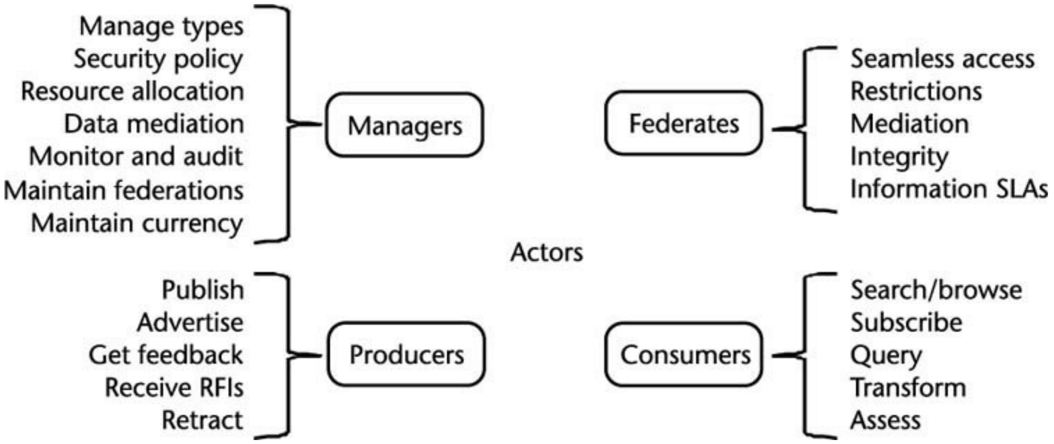


Figure 5.2 Information actors

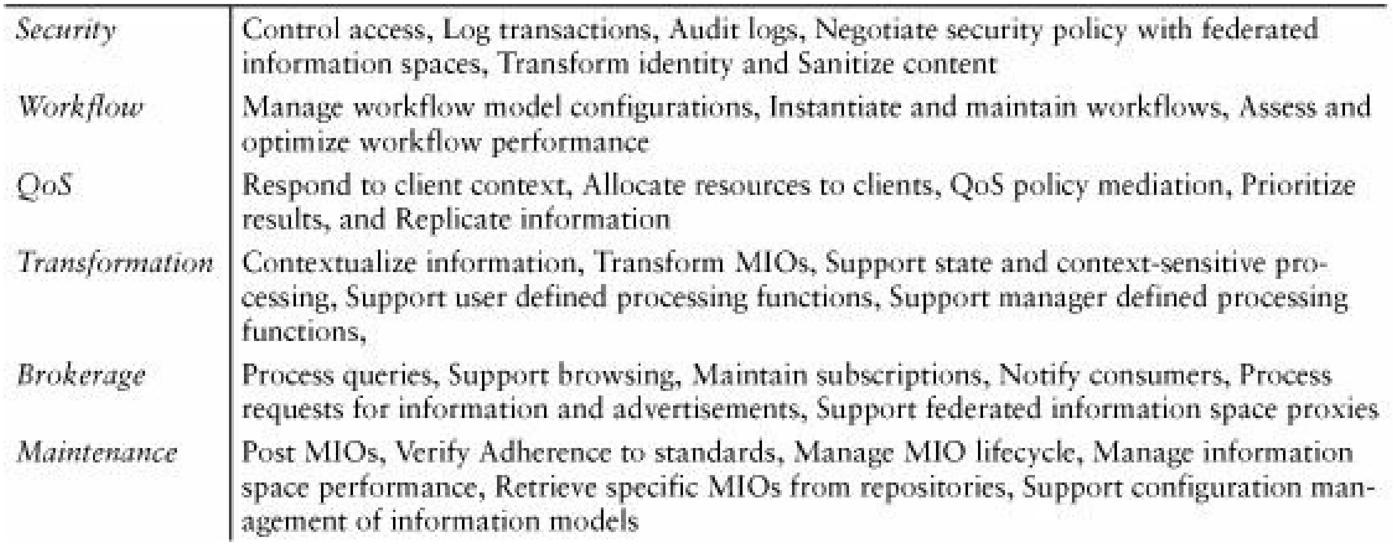
Table 5.1 Elements of Information Actors

Consumers	**Search/Browse** the information space
	Subscribe to new information
	Query the information space to retrieve previously published information
	Transformation, as a part of browsing, searching or query user may filter, set priorities and transform the information prior to delivery
	Provide Feedback on the effectiveness of the information to the information producers
Producers	**Publish** information to the information space and provide appropriate metadata
	Advertise their capabilities and products that may be produced if requested
	Get Feedback to enable them to optimize their service
	Receive RFIs to service new requests
	Retract information that is subsequently deemed to be incorrect. With the exception of a security spill retracted information is not removed from the repository, it is just marked as retracted
Managers	**Manage Types** and the relationships between types and how they support the goals of the enterprise, monitor the introduction of new types:
	Security Policy, including the balance between security concerns and operational effectiveness. The policy may change as a result of the operational context
	Resource Allocation must be managed to provide an optimal outcome
	Data Mediation between evolving information standards and legacy systems
	Monitor & Audit functions must be appropriate for the task
	Maintain Federations with external information spaces
	Maintain Currency by removing or achieving old information, manage the movement of information through the transient, persistent and archive catalogues
Federates	**Seamless Access** is provided to consumers accessing external information spaces
	Restrictions on usage may need to be enforced as the trust relationships will be different for external consumers
	Mediation of inconsistent data standards
	Integrity of information provided by a federated partner
	Information service-level agreement

5.2.3 Service Layers

A set of service layers is defined to perform specific IM activities. The services layers defined by the model are: security, workflow, quality of service (QoS), transformation, brokerage, and maintenance, as shown in Table 5.2.

Table 5.2 Service Layers



Security	Control access, Log transactions, Audit logs, Negotiate security policy with federated information spaces, Transform identity and Sanitize content
Workflow	Manage workflow model configurations, Instantiate and maintain workflows, Assess and optimize workflow performance
QoS	Respond to client context, Allocate resources to clients, QoS policy mediation, Prioritize results, and Replicate information
Transformation	Contextualize information, Transform MIOs, Support state and context-sensitive processing, Support user defined processing functions, Support manager defined processing functions,
Brokerage	Process queries, Support browsing, Maintain subscriptions, Notify consumers, Process requests for information and advertisements, Support federated information space proxies
Maintenance	Post MIOs, Verify Adherence to standards, Manage MIO lifecycle, Manage information space performance, Retrieve specific MIOs from repositories, Support configuration management of information models

5.2.4 Information Spaces

The information space is a collection of catalogues and repositories that provide common functions for storage, retrieval, and lifecycle management. The information space operates on our managed information objects and forms the heart of our managed information environment.

5.2.5 Utility of the Information Management (IM) Model

The model can assist in lifting the quality of dialogue about IM from generalizations and vaguely stated capabilities, to thoughtful consideration of specific activities and appropriate assignment of resources, roles, and responsibilities. Perhaps most importantly, we hope to convey that IM should be an enterprise-level challenge; one that is too important to ignore or relegate to people that are insufficiently resourced, lacking authority, or lacking appreciation of enterprise-level objectives.

The IM model is equally applicable to a range of IM approaches, from manual information exchange processes to wholly automated systems. The model does not prescribe any particular architectural style, such as publish and subscribe, or service-oriented architecture. The IM model is designed to assist in identifying the IM requirements that might need to be supported and implemented following any architectural approach.

We recognize that any IM model needs to include the following critical elements: actors and their roles, processes, and the implementation approach. That being said, given the predominant role of digital content and electronic communication networks, much of the discussion to follow naturally fits within the digital domain. As such, while the model refers to the *actors, producer* and *consumer*, in a manner that could equally refer to persons or computer applications, an instantiation within the digital domain will typically assume that producers and consumers are computer applications that may or may not be acting on behalf of a person.

Instantiations of the model can be constructed upon a variety of architectures from single-tiered, system high operating environments to highly distributed implementations with sparsely and intermittently connected producers and consumers spanning multiple security levels. Some of these may be easier to manage, control, audit, and archive, while the others may be more resilient and scalable.

Information Management Challenges in a Coalition Environment

While we aspire to coherent standards across organizations or national systems, coalition environments present challenges to standardization that are beyond the ability of any one participant to control. Rather than assume away the problem by envisioning a future state where all potential coalition partner's systems are miraculously interoperable, it is more productive to identify challenging realities—stated below as assumptions—and work to ameliorate them. The following assumptions are critical:

1. *Acquisition cannot be standardized or synchronized.* Perhaps a stronger statement of this is that acquisition should not be standardized or synchronized, if the result of doing so is inability to interact with new coalition partners, unacceptable delays in fielding capabilities, or unacceptable high costs for fielding or supporting capabilities.
2. *Universal data standards are unrealistic*. While progress can certainly be made on broad agreements of both high level data standards, such as standard upper-order ontologies, and within domains, such as Command and Control Information Exchange Data Model (C2IEDM), insistence upon a single, universal model is counterproductive. The Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM), which supersedes C2IEDM, is an example of a community-developed model that has become too complex. The authors of the JC3IEDM model are now working to make parts of the model optional so that practical, compliant applications can be more readily developed.
3. *Data standards version skew is the norm not the exception*. While this assumption is primarily a by-product of asynchronous acquisition, it is worth explicitly noting because it has specific implications on the IM environment for coalition operations. If we assume that different participants will migrate to newer standards at different rates, then it is incumbent upon information managers to deal with the problem.
4. *Coalition information spaces must adapt quickly to new partners and processes*. Similar to Assumption 3, this assumption affects where adaptation is likely to take place. Given current software technology, especially as seen in complex military applications, it is unrealistic to assume that applications will be able to adapt to changing processes, products or partners. Today, adaptation is carried out by changing the behavior of people through patchwork fixes and workarounds. From an IM perspective, this situation can be ameliorated by providing managed information tailoring environments and tools to support processes that can be modified by policy rather than with software that cannot be modified.

All of these assumptions imply the need for a flexible information environment. The concept of an information space itself, provided access to it is achieved through a fairly universal set of activities/services (as suggested in the model), is a key enabler of flexibility. New implementations of the information space can emerge without breaking its interactions with producers, consumers, managers, and federated information spaces. Similarly, producers and consumers may evolve without changing the character of the information space. The *brokerage layer* permits information flows to change as needed to allow the incorporation of new partners and processes and to allow the introduction of transformation services in between producers and consumers that can mediate version skew and bridge (albeit, imperfectly) disparate data standards. The *workflow layer* can support new processes necessitated by new partners and operational imperatives. Finally, the *security layer* can allow effective information sharing with new partners and facilitate responsive policy-based control of information rather than hard-coded policies based upon outdated assumptions about the operating environment.

IM is applicable to many military domains, such as coalition logistics, coalition command, and control, including the exchange of the common operational picture, operations other-than-war, and coalition fires. It is also applicable to many different time scales from strategic (months to years), operational (days to weeks), and tactical (seconds to minutes to hours) for users.

In addition to the assumptions introduced above, there exist other unique requirements of IM within a coalition context. Instantiation of the model within a coalition context requires special considerations, particularly with respect to security, auditing, mediation, and replication. In general, IM is constrained in a coalition environment. Policies allowable in a coalition environment may be more restrictive than policies allowable within the national environment. Propagation of information access across coalition partners should be strictly controlled, and information distribution may need to be completely auditable.

While it is true that any system represents a potential vulnerability and a risk to be managed, it is particularly difficult to estimate the risk associated with a system developed and maintained by a coalition partner. Therefore, a coalition environment will need to enforce multiple (potentially nation-specific) security levels. Traditional approaches for discretionary and mandatory access [7], provisions for security downgrades, content filtering, and on-the-fly auditing are applicable to the coalition environment. In a coalition world, each nation is responsible for its own security policy and information disclosure rules. The IM model does not preclude a nation from enforcing national security or information disclosure policies. One of the primary motivations of the services to support federation is to acknowledge and support these information sharing seams in a coalition environment.

The possibility that information may be republished or otherwise divulged to a third party without the originator’s permission presents particular difficulty in this environment. The IM model recognizes the need for auditing to reveal inappropriate release of information. However, this is not possible unless the information is appropriately tagged with pedigree metadata that is intimately bound to the information. While this does not prohibit intentional, or malicious, release of information, it can be a safeguard against inadvertent release.

The Coalition Secure Management and Operations System (COSMOS)[4] international research collaboration project investigated these challenges, and found that implementations could be developed when common representations, such as C2IEDM, were used in conjunction with common policy enforcement mechanisms such as trusted gateways. However, as outlined above, this is not a typical situation.

To achieve information fusion within a complex battle space, we would like to be able to exploit information from all reliable sources, including our coalition partners. We may also need to share information with them to ensure consistent situation awareness (Chapter 2). The key to doing this well is to balance agility with accountability; to accommodate partners joining and leaving the coalition and using a diversity of implementations; and to acknowledge and support the differing trust relationships between the partners.

Information Management Best Practices

Section 5.1 introduced the goal of IM: to maximize the effectiveness of an enterprise by maximizing its ability to act upon information that is produced or consumed within the enterprise. Section 5.3 introduced a set of assumptions about information management in a coalition environment which, unlike most commerical enterprises, may lack coherent governance. These assumptions are that acquisition cannot be standardized or synchronized; that universal data standards are unrealistic; version skew is the norm; and coalition information spaces must adapt quickly to new partners and processes.

In short, the community must learn to accommodate diversity in mission, motivation, technical infrastructure, allegiance, and culture. To a large extent, an IM environment, which can form a bridge between the participants in a coalition operation, provides an ideal place to accommodate diversity. It provides one place in an information architecture where changes should be made. With this in mind, it is essential that our IM environments be designed with adaptability in mind.

In the coalition environment, we must seek ways to:

- Effectively share information within and among nations;
- Reduce complexity (for in challenging coalition environments complexity can mean the difference between success and failure);
- Maintain effective and coherent control of information resources.

The remainder of this section briefly explores means to achieve these goals.

5.4.1 Information Sharing

Effective information sharing requires agreement on how information is shared and how it is retrieved. It also requires that it be simple enough to overcome technical and non-technical barriers to sharing. This can be partially achieved through standardizing IM services, such as publish, subscribe, and query. To be effective, it is vital that we consider the manner in which information is characterized.

In Section 5.2.1, we considered how metadata characterizes information and enables IM. Metadata must be unambiguously interpreted by the information space. Consumers should use predicates over metadata to describe the information they desire. The concept of a predicate language over MIO metadata and accessor services of publish, subscribe, and query form a sufficient set of universal information services to meet most IM needs as described in the model. If a system lacks any one of these components, it is difficult to see how the remaining components operate together. For example, the Department of Defence (DoD) Discovery Metadata Standard [8] specifies one form of metadata, but without a predicate language and a standard set of services, it is difficult to see how the metadata will be used.

> Best Practice: Package information for sharing.

Effective information sharing, however, requires more than just metadata; it requires agreement between consumers and producers about the syntax and semantics of the metadata. Agreement on common metadata elements pays tremendous dividends. This has been demonstrated by *Cursor on Target* [9] demonstrated a minimal, but valuable, IM system for location-base data.

> Best Practice: Adopt common syntax and semantics for common information attributes such as location, time, and subject.

5.4.2 Reducing Complexity

Most technical forms of complexity are encountered during acquisition and fielding, particularly in a coalition environment. The decisions we make regarding the architecture of our edge-user applications, such as to go with a tightly integrated platform centric application instead of a loosely coupled and distributed system, significantly affects the cost and scheduling of capability acquisition and fielding. While a tightly coupled and platform-centric solution may look attractive from a budget and project management complexity perspective, it can have a detrimental effect on our ability to adapt and to respond to changes during the life of a particular application. Changes in nontechnical dimensions, like the global economic and political situation, may have an impact, such as the ability to share information with an unplanned coalition partner, or working with a new non-government organization and their IM capabilities.

The key to managing these risks is moving technical forms of complexity out of the edge-user applications and into shared infrastructure developed and supported specifically for that purpose. IM infrastructures can reduce application complexity and improve the flexibility of edge-user applications. In addition to reducing the burden of implementing IM functionality, loose coupling is achieved because the IM infrastructure provides an abstraction layer between applications, the underlying networks and coalition gateways.

> Best Practice: Adopt a dedicated IM infrastructure.

Traditionally, IM functions have been performed by edge-user applications because there was no infrastructure to take on these responsibilities, including:

- Requester authentication and authorization;

- Access control policy enforcement;
- Prioritization;
- Content tailoring;
- Network provisioning to meet anticipated demand;
- Subscription management.

These are just some of the IM functions that can be subsumed by an IM infrastructure to simplify edge-user applications.

An IM infrastructure provides an information space to hold information published by edge-user applications and a means to make that information available to national and coalition consumers as needed, subject to policy. Once information is delivered to the information space, the publishers are relieved of the responsibility of managing it. The information space will determine who has access to it, under what conditions, and with what priority. The information space behaves according to policies. These policies can be modified at run time, thus affecting the flow of information. If information producing and consuming applications are properly decoupled from the infrastructure a modification of the policies will not require modification of the applications themselves.

In addition to the information space, a fully capable IM infrastructure supports information security, automated workflow, information transformation, quality of service, brokerage, and maintenance.

Establishing, supporting, and using a small number of “universal” information services has the potential to greatly reduce complexity in this domain. These services could include common methods for managing information for example: publish (the act of sharing information), subscribe (requesting notification of information subsequently published), and query (requesting information that has been previously published). These services are universal because, in theory, the same services can be used for any kind of information, meaning that the semantics of the services do not change from one information object to the next. This is not to say that all objects are treated identically: they may have to be handled with differing qualities of service; some may be persisted while others are not and be subject to differing access control policies. The important point is that the manner in which edge-user applications share information does not change from object to object.

Best Practice: Create simple, ubiquitous common services.

Looking at this from a different perspective: if a community of interest must share information among its applications, it must agree on how the information to be shared is structured and interpreted. This typically requires hard human-to-human consensus building, and this problem is largely unavoidable. How that information is exchanged should not be an additional source of complexity. Why should we compound the information semantics challenge with diverse means of exchanging that information? By their nature, the universal IM services have simple and consistent semantics, regardless of the information they service.

Consider as an alternative, how we often share information today in a web services environment. The information provider, occasionally in concert with a community of interest, will create one or more services to expose that information. For example, if the object is an Air Tasking Order (ATO), the provider might create a service that provides methods such as *getATO(date)*, *getMissions(date, wing)*, *getMissions(date, tailNumber)*, and any number of other methods that expose the ATO in as many ways as the provider cares to create. Each of these has different semantics and different parameters that must be understood by any consuming application. This complexity can be reduced if standard forms of exposing and publishing information can be agreed with standard approaches for describing and exploiting information objects and their metadata. If we can achieve this, an information consumer could simply invokes a simplified interface such as *subscribe("namespace.identifier", predicate, filter)* on an information space that has advertised that type of information. While the content of the predicate and filter are specific to the type of information and its structure, their syntax is not. In conclusion, the use of standardized patterns for managing information can reduce complexity by obviating the need for many one-of-a-kind services. Once we start adopting these patterns, we can make use of common instantiations as part of a common information infrastructure, leading to more effective service management and information discovery.

5.4.3 Control and Flexibility

Another reason to consider use of the common information space approach is that it provides more standardized points of control. Rather than trying to modify or influence a myriad of edge-user applications, information managers only need to influence the systems that comprise the information space infrastructure. These systems are designed for this purpose and can expose standard control mechanisms, in contrast to systems built for other purposes.

Any mission-oriented application is written with implicit or explicit assumptions about the environment in which it will operate. Once the application is fielded, it is difficult to modify the application to meet new challenges. Use of a common IM approach and shared information space approach not only provides greater control, but it also allows new systems to be introduced, information flow patterns to be modified and information to be transformed as necessary to mediate differences between old and new systems, potentially across coalition boundaries.

Information Management Support to Information Fusion

This section considers IM topics that are directly relevant to information fusion, beginning with a discussion of the information lifecycle that exists between the sensors that create the information artifacts and the various information exploitation (IX) systems. As information fusion systems often require information exchange among themselves, this section considers syntactic and semantic interoperability, followed by a discussion on the representation and management of the context that information exits within. While some information is available for sensors in a robust structured form, much of the interesting information, such as human intelligence (HUMINT) reporting, is only partially structured and this problem is discussed in more detail. Next, we consider how IM can be provided as a service and how workflow can be used to automate IM tasks. Finally, this section discusses the role of IM supporting information fusion in the context of agent based systems.

5.5.1 Information Lifecycle

Information fusion is one part of a complex feedback system of actors which includes sensors, exploitation systems, and decision makers. [10] A simplistic model showing the interaction between information lifecycle supporting functions and IX capabilities is introduced at Figure 5.3. This model depicts how IM capabilities can be employed to provide an information hub through which information (in whatever form) flows from one actor to another. Using terms of the Model of IM, introduced in Section 5.2.1, the "Sensors" are information "Producers" and the "Exploitation" tools are information "Consumers." The space between the sensors and the exploitation tools is occupied a range of IM functions that operated on an "Information Space," the details depending on the requirements of a particular system.

Using this approach, the IM capability is distinct from the IX capability. With this separation, the IM capability is responsible for managing the lifecycle of information from creation to destruction. This may include dissemination, persistence, retrieval, and disposal, or removal from the working set of available information.

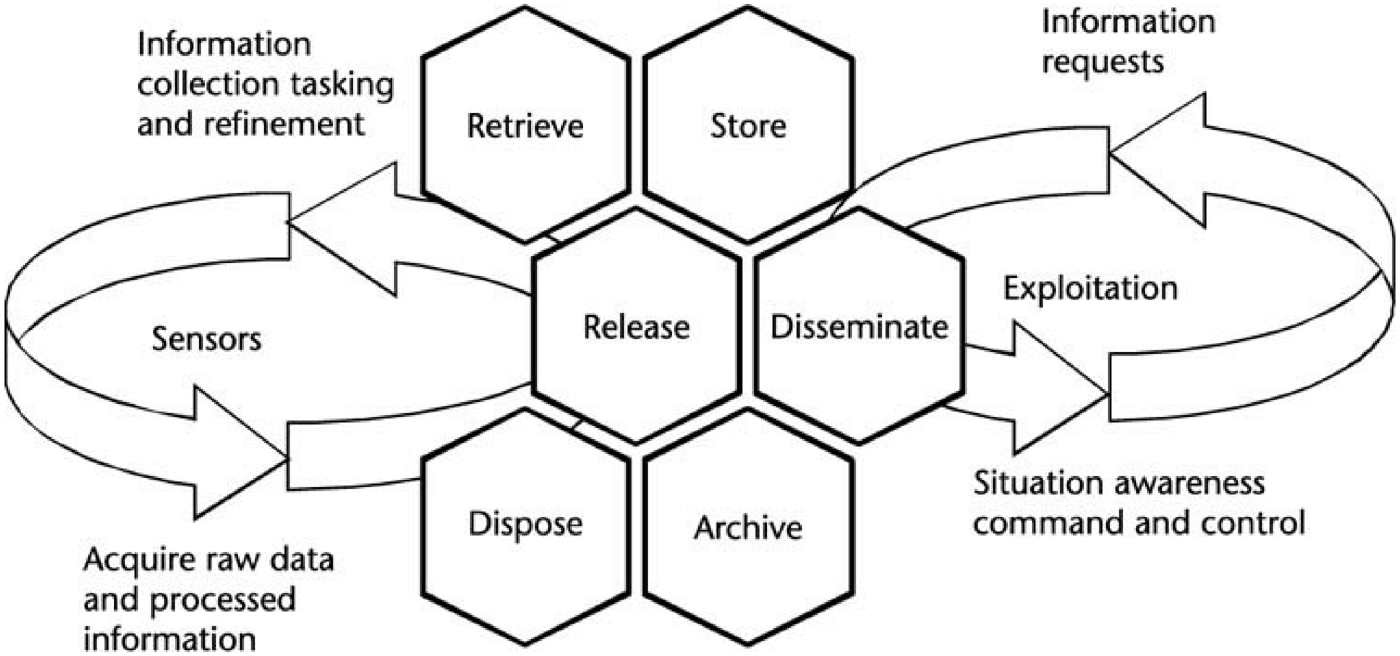


Figure 5.3 Simplistic IM/IX model.

Separating the concerns of managing information from exploiting it allows us to consider the use of common IM services that could include a whole host of other important system functions, such as security and resource management.

By following this approach, the IX capabilities are thus relieved of IM tasks and can focus on the tasks of data preparation, feature extraction, pattern identification and other core information fusion capabilities considered in the rest of this book. This clear separation of roles potentially allows reusable components to be developed and shared, with a view to improving capability while reducing overall system cost.

5.5.2 Syntactic and Semantic Interoperability

By its nature, HLIF generally requires multiple data types that in turn may come from multiple collection systems. The integration of information from various sources and information systems will allow end users to make the best possible use of available resources. In a net-centric context, the heterogeneity of participating systems can make interoperability between these systems difficult. At lower levels of abstraction data format standards, such as National Imagery Transmission Format (NITF), JPEG, STANAG 4607 GMTI, and MPEG, and message format standards, such as ADatP-3, OTH Gold, are well understood and essential enablers of interoperability. At higher levels of abstraction, like named entities, significant events, or potential threats, standardization is much more challenging and yet increasingly important as we strive toward higher levels of fusion. Standardization can take place at the syntactic and semantic level. Syntactic interoperability is the ability of information systems to exchange data in such a way that automatic processing is possible. Semantic interoperability, which builds on syntactic interoperability, is the ability of systems to exchange information and have the meaning of that information accurately and automatically interpreted by the receiving system.

A commonly used approach to information interoperability is the development of common exchange data models. Next, we discuss two examples of achieving syntactic and semantic interoperability: the JC3IEDM developed as part of the Multilateral Interoperability Programme (MIP) [11], and Universal Core (UCore) [12].

MIP JC3IEDM is a large monolithic data model that represents a shared agreement of high-level concepts in support of C2 information exchange requirements between C2 systems in coalition or multinational context. Syntactic interoperability is ensured through consistent encoding of the data. Semantic content is not explicitly represented in the data but is achieved through human-readable documentation of the MIP data model.

UCore relies upon XML schemas that are also used to represent syntax for common concepts. It specifies a set of commonly shared and universally understood concepts across domains representing the Who, What, When, and Where as a basis for information sharing and their semantics are described with Resource Description Framework (RDF) [13] and/or Ontology Web Language (OWL) [14]. As the base elements of UCORE are described by RDF/OWL, semantic interoperability is readily achieved at this level; however, application-specific extensions may not be—potentially leading to future interoperability challenges.

In summary, while syntactic interoperability enables data exchange for automatic processing, this does not mean that information is interpreted with the same meaning between systems that interact with one another. Semantic interoperability is required to ensure that meaning is preserved across system boundaries. In contrast to MIP’s implicit semantics, there is a trend toward representation of data with explicit semantics in a machine understandable form using Semantic Web [15] approaches. The Semantic Web approach builds upon basic RDF with increasingly richer ontology, such as the OWL with first order logics and richer models. These include Cyc and the Suggested Upper Merged Ontology (SUMO), which provide common constructs and a foundation for development of domain models with common constructs. Importantly for information fusion, each of these may potentially promote interoperability and reduce ambiguity albeit at the cost of increased complexity.

The main challenges for semantic interoperability arise at the boundaries of different Communities of Interest (COI). Concepts within different COIs typically require a higher degree of mediation than within the same domain. Common upper ontology, such as SUMO, can ease this mediation. However, even when similar concepts are of interest, differences may arise in the attributes of interest or the granularity of representation. Conversions across such boundaries generally lead to loss of fidelity. While the use of common IM services do not provide the silver bullet to solve this problem, concepts such as the managed information object type, schema repositories and transformation services could offer the infrastructure support to mediation across COI boundaries.

5.5.3 Management and Exploitation of Contextual Information

Context is defined broadly as the environment in which information or a system exists. It can have many implications on the appropriate handling of information as used in Context Aware Access Control (CAAC) and, importantly, on how disparate information should be fused. From an IM perspective, we use a more limited definition of context, which is:

> Information about the environment that is not explicitly referenced by a managed information object but that affects its appropriate handling and use.

Context involves both global attributes not tied to a specific client, such as Force Protection Condition (FPCON) or Information Operations Condition (IN-FOCON), and other attributes such as client connectivity status (e.g., bandwidth, intermittency), threat condition (whether or not this is known to the client), and mission.

Context extraction is the process of identifying and obtaining particular information about the environment (either generally, or pertaining to a particular participant) to support an IM activity. In the IM domain, CAAC is the ability to limit access to information based upon a combination of invariant policies and policies that change depending upon the context of the information producer/consumer, or client.

The need for context aware access control is borne out of concern for both appropriate control of information and efficient use of resources. An invariant policy may specify a set of roles that a client may inhabit. However, the client may be acting in only a few of those roles at any given time. Disseminating information to that client for other roles may unnecessarily expose sensitive information, consume bandwidth for little purpose, and distract the user with irrelevant information. Conversely, a producer faced with the prospect that a client may receive sensitive information when it is not required by the current client role may elect to withhold that information from the client at all times, thus reducing effectiveness of information sharing.

As we are considering CAAC from the perspective of IM, we shall only consider control of information rather than of physical entities. Furthermore, we do not consider aspects of security such as high-assurance devices, secure communications, or authentication; we assume that these capabilities are provided as necessary. Instead, we focus on the challenges of representation, definition, and enforcement of policies.

The core of any effective access control system is the policies that need to be defined, maintained, and enforced. While it is possible (and indeed perhaps prevalent) to encode policies in software, the evolving nature of military operations requires policies that can evolve as well. This suggests that we need policy languages that can be understood by humans and interpreted by machines to encode the policies that we wish to enforce, and that these policies be explicit and distinct from the software of the systems they control. Therefore, it is desirable to enforce a "separation of concerns" that isolates policy enforcement from applications that are affected by it.

In order to achieve the full promise of CAAC, there are many technical challenges that need to be considered and addressed as one designs and implements CAAC within their IM infrastructure. We shall touch briefly on eight such challenges:

1. Extracting context;
2. Policy mediation;
3. Combining contextual policies;
4. CAAC in a federated information environment;
5. Policy comprehension and correct implementation;
6. Addressing relevance and information priority;
7. Suppression of redundant information;
8. Improving prioritization through learning.

In *extracting context*, one must consider how contextual information is updated within a system. One possibility is manual intervention. This may be appropriate for important (but infrequent) changes such as FPCON. There may be databases that contain contextual information that can be monitored. A third possibility is that information passing through an information space can be mined for contextual information. Occasionally this information is structured; in which case it may be straightforward to update stored contextual information, provided that the source of the data is trusted. It is more challenging to extract validated contextual information from unstructured data, which generally involves some degree of human interaction.

Policy mediation is particularly important in complex organizations, such as coalitions, or where different organizations must work in a common information environment. The prevailing policy[5] is the net effect of all enforced policies. These policies may be managed coherently or may interact in a piecemeal manner. Unless there is clear separation of concerns to ensure that different policies are orthogonal to one another, it is clearly preferable to have coherent policies. However, this may require manual or automated identification and resolution of conflicts or inconsistencies between policies sets. An automated mediation capability should be based upon a formal, or unambiguous, model of the policies and a clear understanding of the domain of application.

Combining policies sets is always difficult because there may be redundancies and inconsistencies that are hard to detect, particularly if both positive (granting) and negative (denying) policies are allowed. Contextual policy combination is even more challenging because of the open world that the environment represents. Under such a model, inconsistencies may not only be computationally intractable, but impossible because we lack a complete representation of the environment.

One issue in *combining federated policies*, particularly in a coalition context, is that the policies may be as sensitive (or more so) than the information they apply to. Such sensitivities may necessitate a distributed decision and enforcement mechanism. This exacerbates the mediation and comprehension challenges described above. Different domains (perhaps under the influence of different communities of interest) may use different terms to describe similar concepts. In this case, not only must the semantics of the policy languages match or be mediated, but the definition of the subject, resources, and actions must be as well.

While we aspire to systems that automate policy enforcement, it is still essential that humans possess the tools to *author and understand prevailing*, and potentially prevailing, policies. This is particularly true for policy sets originating from different organizations or functional areas. This requires not only an understanding of one’s own policies, but also of how they interact with those of others. Even with formal models, it is difficult to distinguish between a correct policy and a desirable policy. As has been experienced by countless firewall administrators, a complex ordering of rules—even though enforced according to their specification—may have undesirable behavior. This is because the rules can interact in complex ways, and it is difficult envisioning all implications of even a few rules. Policy languages, such as Organization for the Advancement of Structured Information Standards (OASIS) and eXtensible Access Control Markup Language (XACML) [16] allow predefined and user-defined policy combination functions that further complicate interpretation.

When considering resource allocation— especially resources shared among several clients— it is desirable for a policy engine to return more than simply yes or no; a score could be used to judge relative *importance of information*. Little work has been done to combine context-aware prioritization with queue management necessary to maximize utility of available communications links. It is possible that the priority of information changes as it sits in a queue. It may be as simple as information that gets more or less important with time, but it may be that other contextual information changes have a disruptive effect on the prioritization of information.

Suppression of redundant information can reduce bandwidth consumption and avoid operator overload and confusion. For example, it may limit user access to only the latest version of information products, or to limit access based on bandwidth limitations, operational status level, or need to know. It is desirable to be able to discern, based on context and role of that user, what information needs to be passed and what information should not be passed. An example of this is seen in constructive simulation systems where entity states are only updated when actual state information (e.g., position, velocity, acceleration) deviates sufficiently from state information extrapolated from prior updates.

Of course, *relevance of information* is often in the eye of the beholder, and therefore may not be captured in an infrequently updated or generic policy. Indeed, there may be elements of context that are not available to (or foreseen by) a policy engine. In these cases, it may be possible to learn from observed access patterns and to reprioritize information based upon them.

In conclusion, for the purposes of IM, context is exploited extensively in CAAC. It has applications in other aspects of IM including prioritization, synchronization and replication of information across bandwidth constrained federated information spaces. Information fusion capabilities need context to understand the origin, trustworthiness, and state of systems operating on that information. Currently these two activities appear to be conducted in parallel stove-pipes. There is some opportunity to provide, beyond the core information space capabilities, support for context awareness services exposed from the core IM capabilities. The contextual information used in applications such as CAAC could then be exploited in support of other applications such as higher-level Information fusion.

5.5.4 Management and Exploitation of Unstructured Information

Unstructured information[6] may be characterized as information whose intended meaning is only loosely implied by its form and, therefore, requires interpretation in order to approximate and extract its intended meaning [17]. Unstructured information frequently takes the form of textual content, audio data, or imagery-based data often sourced from sensors, HUMINT, or open source content. Unlike content from typical physics-based (e.g., wind speed sensor) sensors, data from these other sources comes in a variety of formats, varied contexts, and degrees of usage-imposed structure—from none, to self-organized, through semantically structured. It may be more appropriate to label unstructured information as non-traditionally or inconsistently structured information. Such a label recognizes that there is a structure; it is simply challenging for current capabilities to capture this structure in a consistent computational or machine understandable form.

The primary objective of most IM techniques targeting inconsistently structured information is to give it some manageable structure, within as broad a problem context as possible. This objective focuses on deriving implied meaning. However, the ambiguity in the information's structure creates significant challenges for traditional IM approaches. These challenges include difficulties associated with the volume and diversity of the data, as well as impediments to consistent knowledge extraction. In particular, inconsistently structured information poses complex challenges to fusion tasks because the nature of the unstructured information exacerbates problems with interoperability and semantic similarity within similar but different domains.

The nature of unstructured information makes it difficult to effectively perform many of the information management functions described by the model in Figure 5.1 and outlined in Section 5.2. If we consider the unstructured information as a MIO, it may require a number of IM services to act upon it before it enters a management information space. It will likely not have the requisite metadata or encoding to enable those IM services to operate effectively. One of the layers of the model through which it might pass to enrich the unstructured information is the transformation layer. This layer includes services for contextualization of information.

In the case of unstructured textual data (or an oral stream converted to its textual representation), the transformation process may consist of several stages of Natural Language Processing (NLP). These stages will differ depending on the goal of the process, and could include: identification of word and sentences boundaries (text segmentation); assignment of labels to each word indicating its type (part-of-speech tagging); syntactic representation of each sentence according to the language's grammar (parsing); and selection of the meaning of ambiguous words (word sense disambiguation). In order to make the right choices, NLP systems need to use syntactic and semantic knowledge of the language, words meaning and restrictions of use, words relations (ontology), and the document context at all these stages.

When the goal of the transformation process involves understanding of the text, the systems convert human language into more formal representations such as parse trees or first-order logic structures that are easier for computer programs to manipulate. In contrast, if the goal of the transformation process is to automatically fill in structured templates with information contained in the text, extraction patterns or rules are applied to the documents to identify the desired values for the templates' gaps. In this case, the MIO can be annotated with metadata indicating the location where the patterns were discovered, without changing the original text representation. Named entity recognition can also be used to identify specific entities, be they persons, locations, events, times and dates, as well as domain-specific entities. Annotated texts can then be exploited by IM services for information retrieval, but also for HLIF processes supporting automated reasoning.

In the intelligence domain, information extraction and annotation techniques can help support intelligence analysis tasks. As an example, analysts have to ingest significant amounts of unstructured textual documents to identify entities of interest, in particular high-value targets, as well as actions or events, date and location, establish relationships between entities, or between entities and events, and make sense of them. DRDC Valcartier has developed the Multi-Intelligence Tool Suite (MITS) [18] that integrates functionalities of semantic analysis, extraction and geo-referencing of textual information. It uses interchangeable knowledge cartridges comprising knowledge domain descriptors or resources, such as ontologies/taxonomies, inference rules, and grammars, that cover knowledge domains required by intelligence analysts (e.g., terrorism, improvised explosive devices). The documents are analyzed automatically by looking for semantic patterns based on these knowledge domains and producing annotations of entities of interest that are then associated with geographic areas mentioned in the texts.

For effective management and exploitation of multimedia sources, such as imagery, geospatial, audio, and video streams, metadata is key. Descriptive metadata is used to better define the characteristics of the data (e.g., physical properties), while semantic metadata tagging enables the better description of multimedia sources to facilitate more precise content-based retrieval and automated exploitation. Techniques such as automatically tagging the managed information objects with descriptive metadata can be applied (e.g., extracting geospatial data) while extracting content features is more challenging. Moreover, the use of metadata standards is important to facilitate information discovery and exploitation.

The challenge is that the nature of this transformation changes the inconsistently structured information into one that has been defined based on usage or purpose; simply put, it is no longer unstructured. Given current technologies, this transformation is often not lossless. The use of current transformation methodologies shifts the inconsistently structured data into a state that has potentially lost some of its information content in alternative contexts or transformation spaces. The reason behind the loss is that ideally, transformations aim to enrich the original form with meaning (e.g., introducing structure, adding metadata, annotation). This meaning imparts a prescribed context that can alter other perspectives that may exist or provide value within the data; the data is structured according to the problem it is applied to. Even when the original unstructured data is retained after transformation, in its natural state, it also retains problems for access (e.g., locating/searching), resource allocation, quality assessment, and life cycle scoping.

Most current methods for this transformation are measured on their ability to retain as broad a context, post-transformation, as possible. One might view this as retaining multi-purpose information utility. If this view is taken, then the complexity in dealing with inconsistently structured data within the scope of consumers, producers, and federates becomes readily apparent. This is because alternative information spaces or uses may be unknown at the point where transformations are applied from or to one of the scoped areas. As discussed in Section 5.5.3, the policies of information managers are similarly affected because access control, resource allocation, and life cycle management are all intimately tied to data's contextual structure at an operational level.

Flexible ontologies and structuring services/agents can provide foundational mechanisms for addressing the contextual diversity in the application of inconsistently structured information. The architectural structure of service or agent infrastructure requires a problem be decomposed into digestible functionality that is (ideally) inherently reusable by a variety of processes. Ontologies serve as a representation vocabulary that provides a set of terms with which to describe the facts in some domain. By processing inconsistent data against a contextual ontology, the data gains structure are directly related to the applied ontology

In order to effectively exploit unstructured data, it is necessary to tightly integrate the original unstructured data with transformation methodologies and derived semantic understandings. Given the prevalence of unstructured data, this functionality must be a core element of IM architectures. It is this dynamic shifting from non-traditionally or inconsistently structured data to useful, context-relevant information that is the basis for information fusion capabilities. This transformation activity should exist within the tranformation ring as described in the information management model, Figure 5.1.

While it would still be necessary to address the storage and dynamic structuring of unstructured data (e.g., referring to Figure 5.1, the maintenance and information space rings), information fusion capabilities can provide methods to support the transformation layer, providing functionality to assist contextualization. There is merit in considering not only how IM services can support HLIF, but also how information fusion services could be exposed to support this important IM service layer. A service-oriented approach incorporating both the primary IM services, but also providing syntactic and semantic interoperability and transformation services, derived from common information fusion services could provide a much more effective IM and IX capability [2]. This could enable reuseable and extensible systems utilizing orchestrated transformation processes that are flexible and configurable to meet the needs of any particular decision-making.

5.5.5 Information Management as a Service

In a Service Oriented Architecture (SOA), services are generally characterized as modular, loosely coupled and reusable components of logic that may be distributed and executed across the network. There are several claimed benefits to an architecture based on services. These include:

1. *Modularity and Reuse*—Services designed for reuse and composition will ideally provide discrete interfaces that perform specifc business functions which requires no knowledge of the application state for the service consumer to invoke the methods exposed. A number of services may then be chained together with business logic to construct new services or applications. They may be reused, changed-out, and extended without adverse effect to the functionality, although the non-functional requirements, such as performance (including response time and capacity) do need to be considered if an assured level of service is required. By adopting this approach and maintaining interface compatibility, different services with different characteristics may be swapped out as required.
2. *Agility and Flexibility*—When combined with the workflow constructs (which will be discussed in Section 5.5.6), Standard approaches to interface definition and advertisment make them easier to integrate dynamically into workflows as opposed to writing bespoke application code each time a business process changes. This supports the need for rapid development to service the demands of ad hoc coalition groups.
3. *Governance*—The use of an SOA approach requires governance to manage the evolution of the architecture, and manage the demands placed on services to assure a level of performance. Many SOA middleware implementations now provide mechanisms to support governance. Mechanisms include the ability to advertise service specifcations, verify and track service level agreements, as well as managing the service lifecycle, including staging and revoking services.

While there are several technical approaches to implementing an SOA, the current mainstream approach is the use of Extensible Markup Language (XML) Web Services [19]. The recent drive towards SOA naturally leads to the question of how IM capabilities should be expressed as services.

A decomposition of the IM model activities into services should differentiate between services that are visible to the actors (producers, consumers, managers, and federates) and those that operate under the covers. Multiple implementations that can respond to differing needs or environments should be possible. The breakdown of activities should consider the mechanisms through which information is passed from one service to the next. As a point of departure, we will start with a description of the non-SOA IM infrastructure that is used in the Coalition Distributed Information Fusion testbed (see Chapter 6) and then describe how new implementations have incorporated SOA principles.

At the heart of the testbed [20] is the Joint Battlespace Infosphere (JBI) infrastructure built on top of the Java Native Interface (JNI) technology. Reflecting design patterns of the time, the capabilities were presented as an integrated whole rather than decomposed into services. Later JBI implementations were built on **Java 2 Platform, Standard Edition** (J2SE) infrastructure that separated out publish, subscribe and query functions from a user perspective, but they were still tightly coupled within the application server. The Application Programming Interface (API) needed management of client-server application state (i.e., state-full connections) and necessitated a sustained connection between the client and server. Interfacing with different technologies was also nontrivial.

Today's SOA environments encourage integrated systems to be decomposed into reusable component services, each of which may be implemented to meet specific demands. The US Air Force Research Labs has now redeveloped the JBI middleware into a project called JBI Phoenix, which has been re-engineered from the ground-up using an SOA approach to provide a set of IM services offering IM capabilities with four primary services: submission, brokering, persistence, and dissemination. The submission and dissemination services are user facing with producers interacting with the submission service and consumers interacting with the dissemination services. The brokering and persistence services are not considered user-facing, although managers may interact with them. Since the binding is through a service, the architecture allows the introduction of heterogeneous implementations of each service.

There are several supporting constructs that run throughout the architecture, the two most important being contexts and channels. Essentially, a context allows attributes to be associated with information. Context can be used in many ways, including characterization of the information, provenance regarding previous processing or anticipated routing to services, and prioritization based upon factors such as mission relevance. The primary means of communicating among the services is achieved by exchanging contexts and information through channels. Channels provide the mechanism for information to be moved between and among the producers, consumers, and services, or entities. It provides the plumbing that connects entities and enables effective and efficient information flow. It abstracts and encapsulates transport protocols, segregates information and control flows. There are distinct abstractions for control channels and information channels, and they may be implemented using different protocols.

As mentioned earlier, agility is an important attribute in a coalition environment, and SOAs offer the promise of flexibility and agility. The SOA information management services described in this JBI Phoenix architecture are generic in the sense that they are not tied to a specific application domain or service implementation technology. This is because a generic solution is more likely to be flexible enough to permit rapid adaptation to changing environments and missions. However, this is only one of many service architectures that could support information management capabilities.

It should be noted that, when using an SOA middleware approach and related technologies, that one could still build a system that provided services, but did not offer any generic IM services. A system could be contstructed in which all service interfaces are tied into specific data types, systems, and missions. In fact, this is probably more the norm than the exception. Typically, each type of information is supported by a set of services tuned to the particularities of that information. The advantage is that there is no mismatch between the intended uses of the information and the services that support it, unless there are unanticipated uses of course. The disadvantage is that lack of generic capabilities leads to interoperability challenges, limited discovery, and stove-piped communities with less reuse of common core services.

A common problem in resource-constrained environments is that narrowly scoped services rarely manage shared resources well. Indeed, it is almost a corollary of loose-coupling that one service is oblivious to services that it does not directly interact with — even if they are competing for the same resources. As a result, resources are unlikely to be managed well. Indeed, in the JBI Phoenix services described above, much of the complexity of implementation is not in the user facing service interfaces; it is in the ability to allocate finite resources to competing demands. An example of an implementation for resource management is called QoS enabled dissemination [21]. This implementation demonstrates that the architecture can support multiple channel implementations optimized for efficient transport among services, all while maintaining the same user-facing services.

The adoption of common information services, particularly in a coalition context, is the focus of ongoing research within The Technology Cooperation Program (TTCP) Command, Control, Communications and Information Systems (C3I) Group. Efforts are underway to design, establish and evaluate different approaches, considering SOA and workflow using orchestration, building on the efforts of the information fusion test-bed to understand how IM services can also be reused in support of HLIF.

5.5.6 Workfow

A generic reference model for workflow was published by the Workflow Management Coalition [22] and later adapted by the Object Management Group (OMG) [23]. The IM model introduced in Section 5.2 has adopted key aspects of these models. The definition of workflow and workflow management systems from the OMG are as follows:

> Workflow is concerned with the automation of procedures where information and tasks are passed between participants according to a defined set of rules to achieve, or contribute to, an overall business

> goal. Whilst workflow may be manually organized, in practice most workflow is normally organized within the context of an Information technology (IT) system to provide computerized support for the procedural automation. A Workflow Management System provides procedural automation of a business process by management of the sequence of work activities and the invocation of appropriate human and/or IT resources associated with the various activity steps.

From an IM perspective, workflows are the patterns of information flow that support the processes of the enterprise. Specifically, workflows allow for the specification, documentation, governance, and reuse of IM activity patterns. The model of IM treats the services to support workflow at a very high level, as is befitting such a model. These services include the ability to establish, maintain, link and manage access controls and optimize workflows. Clearly, the difficult problems are buried under such high level services. However, the layering of the model allows many challenges for workflow engines to be separated from the core workflow execution and be placed into other layers. For example, many access control issues for the operation of the workflow are indistinguishable from access control to MIOs. Additionally, the definition of users and roles are essentially the same. These commonalities reinforce our belief that adding the workflow layer on top of a managed information space not only allows for greater separation of concerns, it allows much more consistent management between the information space and the workflows that draw upon it.

Workflows are of special importance in information fusion systems because they have the ability to specify and govern interactions between different systems. For instance, a workflow could be created which processes unstructured reports, mediates them to a new format and publishes the data to a partner enterprise. In this case, workflow allows for this task to be automated and it allows for the governance of this activity.

5.5.6.1 Workfow Components

A workflow is made up of activities and control flow primitives that ensure proper sequencing of the activities. Together, the activities and control flow primitives form a directed (perhaps cyclic) activity of greater complexity which is often and readily represented in graph form.

Control flow primitives include simple constructs like sequential routing whereby the output event of one activity is passed directly to another activity, parallel routing/split whereby an output event is sent to several activities that can then run concurrently. Other control primitives include branching and looping. Different workflow languages have different names for the primitives but the concepts are essentially similar. For example, in the Business Processing Execution Language (BPEL) these are *sequence*, *flow*, *pick*, *and while*, respectively [24].

Workflows may be hierarchically decomposed into subworkflows that execute concurrently or sequentially. A subworkflow is represented as a single activity in the parent workflow. Subworkflows may be used merely as a means to simplify presentation of a complex workflow, or they may represent significant boundaries. A subworkflow may have different access-control policies, different owners, and different implementations. For example, a workflow defined for coalition operations may have subworkflows run by each of several coalition partners.

The details of a subworkflow may not be visible to a parent workflow. In this sense, it may be an abstraction of a private workflow. Following the example above, a coalition partner may choose not to expose its internal decision making procedures to the coalition.

Concurrent interacting workflows are workflows that operate concurrently but that are cross-joined by control primitives. Like subworkflows, these may have different implementations or operation on behalf of different principals and may allow only partial visibility into each other's structure. Unlike a subworkflow that is instantiated and dissolved within a single activity of the parent workflow, concurrent interacting workflows may have independent lifecycles.

Both subworkflows and concurrent interacting workflows may be linked either at design-time, instantiation or in the midst of execution.

5.5.6.2 Workfow Management Systems and the Role of Languages

Workflows specified in machine understandable languages allow for greater automation and consistent reuse. The Object Management Group (OMG), Workflow Management Facility (WfMF), and Business Processing Execution Language (BPEL) [24] are used for this purpose. Currently, the language of choice in industry and defense is BPEL. These standards are not compatible in that the OMG specification is built upon a Common Object Request Broker Architecture (CORBA) foundation for procedure invocation whereas BPEL uses web services. It appears that BPEL is gaining industry support, while development on the OMG standard is dropping off.

BPEL expresses workflow as a coordinated set of web service interactions. Aimed primarily at business applications, the language is geared towards abstracting away the implementations of the services to focus upon the sequencing of their invocation.

BPEL does not explicitly support or prohibit the modification of processes, or workflows, at run-time, so-called dynamic workflows. However, there are ways that this could happen. For instance, a BPEL process could provide input to a human interaction whereby the user's actions influence the selection of services. Furthermore, BPEL seems to be intended exclusively for build-time. Researchers [25] are currently working on ways to create orchestrations of services in an ad hoc and intelligent manner using BPEL. It remains to be seen how effectively BPEL can be employed in a coalition environment with diverse workflow engines and trust relationships between partners.

None of the tools seem to support mediation services required to bridge gaps between heterogeneous systems where there may be differences of information representation or syntax. At least BPEL4WS does seem to support heterogeneous implementations of the services themselves.

5.5.6.3 Agents and Humans in Workflows

Agents and humans are both actors in workflows. That is, they are somewhat independent consumers, producers, and managers of information, as described in Section 5.2. They can carry out or direct almost any activity within the IM model themselves. For instance, a human can store information by taking notes on a notepad and then archive it in a desk drawer. However, this is not efficient or beneficial to the enterprise as a whole. So, although actors can carry out many tasks it is often beneficial to have tasks carried out in their stead via other functionalities (low level and high level).

Humans and agents will increasingly be called upon to merge their respective strengths and weaknesses to form integrated workflows in support of hybrid teams. In some cases, the drive to increase machine participation is precipitated by the scarcity of human capital. In other cases, agents offer superior functionality over strictly human actors in certain domains. As capabilities and responsibilities of agents increase, the need to manage disputes and facilitate communication between humans and agents is magnified. As a major point of contact between humans and agents, IM will be a key mediator of human-agent interactions and will increasingly serve a key C2 function of these hybrid teams. For example, deadlock in workflows may result in the case of unclear prioritization between independent human and agents inputs. Humans may be marginalized in team activities if IM structural biases exacerbate agent advantages in machine to machine communication.

The line between humans and agents itself is blurry, as web services may be used to deliver both human and machine presence to hybrid workflows.

5.5.6.4 Agents and Humans as Web Services

The use of BPEL agents, and relatively recently humans, can be modeled like web services. Agents can be instantiated and instructed with web service calls. For instance, several researchers at Whitestein Technologies AG in Switzerland have created a framework called WSIGS to facilitate agents as services [26]. The BPEL-4PEOPLE and WS-Human Task candidate specifications make it possible to abstractly task a human with some function in a workflow. The specifications are abstract in that they leave the implementation and governance of human tasking to the implementer. This may be considered problematic by some, as it does not account for all human performance limitations or issues. Currently several vendor products already support these specifications even though they have not been officially accepted by OASIS [27].

Using these agents and humans as web services presents several challenges to coalition groups. These are mostly governance and interoperability related. This is mainly due to the areas of QoS, or availability, and the fact that agents and humans may have the ability to independently call services themselves. QoS and availability are already issues for services that cross enterprise bounds as it is difficult to enforce in either direction. This potential for problem is inflated by the fact that humans and agents have availability that can change over time and that is difficult to specify. Further research in these areas is necessary to correctly use these potentially powerful paradigms in future coalition systems.

Information Management from an Agent Perspective

Perspectives on agents have also evolved considerably over the past decade. Indeed, the JBI publish/subscribe implementation was part of a larger agent infrastructure based upon JINI developed under a U.S. Defense Research Projects Agency (DARPA) program called Control of Agent Based Systems (CoABS) [28] agent architecture. Agents used the publish and subscribe services to share information. Delegating publish and subscribe responsibility to the agent framework allowed the agents themselves to focus on what they do well rather than finding and managing connections with other agents. Consider a fusion agent that is looking for track data and intelligence data to fuse in an attempt to predict enemy intent [29]. This is a hard research problem. Is it likely that the researcher will also be interested in the details of discovering, characterizing, and interacting with several new data sources? Is it likely that the researcher wants to manage connections to consumers of his agent? How about access control? The publish and subscribe infrastructure gives CoABS the capability to decouple agents, support information discovery, and reduce complexity of managing agent lifecycles while simultaneously providing capability developers a simple programming abstraction.

A key tenet of agents—particularly intelligent agents—is their ability to be autonomous, either with their own intent or an end user's objective as a goal. This autonomous nature, underscores the importance of structured and rigorous IM. To be able to operate autonomously, efficiently, and effectively within an information domain, agents must be able to locate, access, and understand information sources that may be local or distributed. The distribution of information may introduce problems of latency, intermittent connectivity and variable service availability. Additionally, when considering collaborative working environments, versioning and update control complexities are introduced.

An agent's ability to enable collaboration (i.e., communication, cooperation, and coordination), facilitates specialization in essential information tasks such as information retrieval. Information retrieval is a typical agent application and an essential function of IM. Consider the example given above involving the fusion agent. Critical activities in that process were discovering, gaining access and interacting with information sources—all collaborative activities. When this task is considered from the agent's perspective rather than a user's, the dependencies on a descriptively rich, semantically interoperable, and implicitly structured information space are obvious. Often constructed as a cooperative activity where the agent is a proxy for a user or another system component, the agent must operate in and abide by the constraints of the IM environment. As such, agents are actors (producers, consumers, managers, and federators) that interact with each other through the information space and associated management infrastructure that is comprised of resources and other services that perform the needed or necessary activities.

Beyond their mobile, collaborative, and autonomous nature, typical agent characteristics include the ability to interact with their environment and adapt pro-actively. The bounds of an agents' environment frequently extend beyond the computational space to include interaction with users and physical components, such as sensors and devices, that represent information sources. To effectively address the demands of IM, agents need to be able to exhibit proactive behavior, that is, the ability to effect actions that achieve their goals by taking the initiative.This allows agents to operate on the edges and stress the limits of an information architecture, limited only by the architecture's structure and frame. These characteristics make agents uniquely suitable for IM tasks. In this context, the agents' ability to discover new information, to monitor resources or sensors, becomes a significant asset and a desirable quality for managed information environments.

Given the scope and functionality of an agent system, from an external perspective it is not unreasonable for an agent system to look like a service. By taking this approach agent capabilities would appear as dynamic functions that could be included in a service orchestration where approaches for service discovery, mediation, and access control would be applicable. Given this orientation, in a narrow sense, IM services do the heavy lifting of making sure that information gets where it needs to go, and ensure that consumers can find the information needed without regard to where or how it was produced. This is not to say that information quality and provenance are not important, but these attributes should be captured in the metadata for information rather than derived based upon assertions about particular provider end points. Additionally, the services may archive information making it available after the sources have gone offline, and potentially allowing the agents a smaller footprint.

Conclusions

Effective information fusion requires innovative algorithms, domain knowledge and access to information that is available, trusted, accurate, understandable, timely, and properly characterized. Achieving these objectives is the domain of information management. The model of IM described the verbs and nouns of IM that together provide consumers a way to access the information they need to perform information fusion. It starts with an information space that acts as an information clearinghouse within a federated and potentially coalition information environment. It helps information producers flexibly share information, consumers discover it, and managers ensure that it is handled appropriately. HLIF may require many forms of information transformation, processing, and interpretation.

An SOA-based approach to deliver IM capabilities as a service could have wide reaching benefits across the enterprise. In addition to reducing complexity of edge-user applications, IM services can allow fusion agents to exchange information easily. This has the potential to support quicker decisions and actions based on tasks carried out by the agents.

In a broader sense, IM services simplify processes that cross community of interest boundaries through federation. Common IM services designed for reuse and extensibility may support loose coupling that can address many challenges of unpredictable coalition environments.

These common IM services could be incorporated into workflows and orchestration, providing federation of information across coalition boundaries, while supporting the important features of access controls, prioritization, format mediation, and transformation. There is also a part for information fusion to play in providing services that can support contextualization. Semantically linked information can support contextualization and can be considered essential to accurate interpretation to support quality information fusion. Ultimately, it is the richly contextualized federated information spaces that provide the fodder for information fusion. The extent to which it can do so in the trusted and timely manner may directly affect the quality of the resulting information fusion.

The IM services themselves could be used to initiate workflows to trigger events in response to policies, not just for access controls, but also in response to content of information being published to alert users subscribing to the common information space.

Just how we go about achieving these benefits in a coalition context is part of ongoing research and investigation within the TTCP C3I group.

rences

[1] U.S. Office of Management and Budget (OMB) Circular A-130; http://www.whitehouse.gov/omb/circulars/a130/a130trans4.html.

[2] Bray, S., “Framework for Warfighter Information Services - using the concept of a Virtual Knowledge Base,” *International Command and Control Research and Technology Symposium*, (ICCRTS), June 2010.

[3] Farrington, M., “Information Management Capability Development Roadmap (IMCDR),” March 2004.

[4] Linderman, M., S. Haines, B. Siegel, G. Chase, D. Ouellet, J. O’May, and J. Brichacek, “A Reference Model for Information Management to Support Coalition Information Sharing Needs,” *International Command and Control Research and Technology Symposium*, (IC-CRTS), 2005.

[5] Dublin Core; http://dublincore.org.

[6] Department of Defense Discovery Metadata Specification (DDMS), Version 1.3, accessed at http://metadata.dod.mil/mdreg/user/DDMS.cfm.

[7] Bell, D., and L. Lapadula, “Secure Computer Systems: Unified Exposition and MULTICS Interpretation,” *Technical Report ESD-TR-75-306*, MITRE Corporation, 1975.

[8] http://metadata.dod.mil/mdr/irs/DDMS/.

[9] http://cot.mitre.org/.

10] Blasch, E., S. Russell, and G. Seetharaman, “Joint Data Management for MOVINT Data-to-Decision Making,” *International Conference on Information Fusion*, 2011.

11] Multilateral Interoperability Programme (MIP), https://www.mipsite.lsec.dnd.ca.

12] UCore, https://www.ucore.gov.

13] Resource Description Framework (RDF), www.w3.org/TR/rdf-primer.

14] Web Ontology Language (OWL), http://www.w3.org/TR/owl-ref/.

15] W3C Semantic Web, http://www.w3.org/2001/sw/.

16] http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml.

17] Ferrucci, D., and A. Lally, “Building an example application with the Unstructured Information Management Architecture,” *IBM Systems Journal*, Vol. 43, No. 3, pp. 2004, 455–475.

18] Auger, A., “Acquisition and Exploitation of Knowledge for Defence and Security,” *Proc. of the NATO RTO-MP-IST-087 Symposium*, “Information Management and Exploitation,” October 2009.

19] http://www.w3.org/2002/ws.

20] Linderman, M. H., and V. T. Combs, “JINI-Based Publish and Subscribe Capability,” *Proceedings of SPIE*, Vol. 4863, 2002.

21] Loyall, J., et al., “QoS Enabled Dissemination of Managed Information Objects in a Pub-lish-Subscribe-Query Information Broker,” *Proceedings of SPIE*, Vol. 7350, 2009

22] Hollingsworth, D., “The Workflow Reference Model,” Document Number TC001-1003 by the Workflow Management Coalition, 1995 (available at http://www.wfmc.org).

23] Workflow Management Facility Specification, V1.2, Object Management Group (OMG), April 2000, (available at http://www.omg.org, OMG Domain Specifications).

24] Web Services Business Process Execution Language Version 2.0, April 11, 2007, http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf.

25] Armoza, D., J. Ballas, and J. Nevitt, “Intelligent Orchestration Challenges in the DoD,” [In preparation]

26] Greenwood, D., and M. Calisti, “Engineering Web Services—Agent Integration,” *IEEE Conference on Systems, Man and Cybernetics*, 2004.

27] www.oasis-open.org.

28] Kahn, M. L, and C. D. T. Cicalese, “The CoABS Grid,” in *Innovative Concepts for Agent-Based Systems*, *Lecture Notes in Computer Science*, Vol. 2564/2003, 2003, pp. 125–134.

29] Kadar, I., E. Blasch, and C. Yang, “Network and Service Management Effects on Distributed Net-Centric Fusion Data Quality,” *International Conference on Information Fusion*, 2008.

---

1. Defined here as applications that are not part of, or controlled by, the information management infrastructure.

2. A predicate is a logical expression that when evaluated returns TRUE or FALSE.

3. A given type may be encoded in many formats, for example satellite imagery may be encoded in a number of image formats—for example, JPEG. Conversely while the JPEG format may be used for overhead imagery it could also be used for a similar type say, handheld imagery or a substantially different type such as scanned intelligence reports.

4. During CWID 2007, COSMOS demonstrated the Integrated Operations objective by passing C2 data in the international standard format C2IEDM between C2 systems located at NSWC, Dahlgren, USEUCOM, CFEC, and Adelaide http://www.cwid.js.mil/public/CWID07FR/htmlfiles/314int.html

5. A prevailing policy is a function of a defined policy set and an environment that determines which subset of the policy set is enforced at any one time for any given subject, resource, and action.

6. In contrast structured information may be characterized as information whose intended meaning is unambiguous and explicitly represented in the structure or format of the data. A canonical example of structured information is a relational database table.

CHAPTER 6

Coalition Distributed Information Fusion Testbed

Steven Wark (Australia)

The Coalition Distributed Information Fusion Testbed (CDIFT) grew out of a need identified by the defence research organizations in Australia, Canada, the United Kingdom, and the United States to provide an environment for collaborative development, testing, demonstration, and experimentation with higher-level information fusion technologies. These organizations recognized that the broad scope of the higher-level fusion domain, the broad range of technologies needed to address different aspects of this domain, and the relative immaturity of the technologies compared to the lower-level fusion. Key functions required that interoperability between different national capabilities within a coalition needed to be addressed early in the development phase of these capabilities. The CDIFT was established as an initiative of the TTCP Technical Panel on Information Fusion to achieve this.

Models of Collaboration

There are several models of collaboration [1] within a coalition that the CDIFT needed to support.

6.1.1 Technology Showcase

In this model, the developing nation retains exclusive ownership and control of the fusion capability and its products, and only demonstrates selected artifacts, such as a display or Graphical User Interface (GUI), to the other coalition partners. This model supports sharing of knowledge on the feasibility and/or efficacy of particular techniques or approaches, without compromising commercial or national security interests. The role of the CDIFT in this model is to provide known and/or common information feeds to the fusion system in question. In this role, the CDIFT can support comparative evaluation of different national fusion capabilities where high-level metrics such as timeliness and accuracy against known ground truths are relevant.

6.1.2 Technology Demonstration

In this model, the developing nation retains exclusive ownership and control of the fusion capability, but shares the products of the fusion system with the other coalition partners. This model supports sharing of capability without significantly compromising commercial or national security interests. In this model, the CDIFT provides known information feeds to the fusion system, and supports distribution of the fusion products —potentially as inputs to other fusion systems from coalition partners. In this role, the CDIFT can be used to demonstrate/evaluate how complementary fusion capabilities can be integrated into a coalition fusion capability, as well as comparative evaluation of different national fusion capabilities.

6.1.3 Technology Evaluation

In this model, the developing nation retains exclusive ownership of the fusion capability, but shares it for a limited time with the other coalition partners. This could, for example, be to evaluate the capability under a broader range of operating conditions, or for evaluation of the suitability of the capability to meet the needs of the coalition partners. This model supports the development of a national capability and its deployment to coalition partners. In this model, the CDIFT provides a common deployment platform for the fusion capability so that it can be evaluated under controlled conditions. In this role, the CDIFT supports detailed analysis of national fusion capabilities by third parties.

6.1.4 Technology Sharing

In this model, the developing nation gives the fusion capability to the coalition partners. This could, for example, be used for technology evaluation, or to ensure interoperability between coalition partners, or to allow the coalition partners to leverage this capability in their own fusion systems. In this model, the CDIFT provides a common deployment platform for the national fusion capability where the information requirements and flows are understood by all coalition partners. In this role, the CDIFT supports integration of coalition fusion capabilities with national systems. Elements of the CDIFT itself have been provided under this model.

6.1.5 Joint Development

In this model, national fusion capabilities are developed alongside those of the coalition partners to ensure a deep level of interoperability between the systems. The developer retains ownership of the fusion capability developed, it can usually operate independently of the other coalition capabilities, and it can interoperate (hopefully seamlessly) with the coalition systems when needed. This could, for example, be used for development of common information models, adapters, or communication protocols between complementary fusion capabilities. The role of the CDIFT in this model is to provide a common development and testing platform for national fusion capabilities to ensure interoperability with other coalition systems.

6.1.6 Joint Ownership

In this model, national fusion capabilities are developed alongside those of the coalition partners, but ownership of fusion capabilities is shared between the coalition partners. This could, for example, be used for the development of a common fusion infrastructure, or for the integration of complementary fusion capabilities which can not stand on their own. In this model, the CDIFT provides a common development and testing platform to ensure the tight level of integration that may be needed by fusion components. New capabilities within the CDIFT itself could be developed under this model.

Figure 6.1 illustrates these different models of collaboration in a coalition context with different fusion blocks from different coalition partners.

Requirements

To fulfil the various roles of CDIFT with these different coalition collaboration models, the CDIFT needed to meet a number of requirements.

6.2.1 Provide Simulated Information Feeds

Testing and evaluation of information fusion capabilities requires reproducible information feeds with known ground truth. This can be provided by using entirely Synthetic information feeds, by replaying real data that has been suitably analyzed (and sanitized), or a combination of these. As the timing of information can be critical to its interpretation, it is desirable to ensure synchronization of injection of different information feeds against a common reference clock, so that each component deployed in the CDIFT receives information in a reproducible manner to support analysis of the performance of the fusion system.

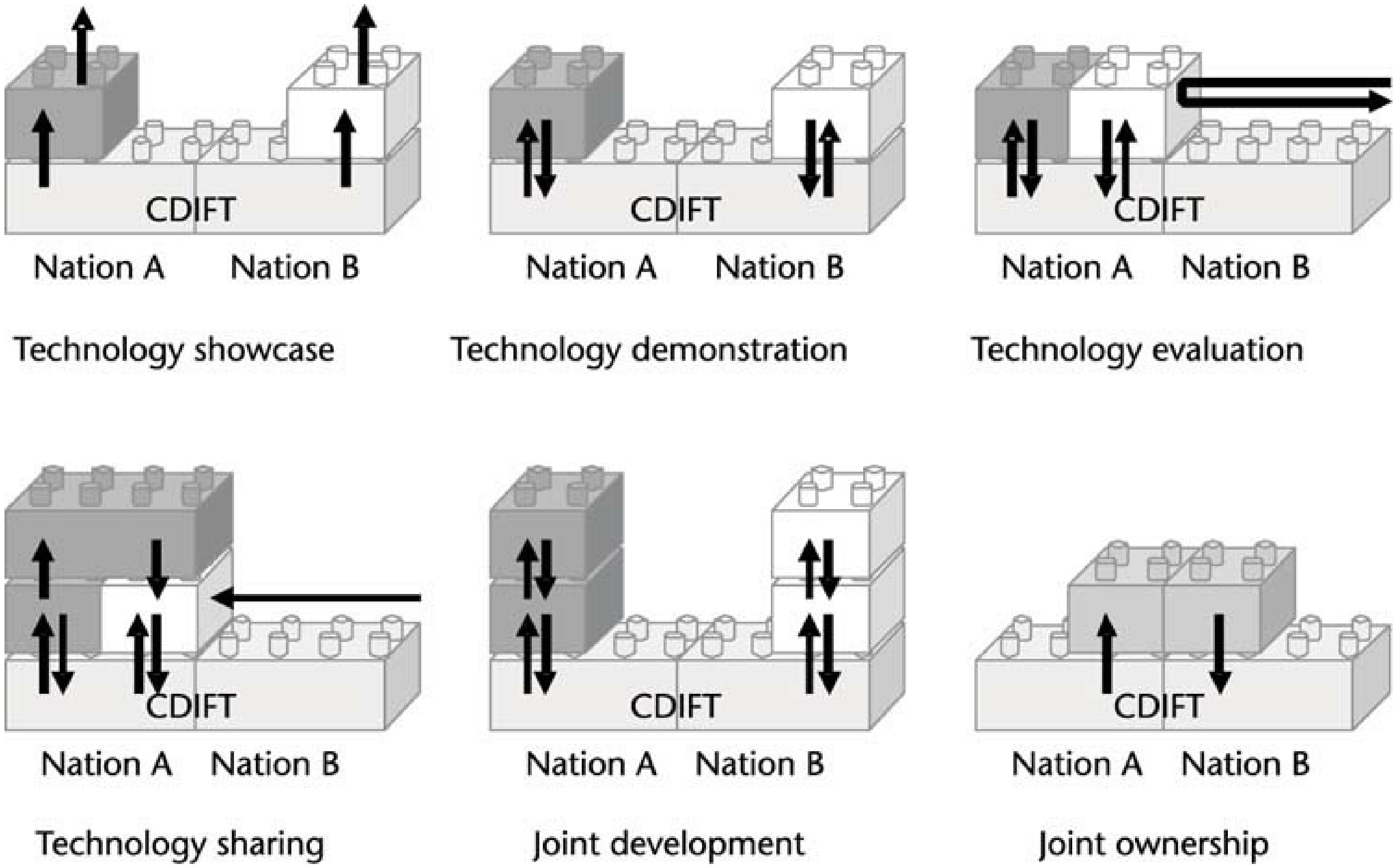


Figure 6.1 Different collaboration models relevant to a coalition context, and how different fusion "blocks" could be exploited across national boundaries.

6.2.2 Real-Time Performance

There is a requirement to aid transition to test and evaluation of information fusion capabilities in operational environments with live information feeds, human-in-the-loop performance evaluation, and demonstration of fusion capabilities.

6.2.3 Distributed Architecture

There is a requirement to facilitate models of collaboration where national fusion capability needs to be protected for intellectual property or national security reasons. This allows national fusion capability to be controlled by the originating nation, with only specific interfaces or products exposed to the coalition via the network. Furthermore, the CDIFT needed to be deployable in a variety of ways:

- On an ad hoc local area network (LAN) during scheduled workshops for demonstration[1], integration, or collaborative development. This constrains the computational resources available to the CDIFT, and has the additional consequence that the CDIFT can not depend on other national infrastructure, such as simulation capabilities or “live” information resources.
- On any particular participant’s LAN for development or integration of national capability. This has the consequence that the CDIFT can not rely on any particular coalition capability that can not be shared with coalition partners (but that may be available during technology showcases or demonstrations).
- On a coalition research and development (R&D) wide area network (WAN) for development, demonstration, and experimentation. Being an R&D environment, this often has limited access to national simulation capabilities and ‘live’ information resources.

6.2.4 Integrate Heterogeneous Systems

The goal of CDIFT1 is to demonstrate, test, and evaluate a variety of information fusion capabilities that could well exploit different underlying technologies. Thus, the CDIFT needs to facilitate integration of a variety of technologies and not mandate any particular solution. This requires the development of common interfaces between systems, but *not necessarily* the use of monolithic common data models if sufficient information exchange mechanisms can be established.

6.2.5 Loose Coupling Between Components

The availability of any particular national capability can not be guaranteed within a coalition system (depending on the collaboration model that is applicable), and as the CDIFT is intended as a testbed for testing and evaluation of new fusion capabilities (including infrastructure), it is expected that the capabilities deployed within the CDIFT may change and evolve as technologies prove suitable or otherwise. Thus, a loose coupling between fusion elements on the CDIFT is required so that no capability depends on another that may or may not be available at any given time.

6.2.6 Dynamic Resource Management and Process Control

Dynamic resource management and process control is not considered to be critical for demonstration and evaluation of higher-level fusion systems on the CDIFT in all cases, but is certainly a desirable capability for coalition systems that should not be precluded.

CoAX (Collaboration 2002 Experiment)

While the preceding requirements are driven by the need for an environment for defence R&D into higher-level fusion in a coalition context, they could also apply to operational higher-level fusion systems involving multiple organizations in other domains, where commercial rather than national security imperatives apply.

The CoAX 2002 experiment [2, 3] demonstrated how multi-agent systems could be used to meet requirements similar to these, using three key technologies:

1. Control of Agent Based System (CoABS) Grid middleware [4], based on Java Jini technology, provided a decentralized, scaleable, and robust framework for integrating distributed agent systems and legacy services that supported dynamic advertisement and discovery of new capabilities as they are added.
2. DAML (DARPA Agent Markup Language) [5], a precursor to Ontology Web Language–Description Logics (OWL-DL), when used to describe agent capabilities, provided a common description language that allowed agents to understand the capabilities available on the CoABS Grid so that they could be exploited, and allowed the rapid generation of interfaces between different agent systems and thus the rapid integration of new capabilities.
3. Knowledge Acquisition in automated Specifcation (KAoS) policy and domain services [6] provided a policy-based framework (using DARPA Agent Markup Language (DAML)) for controlling access to different types and *quality* of information provided on the grid by different coalition participants.

CoAX 2002, shown in Figure 6.2, demonstrated how, with these technologies, a distributed, heterogeneous network of agent systems from more than 14 different organizations could be rapidly integrated and used to provide agile and flexible command and control systems for operations within ad hoc coalitions. The CoSAR-TS program [7] subsequently demonstrated how these capabilities could be adapted to utilize Semantic Web technologies in a coalition search and rescue scenario.

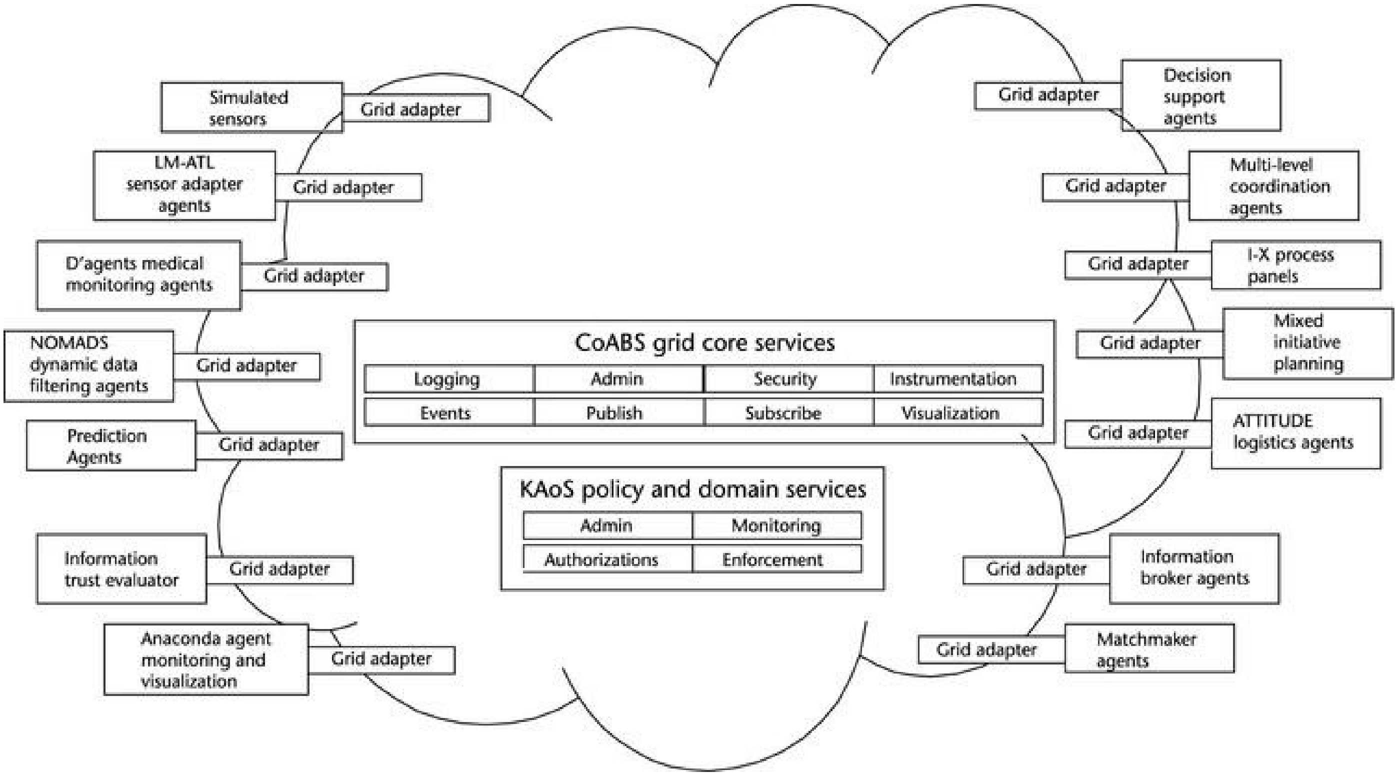


Figure 6.2 CoAX 2002 experiment utilized multiagent systems based on CoABS grid technology to demonstrate dynamic integration of heterogeneous C2 systems in an ad hoc coalition.

The success of the CoAX experiment inspired in part the development of the CDIFT, with some important differences:

- It was not desirable to impose an agent paradigm on higher-level fusion systems deployed in the CDIFT, as one of the roles of CDIFT is to explore the efficacy of different technologies in the higher-level fusion domain.
- It was not critical in the CDIFT to support rapid integration of new capabilities from ad hoc coalition participants, a feature demonstrated in the CoAX experiment and enabled by the agent-based paradigm, coupled with a common semantic description of agent capabilities using DAML.

Architecture

Another view of the CDIFT looks at the functional layers that must be implemented, as illustrated in Figure 6.3.

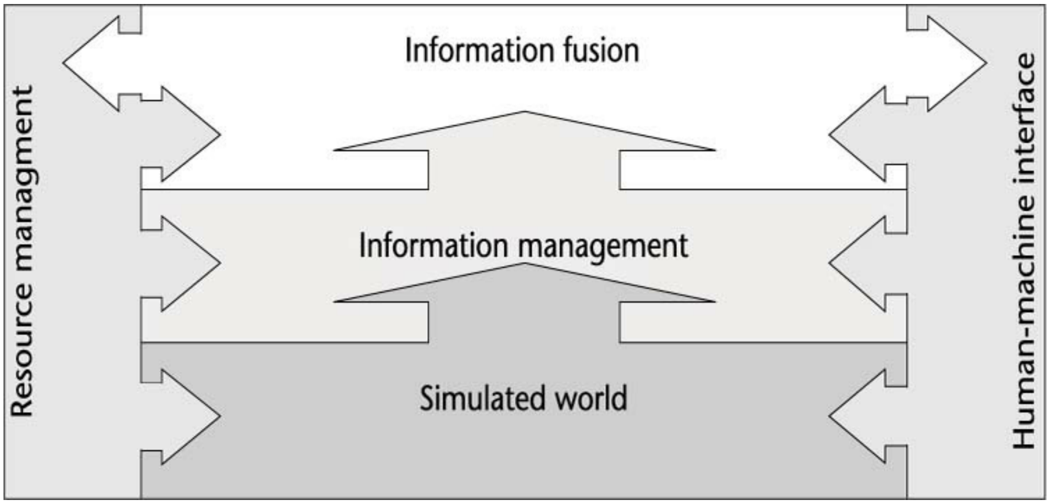


Figure 6.3 Functional elements needed in CDIFT.

6.4.1 Simulation Layer

The generation of a simulated world containing the information sources and resources is needed to stimulate the information fusion capabilities deployed on the CDIFT in a well-known and reproducible manner.

To align with existing coalition capability, a High Level Architecture (HLA) [8] was chosen to provide a synchronous distributed simulation capability using the DMSO RTI-1.3NGv6 run-time infrastructure with the RPR v1.0 FOM [9]. The STAGE [10] simulation software from Presagis is used to generate platform ground truth to suit scenarios of interest. Recorded ground truth can also be replayed when human-in-the-loop or dynamic resource management is not required. Other HLA federates can be deployed in the simulation federation to represent sensors, trackers, observers, and other information sources in the simulated environment. The run-time infrastructure ensures that all HLA federates are synchronized, and allows events to be played in real time, faster than real time, or slower than real time if needed.

For initial development of the CDIFT, a vignette in the NATO North Atlantis scenario [11] (see Chapter 12) was modeled in STAGE, although the general capability provided by CDIFT does not preclude other scenarios being utilized.

6.4.2 Information Management Layer

Infrastructure is needed to manage the flow of information between different fusion capabilities deployed on the CDIFT. A publish/subscribe mechanism was chosen to support efficient dissemination of information and fusion products to multiple consumers. This is important, for example, when multiple fusion elements require the same information sources, or a fusion product may be reused by other capabilities.

The Joint Battlespace Infosphere (JBI) v1.2.6 [12] developed at the Air Force Research Laboratory (AFRL) provides the core information management layer in CDIFT, although publish/subscribe mechanisms using the CoABS grid and Elvin [13] have also been implemented. JBI (see Chapter 5) allows different systems to subscribe to nominated Managed Information Object (MIO) with metadata, and in some cases content, matching nominated conditions. In CDIFT, simulated sensors, trackers, and other information sources publish detections/reports over JBI (or equivalent publish/subscribe layer, shown in Figure 6.4), which are then available to any number of deployed fusion systems subscribing to this information. Fusion products are available in embedded displays associated with the fusion capability, and depending on the collaboration model employed, may also be published as additional information available to other capabilities for further analysis. The use of a common information management layer for dissemination between fusion capabilities does not preclude the use of embedded information management infrastructure within a capability to access, for example, embedded information sources like databases required by a particular capability.

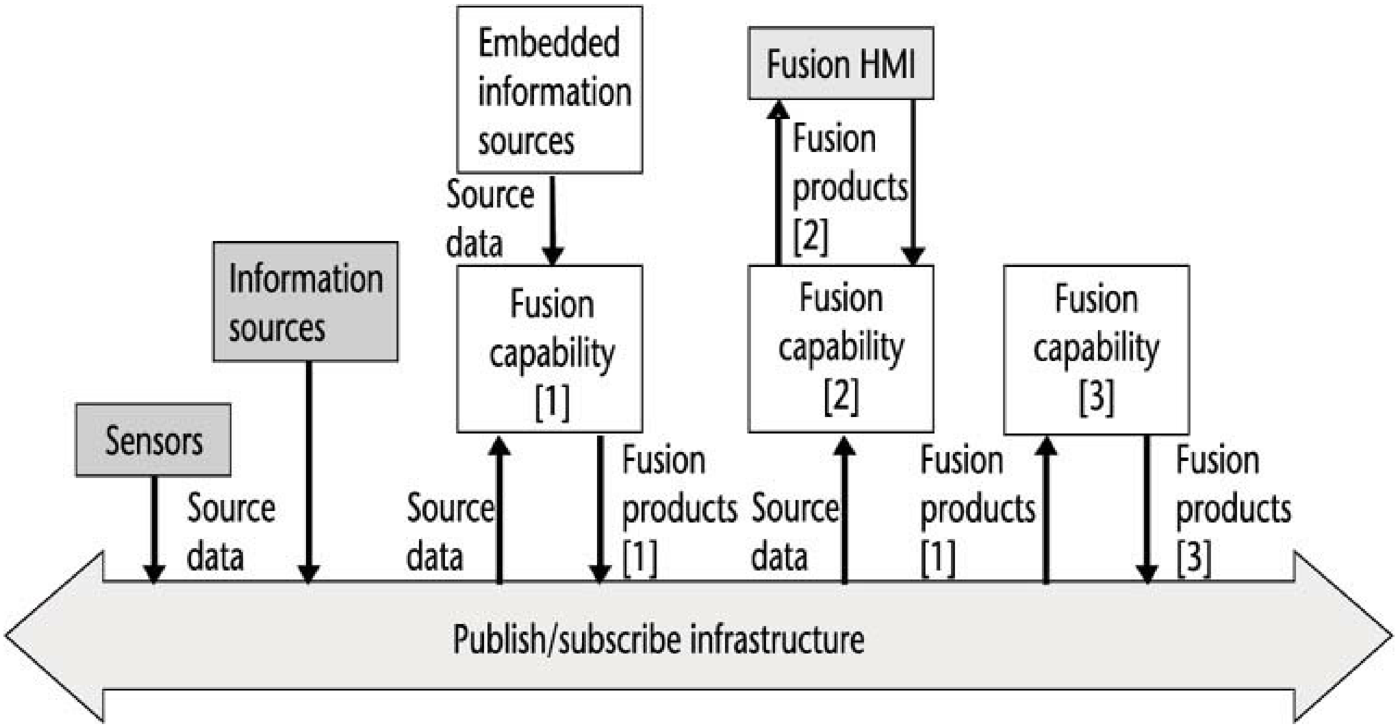


Figure 6.4 A publish/subscribe mechanism allows information to be shared with multiple consumers simultaneously, and chaining of fusion capabilities.

Dynamic policy control of access to information, based on attributes associated with a fusion application or user, has also been demonstrated based on the draft eXtensible Access Control Markup Language (XACML) 3.0 specification [14], and implemented in Axiomatics' Delegent product [15].

XACML defines an XML-based policy specification language and policy management architecture, shown in Figure 6.5, but not the implementation of this architecture. XACML policies permit or deny a request by an application, and can optionally specify advice to be provided to the application (e.g., ”Request was refused because…”) and obligations that must be met (e.g., ”Request permitted provided request logged here”). The draft XACML 3.0 standard also supports delegation, so that authorization rights can be dynamically delegated to other users or applications.

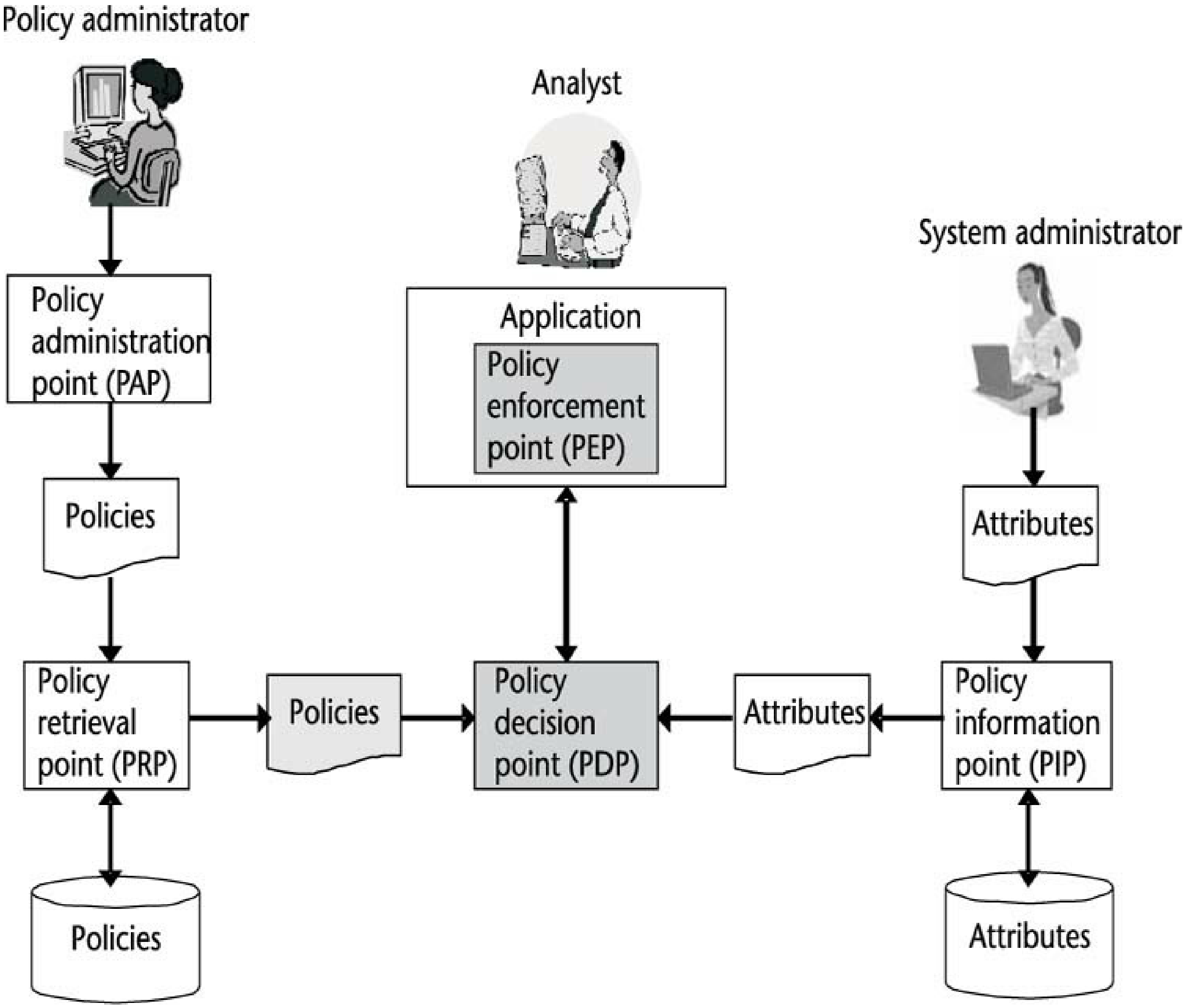


Figure 6.5 XACML policy management and enforcement architecture.

The XACML architecture is designed to operate in a distributed fashion, and has the following key elements:

1. Policy Administration Point (PAP)—an interface to manage XACML policies and specify policy combining algorithms.
2. Policy Retrieval Point (PRP)—manages storage and retrieval of policies.
3. Policy Information Point (PIP)—manages storage and retrieval of attributes associated with the user, application, or environment. This could be as simple as a Lightweight Directory Access Protocol (LDAP) service.
4. Policy Decision Point (PDP)—implements decisions on application requests according to the XACML specification, based on the applicable policies and attributes.
5. Policy Enforcement Point (PEP)—enforces the application requests based on the decisions made by the PDP. This is also responsible for ensuring that policy obligations are met.

XACML was used to demonstrate control of access to information within CDIFT, but could also be used to provide similar capabilities to those demonstrated by KAoS in the CoAX experiment. For example, a policy may permit access to an information source provided that an obligation is met that degrades the information feed or removes national security classified information.

6.4.3 Information Fusion Layer

The information fusion capabilities deployed on the CDIFT may be implemented as single applications that interface directly to the information management layer. However, often the complexity of higher-level information fusion processes means that national higher-level fusion capabilities may themselves require the coordination of multiple distributed components. For example, databases and ontologies may be needed by a particular reasoning engine employed within a fusion capability. Within the CDIFT, there are several different distributed models in use by different fusion capabilities (detailed in Chapter 12):

- AFRL’s Fusion 2+ testbed [16] employs distributed components that communicate via JBI. Each component of the Fusion 2+ testbed publishes its output as another IOT, which is then available to other components as needed.
- The Defence Science and Technology Organization (DSTO) has implemented a multi-agent architecture based on the Attitude [17] agent system using the CoABS grid, Elvin, or eXtensible Messaging and Presence Protocol (XMPP) [18] for

point-to-point and broadcast inter-agent messaging. To achieve this, DSTO implemented a generic agent interface in Java that integrated with Attitude agents (shown in Figure 6.6), or wrapped the standard I/O streams for other (possibly legacy) applications. It also interfaces to one or more of the JBI, CoABS grid, or Elvin publish/subscribe mechanisms by providing a proxy agent that handles transactions with the publish/subscribe infrastructure via agent performatives. This interface abstracts the details of the underlying agent messaging and publish/subscribe infrastructures away from the agent communication protocols and content employed by the deployed agents, so that the multi-agent capability is independent of the underlying CDIFT infrastructure.

Agents and wrapped processes are treated in a similar way by the interface, except that agents are able to dynamically redirect and reform queries and dynamically change communication protocols, while wrapped applications have fixed communication targets and protocols determined at initialization by the configuration parameters used for the wrapper. Where adaptive behavior is required from wrapped applications, a proxy agent is deployed to handle transactions with the wrapped application, as shown in Figure 6.7.

- DSTO has also implemented interfaces to web services (shown in Figure 6.8) providing a geospatial querying and reasoning system. In addition, access to selected components of the CDIFT multi-agent system are also exposed via web services that act as proxy agents, translating, and handing off web service requests to the appropriate agent capability and returning the result.

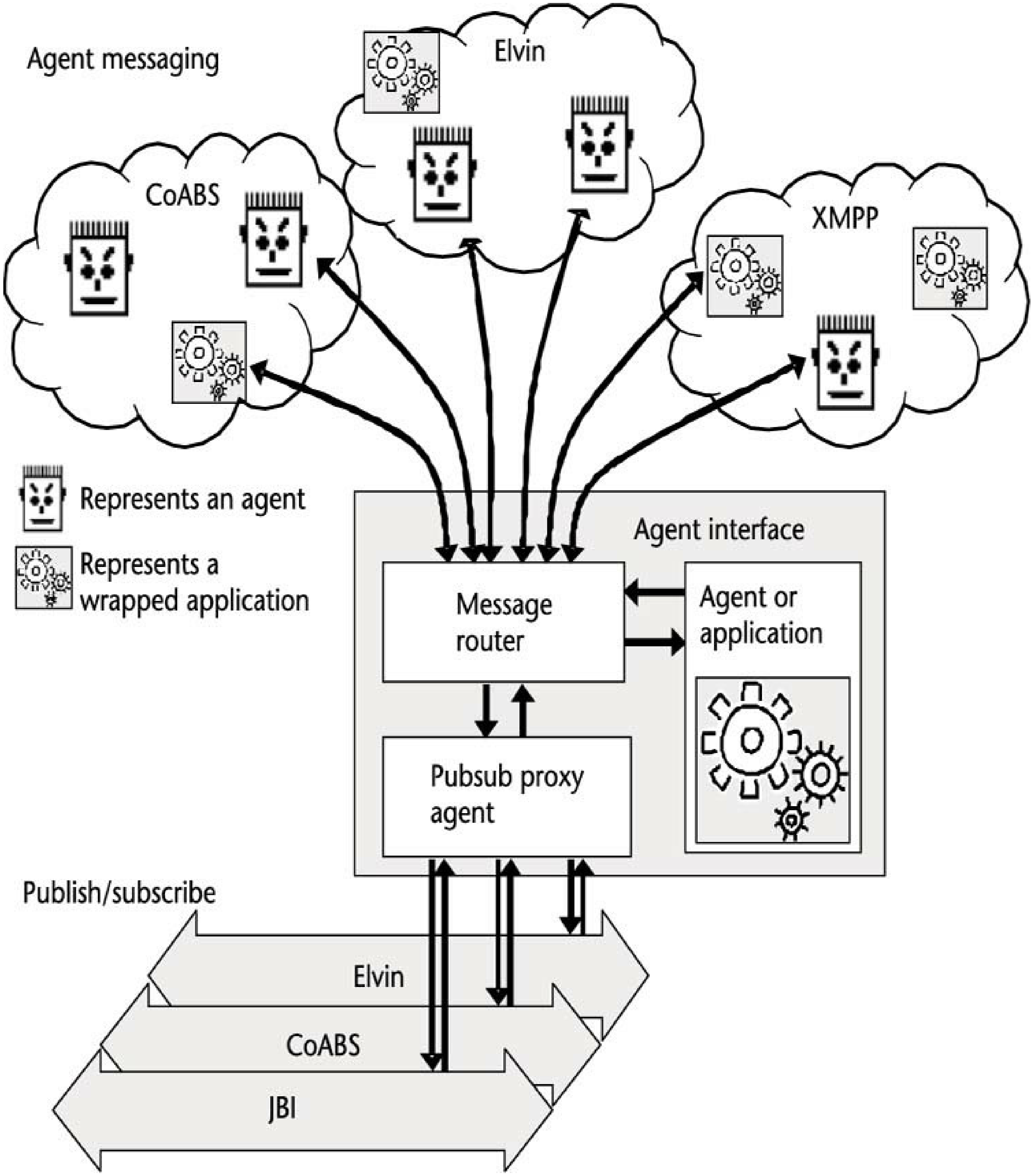


Figure 6.6 Generic agent interface developed by DSTO for CDIFT to wrap applications and abstract interface to publish/subscribe and agent messaging infrastructure.

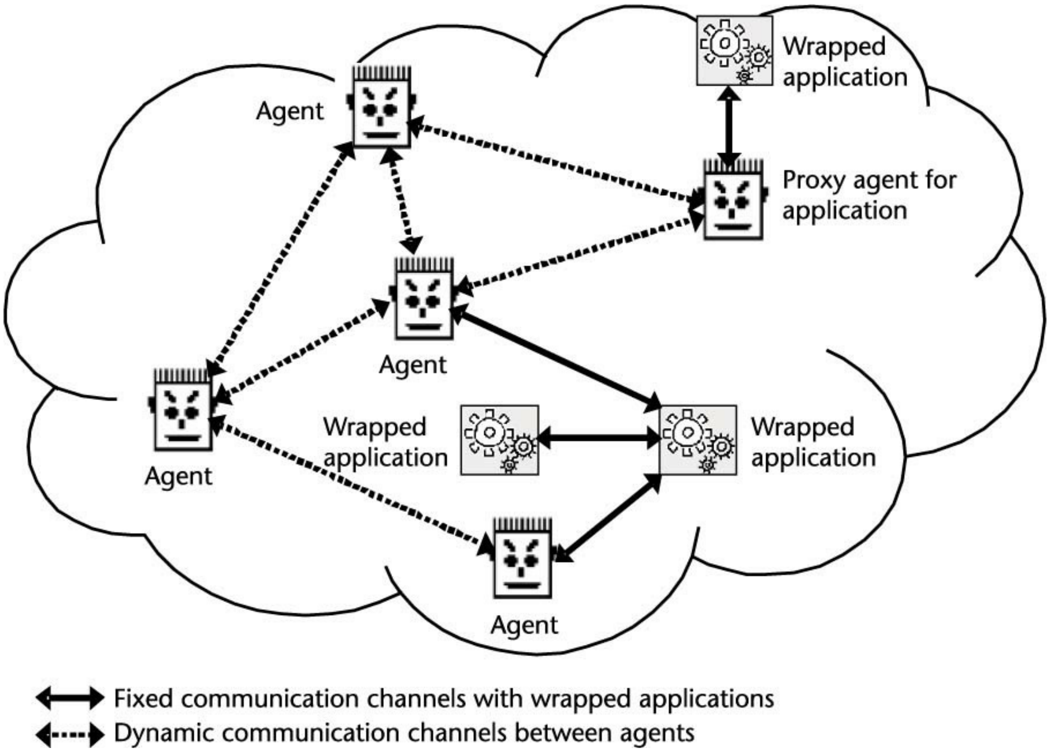


Figure 6.7 Proxy agents can be deployed for managing dynamic interactions with wrapped applications, as wrapped applications are constrained by initial configuration.

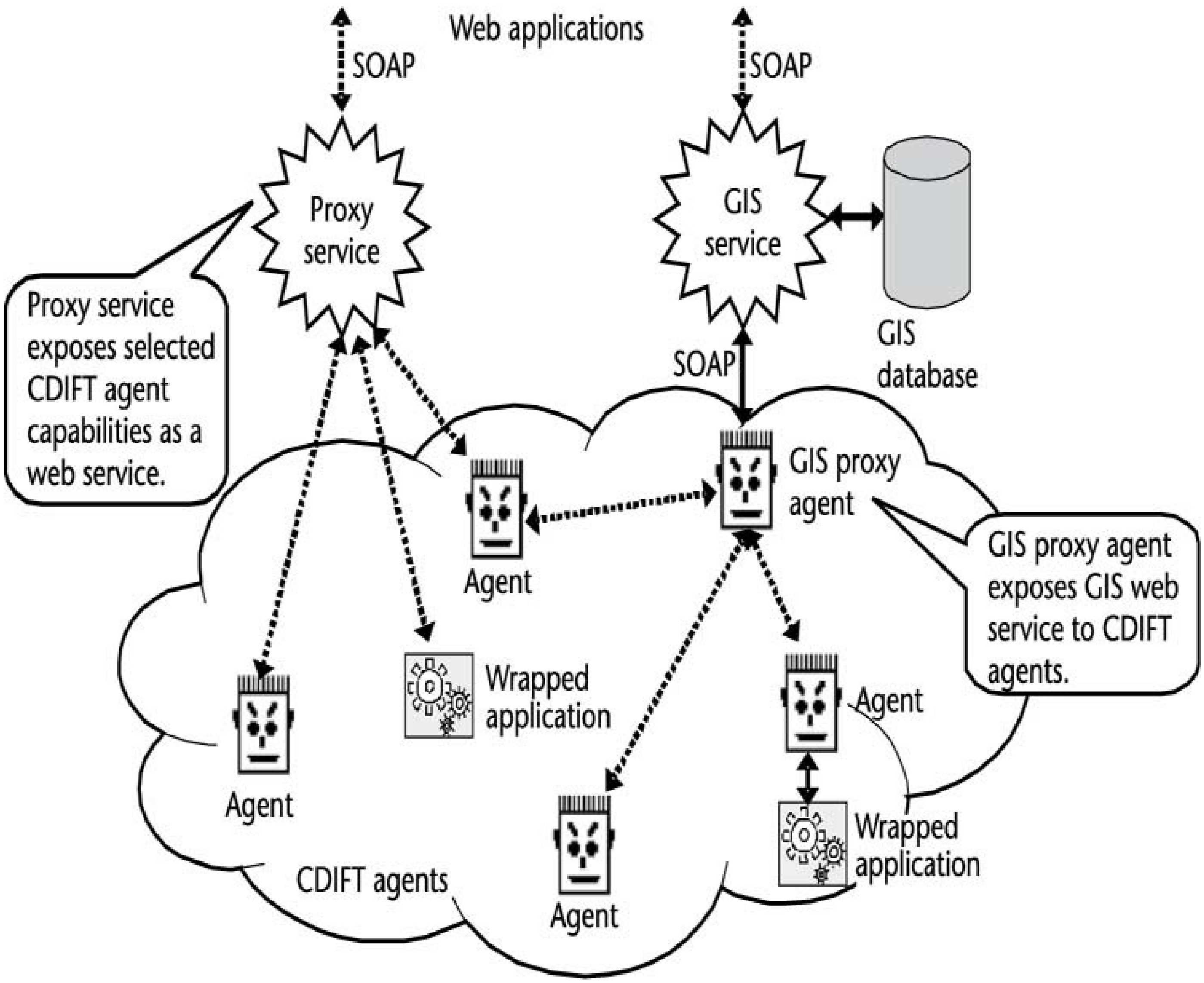


Figure 6.8 Web service interfaces implemented to and from CDIFT agents.

6.4.4 Resource Management Layer

Resource management is needed across each of the information fusion, information management, and simulation layers.

1. Across the information fusion layer, it allows the information fusion processes to be adapted to suit changing operating conditions and to optimize performance of the fusion system.
2. Across the information management layer it allows the information shared with coalition partners to be dynamically changed and the infrastructure managed to maintain the desired quality of service. The use of dynamic policy control by capabilities, such as XACML or KAoS, to manage access to information is an example of resource management in this context.

3. Across the simulation layer (or operational layer, if connected to a live C2 system) resource management involves the tasking of sensors, platforms, or other capabilities to achieve mission objectives. In the case of simulation, this requires deployment of a closed-loop simulation capability such as STAGE, instead of simply replaying pregenerated ground truth.

Resource management can also span all of these layers. For example, an information fusion system may request greater information throughput from the information management layer, or may request a sensor to be retasked so that it can resolve ambiguities in its assessment of a situation. Invariably, the resource management layer will also closely parallel the human-machine interface layer, as capabilities that can not be resolved automatically may be handed off to a human operator.

Different technologies are required for resource management at each layer of the CDIFT architecture.

6.4.5 Human-Machine Interface Layer

As with resource management, Human-machine Interfaces (HMI) are needed for each of the information fusion, information management, and simulation layers.

1. Across the information fusion layer, HMI allows an analyst to employ an information fusion system to address a particular problem of interest, and to fine-tune its operating parameters to suit that problem. Ultimately, the role of a fusion system is to support the situational awareness of a decision maker, and the fusion system needs to be able to effectively display its output to the decision maker to achieve this. The particular format of the HMI needed is highly dependent on the capabilities of the fusion system and the requirements of the analyst and decision maker. The HMI may be embedded within a particular fusion capability (indeed, the analyst may well be an integral part of that capability), or be a generic capability that the analyst/ decision maker tunes to suit their particular needs. This is the concept behind the User Defined Operating Picture (UDOP) in Chapter 9 [19].
2. Across the information management layer, HMI allows an administrator to manage the performance of the system, establish the credentials and authorizations of coalition partners, set up policies for information sharing within the coalition, and set up information object specifications in JBI or equivalent publish/subscribe infrastructure (see Chapter 5).
3. Across the simulation layer HMI allows the simulation engine or playback application to be managed to suit particular demonstrations or experiments, and

scenario ground-truth to be displayed for comparison with information fusion system outputs (Chapter 13). When human-in-the-loop experimentation is required, HMI allows an operator to interact with the simulated world and effect change in response to changing conditions; for example, if a sensor platform needs to be manually re-tasked.

As with resource management, the HMI technologies and formats required for particular roles vary greatly. However, in some cases a User Defined Operating Picture (UDOP) capability can be used to provide a common HMI spanning some of these requirements, by utilising the capabilities of JBI, or equivalent publish/ subscribe infrastructure. There are two approaches to this that have been pursued in the CDIFT.

1. Provide a common display that can subscribe to different MIO published over JBI. The MIO of interest (determined by the metadata, and in some cases content, associated with the MIO) can be either preconfgured or determined by the user at run-time. This approach has been incorporated into DSTO's Higher-level COP (HICOP) capability [20], which is shown in Figure 6.9. In CDIFT, HiCOP has been used as a UDOP to display simulation ground truth, tracker plots, or track clusters based on the MIO subscriptions selected, and give simple verbal summaries from animated characters of particular events of interest to the user. With technologies such as the HICOP a much wider range of MIO content can also be presented, depending on the particular role needed.
2. A simple report subscriber application has been developed by DSTO to subscribe to a tailored MIO developed to represent fusion reports. This IOT can contain different types of content, which can be rendered in different ways based on the users' configuration to suit their information needs. The metadata associated with this MIO includes:
   - *Context* a collection of tags associated with the content. By choosing to subscribe to reports with only certain tags, the user is able to tailor the information presented to them.
   - *Type* is a particular tag describing the type of report. This is used to provide finer grained control of the subscription. For example, the user may only be interested in status reports and incident reports.
   - *Format* along with the user configuration, the format of the report determines how it will be rendered and presented to them. For example, text reports could be rendered as a text display, or presented verbally by an animated character. The user can also choose to set up the subscription so that only reports with particular formats are rendered. For example, they may not be interested in location reports.
   - *Priority* reports with priority less than or equal to 1 are queued based on the

priority, while reports with priority greater than 1 will interrupt reports of lower priority. This is an important distinction, for example, if the report is rendered as a verbal notification by an animated character. The user can set a threshold for subscription to reports based on the priority so that only reports of higher priority are presented to them.

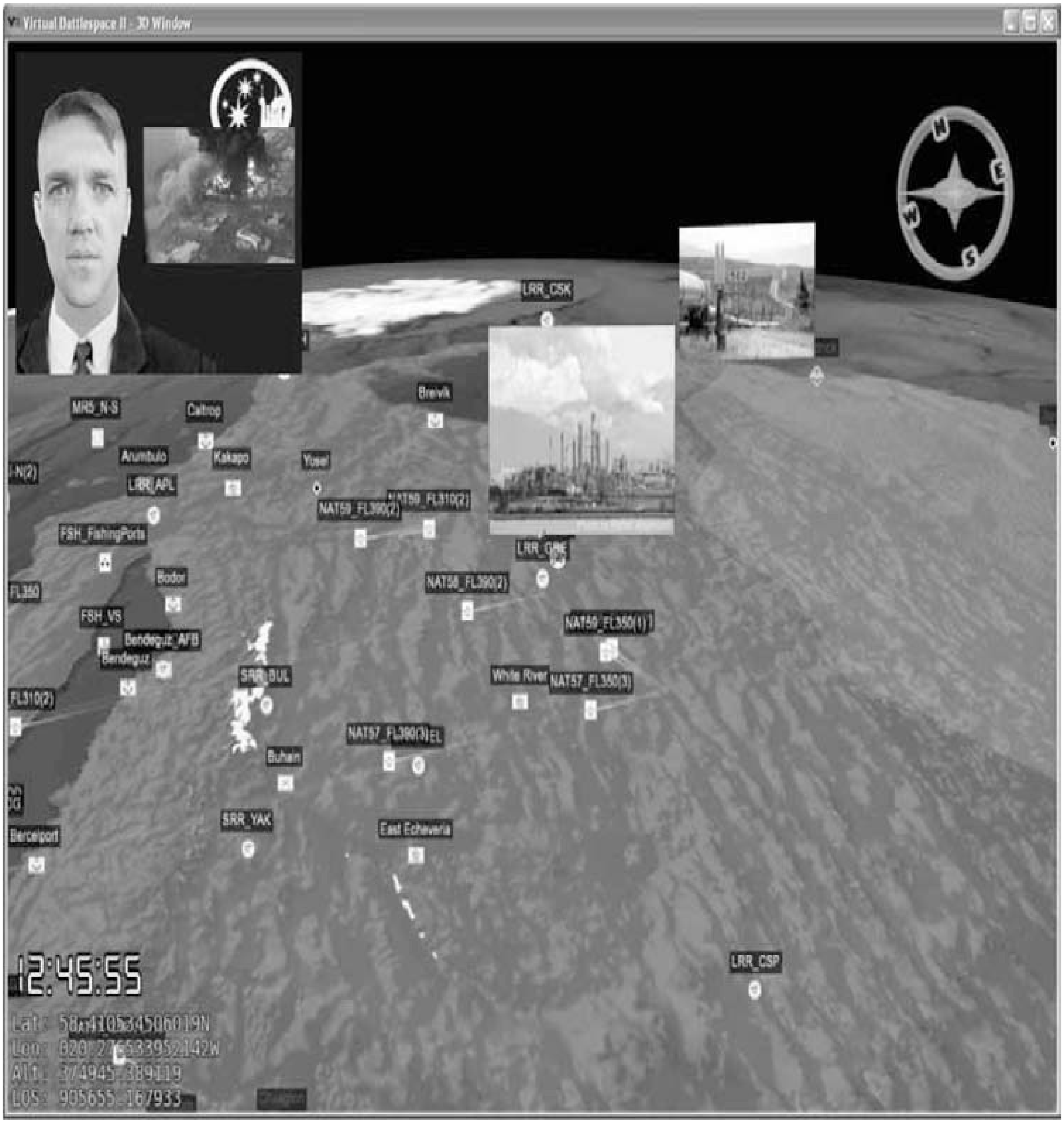


Figure 6.9 Higher-level COP incorporates geospatial, track, and embedded multimedia information with animated characters to tell the story behind the data. In CDIFT the information displayed is user-defned through subscription to

JBI. This example shows tracks and media relevant to the TTCP "Military Strikes in Atlantis" scenario ([11]).

The display and reporting concept can be quite powerful as a UDOP capability when coupled with a publish/subscribe mechanism, as shown in Figure 6.10.

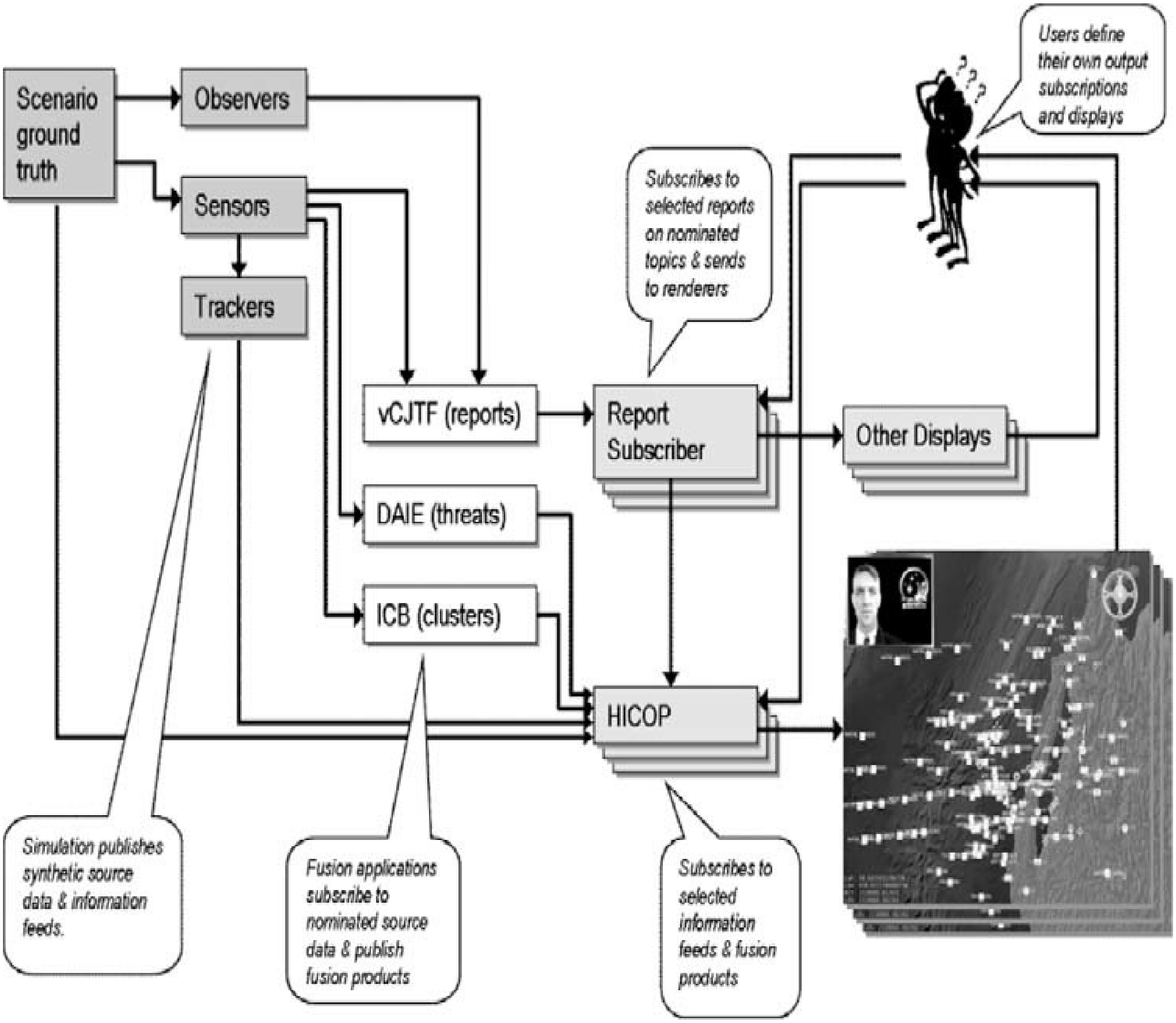


Figure 6.10 Example information flows in CDIFT support users defning their own view of the operating picture.

Most of the HMI work currently demonstrated in CDIFT has focussed on providing a display capability, with control interfaces designed specifically for each particular capability.

Conclusion

The CDIFT provides a flexible distributed environment that supports collaborative development and testing of fusion capabilities in a coalition context. It provides a simulation framework to support the generation of synthetic data, and it provides a publish/subscribe infrastructure that facilitates the integration of coalition fusion capabilities. CDIFT also provides a multi-agent infrastructure that can be used for the integration of fusion components based on software agents.

Current work in the CDIFT environment includes the incorporation of additional higher-level fusion systems developed at DSTO based on the State Transition Data Fusion (STDF) model [21], using agents developed with the Mephisto conceptual framework [22] and coordinated through the Linda process communication model [23]. In this system, the various sensor, effector, and cognition components for each agent communicate through separate processes. These agent components interface to the CDIFT infrastructure via the generic agent interface developed at DSTO.

Some preliminary work has been done in CDIFT on the use of policies to dynamically control access to information within a coalition, using the XACML framework. This could support tuning of the information shared within a coalition based on the current roles and immediate information needs (i.e., need-to-know). An interesting area of further investigation could be to use policies to dynamically filter the information provided to coalition systems so as to restrict it to particular topics or classification levels based on, for example, the current region or activity of interest (AOI), instead of simply denying or permitting access to the entire body of information. A higher-level fusion system could inform the AOI process, as well as use it. Further discussion of the CDIFT is found in Chapter 12

rences

[1] Lambert, D. A., “TTCP C3I TP1 Annual Report,” *TTCP C3I Annual Meeting*, Williamsburg, , 2008.

[2] Allsopp, D. N., et al., “Coalition Agents Experiments: Multiagent Cooperation in International Coalitions,” *IEEE Intelligent Systems*, May/June 2002, pp. 26–35.

[3] Wark, S., et al., “Dynamic Agent Systems in the CoAX Binni 2002 Experiment,” *Intl Conf. on Information Fusion*, 2003.

[4] Brake, D. E., and G. Emami, “Control of Agent Based Systems (CoABS) Grid,” AFRL Technical Report, *AFRL-IF-RS-TR-2004-169*, June 2004.

[5] Hendler, J., and D. L. McGuiness, “The DARPA Agent Markup Language,” *IEEE Intelligent Systems*, Vol. 15, No. 6, November 2000, pp. 67–73.

[6] Uszok, A., et al., “New Developments in Ontology-Based Policy Management: Increasing the Practicality and Comprehensiveness of KAoS,” *IEEE Workshop on Policies for Distributed Systems and Networks*, 2008.

[7] Tate, A., et al., “Coalition Search and Rescue—Task Support. Intelligent Task Achieving Agents on the Semantic Web,” AFRL Technical Report, *AFRL-IF-RS-TR-2006-91*, March 2006.

[8] IEEE Standard 1516.

[9] SISO Standard SISO-STD-001.1-1999.

10] http://www.presagis.com/products_services/products/ms/simulation/stage/.

11] Blanchette, M., “Military Strikes in Atlantis—A Baseline Scenario for Coalition Situation Analysis,” The Technical Cooperation Panel, *Technical Report TR-C3I-TP1-1-2005*, 2005.

12] Combs, V. T., et al., “Joint Battlespace Infosphere: Information Management within a C2 Enterprise,” *International Command and Control Symposium*, 2005.

13] Segall, B., and D. Arnold, “Elvin Has Left the Building: A Publish/Subscribe Notification Service with Quenching,” *Proceedings of the AUUG*, 1997.

14] http://www.oasis-open.org/committtees/tc_home/php?wg_abbrev=xacml.

15] Seitz, L., et al., “Policy Administration Control and Delegation using XACML and Delegent,” *Proceedings IEEE/ACM Intl Workshop on Grid Computing*, 2005.

16] Salerno, J. J., M. Hinman, and D. Boulware, “A Situation Awareness Model Applied To Multiple Domains,” *Proceedings of SPIE*, Vol. 5813, 2005.

17] Lambert, D., “Advisers with Attitude for Situation Awareness,” *Workshop on Defense Applications of Signal Processing*, La Salle, IL, 1999.

18] Extensible Messaging and Presence Protocol Standards Fountation, http://wmpp.org.

19] Mulgund, S., and S. Landsman, “User Defined Operational Pictures for Tailored Situation Awareness,” *International Command and Control Research and Technology Symposium*, 2007.

20] Wark, S., et al., “Situational Awareness: Beyond Dots on Maps to Virtually Anywhere,”*SimTecT*, Adelaide, Australia, 2009.

21] Lambert, D., “A Unification of Sensor and Higher-Level Fusion,” *International Conference on Information Fusion*, 2006.

22] Lambert, D., and C. Nowak, “Mephisto I: Towards a Formal Theory,” *Proceedings of the 2nd Australasian workshop on Advance in Ontologies (AOW’06)*, Darlinghurst, Australia, 2006.

23] Carriero, N., and D. Gelernter, “Linda in Context,” *Commission of the ACM*, Vol. 32, No. 4, April 1989, pp. 444–458.

---

[1] For example, the CDIFT was demonstrated at the International Conference on Information Fusion (Fusion 2009), July 2009, Seattle, WA.

CHAPTER 7

Information Fusion and Resource Management Testbed

Adel Guitouni, Pierre Valin, Éloi Bossé, and Hans Wehn (Canada)

The Information Fusion and Resource Management Laboratory (INFORM Lab; previously CanCoastWatch) is an advanced simulation testbed for the purpose of evaluating the effectiveness of network enabled operations in a coastal wide area surveillance situation, with algorithms provided by several universities. This INFORM Lab testbed allows experimenting with high-level distributed information fusion, dynamic resource management, and configuration management, given multiple constraints on the resources and their communications networks. This chapter describes the architecture of INFORM Lab, the essential concepts of goals and situation evidence, a selected set of algorithms for distributed information fusion and dynamic resource management, as well as auto-configurable information fusion architectures. The testbed provides general services which include a multilayer plug-and-play architecture, and a general multi-agent framework based on John Boyd's OODA loop described in Chapter 2 and 10. The testbed's performance is illustrated on a noncooperative search scenario involving many target ships and various methods of deceit.

Introduction

The INFORM Lab testbed [1] is a plug-and-play environment using a highly adaptive and autoconfigurable, multilayer network architecture for large-volume distributed information fusion and resource management. It addresses challenges in military intelligence, surveillance and reconnaissance, assuming a multitude of different sensor types on multiple mobile platforms. INFORM Lab allows users to study and compare the performance of different algorithms for higher-level distributed data/information fusion, dynamic resource management, and configuration management under various multiple resource and communication constraints.

The problem of testing information fusion (IF) and resource management (RM) algorithms and different approaches requires the use of a testbed sufficiently flexible to perform a variety of what-if analyses and risk assessment. As will be shown in the next section, a search of the literature concerning existing testbeds shows that there is lack of a closed-loop distributed information fusion and dynamic resource management that is not done anywhere else. The testbed must be able to do open and, more importantly, closed loop testing. It should be flexible enough to allow easy generation of scenarios, and instantiations of such scenarios, called vignettes.

Succinctly stated, it should provide partial answers the following questions:

- How does one control and optimize IF?
- How does one manage networks, communications, platforms and distributed IF?
- How does one compare different solutions?
- How does one provide validated advice?

In this chapter, the main characteristics of this distributed research testbed are described.

INFORM Lab Architecture

The testbed provides a general multi-agent architecture based on John Boyd's Observe, Orient, Decide, Act (OODA) loop [2]. Due to the general nature of the OODA agents, very diverse elements can be modeled by it. They include ships, airplanes and fixed radar stations, but also collections of assets such as squadrons, or individual sensors if required. This great flexibility is further enhanced by a multilayer plug-and play architecture that lets researchers easily add their own algorithms to INFORM Lab. The testbed provides general services that are useful for testing surveillance applications. The main components of INFORM Lab are shown in Figure 7.1.

The user-defined *OODA agents*, called nodes in INFORM Lab, are at the centre of the system. They are supported by an editor, a testbed proper, and a viewer. The editor makes it easy to configure the simulation. For example, it allows for the specification of the node behavior, and the setting up of the scenario and the environmental conditions. The editor also allows specification of the relationships between nodes, such as one node being the superior of another node. The output of the editor is an eXtensible Markup Language (XML) file that contains all the information needed to run the simulation. This configuration file is passed to the testbed, which then runs the simulation. The testbed also provides convenient services to the nodes. For example, the testbed maintains the simulation time, and other global run-time information and metrics, which can be accessed by the nodes via a convenient Application Programming Interface (API). The output of the testbed is a log file, again in XML format, that can be passed on to a viewer. The viewer allows visualization of the movements of the nodes as a function of time. It also shows the nodes against a Geographic Information System (GIS) background and environmental factors, such as developing fog banks.

Externally, nodes are characterized by their ability to communicate with other nodes by sending messages via a simulated communication network. For realism, the communication links can be given the characteristics of known standards, such as link-11 or radio. In INFORM Lab, the nodes usually exchange messages that contain orders, requests, or information. The situation is illustrated in Figure 7.2.

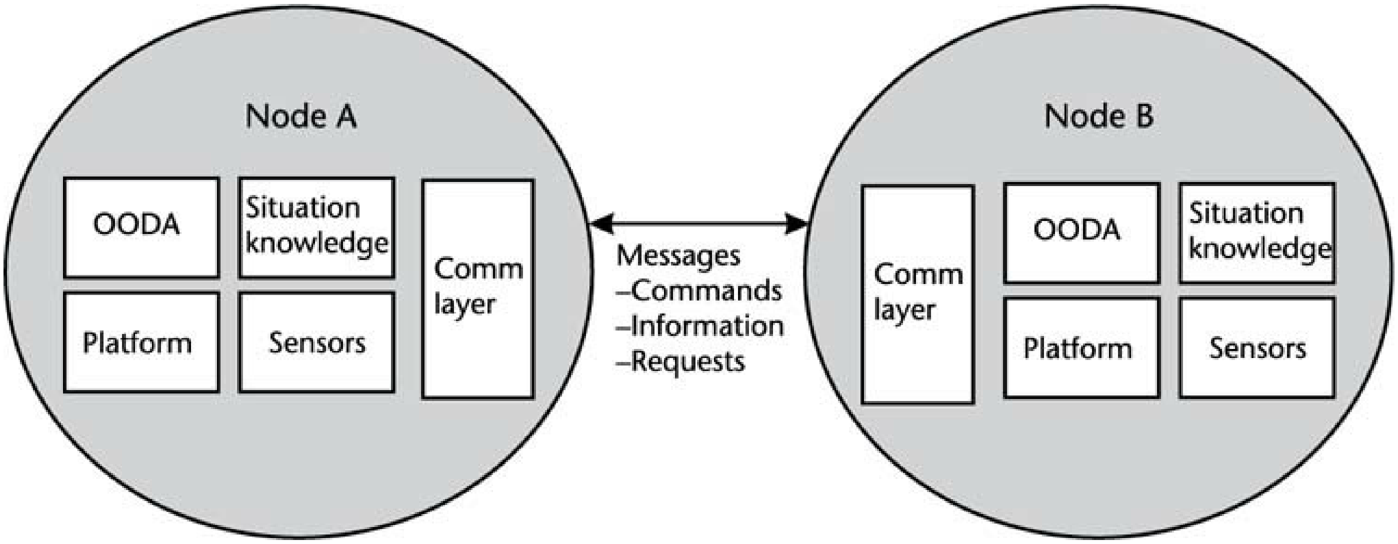


Figure 7.2 OODA nodes overview.

Also, the nodes have other internal components besides communication equipment. A node can sense its surroundings via its built-in sensor(s), and it can move using its built-in platform. It also has situation knowledge, which is all that the node knows about the world. A particularly important part of the node's situation knowledge is the measurement-derived situation evidence, which will be discussed later in more detail. A node's behavior is determined by its OODA components: Observe, Orient, Decide, and Act:

- The Observing function of a node corresponds to a Level-1 data fusion capability;
- The Orienting function corresponds to Level-2 and Level-3 information fusion;
- The Deciding function performs the Resource Management task;
- The Acting function implements the decisions made by the Deciding function of the node.

INFORM Lab uses two architectural approaches that support future research into large area surveillance algorithms, and facilitate addition and modification of the testbed:

- Component-based standard node architecture allows researchers to design replacement components at whatever architectural level of interest to them and then use the new algorithms they wish to explore. This is achieved via a standard interface defined for each component in the architecture.
- Plug-and-play mechanism that provides standard component addition or replacement. This includes OODA components such as the deciding box, or sub-

components, such as the planner. But it also includes the Environmental model, or components of the communication model.

The flexibility of this architecture will allow researchers to easily update or replace components and hence investigate many different approaches to data/information fusion and resource management with no software development beyond the content of the component they wish to redesign.

Other systems have been developed which support some of the functionalities of INFORM Lab. A Markov random field model of context for high-level information fusion has been proposed in High-Level Information Fusion Evaluation (HiLIFE) system [3], a maritime surveillance application which is pursued by the North Atlantic Treaty Organization (NATO) Undersea Research Center using real data offers an overall closed-loop surveillance system [4], and finally the Coalition Distributed Information Fusion Testbed (CDIFT) (see Chapters 6 and 12), where concepts of INFORM Lab could guide the development of a closed-loop version of CDIFT to address Command and Control (C2) problems.

7.2.1 OODA Agent Components

The OODA concept was introduced in Chapter 2. INFORM lab maps the OODA concept into the following functionality:

- Observe: track-level fusion, mostly Level-1 fusion;
- Orient: mostly Level-2 and Level-3 fusion;
- Decide: resource management, planning, scheduling, decision making;
- Act: implement decisions and turn them into actions.

OODA provides the processing functions of the agent. OODA has the advantage of being widely accepted in the military research community. OODA agents are also very flexible and can model anything from a camera to a fleet.

The components of an OODA agent are shown graphically in the Unified Modeling Language (UML) diagram Figure 7.3. The INFORM Lab OODA agents have the following, mostly optional, properties. An OODA agent:

- Is located on a platform. But multiple agents can be located on the same platform.
- May directly control a platform, for example by commanding it to move. But by convention, a platform is controlled directly by only one agent.
- Can have sensors, which allow the agent to learn some of the properties of other

platforms that are represented in the simulation.

- Can have actuators which allow an agent to impact other platforms and agents.
- Can have capabilities which are propositions expressing what an agent can do. This is useful for decision making.
- Can have decisions which the agent has not yet to act on.
- Can have a communicator, which allows an agent to communicate with other agents.
- Has a goal manager, which keeps track of the agent's goals. Goals are mission statements of what the agent should do.
- Has situation evidence, which represents what the agent has learned. It represents an agent's view of the current situation.
- Can have an Observing component.
- Can have an Orienting component.
- Can have an Deciding component.
- Can have an Acting component.

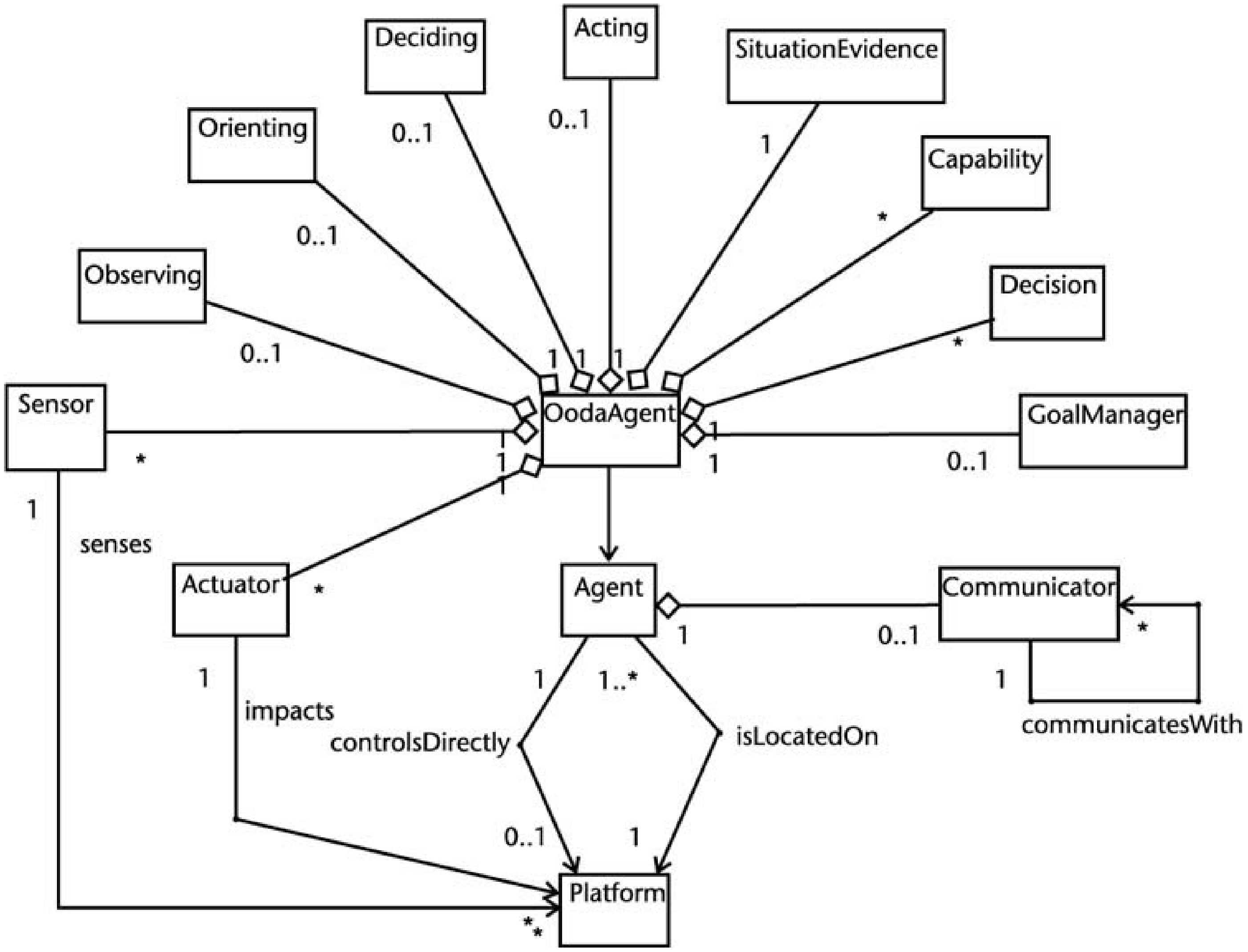


Figure 7.3 Parts of an OODA agent.

An OODA agent coordinates its multiple (and often optional) subcomponents, which include a goal manager, communicator, OODA, sensors, and actuators. The sensors of the agent, together with messages from other agents, are sources of information for the agent's situation evidence which then input to the observing component. The observing component aligns the provided situation evidence in space and time, generates tracks that are consistent among multiple agents, and provides other Level-1 information fusion services. The consolidated situation evidence is then passed on to the orienting component, where the intentions, impact, and relationships of the evidence objects are evaluated, resulting in a description of the current situation. The resulting situation evidence is subsequently passed on to the deciding component where it serves as a basis for decisions; for example, decisions are made where the platforms should move to in order to improve the sensor performance. Finally, the acting component implements the decisions; for example, it issues commands to the platform to move, or it sends goal or tasking orders to other agents via the communicator.

INFORM Lab is focused on providing fusion and resource management researchers with what they need to conveniently test their algorithms. For this reason, the INFORM Lab provides default implementations for all OODA components. In particular, it provides default implementations of platforms, sensors and communication networks. These implementations aim at a mid-range of model fidelity that is most appropriate for supporting the fusion and resource management algorithm research. Thus, error models tend to be simple Gaussian statistics. The models are simple to adapt to the users' needs, usually simply by providing parameters. For very specialized needs, only a small function needs to be added by the user in a plug-and-play fashion, rather than having to write a new implementation of these components from scratch. Nevertheless, all these components could in principle be swapped by totally new implementations. In what follows, the default implementations for these two components are briefly introduced.

7.2.2 Platforms

Platforms model the physical aspects of OODA agents as illustrated in Figure 7.4. They are the vehicles on which other elements, such as agents and sensors, are located. They are also the objects on which many tasking orders, like motion, refueling, and engagement, are performed. Hence, platforms are physical resources that have a physical appearance and other physical properties. These properties are defined via configurable attributes, such as width, height, or shape, and can be detected by some sensors.

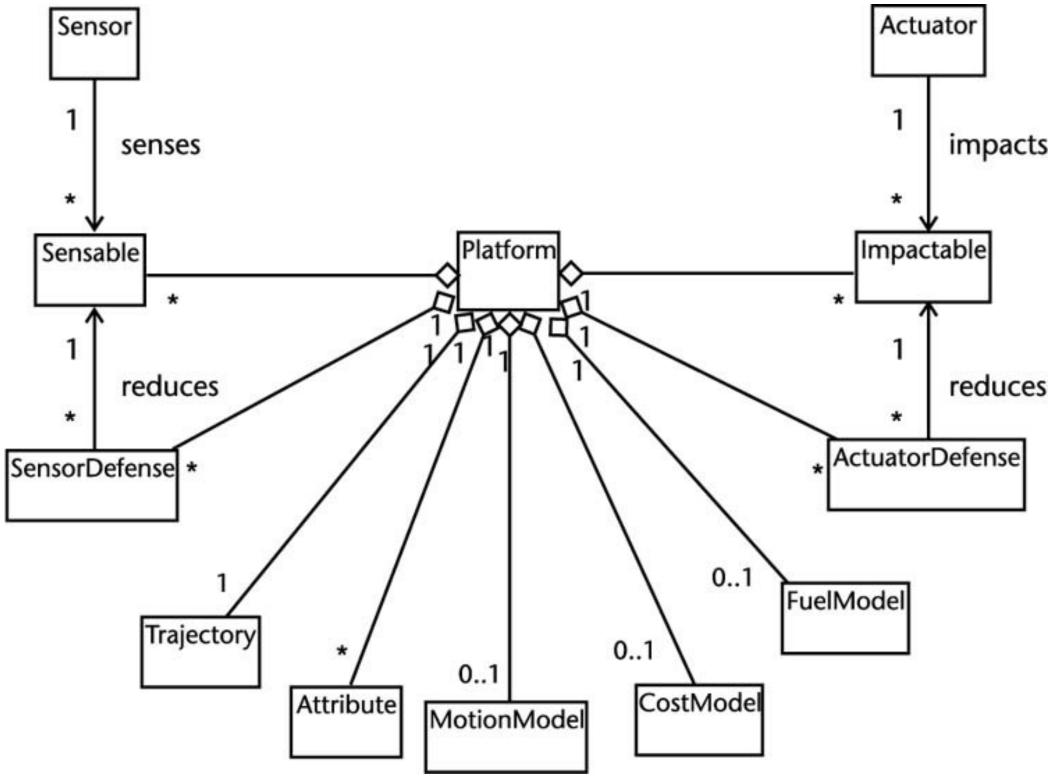


Figure 7.4 Platform context class diagram.

Most platforms are mobile; hence a platform has a trajectory that characterizing its motion. For stationary platforms, this trajectory consists of just one point. The platform motion information can be further refined by associating a motion model, fuel model, and operating cost model, which are particularly important for resource planning and scheduling.

Platforms also provide sensors with “sensables,” like properties of a certain type that can be sensed by a certain type of sensor; for example, a submarine platform has a sonar sensable that makes it possible for a sonar sensor to sense it. The probability of detection of a platform can be reduced if counter measures, or “sensordefendables,” like camouflage, are used by the platform to hide itself from a sensor.

The counterpart to sensable for actuators is an “impactable,” which indicates that a certain type of actuator can impact the platform in some way. Analogously, there are counter measures, or “actuatordefendables,” like shields, that reduce the probability of impact.

The default platform implementation allows platforms to be classified into types. A platform type defines the characteristics of a group of platforms of the same type. Each platform is associated with exactly one platform type. The platform manager maintains all the platform types and platforms used in an INFORM Lab simulation. It is also responsible for parsing all the different platform types and platforms from the vignette configuration. The platform manager provides methods for retrieving platforms by name, as well as all platforms, of a given type.

A new sensor can be created simply by selecting in the configuration editor the desired:

- *Sampling model:* the timing of the sensor measurements;
- *Search volume type:* the shape of the area that the sensor can see;
- *Visibility model:* the ability of the sensor to function in certain environmental conditions and the resulting probability of detection of the target;
- *Measurement model:* performed, like position and velocity.

Once this selection is made, further parameters for each of these components can be provided to further customize the sensor’s characteristics. INFORM Lab already provides implementation for the most common of these classes.

7.2.3 Default Communicator

The default communication model provides an agent with a single communicator, which allows an agent to talk to other agents. The communicator has one or more CommDevices, which model the physical communication capabilities of an agent. They are associated with "CommLinkTypes" that determine the communication protocols, such as link-11. A CommLinkType in turn has several modes represented by CommMode, representing the different communication modes a link supports, and their characteristics, which are data rate, range, and line-of-sight restriction.

The messages that the communicator forwards and stores can consist of orders, requests, decisions or information. An agent's communicator is its gateway to the INFORM Lab communication model, which offers the following features:

- Distinction between the network and individual communication devices;
- Point-to-point communication;
- Different message types;
- Communication range restrictions;
- Latency;
- Asynchronous message delivery;
- Different types of data links and (in)compatibility between communication devices of different types;
- Full simulation of routing: the path of messages and their acknowledgements through the communication models is explicitly modeled; where each hop constitutes a communication action and the agents apply routing strategies to find the most (cost- or time-) efficient route through the network;
- Bandwidth limitation: each communication channel can transmit up to a certain amount of information within a fixed time interval, depending on the channel characteristics. The amount of bandwidth needed to transmit a message differs between the types of messages which will allow the development and testing of policies to reduce bandwidth consumption and optimize the use of communication resources;
- Failures and timeouts: messages are assigned a time interval, within which they have to arrive at their destination. Because of bandwidth limitations and communication interruptions, messages may time out. As well, unreliable channels will be modeled, in which messages arrive intact only with a certain probability less than 1. This will allow investigations into the impact of imperfect communication channels.

These features are quite sufficient for most high-level fusion research. For more specific research needs on communication-related issues, more details and additional features are necessary, but the model is generic enough to support such extensions.

7.2.4 Goals

The goal manager has the task of administering the goals of an OODA agent. An agent can have multiple goals. A goal is a mission statement. Alternatively, it could also be described as a special, very high-level plan that has only a single task. Thus, a goal does not have any precedence constraints. It also does not prescribe any resources that may execute the goal. A goal is primarily characterized by:

- Proposition (what to do);
- Area (where to it);
- Time window (when to do it);
- Object identfiications (ObjectIDs) (optional – to whom to do it);
- Priority (how important is it);
- Objective (how to tell when it is done).

7.2.5 Situation Evidence

Situation evidence (SE) is an agent's description of the current situation. A flexible way of representing a situation is by simply providing a list of propositions that describe the properties of the situation at a given time. Each proposition is also qualified by the degree of accuracy with which it is known so that propositions can be compared and fused.

In INFORM Lab, the situation evidence data structure captures what an agent has learned by sensing its environment, communicating with other agents, and what the agent has inferred by reasoning. In INFORM Lab, the situation evidence is represented as a collection of pieces of evidence. Each evidence item has four data elements. Note the symmetry with Goals:

1. Proposition (what?): e.g., *isSmuggling;*
2. Set of proposition quantifiers (how well? where? when?): such as probability, area, and time interval;
3. Set of situation objects that the proposition refers to (who?): a set of tracks, (track1, track2, etc.);
4. Time stamp (when asserted?).

For example, at time t_1 the proposition *isRendezvousing* is asserted. There is a single proposition qualifier that records the certainty with which this proposition is true. For example, this could be represented as a Dempster-Shafer value. Suppose in our example, there are two ships that are rendezvousing, then there are two objects of interest listed in the evidence structure, namely the tracks for the two ships, which are represented by their respective track-IDs.

Although the emphasis is on higher-level fusion, no distinction in principle is made between Level-0 contact information, Level-1 track information, and Level-2 logical propositions. They are all treated as propositions for which there exists direct evidence at certain points in time. All evidence is treated as partially uncertain. For our current version of INFORM Lab, there are only two types of uncertainty: Gaussian covariances, and Dempster-Shafer (DS) masses. The former is usually used for properties such as position, velocity, or shape, and the latter is often used for logical propositions, such as *isSmugglingOperation*.

In the case of Level-1 propositions, such as position, the proposition qualifiers also contain the actual measurement values and information, such as a sensor ID that allows avoidance of data loops. Thus, a piece of evidence can have a Level-1 proposition, for example velocity, with at least two qualifiers. One qualifier reports the observed value of the velocity and the other its covariance. In this example, there may be only one object of interest, namely the track of the ship whose velocity is reported.

More elaborate qualifiers may also be accommodated. For example, a qualifier could represent the Probability Distribution Function (PDF) of a proposition as a function of space at the given time. This also allows storing information about measurements that failed to detect a target in a certain area.

The situation evidence provides key information for decision-making. For example, if the goal was to find a fishing boat in distress, then all that is required is to query the situation evidence for the piece of evidence that has asserted the proposition *isFishingBoatlnDistress* with the highest probability. If the value is high enough, then the search could be declared complete.

7.2.6 Agent Affliations and Relationships

INFORM Lab supports arbitrary, user-defined affiliations of agents. For example, important affiliations are: Friendly, Hostile, and Neutral. Although some of the affiliations can be thought of as mutually exclusive, INFORM Lab has no restrictions as to assigning multiple affiliations to agents.

INFORM Lab also allows arbitrary user-defined agent relationships, hierarchical and non- hierarchical. For example, in the default implementation of the OODA agent, the agents have *superiorOf* relationships that establish a command and control hierarchy.

The agent relationships, particularly agent hierarchies, can be used to flexibly model agents that represent groups of agents. For example, an admiral is an agent that can be thought of as representing a fleet of ships, despite the fact that he is physically located on a flagship which may have a separate captain agent. The admiral agent can command the fleet and has the fused situation evidence of all the ships in the fleet. Thus, the admiral makes decisions on another level of abstraction than, for example, a Unmanned Air Vehicle (UAV). INFORM Lab allows commander agents to be abstract in the sense that they may have no sensors or actuators of their own, and may not command directly the platform they are traveling on. The commander just issues goals and receive situation evidence.

For example, a commander agent commands a subordinate agent by issuing a goal. To do this, the commander may need to break his own goal into subgoals that are subsequently sent to his inferiors. In principle, goals can be broken down in at least three different manners:

- Decompose the original goal proposition into subpropositions;
- Decompose the original goal area into smaller subareas;
- Decompose the original goal time interval into subintervals.

For example, a commander has the goal to search for smugglers in a large area A. The commander may decide to break his goal into, say, two subgoals: Search for smugglers in subarea A1 and Search for smugglers in subarea A2.

Each of these subgoals is sent to a different subordinate wing commander to execute. In our OODA implementation, a wing has certain capabilities and modes that can be scheduled on a high-level as if it was an individual agent. Thus, an agent can also represent a subgroup of agents in a hierarchy. For example, a squadron leader may be represented as an agent that inherits the capabilities of an entire squadron. Such an agent may decompose a given goal into further subgoals, which it can then send to subordinate agents. Each separate subgoal can then be handled by the specialized inferior agent that is best suited to deal with it. The situation is depicted in Figure 7.5.

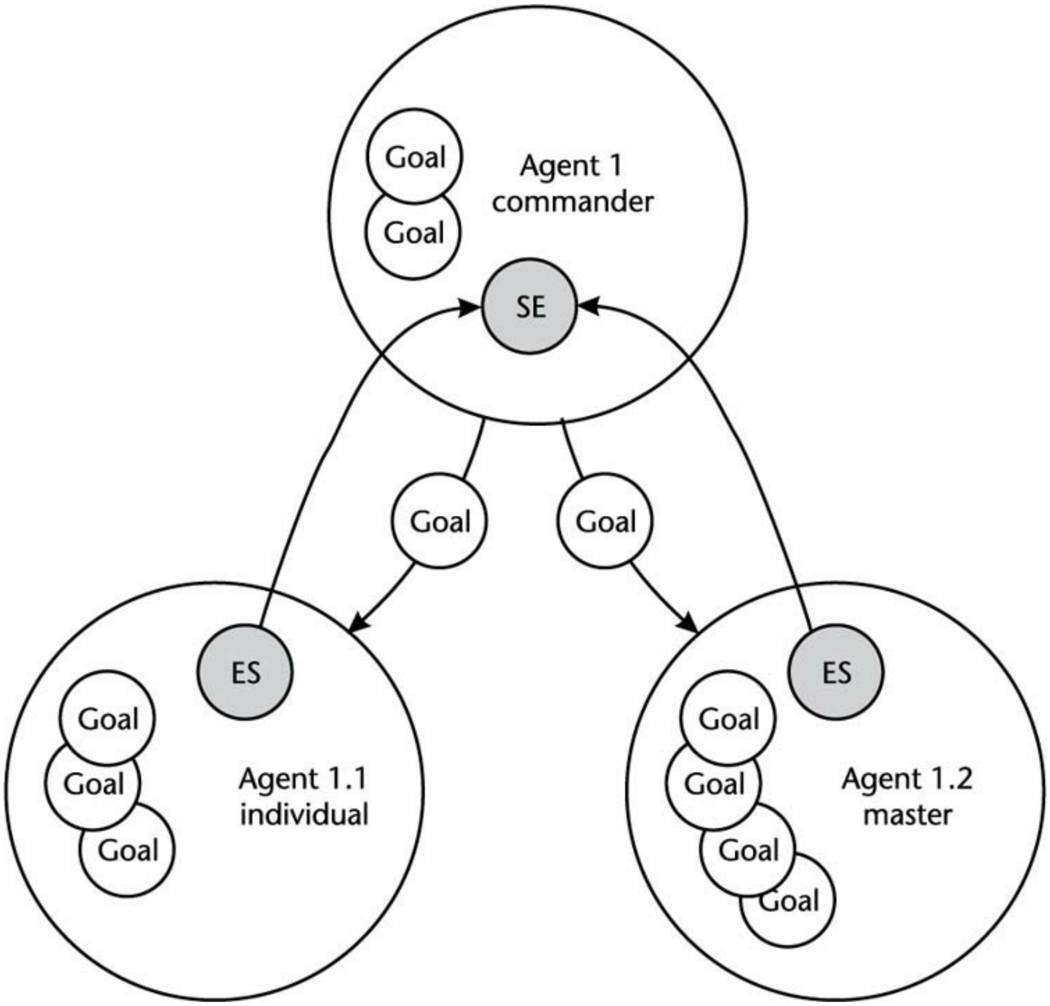


Figure 7.5 Goals and situation evidence.

As is shown in Figure 7.5, the parent agent passes goals to its child agents, and the child agents pass situation evidence up to the parent. When the goals travel down they become more and more detailed and concrete, when the evidence travels up, being fused along the way, it becomes more and more abstract and high-level. This situation could be viewed as follows: the goal is a question that an agent is asked, and the situation evidence is the answer that the agent gives. In the current surveillance context, the questions are always of the type: find pertinent information about something. The answer is then the information found (see Figure 7.1).

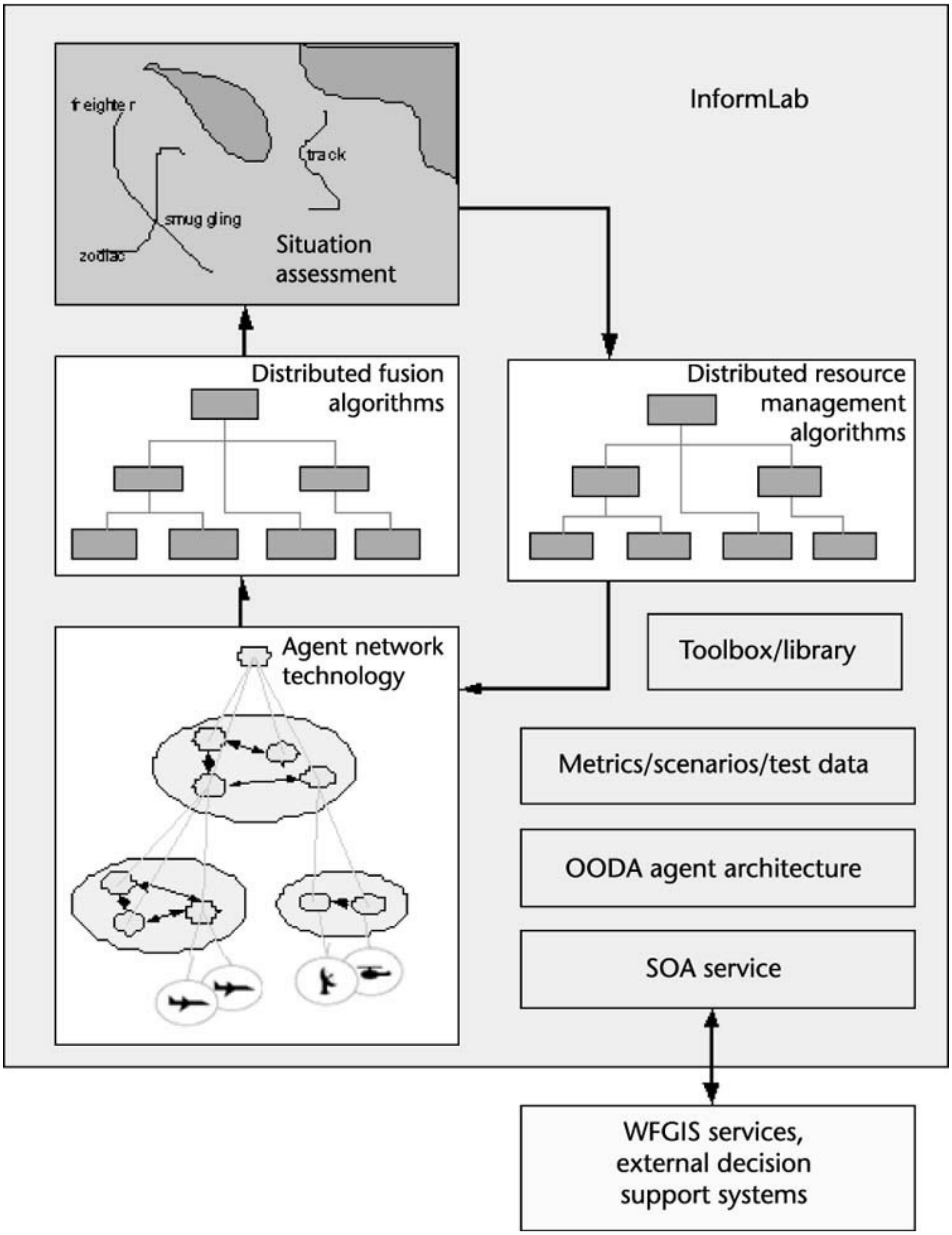


Figure 7.1 Components of INFORM Lab.

7.2.7 Services

INFORM Lab provides algorithm developers with access to relevant services. The services provide vital global information about other agents and their relationships, and about the simulation geography and the environment, about the simulation time and the communication networks. Since these global functions are designed as services, they also provide INFORM Lab with some of the advantages of a service-oriented architecture, such as a high degree of decoupling between components.

7.2.8 Extension Mechanisms

INFORM Lab is explicitly designed to be extendable, which is essential for it to fulfill its function as a testbed. There are three main extension mechanisms in INFORM Lab

Parameterization

Users can customize platforms, sensors, communication devices and many other elements of INFORM Lab by changing parameters for existing INFORM Lab components. Many of the extension needs for INFORM Lab can be met in this convenient manner, which does not require programming. In fact, users can define complete scenarios and vignettes by providing an entirely new vignette configuration file, or they can just change a parameter in a subcomponent, all without programming.

Pluggable Objects

Algorithm developers need to be able to add their algorithms to INFORM Lab with as little effort as possible. INFORM Lab accommodates this need. Existing modules can be replaced by alternative implementations with little integration effort. The INFORM Lab plug-and-play mechanism has many advantages. It maximizes the mutability of a simulation through a strictly compartmentalized architecture, allowing different algorithms or physical simulation entities to be swapped for the purposes of experimentation. It allows for a flexible configuration file format, which can be dynamically constructed by the visual editor tool and built up from pre-configured components. It minimizes the impact of plug-and-play conformance on individual development efforts of experimenters wishing to plug in custom components.

XML Interfaces

INFORM Lab provides XML interfaces between the editor and the test bed, and between the test bed and the viewer. This standard format allows the default editors, and viewers to be more easily extended or replaced by alternative implementations.

INFORM Lab Implementation

The computing environment is described as the following:

INFORM Lab is written entirely in Java. However there are plug-ins that are written in C++ and connected to INFORM Lab via JNI. But these plug-ins have alternative Java implementations and thus INFORM lab can run without linking to them.

Thus, INFORM Lab is platform independent, and only requires a recent Java virtual machine. The code was developed for Java 2 Platform, Standard Edition (J2SE) Version 1.6, using Eclipse.

The graphical user interface is summarily described by the following editors:

Configuration editor allows the convenient setting of pluggable component parameters in a hierarchical manner. Pull-down menus guide the users through the available choices for the pluggable components. At the highest level, the user can easily configure events, agents, platforms, agent relationships and affiliations, the environment, the GIS elements, and the communication links.

Relationship editor provides a graphical display and editing capability of the social layer. Thus, it allows graphically setting the command and control hierarchies between agents, set agent affiliations, and set arbitrary relationships between agents; however, it should be noted that this is currently somewhat limited.

Trajectory editor allows graphical editing of initial trajectories of agents. The edited trajectories are written back into configuration file. Agents may later change trajectories as the simulation progresses.

Batch run editor enables the automatic execution of a large number of runs, such as the Monte Carlo analysis. As such, it allows the convenient execution of a set of different configuration files, and also the automatic execution of a large number of runs with different random seeds. Results and metrics of interest can be conveniently logged code into MATLAB files for later analysis and display in MATLAB.

Vignette viewer allows to graphically display the temporal execution of a vignette from the XML log file. The Display features include GIS elements such as coastline, cities, rivers, roads, sea lanes, and fishing areas; time slider and special event markers; zoom and pan; agent label, affiliation and military symbols; agent trajectories; sensor coverage footprints; communication ranges; message passing; goal areas and special event areas; weather conditions such as changing foggy areas; fueling stations; evidence maps; and more.

Other displays: INFORM Lab features a complete set of convenient graphical displays and viewers that allow the user to understand more easily the simulation and its results, such as an agent details view that shows an agent's goals, tasking orders, subordinate agents, superiors, and capabilities, and a situation evidence view, which is a summary of the agent's SE, including the time of logging, the evidence type, the tracks and targets, and where in the OODA cycle the SE was logged.

Tests and Validation

INFORM Lab comes with several vignettes that allow the users to make use of ready-made data sets. The currently implemented vignettes are:

- Vancouver Island Scenario:
  - Cooperative Vignette: A sinking fishing boat needs to be located;
  - Non-cooperative Vignette: A costal smuggling operation needs to be detected.
- Atlantis Scenario:
  - Strike Vignette: A military navy convoy is attacked by hostile forces.

The vignettes (and INFORM Lab) are designed particularly with Level-2 and higher-level data fusion in mind, for example, detecting a smuggling pattern. INFORM Lab can conveniently generate specific forms of uncertainty to best challenge the algorithms, including low-level noise, or sensor noise, as well high-level noise, such as cluttering by white traffic and variations of the smuggling behavior. Also, unforeseen changes in resource availability and capability and in environmental conditions can also be simulated to challenge the resource management algorithms.

The OODA agents in these vignettes operate in a closed loop. The agents' goals lead to resources being allocated to certain tasks, which in turn cause situation evidence to be collected and fused. The new situation evidence is then compared with the goals, which then may lead to new replanning of resources.

As an example, a noncooperative search scenario is described in more detail in what follows. The noncooperative search vignette features targets that intentionally avoid being detected. In our example, it involves a freighter carrying illegal immigrants, which are offloaded to zodiacs. This scenario features deceptive maneuvers and potentially intentionally false sensor data, and requires sophisticated distributed information fusion and resource management capability.

The mission focuses on a threat situation that develops off the northwest tip of Vancouver Island. A freighter coming from the eastern Pacific carrying illegal immigrants arrives near Cape Scott on northern Vancouver Island. It leaves a known sea-lane off Cape Scott to begin a manoeuvre to offload the illegal immigrants. It does not use the Automatic Identification System (AIS) to identify itself [5], rather— when it suspects it is being watched—it may use the AIS identification of another freighter scheduled to be in the area, in an attempt to confuse surveillance. It uses two land-based zodiacs to offload the illegal immigrants by making multiple trips to and from the freighter to ferry persons to the coast. The intended drop point is either Guise Bay or Experiment Bight in the Cape Scott Provincial Park, depending on conditions. The scenario is mapped in Figure 7.6.

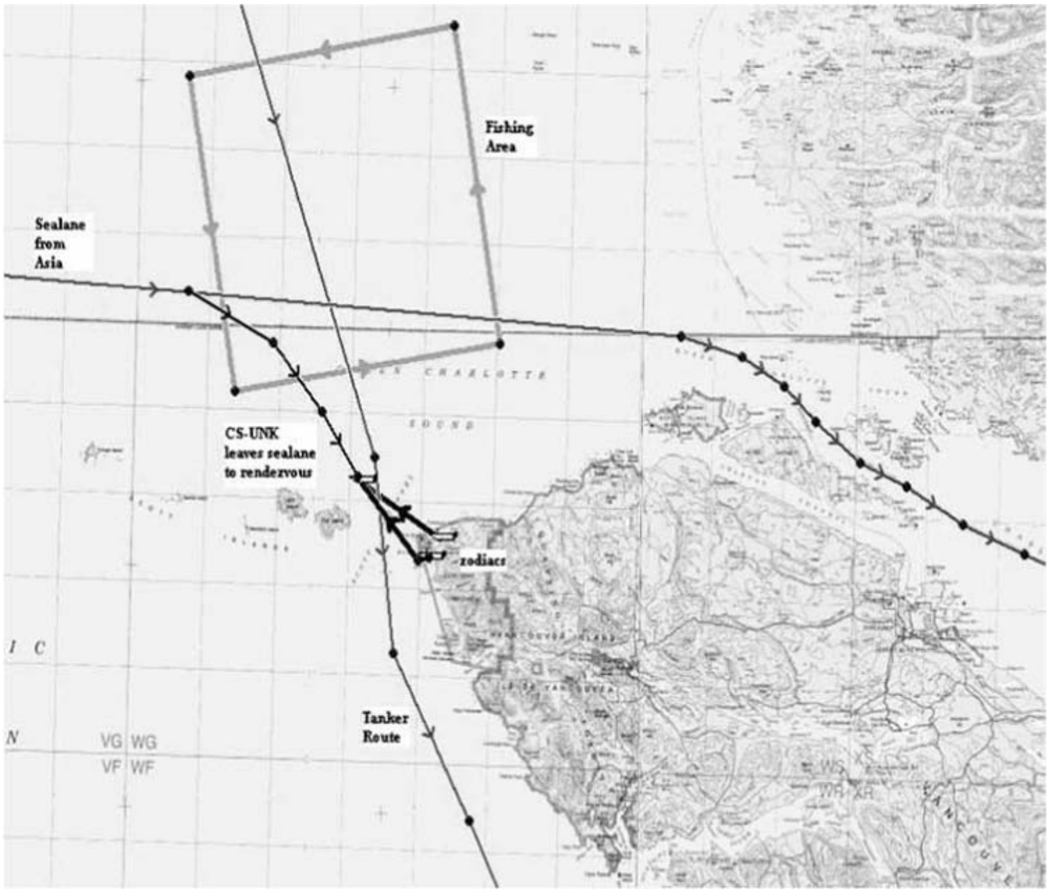


Figure 7.6 Map of noncooperative search vignette.

The complexity of the noncooperative search is captured by a network of evidence with multiple possible confirmatory patterns. For example, a DataInforma-tionFusion agent may fuse information from various sensors on various platforms to identify a smuggling operation by:

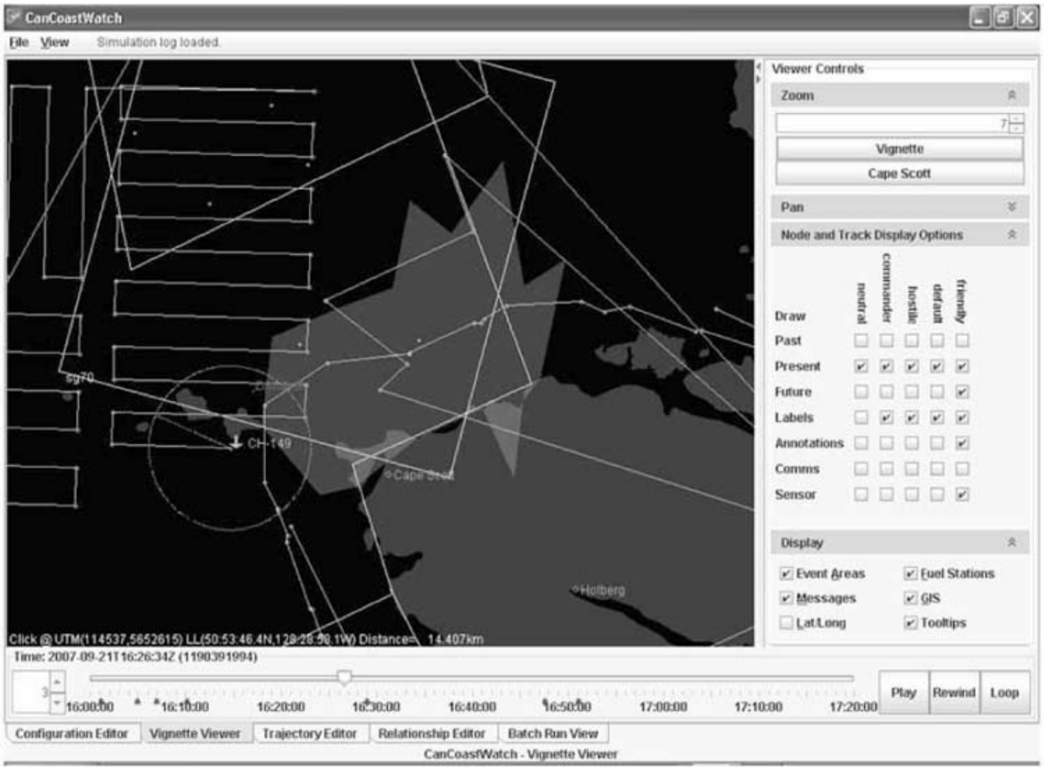
> Identifying an isFerrying activity that in turn requires confirmation of <isLargeShip(s1)^isShipNearShore(s1)isMovingSlowly(s1)^isSmallShip(s2) isMovingBetweenBeachAndLargeShip(s2,b,s1)>.
>
> Identifying an isRendezvousing activity that in turn requires confirmation of <isLargeShip^isSmallShip^isTandemMotionBetweenShips> where the tandem motion is defined by <isShipsHaveSameHeading(s1,s2) ^isShipMovingSlowly(s1)^isShipMovignSlowly(s2)^areNear(s1,s2)>

The reasoning here will require working through threads of evidence where, depending on the situation, what can be sensed when a resource arrives; as there will be different pathways through the evidence to make the conclusion *isSmuggling*.

The resource management is also complex. The command center in Comox receives an intelligence report that a smuggling operation is to commence somewhere in northern

Vancouver Island. This is modeled in INFORM Lab as a commander agent receiving a goal to search for a smuggling operation. Given the large size of the area, and the resources available, the commander node decides to split the search area into two large, independent subareas. This is modeled as two subgoals, each being sent to a separate squadron-leader agent. Each squadron-leader agent, in turn, generates a detailed search plan for the resources under its control and sends tasking orders to their subservient nodes. In the process, one of them has to make a decision if an Aurora aircraft that is already on a routine background surveillance mission around Vancouver Island should be ordered to interrupt its mission and participate in the hunt for the smugglers. As usual, the background mission is modeled as just another goal that the responsible agent has to satisfy. Later, the other squadron-leader agent is faced with the dilemma of satisfying a high-priority request to find a sinking fishing boat while the search for the smugglers is going on. Again, two goals need to be simultaneously satisfied. Depending on the situation and on the resource availability, different decisions will be made by the agents.

The deciding function of the OODA agents periodically evaluates if their goals are met or not. If the situation evidence of an agent indicates that one of its goals is met, then the agent removes the goal. Alternatively, if the situation evidence indicates that additional actions need to be taken to facilitate the achievement of a goal, then new subgoals may be created. The removal or creation of goals would of course lead to a distributed replanning, leading to modified resource allocations and schedules. This is illustrated in Figure 7.7.



Simulation log loaded.

Click @ UTM(114537,5652615) LL(50:53:46.4N,128:28:58.1W) Distance= 14.407km

Time: 2007-09-21T16:26:34Z (1190391994)

Node and Track Display Options

Draw	neutral	commander	hostile	default	friendly
Past	☐	☐	☐	☐	☐
Present	☑	☑	☑	☑	☑
Future	☐	☐	☐	☐	☑
Labels	☐	☑	☑	☑	☑
Annotations	☐	☐	☐	☐	☑
Comms	☐	☐	☐	☐	☐
Sensor	☐	☐	☐	☐	☑

Display

☑ Event Areas
☑ Messages
☐ Lat/Long
☑ Fuel Stations
☑ GIS
☑ Tooltips

Figure 7.7 The CH-149 helicopter is detecting the missing key evidence for smuggling, the rendezvousing pattern of the suspect agents.

Thus, the OODA agents operate in a distributed feedback loop. The goals lead to resources being allocated to certain tasks which, in turn, cause situation evidence to be collected and fused. The new situation evidence is then compared with the goals, which may lead to new replanning of resources. Thus, INFORM Lab provides the means for the user to test their algorithms in a closed-loop setting.

The testbed allows the evaluation and comparison of different information fusion and resource management algorithms with respect to several criteria, for example, the elapsed time until the sinking fishing boat or the smuggling operation is discovered. Other criteria may include the cost for the surveillance operation, or the weighted sum of the detection time and the cost of the operation.

Conclusion

An advanced simulation testbed (INFORM Lab) for the purpose of evaluating the effectiveness of information fusion and resource management technologies in a coastal wide area surveillance situation has been described in details. INFORM Lab testbed allows experimenting with high-level distributed information fusion, dynamic resource management, and configuration management, given multiple constraints on the resources and their communications networks. The testbed’s performance has been illustrated on a noncooperative search scenario involving many target ships and various methods of deceit. Chapter 16 will build on these ideas of the development of high-level information fusion measures of effectiveness.

rences

[1] Valin, P., E. Bossé, A. Guitouni, H. Wehn, and J. Happe, "Testbed for Distributed High– Level Information Fusion and Dynamic Resource Management," *Proceedings of International Conference on Information Fusion,* 2010.

[2] Fadok, D. S., J. Boyd, and J. Warden, "Air Power's Quest for Strategic Paralysis," Maxwell Air Force Base AL: Air University Press, (AD–A291621), 1995. – or Col. John R. Boyd, Patterns of Confict, December 1986.

[3] Glinton, R., J. Giampapa, and K. Sycara, "A Markov Random Field Model of Context for High Level Information Fusion," *Proceedings of International Conference on Information Fusion*, 2006.

[4] Carthel, C., S. Coraluppi, and P. Grignan, "Multisensor Tracking and Fusion for Maritime Surveillance," *Proceedings. of International Conference on Information Fusion*, 2007.

[5] Blasch, E. P., P. Valin, E. Bossé, and J. Roy, "Ontology Alignment in Geographical Hard-Soft Information Fusion Systems," *International Conference on Information Fusion*, 2010.

CHAPTER 8

The Legal Agreement Protocol

Dale A. Lambert and Ashley G. Lambert (Australia)

The dominant demarcation of labor within the fusion community promotes machines for the conduct of lower-level fusion and people for the conduct of higher-level fusion. Meanwhile, the maturing Information Age is increasing both the volume of relevant information and the volume of information to sort through to find the relevant information. So as the Information Age matures, it is inevitable that machines will need to shoulder a greater share of the information processing and analysis burden. Chapter 3 indicated how an individual cognitive machine can deliver observable, object, situation, and scenario assessments that correspond to the sensation, perception, comprehension, and projection awareness obtained by people. An unaddressed question in that chapter, however, is how to organize collections of people and cognitive machines to meet fusion problems.

The range limitations associated with sensors means that sensors will generally need to be spatiotemporally distributed. The first author's Attitude too cognitive model allows a single automated cognition to support sensors and effectors located anywhere across the globe, but too many persistent sensors and effectors associated with a single cognition will cause cognitive overload for that cognition. For large scale fusion problems, a society of distributed automated cognitions will be required; more generally, a society of distributed cognitions of both human and machine persuasion.

Conceptualization

One of the earliest proposals for coordination of multiple fusion systems was Bowman's data fusion tree architecture model, which advocated a hierarchical tree of interacting data fusion and resource management node pairs ascending the JDL levels. It was accompanied by a four-phase process for developing data fusion processes within an information processing system [1]. However, the distributed networked Information Age is trending away from this hierarchical approach. In response, Steinberg and Bowman recommended the Dual Node Network (DNN) technical architecture for data fusion and resource management [2] which allows for a network of interacting fusion and management nodes, in which each data fusion node performs data preparation, data association, and state estimation, and each resource management node performs the dual functions of task preparation, task planning and resource state control. Distributed data fusion and resource management nodes can then be applied for each JDL level and the networks can be adaptive.

This section provides an alternative conceptualization to the Steinberg and Bowman DNN proposal. In [3], the concept of Ubiquitous Command and Control (UC^2) was introduced, pronounced "you see too" with pun intended. An expanded explanation of UC^2 was subsequently prepared and presented in [4]. UC^2 is about a transformation in the nature of command and control practices due to technological advances spawned by the Information Age. In Chapter 2, a shift in fusion community emphasis to higher-level information fusion to address aspects of the Information Age was argued for. Consequently, the UC^2 outlook serves as an appropriate basis for engineering fusion systems in the Information Age. The UC^2 framework derives from a number of related themes.

8.1.1 Decentralization

In [4], a sequence of Information Age transformations was noted. The transformations begin with an adaptation of location. Advances in transportation and telecommunications have altered the extent to which presence is influenced by distance; advances in transportation technology have allowed physical presence to occur more rapidly and more frequently, while advances in telecommunications technology have allowed virtual presence to occur more rapidly and more frequently. The combined increased presence facilitates a shift in the sphere of influence from localization to globalization.

The adaptation of location in turn engenders an adaptation of function. The role performed by individuals and organizations changes. As was noted in [4]:

> Increased [physical] presence and virtual presence has increased both the nature and number of players that can influence a function. The effect is to increase the scope for both competition and collaboration.
>
> - Competition increases because an external presence can be more easily imported to perform a function.
> - Collaboration increases because one can more easily export functional expertise as a component within broader functions.
>
> The outcome of these competitive and collaborative forces has produced an increased focus on competitive strength. Individuals and organizations have been compelled to understand the functions that they can competitively perform and then apply them collaboratively in strategic alliances with individuals and organizations possessing expertise in complementary functions.

Functionality is increasingly being understood as a building block commodity within a global economy.

The adaptation of function induces an adaptation of structure. Industrial Age organizations relied upon centralized, hierarchical integration of differentiated function based on specialization in production and a division of labor. The effect of an increase in competition and strategic alliance erodes this classical hierarchical structure to produce Information Age organizations that include networked structures. The following consequences were observed in [4]:

1. Command and subjective rational decision-making will be tempered by negotiation. Command and control will be supplemented by collaboration.
2. Single minded emphases, such as the executive's focus on profit or the militarist's focus on force, will increasingly need to be understood and applied against a broader diversity of motivating goals.
3. The presumption of control will increasingly be understood as a question of managing change in a complex environment.

The classical hierarchical control structures are to a degree subsumed by self-monitoring and self-regulating processes operating across the collective.

The adaptation of structure in turn generates an adaptation in change. The pace of change intensifies. The identity of an organization becomes the perdurantist's process identity, in which identity is understood in terms of change. In [4], the following insight was offered:

> One outcome of a process conception of organizational identity is that organizational change becomes less a centralized decision and more of an environmental effect of adaptations in location, function and structure. A second outcome is that competitive strength no longer lies solely in knowledge and strategic alliance, but also in their adaptation. Innovation and the ability to form dynamic relationships become the basis for competitive strength. In an information economy in which information is rapidly traded, innovation becomes

the new means of production.

The foregoing account of Information Age organization recommends a decentralized approach to the organization of Information Age fusion systems. Decentralized location means that fusion should be conducted across a variety of distributed platforms rather than at a centralized location. Decentralized function means that the distributed components of a fusion system should be collaboratively integrated through alliances based on competitive advantage. Decentralized structure means that the distributed components should be dynamically combined to facilitate a competitive advantage through self-regulating networks. Decentralized change means that these networks will be highly dynamic, with components contracted when there is advantage in doing so. Flexibility and redundancy are the potential benefits of this decentralization. Flexibility can arise through the ability to dynamically share the load throughout the collective capability. Redundancy ensues because the contracting of an alternative component can occur in response to component failure.

Decentralization exposes a difference in design thinking between the DNN and UC^2 conceptualizations. There is no a priori design architecture for a particular fusion system under the UC^2 approach. UC^2 trades away an a priori agreed organizational structure for an a priori agreed social protocol through which different organizational structures can be dynamically negotiated. Fusion capability is then assembled like Lego™ building blocks on the fly as required, through contracted agreements that propagate between available agents through the network at that time, with social policies and protocols providing social governance. Fusion expertise is then dynamically contracted from the available pool of expertise, not prescribed in advance. This accommodates a network in which the constitution of agents is dynamic and in which the fusion product can be arrived at through a combination of competitive and collaborative processes.

8.1.2 Ubiquity

In [3], ubiquitous command and control systems were so named because of their promotion of ubiquity. A ubiquitous fusion system is a system in which a fusion capability resides on every platform and each of these fusion capabilities are similar rather than identical.

A fusion capability on every platform extends the decentralized location theme to a ubiquity theme by requiring the collective fusion capability to be distributed “almost everywhere,” to use the mathematicians’ turn of phrase. Ubiquity delivers the ultimate form of redundancy, by allowing redundancy to still surface after numerous levels of failure. Ubiquity provides for a graceful degradation of performance under failure by allowing the system to reconfigure so that (possibly less accomplished) fusion components can assume the role of other fusion components when failure occurs.

Having similar, rather than identical, components offers a balance between unity and diversity. Complete uniformity throughout the fusion system risks redundancy, as the source of failure in one fusion component could also be the source of failure for *all* fusion components. A degree of diversity in the fusion system guards against this prospect. Ubiquity operates by allowing individual fusion components to harbor both public and private views, with the former being a product of agreement with other individuals, while the latter retains alternatives should they be required. If a fusion component Z receives advice that α is the case from fusion component X and that not(α) is the case from fusion component Y, then an agreement might be formed to proceed under the assumption that α is the case. Y would then participate in that agreement as if α was the case, but would privately retain the reasons for endorsing not(α). This might subsequently prove to be invaluable if it turns out that not(α) is, in fact, correct. Under the UC^2 approach, inconsistencies are managed, not discarded. Agreements facilitate social unity while retaining the robustness of diversity.

In contrast to the DNN model, having a similar fusion capability within each fusion component under UC^2 means that every fusion component is capable of performing Level 0, Level 1, Level 2 and Level 3 fusion. Fusion components are cognitive agents. The functions that a fusion component actually performs at a given time will then vary with the agreements it has in place at that time. A fusion component might only provide an observable assessment role in the context of one agreement that it is engaged in, while concurrently providing a situation and scenario assessment role in the context of another agreement that it is engaged in.

8.1.3 Automation

The advantage of automated software expertise is that it is easily replicated, adapted and distributed. The transferability of automated software expertise is the attribute that enables the ubiquity of our fusion capability. Fusion components can become ubiquitous because we can embed fusion capability in transferable software, as was noted in [3]:

> The fact that we can readily duplicate software then becomes the crucial attribute, because duplicated software encoded human expertise is the mechanism that facilitates the ubiquitous capability.

Lower-level fusion automation is currently widely practiced. In Chapter 3, the first author agitated for a similar automation of higher-level fusion by noting the parallel between the levels of human situation awareness and the levels of machine data fusion, together with the complementary strengths and weaknesses of each. A key aspect to the implemented ATTITUDE TOO cognitive model is that the machine's cognitive routines become social routines when the subject of a propositional attitude instruction is other than oneself. In that, event the cognitive routines specify a role to be played by an agent within a pattern of social behavior.

The DNN proposal canvasses a fusion system with fusion and resource management components. The UC^2 ideology promotes automation through software agent components, which encourages a deep integration of fusion and resource management elements within the each component. Under UC^2, the integration of fusion and resource management can be very tightly coupled. In a previous Airborne Early Warning and Control (AEW&C) application, the integration of fusion and resource management elements often occurred within a single cognitive routine. This is why, from the fusion perspective, the author absorbs JDL resource management (level 4) within each of the JDL Levels 0 through 3, as discussed in Chapter 3. Resource and fusion information can be shared at any of these levels. Indeed, even cognitive routines can be passed between agents under the ATTITUDE TOO cognitive model.

Automation additionally advocates a mixed initiative strategy in which intent, in addition to awareness and capability, are distributed between human and machine. The decentralization of structure introduces an intent fusion problem in addition to the traditional awareness fusion problem, since the strategic alliance process allows an individual fusion component to receive multiple, possibly conflicting, intents from other fusion components. Automation extends that complexity further by allowing machine based fusion components to be among those fusion components issuing intent to form dynamic strategic alliances. People and machines potentially assume a more equal footing under the UC^2 model, with the role played by each on any given occasion determined by the competitive advantage afforded. Self-regulation is then maintained across a human-machine collective. As was noted in [3]:

> In UC^2 systems, the automated and human decision making is fully integrated, with each assessed equally on it merits. This includes the currently controversial option of allowing the machine to at times override the human. The introduction of automated rules of engagement components (essentially legal expert systems) within weapons and weapon systems illustrate the point. The resulting “moral weapons” will have the ability to assess and decline targeting requests when rules of engagement violations are deduced. Decisions to override these moral weapons can be logged for subsequent review.

8.1.4 Integration

In UC^2 systems, the automated and human decision-making is fully integrated. Integration exists to complement the weaknesses in some parts of a UC^2 system with strengths in other parts of a UC^2 system. This includes the division of labor between people and machines, ideally to amplify “the human as hero” and minimize “the human as hazard” [5]. Integration requires integration between human fusion components, between machine fusion components, and between human and machine fusion components. The latter requires improved interfaces between human and machine, particularly in relation to higher-level fusion. Some of these are featured in Chapter 12.

The management of integration rests with an agreement construct, as this is the means by which social coordination is obtained between different cognitions, be they human or machine based. A unifying agreement protocol could thus serve as the basis for dynamic strategic alliances to perform fusion tasks to competitive advantage across a human-machine collective. The inclusion of machines within the collective alters the way in which Pigeau and McCann's command and control attributes of authority, responsibility and competency are practiced [6]. In [4] the first author observed the following.

- An automated agent's competency will depend on the expertise embedded within it, and the agreements it forms should primarily derive from its competencies.
- An automated agent's responsibility will follow from the social agreements it forms, given available competencies.
- An automated agent's authority is not determined by a priori rank, but depends upon the role it assumes in social agreements, given available competencies. …, in the end, authority is a matter of agreement.

A unifying agreement protocol was canvassed in [4]:

> UC^2 systems can achieve social coordination by instituting social agreement protocols that coordinate collectives composed of both people and machines. The social coordination can be instituted through software, as more sophisticated variants of existing workflow systems. In essence, eBay is a social agreement protocol implemented through software. The cost of finding information and expertise in this system is low and the agreement and monitoring mechanisms provide feedback for self-regulation. Examples of social contracting protocols, include:
>
> 1. Contract net protocol (CNP, Smith 1980 [7]);
> 2. Extended contract net protocol (ECNP, Fischer 1996 [8]);
> 3. Provisional agreement protocol (PAP, Perugini et al, 2003 [9]);
> 4. Legal agreement protocol (LAP) [10].
>
> The above protocol ordering reflects an increase in computational complexity, and an increase in rights for the proletariat. The legal agreement protocol being introduced here, offers a facility for full contract law agreements between agents, be they human or machine.

The remainder of this chapter expands upon the legal agreement protocol initially proposed by the first author. The emphasis is on strengthening its legal credentials, formalising the framework, and implementing it as a set of cognitive routines to facilitate social routines for forming fusion systems through dynamic contract coordination.

Formalization

The ATTITUDE TOO cognitive model provides the basis for software identities to be able to intelligibly collaborate with one another. To ensure that functional, enforceable, and ethical transactions occur between these software identities, a governing framework is necessary to guide behavior. The aforementioned suggests that societal functions can be reduced to agreement constructs between parties. Legal analysis purports this too through contractarian analysis [11] which asserts that consensual formation and consensual terms [12] lie at the heart of legal contracts. Given this existing framework in law and the future ability for these software identities to perform as agents in the legal sense of the term "agent," and so contract on behalf of people, there is much utility in adapting contract law as the basis for the collaborative principles governing cognitive software behavior. This motivates the legal agreement protocol defined in this chapter for forming contracts between software agents. The legal agreement protocol is designed to adhere to existing legal principles while ensuring clarity in the contract formation process. In this way the protocol can be equated with a standard form contract. The legal agreement protocol needs to consider all three aspects of the contract process: contract formation; contract performance; and contract remedies.

8.2.1 Contract Formation

For the successful formation of a contract between two software agents under the legal agreement protocol, the legal requirements of invitation, capacity, agreement, and certainty must all be satisfied in this order. This structure notably omits the legal requirement of consideration, for reasons discussed in Section 8.2.1.5.

8.2.1.1 Invitation

Agent negotiations start with what is legally referred to as an invitation to treat, being an invitation for agents to enter into further negotiations. This consists of an agent informing other agents of their intent to have a given task performed within a certain time, without yet imposing any obligations to do so. Everyday examples of an invitation to treat include: an owner's indication that he or she might be interested in selling at a certain price [13]; the display of goods for sale [14], the holding of a public auction [15], and a call for written tenders [16]. In the current context, invitations to treat will primarily relate to the provision of fusion services.

8.2.1.2 Capacity

Following a reply to an invitation to treat, agents assess the capacity of one another to ensure agreement is possible. Capacity is the capability of parties to form contracts. Under current Australian law, minors (persons under 18) [17], bankrupts [18], and persons intoxicated or of unsound mind (for example, having limited capacity) [19].

For fusion agents, capacity can be performance based. An agent's ongoing level of performance can be monitored and updated based on successful contract completion, and in accordance with David Hume's philosophy, this can be used as a trust level measure of their likely reliability for successfully completing agreements. Having publicly viewable trust levels enables agents to assess the trustworthiness of all other agents before forming agreements. This facilitates a market structure protected by disclosure requirements, whereby good performance is rewarded. eBay™ operates through such a mechanism with human agents. The mechanism for publicly viewable trust levels could also be accompanied by a regulatory legal framework. Poor performance could result in an agent losing his or her capacity to contract for a period of time. This both protects other agents from poor performance and temporarily excludes the agent in question from fulfilling his or her trading desires.

The protocol is also designed to not allow any agents the capacity to contract when they believe the terms to be illegal. This mitigates, rather than guarantees, the possibility of contracts being void due to illegality [20].

As the sophistication of software agents develops, there is potential for the law to recognize them (as has been done with corporate bodies) as legal persons. While there is much speculation as to how exactly this would be achieved [21], the protocol should ensure that an agreement constitutes legal obligations being made. This potentially allows the legal agreement protocol to not only service legally enforceable contracts between software agents, but also between people and software agents. The latter includes the capacity for software agents to serve as legal agents on behalf of people, and for the capacity for people to serve as legal agents on behalf of software agents. The legal agreement protocol could also be implemented as an interface framework to manage the formation of legally enforceable contracts between people, without the involvement of software agents.

8.2.1.3 Agreement

In accordance with commercial practice, after the invitation to treat and capacity assessments, two main stages are required to form an agreement: a preliminary proposal and formal offer and acceptance. No obligations are imposed until after acceptance.

Preliminary Proposal

Agents respond to an invitation to treat with a preliminary proposal. Following this, the agent who sent the initial invitation informs selected agents that their proposals are preferred, possibly with some refinement required, and informs other agents that they have been unsuccessful. This shortlisting process has the effect of narrowing down parties involved in the bargaining process. Under Australian law, preliminary agreements can be binding when objectively interpreted if the parties intended it to be binding [22]. Under the legal agreement protocol such agreements are accepted to be non-binding.

Offer and Acceptance

When informed that the proposal is preferred, the agent can then send a formal offer. This explicitly stipulates all conditions of the agreement, and the time within which completion must occur. Upon receiving a formal offer, it follows that an agent may reply with formal acceptance. As will be seen below, all conditions of the agreement under acceptance must be identical to the offer for the agreement to be effective. Any variation, however slight, will cause the transaction to be a counteroffer, which then requires acceptance from the other agent. Acceptance, when communicated to the offeror [23], marks the successful formation of a contract. As these agent systems are designated for the purpose of receiving communications, acceptance is held to have been communicated as soon as it enters the system [24]. From the moment acceptance is received, the agents' conditions become enforceable obligations, and performance against those obligations will be monitored.

8.2.1.4 Certainty

For an agreement to be legally effective, the contractual terms must be certain. In practice, this means the contractual terms must be complete, certain and clear, and free of illusory promises (where one party is given unfettered discretion as to performance of a promise).

The Mephisto semantic framework [25] delivers the capability for certainty and clarity through semantics grounded in formal logics. The Mephisto definition of "offers" and "agrees" defines offer and agreement for the Legal Agreement Protocol. The definition of offers is replicated below:

Definition (offers): $\mathrm{offers}(@(Y, t, s_1), @(X, t, s_2), \alpha) =_{df} \exists\, t_2\ \exists\, s_3\ (\mathrm{before}(t, t_2)\ \&\ \mathrm{intends}(@(Y, t, s_1), \mathrm{intends}(@(X, t_2, s_3), \alpha))\ \&\ \mathrm{informs}(@(Y, t, s_1), @(X, t, s_2), \mathrm{intends}(@(Y, t, s_1), \mathrm{intends}(@(X, t_2, s_3), \alpha))))$.

The definitions of offers and agrees are primarily based upon the concepts of "informs" and "intends" which are also unambiguously defined in Mephisto through the use of a formal logic. Completeness, and the safety net against illusory promises, is handled by the way the terms of an offer are processed. The contract content α can be expressed in conjunctive normal form as cnf(α) = (α_1 & … & α_n) where each α_i is a disjunction with the general form ($\alpha_{i,1} \vee \ldots \vee \alpha_{i,k} \vee \neg\alpha_{i,k+1} \vee \ldots \vee \neg\alpha_{i,m}$). Each disjunct α_i, can therefore be written as ($P_i \vee \neg q_i$) where $P_i = (\alpha_{i,1} \vee \ldots \vee \alpha_{i,k})$ and $q_i = \neg(\alpha_{i,k+1}$ & … & $\alpha_{i,m})$, or equivalently as (q_i => p_i). Thus α can be equivalently expressed as ($q_1 \Rightarrow p_1$) & … & ($q_n \Rightarrow p_n$). The completeness of the contractual content α is therefore determined by whether the conditions qi cover all eventualities; that is, $q_1 \vee \ldots \vee q_n$, is tautological. If $q_1 \vee \ldots \vee q_n$ fails to hold, then the contract is void under the legal agreement protocol, and parties are returned to their original positions.

8.2.1.5 Consideration

Under current Australian common law derived from English common law, a valid contract requires that each party exchanges or promises to exchange something of legal value. There are several possible rationales for this doctrine; however, none of the arguments are persuasive in the context of contracting software agents. Some of the arguments are rebutted below:

Consideration distinguishes between fair and unfair transactions. Since the exchanged goods or services require only some legal value, grossly uneven items and unfair transactions are still possible.

Consideration ensures that only economically efficient transactions are enforced. Gratuitous promises can also have significant economic benefits.

Consideration fulfils the function of a formal requirement such as writing, a seal or notarization, without subjecting the parties to the inconvenience of those requirements. The legal agreement protocol already ensures that agents interact in a highly formal manner.

There is also much academic support for the removal of consideration in a general legal context, with some commentators describing it as a mere encumbrance [26] to contract law.

8.2.2 Contract Performance

Once a contractual agreement has been established, legal obligations are conferred. Contract performance is about whether those legal obligations are met. Contract performance management involves monitoring the contract activities in relation to quality, quantity, and timeliness. Performance milestones identified in the contract provide the basis for assessing contract performance, and these often require communication between the contracted parties. The milestones can specify criteria that must be true, and criteria that must not be true, at a given time. Under the legal agreement protocol, the basis for assessing contract performance is specified during contract formation.

8.2.3 Contract Remedies

The legal agreement protocol provides remedies for an unsatisfied agreement through two mechanisms concurrently: preagreed monetary compensation, and trust ratings ("trust" in the general, as opposed to legal sense of the word). The trust rating allows contractors to assess the trustworthiness of prospective contract partners. The monetary compensation allows recompense for failure to complete terms of a contract. While this structure handles compensation within the protocol, in the case that a party pursued litigation, the court would determine the appropriate remedy in its application of the law.

8.2.3.1 Pre-Agreed Monetary Compensation

Throughout the formation stages of the agreement, agents specify (and ultimately agree upon) a monetary amount that is payable for failure to complete any individual term. It must be clear that there is sufficient security to provide the total amount payable, which can be achieved through a variety of means, such as a trust fund, insurance policy, etc.). This form of compensation provides an internal cap on liability from the outset, and allows the parties to the contract to understand the monetary risk of their endeavor in precise terms; in some instances, the monetary risk could be agreed at $0. The inclusion of monetary compensation also removes some of the hurdles to recognizing software agents as separate legal entities, in that they can, to some degree, be held accountable for their actions.

8.2.3.2 Trust Ratings

As mentioned earlier, a publicly viewable trust rating for each agent creates a transparent—and therefore fairer, and more efficient—market [27]. This can be calculated along the lines of mathematical expectation, in which the ratio of success to successes and failures is combined with the amount of reserved compensation agreed upon in each of those cases.

Computation

Computational elements are required for contract formation, contract performance and contract remedies.

8.3.1 Contract Formation

The contract negotiation process moves through a number of states as it progresses toward agreement. In the implemented Attitude too, each state can be monitored through a cognitive routine, and so the social routine of contract negotiation is implemented through various cognitive routines that rely on certain communications from the other party in the negotiation. The communications assume the form of formal speech acts that are used to transition the negotiation through the states. Figure 8.1 coarsely illustrates the contract formation process by identifying the states, the cognitive routine required for each state, and the speech act type required for state transition. Each agent possesses all of the cognitive routines mentioned. When a contractual negotiation occurs, the cognitive routines on the left of Figure 8.1 are exercised by the contracting party, and the cognitive routines on the right of Figure 8.1 are exercised by the contracted party. The following sections include some indicative code.

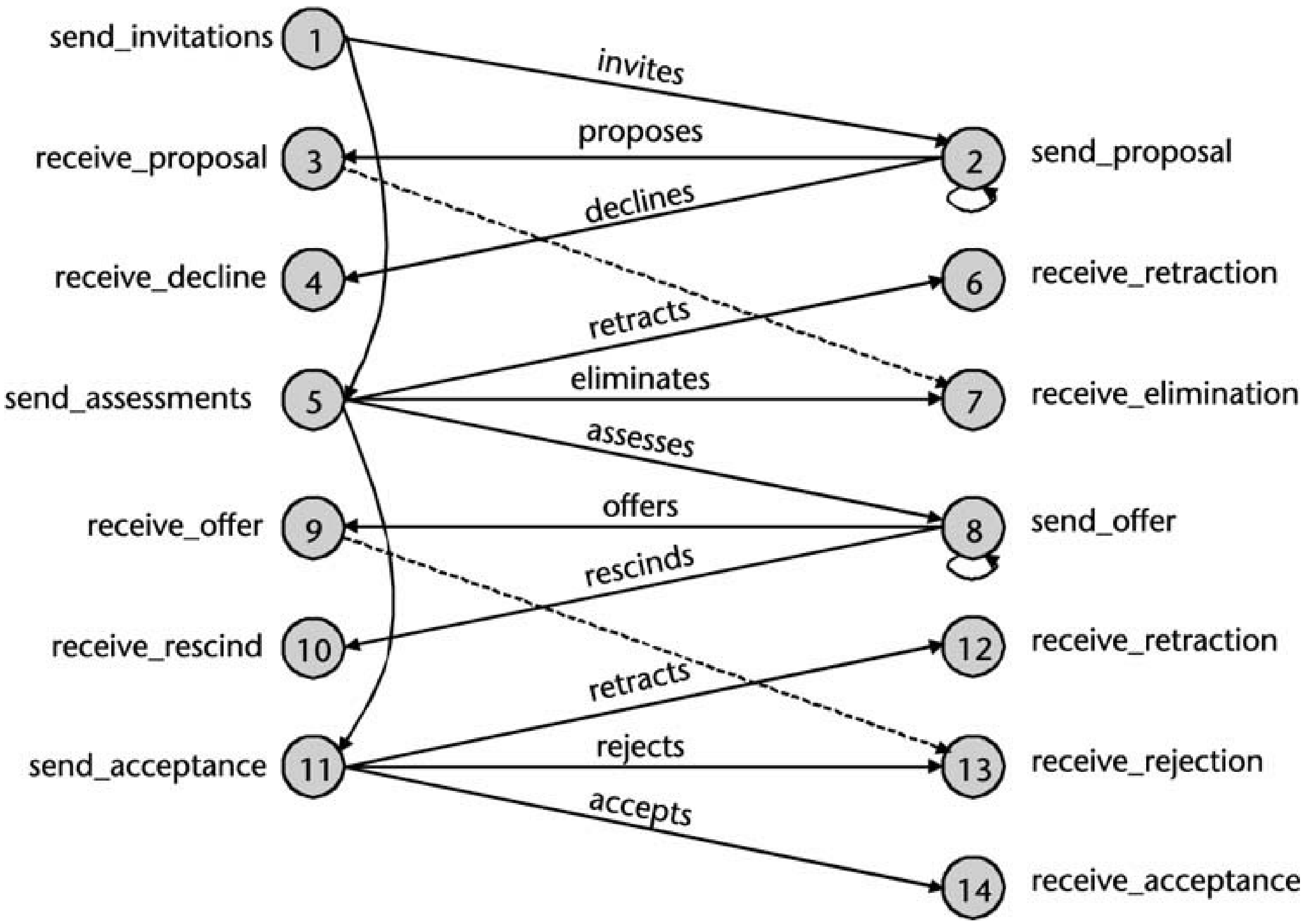


Figure 8.1 Contract formation computation.

8.3.1.1 Invitations

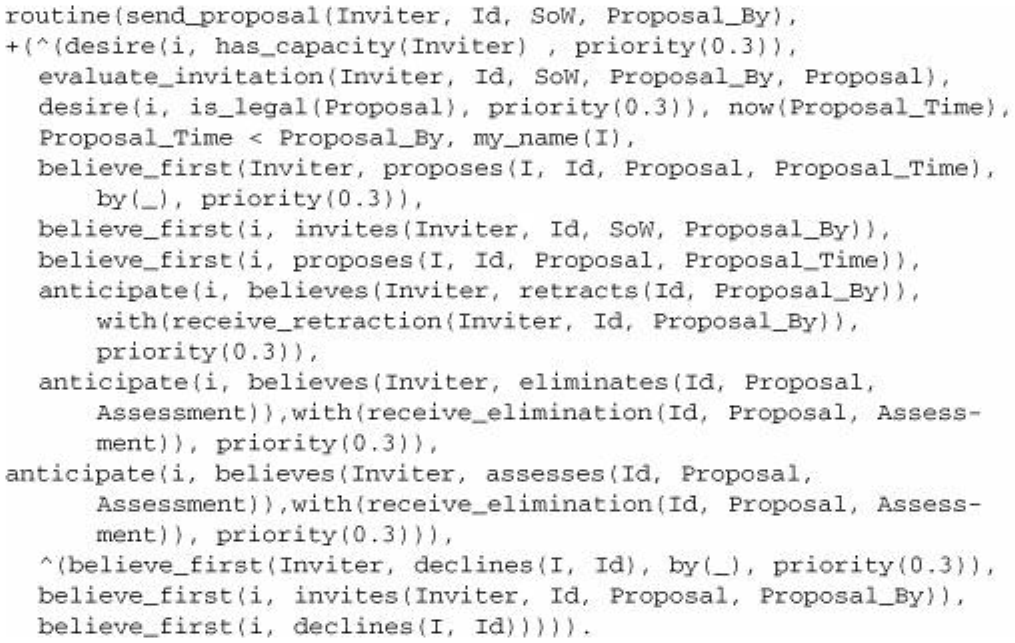
The contract formation process begins with the contracting agent exercising a send_ invitations cognitive routine.

```
routine(send_invitations(SoW),
^(outcome(SoW, Outcome), -(ask_if_believe(i, Outcome)), desire(i,
     is_legal(SoW), priority(0.3)),now(Now), Proposal_By is Now +
     120.0, desire(i, identify_market(SoW, Proposal_By, Market),
     priority(0.3)), \+(Market = []),
  generate_proposal_id(Id), believe_first(i, invited_proposals(Id,
     SoW, Proposal_By, Market)),
  anticipate(i, believes(Proposer, proposes(Proposer, Id, Proposal,
     Proposal_Time)), with(receive_proposal(Proposer, Id, Propos-
     al, Proposal_Time)), priority(0.3)),
  anticipate(i, believes(Proposer, declines(Proposer, Id)),
     with(receive_decline(Proposer, Id)), priority(0.3)),
  my_name(I),
  foreach(member(X, Market),believe_first(X, invites(I, Id, SoW,
     Proposal_By), by(_), priority(0.3))),
  intends(i, send_assessments(Id, Proposal_By), priority(0.3)))).
```

send_invitations is applied to an intended statement of work SoW. The SoW specifies the outcome to be achieved, the deadline by which it is to be achieved, other constraints deemed relevant, and the means by which proposals will be assessed against the statement of work. The first step is to check that the statement of work outcome has not already been achieved. The legality of the statement of work is then assessed. After this the market is assessed to identify prospective contractual partners that could achieve the statement of work, and a unique proposal identifier is then generated. Anticipations of proposals and declines are then established for the invitation before the invitation is sent to each agent in the identified market. The send_assessments routine is then invoked.

8.3.1.2 Proposals

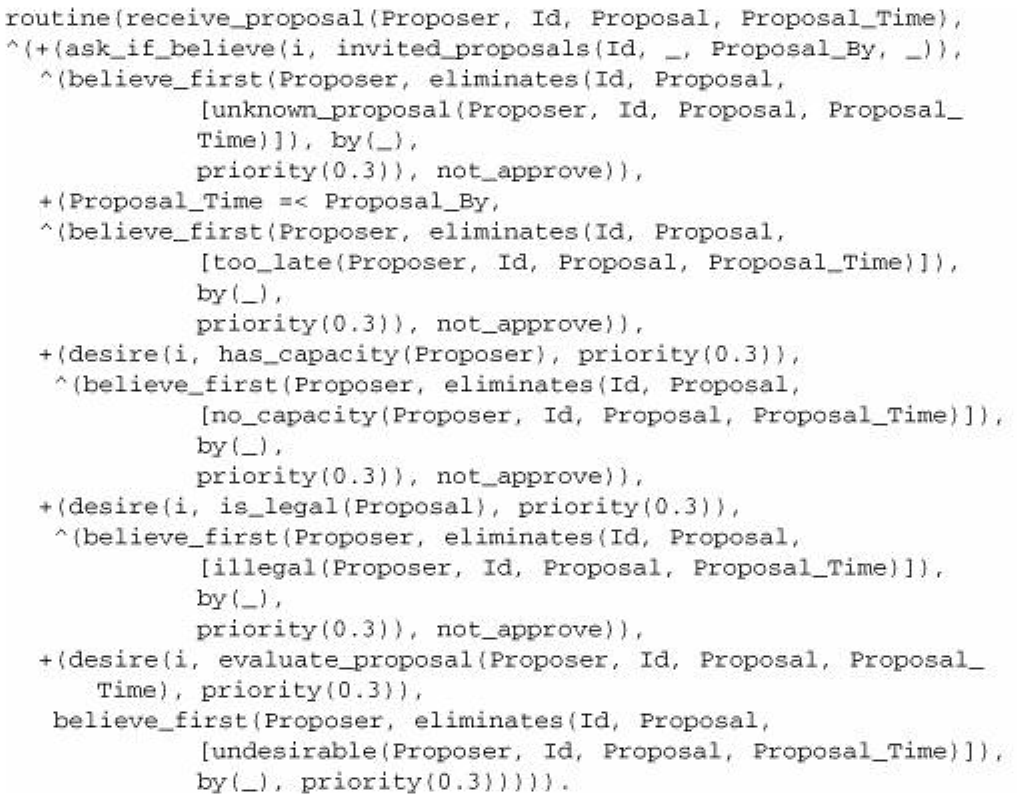
Recipients of an invitation to treat respond through the send_proposal cognitive routine.



```
routine(send_proposal(Inviter, Id, SoW, Proposal_By),
+(^(desire(i, has_capacity(Inviter) , priority(0.3)),
  evaluate_invitation(Inviter, Id, SoW, Proposal_By, Proposal),
  desire(i, is_legal(Proposal), priority(0.3)), now(Proposal_Time),
  Proposal_Time < Proposal_By, my_name(I),
  believe_first(Inviter, proposes(I, Id, Proposal, Proposal_Time),
      by(_), priority(0.3)),
  believe_first(i, invites(Inviter, Id, SoW, Proposal_By)),
  believe_first(i, proposes(I, Id, Proposal, Proposal_Time)),
  anticipate(i, believes(Inviter, retracts(Id, Proposal_By)),
      with(receive_retraction(Inviter, Id, Proposal_By)),
      priority(0.3)),
  anticipate(i, believes(Inviter, eliminates(Id, Proposal,
      Assessment)),with(receive_elimination(Id, Proposal, Assess-
      ment)), priority(0.3)),
anticipate(i, believes(Inviter, assesses(Id, Proposal,
      Assessment)),with(receive_elimination(Id, Proposal, Assess-
      ment)), priority(0.3))),
  ^(believe_first(Inviter, declines(I, Id), by(_), priority(0.3)),
  believe_first(i, invites(Inviter, Id, Proposal, Proposal_By)),
  believe_first(i, declines(I, Id))))).
```

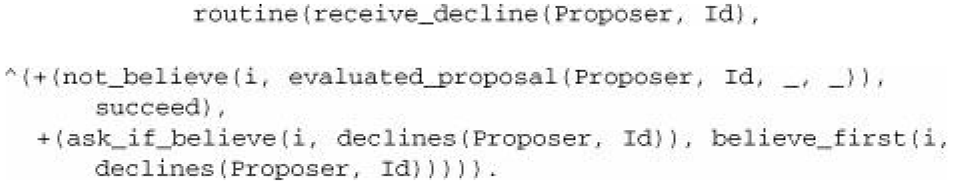
The send_proposal routine is anticipated by receipt of an invites(Inviter, Id, SoW, Proposal_By) belief where: Inviter identifies the issuer of the invitation to treat; Id is the unique identifier associated with the invitation to treat; SoW is the statement of work associated with the invitation; and Proposal_By is the deadline for a response. The receiving agent first examines the capacity of the inviting agent, and if suitable, evaluates the invitation. A successful evaluation results in a proposal. The legality of that proposal is considered and a check that the contracting deadline has not passed is also undertaken. The proposal is then sent to the inviting agent. It has the form proposes(I, Id, Proposal, Proposal_Time), where: I records the identity of the proposing agent; Id is the unique identifier for the invitation to treat; Proposal is the proposal being made; and Proposal_Time is the approximate time of the proposal. The proposing agent then records both the invitation that it received and the proposal that it responded with. It then anticipates a retraction, elimination, or assessment response toward the proposal from the inviting agent. If failure occurs at any stage during this sequence, the proposing agent instead issues a declines response to the invitation and records both the invitation and the decline response. The proposing agent may also change its mind by issuing a new decline or a new proposal prior to the proposal deadline.

The inviting agent should receive proposal or decline responses from agents receiving an invitation. The receive_proposal cognitive routine manages receipt of a proposal (Proposal) from proposer (Proposer) for the invitation with identifier (Id) and assesses the proposal's eligibility. If it is eligible, then it evaluates the proposal. Eligibility consists of a sequence of checks for: the proposal identifier (Id) being unknown; the proposal time (Proposal_Time) being too late; the proposer (Proposer) not having the legal capacity to contract; and the proposal (Proposal) not being legal. If the proposal is eligible it is evaluated. If it is evaluated unfavorably, then proposer (Proposer) is immediately notified of the proposal being eliminated as undesirable. This gives the proposer (Proposer) time to resubmit another proposal before the deadline By_Proposal is reached. If proposal (Proposal) is evaluated favorably, then the evaluate_proposal code results in an evaluated_proposal(Proposer, Id, Proposal, Assessment) belief, where Assessment is an assessment of the proposal. This is used for shortlisting.



```
routine(receive_proposal(Proposer, Id, Proposal, Proposal_Time),
^(+(ask_if_believe(i, invited_proposals(Id, _, Proposal_By, _)),
  ^(believe_first(Proposer, eliminates(Id, Proposal,
            [unknown_proposal(Proposer, Id, Proposal, Proposal_
            Time)]), by(_),
            priority(0.3)), not_approve)),
  +(Proposal_Time =< Proposal_By,
  ^(believe_first(Proposer, eliminates(Id, Proposal,
            [too_late(Proposer, Id, Proposal, Proposal_Time)]),
            by(_),
            priority(0.3)), not_approve)),
  +(desire(i, has_capacity(Proposer), priority(0.3)),
   ^(believe_first(Proposer, eliminates(Id, Proposal,
            [no_capacity(Proposer, Id, Proposal, Proposal_Time)]),
            by(_),
            priority(0.3)), not_approve)),
  +(desire(i, is_legal(Proposal), priority(0.3)),
   ^(believe_first(Proposer, eliminates(Id, Proposal,
            [illegal(Proposer, Id, Proposal, Proposal_Time)]),
            by(_),
            priority(0.3)), not_approve)),
  +(desire(i, evaluate_proposal(Proposer, Id, Proposal, Proposal_
      Time), priority(0.3)),
   believe_first(Proposer, eliminates(Id, Proposal,
            [undesirable(Proposer, Id, Proposal, Proposal_Time)]),
            by(_), priority(0.3)))))).
```

The receive_decline cognitive routine manages receipt of decline speech acts from the proposing agent Proposer with respect to the invitation designated by identifier (Id). The decline may be an immediate decline after receipt of the invitation, or it could be a subsequent decline after a proposal has already been sent, in which case the intent is to withdraw the proposal. receive_decline removes the evaluation of any previous proposals from Proposer for the invitation associated with Id and believes the decline information from Proposer unless it is already recorded.

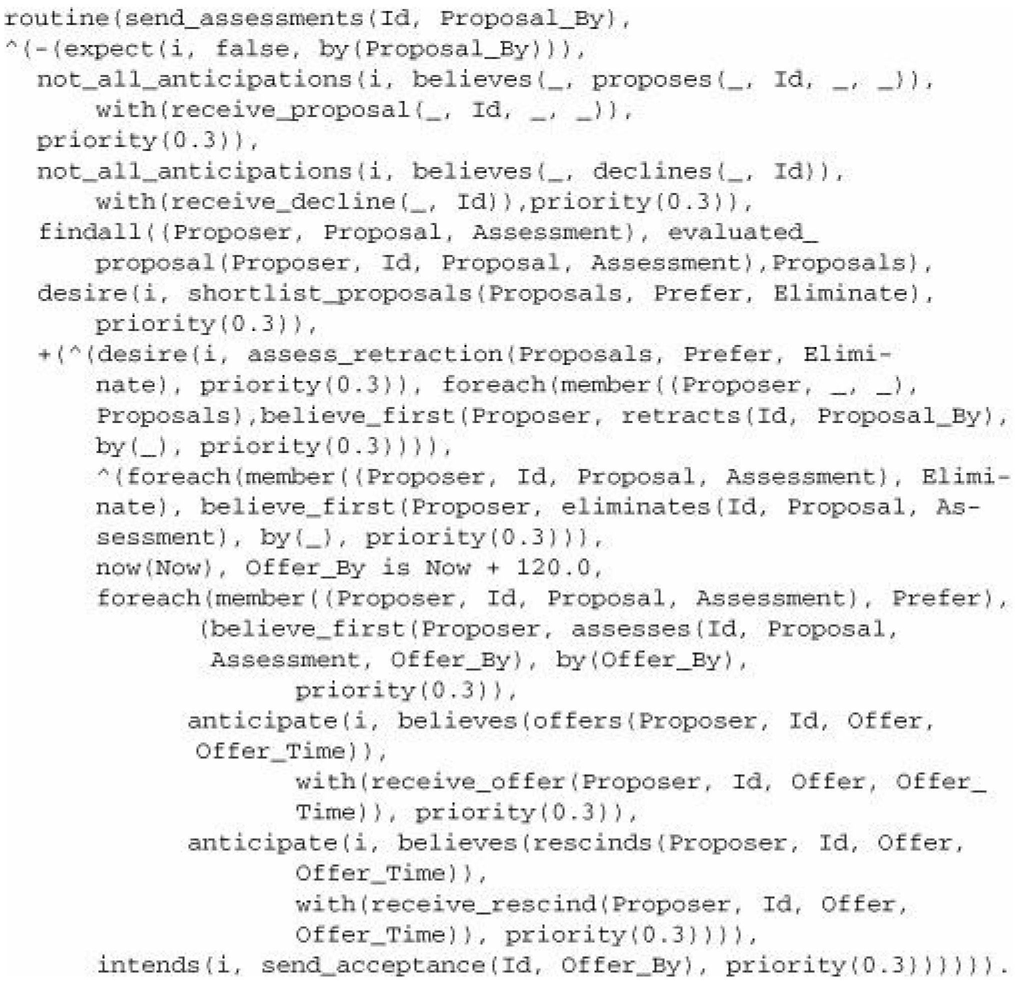


```
              routine(receive_decline(Proposer, Id),

^(+(not_believe(i, evaluated_proposal(Proposer, Id, _, _)),
     succeed),
  +(ask_if_believe(i, declines(Proposer, Id)), believe_first(i,
     declines(Proposer, Id))))).
```

8.3.1.3 Assessments

The send_assessments cognitive routine is invoked at the end of execution of the send_invitations routine with the unique identifier Id and the proposal deadline Proposal_By. It initially waits until the Proposal_By deadline has been reached, after which it removes the current anticipations for proposals and declines for the Id, and retrieves all evaluated proposals. These proposals are then shortlisted into the set that is preferred and the set that is to be eliminated. Based on this partitioning, a decision is then made on whether to retract the initial invitation. In that event, each proposer is notified of the retraction. Otherwise, each proposer of an eliminated proposal is notified that their proposal has been eliminated, while each proposer of a preferred proposal receives an assessment of their proposal, and subsequent offers and offer rescinds are anticipated from that proposer. send_acceptance is then invoked with an offer deadline.



```
routine(send_assessments(Id, Proposal_By),
^(-(expect(i, false, by(Proposal_By))),
  not_all_anticipations(i, believes(_, proposes(_, Id, _, _)),
      with(receive_proposal(_, Id, _, _)),
  priority(0.3)),
  not_all_anticipations(i, believes(_, declines(_, Id)),
      with(receive_decline(_, Id)),priority(0.3)),
  findall((Proposer, Proposal, Assessment), evaluated_
      proposal(Proposer, Id, Proposal, Assessment),Proposals),
  desire(i, shortlist_proposals(Proposals, Prefer, Eliminate),
      priority(0.3)),
  +(^(desire(i, assess_retraction(Proposals, Prefer, Elimi-
      nate), priority(0.3)), foreach(member((Proposer, _, _),
      Proposals),believe_first(Proposer, retracts(Id, Proposal_By),
      by(_), priority(0.3)))),
      ^(foreach(member((Proposer, Id, Proposal, Assessment), Elimi-
      nate), believe_first(Proposer, eliminates(Id, Proposal, As-
      sessment), by(_), priority(0.3))),
      now(Now), Offer_By is Now + 120.0,
      foreach(member((Proposer, Id, Proposal, Assessment), Prefer),
             (believe_first(Proposer, assesses(Id, Proposal,
              Assessment, Offer_By), by(Offer_By),
                    priority(0.3)),
            anticipate(i, believes(offers(Proposer, Id, Offer,
             Offer_Time)),
                    with(receive_offer(Proposer, Id, Offer, Offer_
                    Time)), priority(0.3)),
            anticipate(i, believes(rescinds(Proposer, Id, Offer,
                    Offer_Time)),
                    with(receive_rescind(Proposer, Id, Offer,
                    Offer_Time)), priority(0.3)))),
      intends(i, send_acceptance(Id, Offer_By), priority(0.3)))))).
```

8.3.1.4 Offer and Acceptance

The remainder of the negotiation mirrors the portion already discussed. offers speech acts parallel proposes speech acts; rescinds speech acts parallel declines speech acts; rejects speech acts parallel eliminates speech acts; accepts speech acts parallel assesses speech acts. Shortlisting is replaced by evaluation for acceptance.

8.3.2 Contract Performance

Cognitive routines for contract performance are not discussed here, other than to note that acceptance of an offer will result in a monitoring routine being invoked with expectations for each of the milestones. Whenever any of these expectations are not satisfied, contract remedies may be invoked.

8.3.3 Contract Remedies

Contract remedies require the establishment of two computational mechanisms.

The first is a trust rating system that monitors the contractual performance of an agent over time. A mathematical expectation approach was suggested in Section 8.2.3.2.

The monitoring of these ratings requires a centralized rating agency or a universally applied ratings policy. The computational details are not presented here.

The second is a framework for monetary compensation. This would require the electronic transfer of funds or insurance credentials. The computational details are also not presented here.

Sample Vignette

In the North Atlantis vignette, a ship is suspected of carrying resupply munitions. Persistent surveillance and analysis of the ship's movement, and vessels which might come into contact with it, is therefore warranted. The legal agreement protocol can be used to manage various fusion systems to meet this challenge.

Given the ship's current and likely projected routes, a list of trusted fusion agents (people and machines) can by identified. This could include: Navy frigates, Navy patrol boats, Navy submarines, Air Force surveillance aircrafts, border patrol aircrafts, unmanned aerial vehicles, ally flagged merchant ships, allied fishing vessels, lighthouse keepers, military command centres, and intelligence agencies. An invitation to treat could be issued to the above to solicit proposals for assistance from them. Some of the recipients will evaluate and decline the invitation, while others will respond with proposals that usually only partially meet the overall objective. Some of these proposals may be received too late or may even be illegal, perhaps by seeking to violate the Telecommunications Act [28]. These proposals will be eliminated immediately, and the remainder will be evaluated. The evaluated proposals are then shortlisted. Shortlisting includes scheduling which fusion components are able to provide surveillance or analysis during which times and/or locations, the result being a fusion package aimed at meeting the objective. If a fusion package is not possible, then the invitation is retracted.

Each fusion agent allocated a task in the fusion package is notified of the assessment of their proposal. The assessment may differ from their proposal to some extent. For some fusion agents the tasking assessment may involve them having to fly new routes or allocate a human or machine analyst, while for others it may simply be a matter of keeping a watchful eye while continuing on with their current and planned activities. The execution of some tasks will be conditional on certain events occurring. Some tasks will be about the structure of information sharing, identifying how Level 0 to Level 3 fusion tasks are to be distributed. Where it is sensible to do so, some equipment could negotiate tasks without consulting with human operators. Each fusion agent receiving an assessment can either make an offer to that effect, or rescind their proposals and offers. Ultimately the contracting agent retracts the invitation, or else rejects and accepts offers from different fusion agents. Rejection could occur in order to formulate a counter-offer, initially via an invitation to perform a slightly different task. Acceptance leads to an initial fusion package, but dynamic negotiation of the sort discussed will continually modify the package as events unfold.

rences

[1] Steinberg, A. N., C. L. Bowman, and F. E. White, "Revisions to the JDL Data Fusion Model," The *Joint NATO/IRIS Conference*, 1998.

[2] Steinberg, A. N., and C. L. Bowman, C. L., "Rethinking the JDL Data Fusion Levels," *Proceedings of the National Symposium on Sensor and Data Fusion*, 2004.

[3] Lambert, D. A., "Ubiquitous Command and Control," *Proceedings of IEEE Information, Decision and Control Conference*, 1999.

[4] Lambert, D. A., and J. Scholz, "A Dialectic for Network Centric Warfare," *Proceedings of the International Command and Control Research and Technology Symposium (ICCRTS)*, 2005.

[5] Reason, J., *Human Error*, Cambridge University Press, 1990.

[6] McCann, C. (ed.), *The Human in Command: Exploring the Modern Military Experience*, Kluwer Academic Publishers, 2000.

[7] Smith, R. G., "The Contract Net Protocol: High Level Communication and Control in a Distributed Problem Solver.," *IEEE Tr. on Computers*, Vol. C-29, No. 12, 1980, pp. 1104–1113.

[8] Fischer, K., J. P. Muller, and M. Pischel, "Cooperative Transportation Scheduling: An Application Domain for DAI." *Applied Artificial Intelligence*, Vol. 10, No. 1, 1996, pp. 1–33.

[9] Perugini, D., D. A. Lambert, L. Sterling, and A. Pearce, "Distributed Information Fusion Agents," *International Conference on Information Fusion*, 2003.

10] Lambert, D. A., "A Blueprint for Higher-Level Fusion Systems," *Journal of Information Fusion*, Vol. 10, 2009, pp. 6–24.

11] Schepple, K., and J. Waldron, "Contractarian Methods in Political and Legal Evaluation," *Yale Journal of Law & the Humanities*, Vol 3, 1991, p. 195.

12] Langbein, J. H. "The Contractarian Basis of the Law of Trusts," *Yale Law Journal*, Vol. 105, 1995, pp. 625–671.

13] *Harvey v Facey* [1893] AC 552.

14] Winfield, P. H. "Some Aspects of Offer and Acceptance," *Law Quarterly Review*, Vol. 55, 1939, pp. 499–517.

15] *Payne v Cave* (1789) 3 TR 148.

16] Paterson, J., A. Robertson, and A. Duke, *Principles of Contract Law (3rd Edition)*, Thomas Reuters, 2009, p. 51.

17] *Minors' Contracts (Miscellaneous Provisions) Act 1979; Sultman v Bond* [1956] QST 180.

18] Bankruptcy Act 1966 (Cth).

19] Mental Health Act 1974.

20] Hill, J., and J. W. Carter, "Severance, Illegal Contract and Company Law," *Company and Securities Law Journal*, Vol. 4, p. 183, 1986.

21] Andrade, F., et al., "Contracting Agents: Legal Personality and Representation," *Artificial Intelligence and Law*, Vol. 15, p. 357, 2007.

22] *Masters v Cameron* (1954) 91 CLR 353.

23] *Batt v Onslow* (1892) 13 LR (NSW) Eq 79.

24] *United Nations Commission on International Trade Law (UNCITRAL) Model Law on Electronic Commerce*, 1996, (www.uncitral.org).

25] Lambert, D. A., and C. Nowak, “The Mephisto Conceptual Framework,” *DSTO Technical Report TR-2162*, Department of Defence, 2008.

26] Wright, L., “Ought the Doctrine of Consideration to be Abolished from the Common Law?” *Harvard Law Review*, Vol. 49, p. 1225 – 1251, 1936.

27] Fama, E. F., “Efficient Captial Markets: A Review of Theory and Empirical Work,” *Journal of Finance*, Vol. 25, p. 383.

28] Telecommunications (Interception and Access) Act 1979 (Cth).

Part III
Human-System Interaction

CHAPTER 9

User-Defined Operating Picture (UDOP)

Peter D. Houghton (United Kingdom) and Elizabeth K. Bowman (United States)

The concept of the user-defined operating picture (UDOP) has risen as a response to observed limitations of current military "picturing" capabilities in coping with today's and tomorrow's evolving operational needs. Many of these challenges can only be adequately addressed through the provision of a more flexible and end-user community driven capability; one which supports a more rapid and agile configuration of picturing systems. In this chapter, we discuss the nature of the challenges and how these affect current picturing systems. We describe what UDOP characteristics could address these challenges, and consider what could be done to realize a UDOP capability by exploiting understanding gained from the development of prototypical UDOP implementations. This chapter concludes with a discussion on a few of the remaining issues that need to be addressed before UDOP could become a reality. The discussion in this chapter is an expression of a current position with respect to the concepts and is intended to stimulate a discussion on the needs for and the nature of UDOP.

Introduction

If high-level information fusion (HLIF) systems are to remain effective in addressing the diversity of future problems faced by defence enterprises, then this will require the provision of a flexible and end-user community-driven capability; one which supports a more rapid and agile configuration of picturing systems. Such capabilities have started to appear and are frequently referred to as a User Defined Operating[1] Picture (UDOP). The current perceived purpose of military operating pictures is situational awareness; but the potential role of highly flexible operating picture capabilities such as UDOP is much more complex and comprehensive than this simple conception implies.

To give a sense of what this more comprehensive role might be, it is proposed here that UDOP could provide a means to (a) improve individual awareness and sensemaking, (b) support more sophisticated interactive tasks such as analysis, exploration, pattern, and trend spotting, (c) support decision making by feeding inputs into modeling and decision aids, (d) support sensemaking and decision making in collaborative team contexts, (e) provide a common frame of reference, (f) reduce cognitive load via exploitation of visual techniques, and (g) enable effective workflow experiences.

Given this broader representation of picturing needs, one can then pose the question as to whether current picturing capabilities can support these needs, especially in response to the new and emerging operational challenges[2]. To enumerate some examples of the sorts of challenges envisaged, it is suggested that future picturing capabilities need to be able to operate effectively in an environment where there is:

- A more varied set of problems to address, many of which will be complex and wicked in nature;
- A greater variety of tasks to support, with a greater preponderance that are analytic and exploratory (rather than focusing on monitoring and control);
- A varied set of actor organizations to involve in picturing activities, many of which will not be familiar with military approaches and who have different picturing problems to address within a comprehensive approach;
- An increasing number of data sources, which contain a greater variety of information about a greater variety of things of interest, many of which will not be purely military.

The UDOP is defined as user-modifiable configuration of services that provide data access, processing and visualization mechanisms for a wide variety of data/information sources with the intent of supporting user communities in understanding complex situations from historical, current, or anticipatory perspectives. The flexibility implicit in UDOP is intended to enable user communities to specify domain-specific business logic to create value-added information products derived from the raw data and to visualize these products to extract insights not necessarily attainable from the use of any single source. Ideally, the capability should also include support to collaboration to enable the sharing of information [1], improved consistency of situation awareness [2], and decision making that is able to evolve with the dynamic nature of the environment represented by the UDOP.

In the remainder of this chapter, we will explore the above issues in more detail, including where the needs have come from and why current systems struggle. We will define what characterizes a UDOP, as opposed to existing capabilities, and will also explore possible ways of providing future UDOP capabilities, and we provide further illustrations that explain our conceptual model of a UDOP.

The Need for a New Picturing Capability: UDOP

9.2.1 Challenges with Picturing Capabilities

Arguably, before considering in detail what a UDOP is, or should be, it is necessary to first consider why there may be a need to progress beyond what is currently provided by military ‘picturing’ capabilities [2]. Thus, this section discusses the question of what the challenges, issues and limitations are with respect to current capabilities and why they may exist.

UDOP has its origins in the concept of the common operational picture (COP), which the *United States Department of Defense Dictionary of Military and Associated Terms* defined as “a single identical display of relevant information shared by more than one command. A common operational picture facilitates collaborative planning and assists all echelons to achieve situational awareness.” Another definition, espoused by Defense Research Development Canada-Valcartier, is “the Common Operational Picture is the integrated capability to receive, correlate and display heterogeneous sources of information in order to provide a consistent view of the battlespace.” Note that this latter definition focuses on the integration capability versus the end product. Moreover, the end product is not simply a picture but an understanding and a consistent view. The COP is not the only term used to refer to such picturing capabilities; whilst it is arguably the most popular, there are many others in current use. Historically, there has been a plethora of such terms, including the Common Recognized Operational Picture (CROP) and the Joint Operations Picture (JOP).

While there are differences among these picturing concepts, there is also a considerable degree of similarity. The following issues are arguably common to all of the above picturing system/concept types:

- They seek to embody and sustain single authoritative views of aspects of the environment that can be very limiting for current and future operations;
- They tend to focus on one type of view, typically positional and geospatial, with complex military symbology placed over mapping of various forms;
- It is often not clear whose purpose such single views of reality are intended to aid and how they are expected to provide this aid;
- They are often poor at adapting to the changing needs of operations where significant aspects are frequently changing;
- They run into considerable difficulties when constructing composite views (i.e., when they are not focused on small parts of single service views of a situation);
- They tend to focus on inputs to a process rather than assisting in the individual and collaborative creation of process outputs.

One might ask why these issues are endemic, but it appears that a primary reason is that these picturing capabilities were either constructed in the Cold War era, or based on the longer-lasting assumptions that were embodied within it. They are thus optimized to support commanders operating in military environments that are complicated, but contain relatively stable, well understood and tractable, problem types.

In contrast to the Cold War era, modern crisis and conflict environments are likely to be considerably different. They are not purely military in nature, but instead involve many other actors, both supporting and in opposition to desired political outcomes. The problems faced by the military and other collaborating organizations in such environments are also complex and wicked [3], in that there is often no solution as such and, thus, aids designed to support "mechanistic" thinking and processes are likely to be more of a hindrance than a benefit. Problems will also be very varied and varying, in that what is of significance in one crisis situation, will be different from the previous one, and even different from a previous phase of a crisis situation which is ongoing. So from a military contextual point of view, future "picturing capabilities" need to be able to operate effectively in this environment where there is a more varied set of problems to address, and a greater variety of tasks to support. This is likely to lead to a greater preponderance of command activity which is analytic and exploratory rather than more traditional monitoring and control of discrete elements. There will also be a more varied set of actor organizations to involve in picturing activities, many of which will not be familiar with military approaches and who will have different problems to address within a Comprehensive Approach [4]. As a consequence there will be an increasing number of associated data sources that describe a more varied set of military and non-military aspects of the environment.

The COP, which is based on geospatial and tracking assumptions such as road constraints [5], may be sufficient to support simpler problems and better support lower-levels of military operational capability. However, for more complex situations, and to support the more advanced capabilities required by higher-levels of command, a much more sophisticated capability will be required, for example, one which includes the ability to facilitate tasks such as cross-agency collaborative planning.

Future technical and information environments will also likely be somewhat different, with a greater number of technical systems, and increased connections to external systems. These are likely to result in the need to consume a larger number of more diverse information feeds and services. With these come a greater variety of data formats and types, and a greater variety of information properties (e.g., wider range of uncertainty types). Finally, there will be an increasing number of information sources provided in the form of information services, some of which will be fleeting in nature and will use more recent technical system approaches, such as Service Oriented

Architectures (SOA) [2, 6, 7].

Changes to the underlying technical infrastructure will create different demands on components supporting interaction between the human and the information (i.e., including visualization and presentation services). For example, because of the nature of the data sources and the variety of tasks to be supported, the preponderance of geo-based visualizations in the COP will have to change. Instead, there will need to be increased support to a wider range of visualizations and in particular of abstract data types (e.g., abstract concepts such as population sentiment). Further, the exploratory and investigative nature of tasks will require a capability to visualize multiple viewpoints simultaneously, and hence employ approaches such as multiple-linked-views [8], which are intended to enable end users to gain higher-order insights for their specific set of tasks.

It should be obvious from the above characterization of the problems with current operating pictures, and what might be needed to address the new demands, that a UDOP will need to diverge considerably from earlier picturing concepts and implementations. The primary reason for this is the considerable difference in the context of use. It could be argued that rather than being an evolution of the COP that UDOP is potentially a revolutionary change that could be inspired by the high-level information fusion community [9] as expressed in Chapter 12.

It is recognized that the above characterization of the problems with current operating pictures, and potential future demands, is neither detailed nor comprehensive enough to adequately define and justify a UDOP. Thus it is imperative that an improved understanding of the present and potential future dissatisfaction with current COP concepts and instantiations be developed, providing a sounder foundation on which to base future UDOP development. While some evaluations of picturing capabilities have been carried out [10], they have been quite limited in nature. Thus, the required evidence to support an adequate characterization of the problems and future needs does not presently appear to exist.

9.2.2 Potential Universality of Picturing Challenges and Issues

The question addressed in this section is whether the challenges and issues with current picturing capabilities are universal at the individual, team/group, or higher echelon levels. Unfortunately, as there has not been sufficient characterization and evaluation of picturing capabilities, a definitive and evidence-based response cannot currently be provided. The following thus represents a series of hypothesized responses to the universality question. It is proposed that:

1. While there may be some similar generic concerns, it is highly unlikely that

precisely the same set of challenges and issues will emerge at every echelon's use of picturing capabilities.

2. The challenges and issues which are likely to be extant at any point in time will be dependent on contextual factors such as:
   - The nature of the operation/crisis/conflict and, thus, the types of problem being addressed;
   - The degree and rate of change in the operational situation;
   - The number and diversity of actors involved;
   - The command levels being supported;
   - The nature of the tasks to be supported—related to roles and responsibilities;
   - The communications, processing and human interaction capabilities available to support the delivery of data, computation, analysis and fusion processes, and the rendering of the required visualizations.

9.2.3 Impact of Picturing Challenges and Issues

The current picturing challenges and issues are likely to impact military capabilities quite broadly, covering timeliness, quality, understanding, sharing/collaboration, interpretation, and decision making. Despite the unknown nature of these, there are considerable assumptions made about the benefits of picturing capabilities; in fact, such benefits are implicit in the network centric warfare benefits chain [11]. However, finding evidence to support the veracity of such assumptions is very difficult, as is locating appropriate techniques for determining and employing appropriate measures to help develop such evidence.

It can therefore be concluded that there is a pressing requirement for future experimentation on both current COP and future UDOP picturing capabilities to examine the needs and costs, risks, benefits, and issues associated with picturing provisions. This experimentation should not focus entirely on the issues and concerns, but should also attempt to put a more quantitative/qualitative assessment on the benefits of particular features of picturing capabilities being employed.

Before such assessments can be conducted, appropriate measures also need to be devised. Due to insufficient attention being paid, tried and tested measures which enable reliable evaluation of the individual and organizational benefits of picturing capabilities are in short supply. Even if these measures were to be developed, there would be reasonable questions raised in regard to the extent to which such measures would be valid and to what degree conclusions derived from them could be safely used outside of the context in which they were taken. The fact that UDOP is intended to be flexible and adaptable enough to operate in a multitude of organizational settings adds yet further to the challenges faced in making reliable and repeatable measurements.

9.2.4 Defining Users and User Needs

The question addressed in this section is about who the real users of picturing capabilities actually are and the nature of their roles, responsibilities, tasks, activities, and information, analytic, communication, and collaboration needs.

Typically, it is possible to assess whether a new or "improved" capability is adding value by considering analytically whether such a capability is a good fit to the individual and collective needs of the end-user communities that the capability is intended to support. Thus, in theory, if we knew more about the nature and needs of UDOP users, this would provide an indication of their information needs and help determine whether the UDOP concept would be a good fit to address those needs. However, such an analytic approach is undermined by the very essence of what a UDOP is intended to be. That is, UDOP is about flexibility, adaptivity, change, and customization. It is therefore not a static capability addressing a static need.

However, perhaps one way of addressing this problem is to try and express what the outer envelope of UDOP capabilities is; that is, to define:

- The extent of its organizational coverage—from lower tactical to grand strategic, and extending breadth-wise to include all services, all functions, and also agencies and other government departments (OGDs) participating in a comprehensive approach.
- The nature of its adaptivity—what things are easy to change that most users will be able to effect, what things are more difficult to change and require an expert user, and what things are fixed and/or require system developer expertise to amend or fix.
- The extent of its inbuilt information management/information fusion capabilities—hence defining what might be provided separately, perhaps as collections of interconnected and interdependent services.
- The extent of its visualization capabilities—what sorts of graphical capabilities might be built-in, and what in the future might possibly be added, perhaps through concepts such as plug-ins.
- The extent of its support to collaboration—which parts may be built in and which parts may rely on more general purpose external collaboration services.

Because of the potential high degree of flexibility of UDOP, it cannot be considered from a purely technical standpoint (i.e., it should not be viewed purely as a potential future technical system). Ideally, and using the UK concept of Defence Lines of Development (DLOD[3]) [12], as an exemplar multi-perspective framework, UDOP should be considered from the viewpoints of all relevant DLODs. Thus, to effectively exploit UDOP, there would need to be appropriate consideration paid to training and doctrine lines especially, and also standard operating procedures (SOPs) and concepts of employment (CONEMP). For example, we might presume that, in certain circumstances, the sharing of UDOP views would be of direct benefit to others, especially in team or headquarters settings. However, this view sharing cannot be undertaken out of context, as one needs to know which views it makes sense to share within these organizational settings, and why, and what benefits are likely to accrue. What needs to be avoided is the creation of views, simply because UDOP technology might make this easy, while ignoring the fact that these views might not be conducive to good team and organizational processes.

It should be recognized that there is a contrary way of enabling UDOP view development, which is to permit all manner of views to be created, based on the assumption that only those that are demonstrated to have long term value will "survive." However, before considering such an ecological approach, it should be noted that it will only work if there is (a) freedom to experiment and (b) an effective feedback process that ensures only the "fittest" (from a perspective of what is best for operational outcomes) survive rather than those which might be personally preferred by a particular commander and/or his staff. Such approaches have arguably worked successfully in other domains, for example in the design and exploitation of applications on mobile devices such as the iPhone.

Finally, a potential third way, which has elements of the two suggestions above, would be to require all view creators to carefully consider the intended purpose of their new view. This might then be technically enforced, for example by requiring valid meta-data to be attached to all new views. Thus, when someone is considering the need to create a new view, a metadata matching process might suggest views which already exist and may serve the intended purpose, or ones which are sufficiently similar that they can be readily adapted.

9.2.5 Current Abilities to Define Own Pictures

Before overly laboring on the key point of difference for UDOP being the ability to define custom pictures and views, it is necessary to address the question as to whether users are currently able to define their own pictures. The brief answer to this question is that "it depends"; especially as there are a number of levels within a "definition" capability. Some picturing capabilities tightly constrain both what is displayed and how the displays are managed. For example, there may be strict rules on what is relevant to the picture and thus rules may be predefined and hard-coded into the system. In contrast, other picturing tools provide a degree of customization in the form of filtering options, so that given a set of feeds, only certain items will appear in the visualizations based on filtering rules and selected symbology sets.

Despite user definable options, such as filtering, COP views are typically quite constrained; they are limited either to relatively simple symbols/overlays placed over mapping or terrain backgrounds (effectively mirroring what can be done with paper maps), or alternatively, are provided with a few additional specialized views such as text-oriented tabular entry (TOTE) displays. Full control of data to be exploited, processing undertaken on that data and visual presentations are only in very rare circumstances under the direct control of the end-user. The more frequently observed situation is one where the end-user is considerably constrained by what the system designer deemed it was necessary to have.

Typical picturing systems don't just place constraints on the selection, processing and presentational aspects; rather they similarly tend to constrain the input data sources and types. The reasons for this are not purely technical, but also include historic policy and precedent, such as the need to know principle. Some of these policies are only just starting to change, despite acceptance of the need for greater information sharing as promoted by the network centric warfare concept of the need to share. UDOP is thus implicitly dependent on the enactment of power to the edge principles [13], supporting the sensemaking needs of all actors, even those at the edge of the network, in being able to select from a wide range of available information.

9.2.6 Purposes of Picturing Capabilities

The question to be addressed in this section is whether it is possible to define what the primary purposes are for constructing and using a picture. For example, is it is the intent to provide a common frame of reference to aid with decision making, analysis, information space exploration, problem exploration and resolution, and organizational coordination? It would seem that the simple answer to this question is "yes" for each of the reasons listed above. The following provides a brief discussion on a potential set of purposes.

A primary initial purpose must be to improve *individual awareness* and *sense-making*. What the picturing system is providing is a means to obtain a collection of information relevant to a particular individual's tasks, roles and responsibilities and to render this information collection in such a way that it improves their awareness and understanding of a particular real-world situation.

A second purpose is to provide individuals with a capability which is beyond mere presentation of information, and instead supports more sophisticated interactive tasks such as analysis, exploration, pattern spotting, trend spotting, anomaly detection, and comparison.

A third purpose is to support decision making, so picturing content can perhaps be fed into modeling and decision making aids, supporting what-if exploration, planning, and option comparison. Each of these may have their sets of visualization views to support them.

A fourth purpose could be as a common frame of reference, where the picture is not provided for any one individual or even any one team, but as a single common artifact to which many people can refer. Typically, common frames of reference pictures exploit the traditional symbology overlaid on maps and are based on the assumption that it is safe and sufficient to provide a single representation of reality. However—especially in the context of effects or outcome based approaches, and in the Canadian military perspective referred to earlier—there ought to be alternative frames of reference which provide a more nuanced and meaningful perception of what is happening in the external situation and within the team or wider enterprise. However, it is far from clear what the nature of these should be. The question of what the most effective forms of visualization are for common frame of reference purposes within complex multi-actor operational contexts has not been sufficiently explored.

A fifth purpose could be to reduce cognitive load. Visualizations, if chosen carefully, can reduce load on memory and assist with problem solving [14]. Likewise, workflow can be potentially improved either individually or as a part of team flow. Note that here we are extending the concept of individual flow [15] to refer to a team when it is working to coordinate activities without effort and conscious thought. Arguably, such a state is a desired outcome of intense collaboration [16].

It needs to be understood that the creation of a UDOP is not an end-objective in its own right. The intent is to significantly enhance both individual and collective sensemaking, leading to better individual and shared understanding. This improved understanding has the potential to enable greater organizational effectiveness via consequent improvements in coordination, collaboration, learning, and decision making. Arguably, the greatest potential benefit will accrue from supporting analysis tasks within complex situations, with wicked problems [3] where there is no single obvious, analytically derivable answer.

Characteristics of a UDOP

The question to be addressed in this section is whether it is possible to use the understanding of the challenges, needs, and requirements to better conceptualize a UDOP. In starting this endeavor, we first consider whether we can devise a concise and meaningful definition of what we perceive to be a UDOP[4]. The following are offered as an initial set of UDOP characteristics, whereby it is suggested that if a capability exhibits the majority of these, then it could quite reasonably be referred to as a UDOP:

- The data sources to be analysed, processed and displayed are user-selectable;
- The types of visualization employed to present either raw or processed data sources are also user-selectable and configurable;
- It is possible for users to create multiple associations of data to visualizations, such that many data sources can feed a single visualization and many views can simultaneously be deployed from the same data source;
- There is good support for rapid and iterative development of visualizations;
- There is strong support for interaction with the information, and the associated visualizations, to support effective exploration and discovery;
- There is user-configurable data processing, transformation and filtering;
- There are user-configurable multiple linked views[5];
- There is support for the sharing of UDOP configurations and components, including views, pictures, templates, overlays, and configurations;
- There are integrated collaboration capabilities, such as shareable pointers, and textual annotations that relate to either views, sets of views, or view elements.

Figure 9.1 provides a graphic portrayal of some of the user configurable aspects discussed above, represented in the form of layered services, where an architectural approach which is likely to be employed to construct a UDOP.

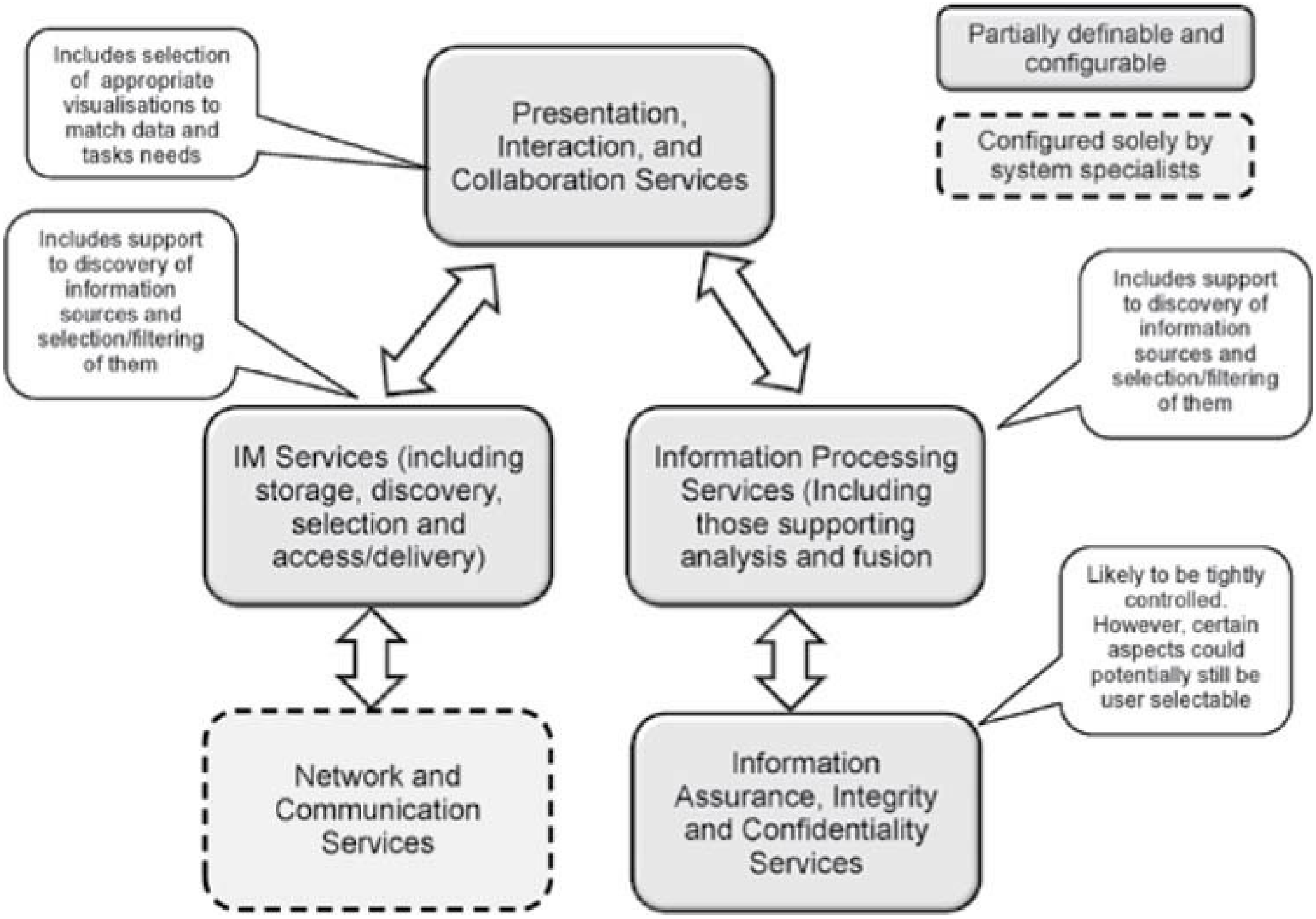


Figure 9.1 Concept of user configurable layered services supporting UDOP.

The initial question we posed in this section was “What makes a UDOP?” One might ask whether it is having a button bar, or whether it is about having layers and pluggable components, or having layers and the ability to configure the inputs. In reality, none of these are enough on their own. In addition to thinking about the technical characteristics of a UDOP, we must also consider some of the human interface aspects. These include:

- The fact that agreed businesses processes and the sharing of good practice will all still be needed to effectively exploit UDOP capabilities. The existence of UDOP is not a licence for data and visualization anarchy.
- That processes and concepts of use [17] will need to define the degree of flexibility in the use of UDOP capabilities that is safe to use in particular circumstances.
- That we need a clear understanding of what the term ‘user’ is referring to in each particular employment of a UDOP. For example, this term may refer to individuals, teams, organizations, or even multiple organizations such as federations or

coalitions.

- A recognition that UDOP is not necessarily constrained to be an operational capability, as in military operational level[6] of command [18], as the use of such picturing capabilities could be beneficial at multiple levels of command and in different domains. This applies (albeit differently) to tactical, operational, strategic, cyber, whole of government, and comprehensive approach contexts.
- The fact that there are a number of implicit risks in employing the UDOP, some of which can only be mitigated through organizational means. For example, the UDOP could provide ready access to information that is not appropriate to employ for particular tasks or at certain levels of command. As a consequence, individuals and organizations would need to be educated to use the potential flexibility of the UDOP capability with care.

Realizing a Future UDOP Capability

9.4.1 Developing an Understanding of Components and Architectures

If there is a desire to realize the UDOP concept in the form of a future fully working system, we will need to develop a much better understanding of the primary features of a future UDOP and conduct explorations into feasible and desirable components and architectures. One way of starting this process is to consider some of the partial and prototypical UDOP implementations that already exist. For example, Mitre Corp. was developing a UDOP technical implementation in 2007 [19]. Their architecture was divided into an information assurance layer, an information infrastructure layer, and an application layer (see Figure 9.2).

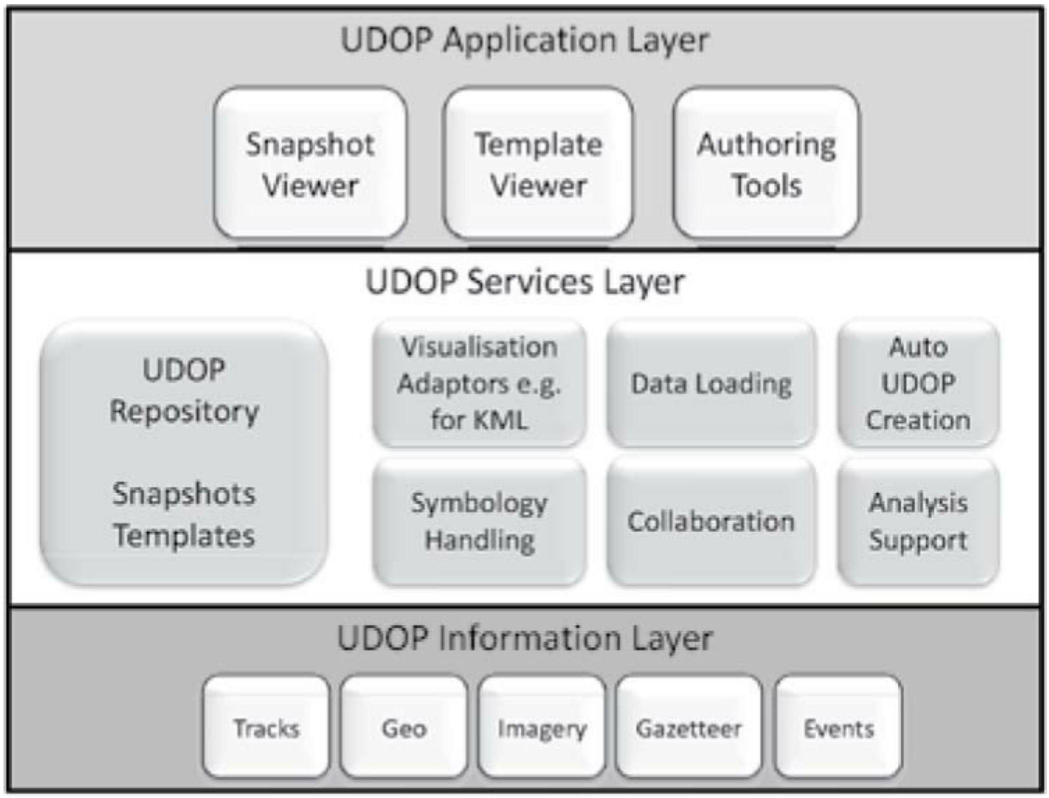


Figure 9.2 Mitre view of a UDOP architecture. (Adapted from [19].)

Another important architectural concept for a UDOP is the pipeline. The Mitre prototype system described above also used the concept of a pipeline as shown in Figure 9.3.

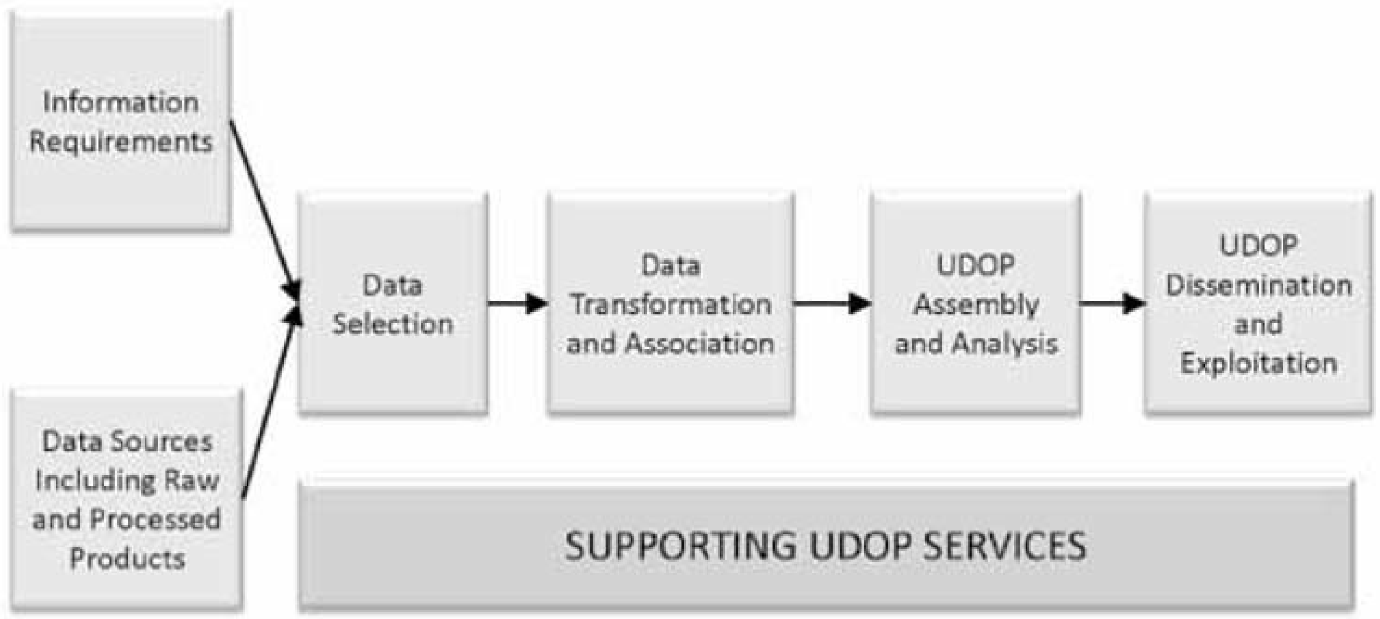


Figure 9.3 Mitre view of a UDOP pipeline. (Adapted from [19].)

However, it should be recognized that there are many potential architectures and pipelines that could be used, such the one shown in Figure 9.4.

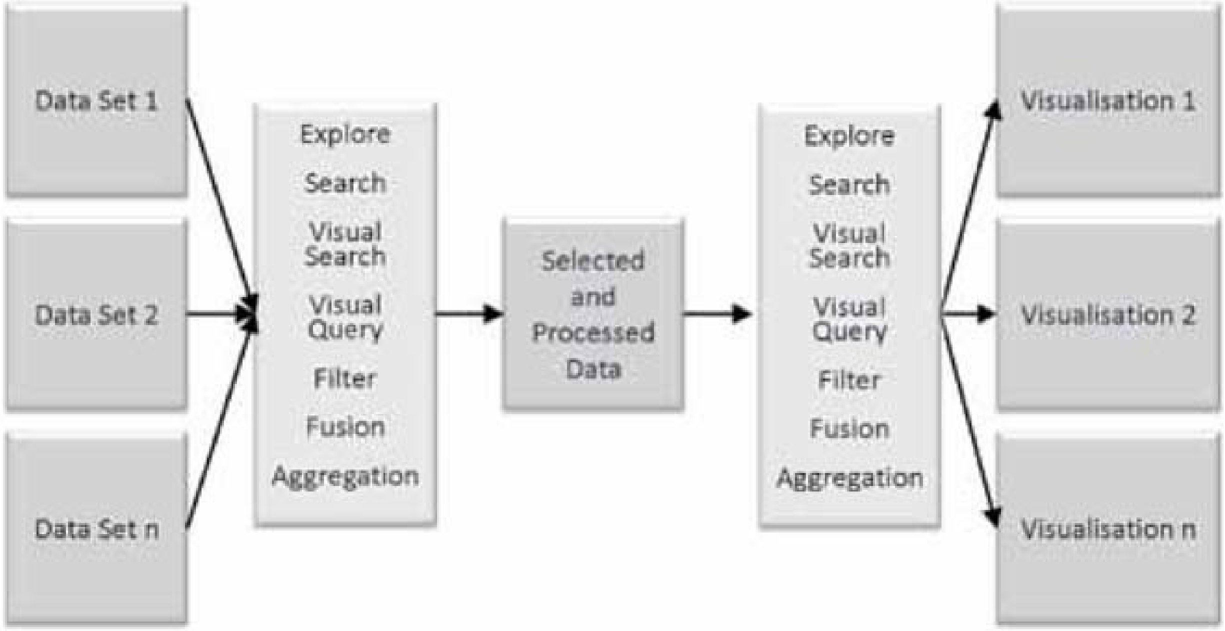


Figure 9.4 An alternative pipeline model.

An important feature that has to be supported in the architecture, and is critical for effective exploitation of the UDOP concept, is the degree of help provided to the user with the UDOP configuration process. There is a multitude of ways that this can be achieved, a prime example being that developed for the U.S. Air Force Research Lab (AFRL) Airspace Concepts Evaluation System (ACES) Viewer prototype [20]. ACES

Viewer starts the process of data processing and visualization configuration via a tabular style interface. However, once the configuration has been built, it can be visualized and subsequently modified using a visual control, referred to as a visualization composition graph. This exposes the detail of the pipeline models for a specific configuration thread through the system. An example of such a graph is shown in Figure 9.5.

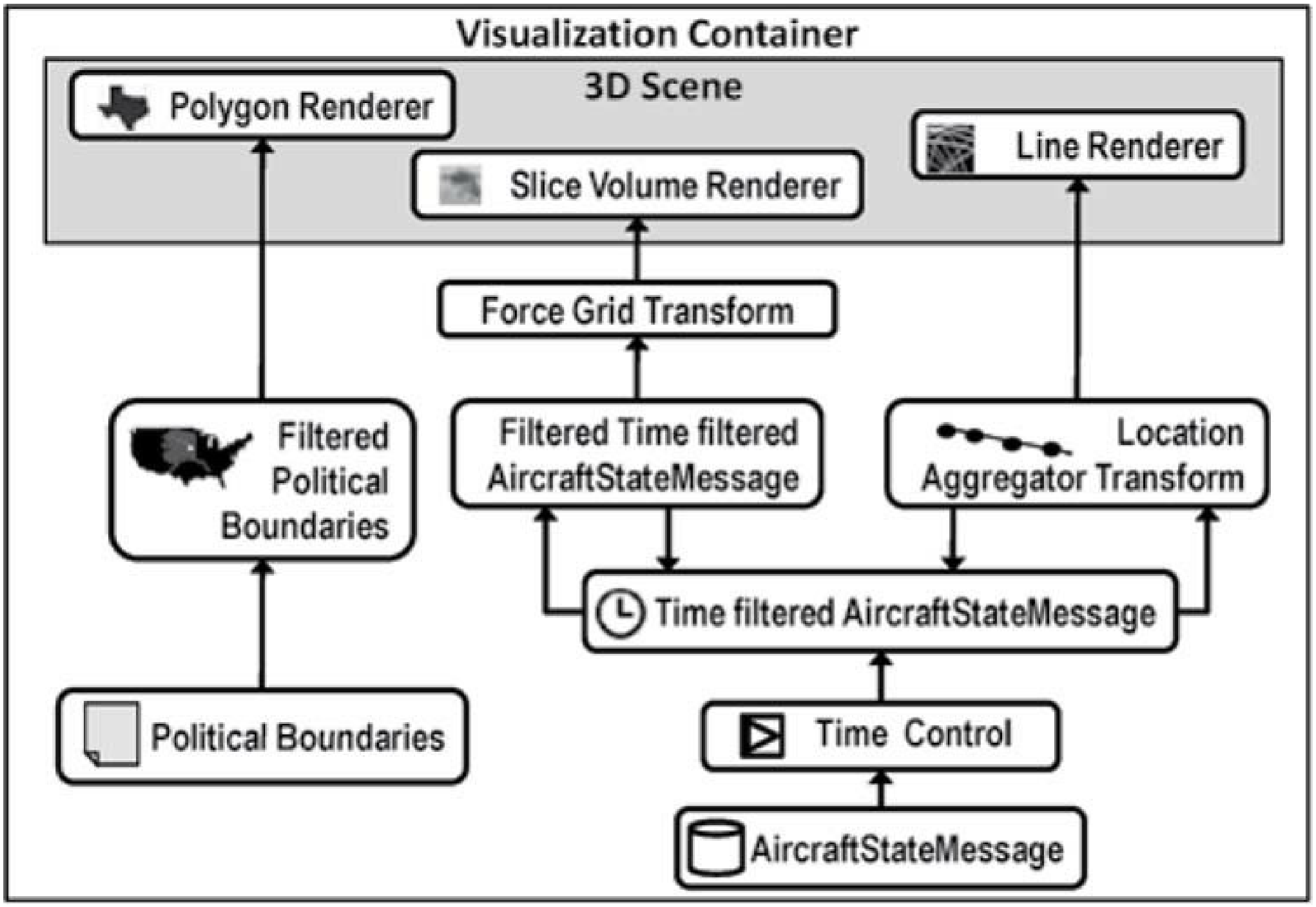


Figure 9.5 Visualization composition graph. (Adapted from [20].)

A comprehensive analysis of the variety of UDOP implementations, their architectures and components, strengths and weaknesses, and likely fit with evolving infrastructural approaches is beyond the scope of this chapter. However, it nonetheless is a necessary step that will need to be taken if UDOP is to emerge as the replacement for the prior COP concepts. Thus, rather than expand on components and architectures further, we conclude with a revisit to one final aspect of the "aid-to-user configuration" topic discussed above; in this case, focusing on the visualization component.

9.4.2 Providing Guidance for Exploitation of UDOP Visualizations

Configuration of the visualization component of the UDOP pipeline is particularly challenging because the most likely end users will not be experts in the design and application of visualizations to enhance human sensemaking processes. This raises an interesting question as to whether it would be feasible to provide “within-tool” guidance to UDOP users as to which visualizations are best suited for which purposes. Our analysis of the sociotechnical aspects of UDOP suggests that we are a considerable distance away from being able to solve this problem effectively. The following discussion considers the scale and nature of the issues.

We are attempting to find an approach which can address the likely effectiveness and validity of particular visualizations based both on the data to which they purport to represent and the task and context within which the visualizations are being used. Figure 9.6 provides a sample simplistic model as to how guidance could, in principle, be provided.

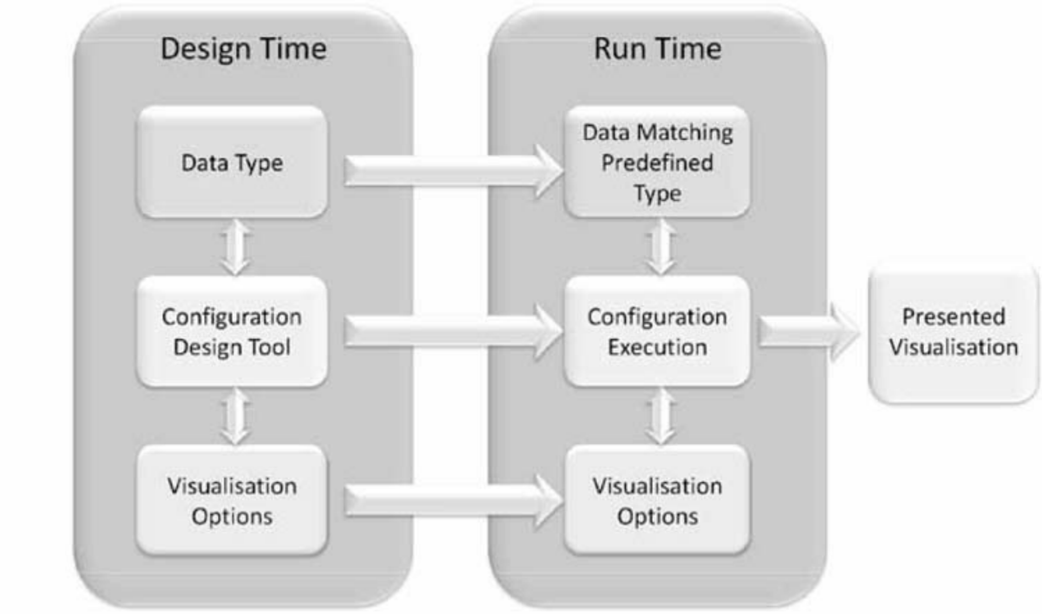


Figure 9.6 Provision of guidance on valid visualizations.

The concept portrayed in Figure 9.6 suggests that it may be possible to encode guidance on which visualization options are valid for particular data types at design time. This could then be exploited at run time via a component that would exploit the selected data types to provide, or suggest, appropriate/valid visualizations based on a set which was encoded at design time.

While this concept is superficially attractive and theoretically possible, there is in practice a plethora of problems to overcome. For example, if the UDOP system provides the ability to write customized scripts for any of the analytic/fusion components within the pipeline, then the approach would fail, as the pre-configuration process would have no knowledge of later customizations. One means of dealing with this problem would be to prevent such customizations being made by anyone other than expert users, who would have sufficient knowledge to construct appropriate links to valid visualizations. Another approach would be for in-tool automation to take account of the context of the tool user. However, this would then require the supply of information on user context, which is beyond current technical means to reliably ascertain and provide.

Previous bodies of research which are potentially relevant to this question include that on the Reference Model for Visualization (RM-Vis) [21]. RM-Vis is potentially relevant as it includes the notion of locating appropriate visualizations to meet a range of different information needs. As such, it potentially provides encyclopaedic information of which decisions on appropriate data and visualization pairings could be made. Subsequent research work extended RM-Vis is to develop the IMAGO[7] system to present the user with a series of different options which could be explored in a testing environment before real use. Defence Research and Development Canada (DRDC) used IMAGO to provide a maritime intelligence, surveillance and reconnaissance testbed, focused on demonstrating, designing, and prototyping novel information visualization techniques for a Recognized Maritime Picture (RMP) [18].

As has been noted previously, if the UDOP concept were to be properly instantiated, then it would in principle be easy and quick to create new views. One way of providing some degree of order to a situation where too great a variety of views is being created would be to encourage all view creators to consider the intended purpose of their new view. Note that, from a technical perspective, this might be enforced by requiring valid metadata to be attached to all new views. Then when someone is considering the need to create a new view, they may either find a view already exists for the intended purpose, or that there is one sufficiently similar that can be safely adapted.

9.4.3 Feasibility of UDOP

Current and future operations demonstrate a pressing need for instantiations of the UDOP concept. Existing approaches have been left wanting, and we have described some of the necessary characteristics a UDOP might need to have to overcome problems observed in prior COP solutions. It is apparent there have been many attempts to develop either prototypical or real UDOP implementations. However, it is also apparent that there has yet to be any substantial adoption of UDOP concepts and technologies across multiple organizations and for multiple purposes, contrary to what might have been expected given the urgency of the need and the maturity of some of the existing technical work. Therefore, may be quite reasonably asked what is currently preventing further progress on UDOP employment.

First, the scale of the challenge must be recognized. To address all or even the majority of the issues for UDOP that have been raised in this chapter would be a considerable endeavor. It may thus be the case that no current implementation is appropriate for universal adoption. In addition, the risks of moving from 'tried and trusted' COP concepts may also be perceived to be too great.

Second, we also need to recognize that to bring UDOP into existence, there is a need to make progress in both the technical and organizational domains (the latter including political, cultural, policy, and security concerns). Thus, in progressing an objective to instantiate UDOP it is important that we identify the primary barriers in each the domains. For example, it is a well known fact that it is difficult to enable intra-organization and cross-organization sharing of data. UDOP potentially amplifies this problem when one considers the potential degree of access and analytic capability over the data that it could provide. Thus it would appear that the some of the strengths we seek from UDOP, could at the same time be the very things that might prevent its wider adoption.

9.4.4 Way Forward

Given the number and scale of the challenges and the potential barriers to adoption of a UDOP, it would be quite reasonable to ask whether UDOP solutions are indeed the way forward. To answer this question, we consider the experience from a recent implementation of a UDOP built to coordinate humanitarian relief in Haiti following the January 12, 2010, earthquake [23]. This natural disaster was complicated by the chaotic and uncoordinated relief and aid flowing into the country by many different governmental and humanitarian agencies. A user-friendly Geographic Information System (GIS) was produced in a short period of time to unite over 2,000 users to share data aggregated through GIS tools, associated databases, and mobile applications in a geospatial perspective. The Haiti UDOP was built on the browser-based Google Earth plug-in and the iSpatial framework [24]. This framework allowed users to integrate tools for their unique purposes and to share data. The GIS-based functionality allowed users to overcome challenges associated with a multiplexity of languages across the user community, time pressure, and the lack of a central controlling body. For example, as ad hoc tent cities or aid stations were identified in duplicate reports, these were resolved through the "universal language of geoloation" [24, p. 4]. The Haiti UDOP included three core features: data import, links to Uniform Resource Locator (URL)s for sending/rendering data in the UDOP, and a spatial content export capability that allowed users to exploit the tool as a repository for creating content and as a means of indexing of existing content. Additional tools were built into the UDOP over time to allow users to label and annotate images. Underlying these capabilities was an automatic recording feature that provided a means to determine the lineage and pedigree of all data. The UDOP developers noted that the user community learned tool features quickly and experience in interacting in a spatial collaborative environment.

The Haiti UDOP was accessed via mobile devices for collecting and storing data with good results. However, the authors acknowledge that managing the large data stores was a challenge. These challenges included data validity, redundancy, uncertainty of whether data was outdated, and dealing with the delicate balance required between limiting user permissions and allowing unfettered access to data. A study group employed by the UDOP developer examined the lessons learned by the Haiti UDOP to document considerations for future UDOP development [23]. These findings included:

- The lack of a scalable paradigm for content management;
- The limitations of the system architecture that relied on a file-based approach for representing geographic layers;
- The need to represent data provenance, accuracy, and reliability;
- The need for processes for data input and management;
- An approach to symbology appropriate for the user group [24].

This study group recommended a three-tier architecture similar to the diagrams shown previously in this chapter. These comprised a data tier, a web services tier, and a client tier. However, the demonstrated ability to compose a UDOP for the Haiti earthquake disaster in short order for over 2,000 users is a feat that suggests the UDOP concept is both feasible and attainable.

Another example of a UDOP is provided by the MightyMoRiver project [25], which integrated various multisource media, social media, and GIS system information into a UDOP. This UDOP instance provided overlays, textual reports, images, user collaboration tools, and user queries to tailor available information in support of emergency response. Additionally, information products developed in the UDOP could be quickly aggregated for dissemination to various media sources.

A Few Examples of Remaining Issues

The phase of research work which led to the development of this chapter concluded with a substantial number of issues remaining, and a plethora of questions left open and unanswered. It is suggested that these issues represent a set of outstanding research challenges that ideally ought to be addressed, as the majority are fundamental in nature, and are therefore not specific to any one technical implementation.

9.5.1 Awareness of Information Sources

The first issue raised here is one of how to effectively enable users to be made aware of what information sources are potentially available and accessible and understand what their respective properties are. It seems that there is indeed a significant problem with providing a view of available data, especially when it is voluminous and has significant variety, potentially varying substantially even within one information type. Current UDOP implementations may be avoiding this issue by assuming good prior knowledge of information sources and focusing their efforts on forming the data processing to visualization interconnection in the UDOP pipeline. This problem is compounded when there is a variety of different communities that are both providing data and consuming it, where there is even more variety and less implicit understanding of data availability and properties. Previous research has examined this issue of provenance, for example that on accountable visualization. The intent of accountable visualization is to make it possible to trace back from a visualization to data sources to explore and understand provenance [26]. In addition to understanding data properties there will also be complex "need to know" concerns to address, even regarding whether certain types of data exist.

In respect of data properties there will be a need to understand the validity classes that presented data belongs to within a single UDOP view. For example, a view might contain:

- Only "validated" data (e.g., the recognized air picture);
- Only raw data which has not been validated;
- A mix of validated and raw data with some means (ideally, a visual indication) to distinguish between them.

In addition to raw and validated data, there will also be a need to present hypothetical data, such as a view that might contain the results of a "what-if" type of exploration mixed with "real" data. Alternatively, a view might represent an entirely hypothetical situation. As with the validity classes described above, there would need to be an obvious means for end-users to recognize which type of data is contained in each view.

Ideally, it should also be possible to provide a means of defining and presenting the overall validity of a view (i.e., distinguishing between one based mostly on raw data and one based entirely on assessed data). The mechanisms which support validation of data and views could potentially be provided as a specific subset of the configurable information processing services that comprise one of the primary service layers of UDOP.

9.5.2 Selecting Information Sources

A second issue to address is the mechanism by which one can select the information resources required and have these selections maintained through time, even when the availability of the data sources and their content is varying. We must therefore consider quite carefully what it means to select and have a view maintained. It might be necessary, for example, for the system to warn end-users if any fundamental properties of the data (perhaps lodged in meta-data) that have changed from the previous time that similar data was displayed. If this was a requirement, the UDOP system would need to be able to track changes in the metadata, understand what the end-user history has been, and have a nonintrusive mechanism for warning about changes.

9.5.3 Dealing with Remaining Need-to-Know Constraints

In relation to earlier comments about UDOP being dependent on changes in information sharing culture, it remains the fact that some need-to-know constraints are still likely to be in operation. Therefore, UDOP concepts and implementations will need to have a means for handling these appropriately. One might be unaware of the potential availability of information that is critical for understanding the question or issue at hand. However, the source may not understand potential users' needs, for if it did, exceptional access might be provided, dependent on the legitimacy of the need and its criticality. This raises the question of how to differentiate between legitimate and important needs, versus illegitimate and nonessential. One potential means might be a formal mechanism for passing information needs back down the UDOP pipeline, eventually to the data sources.

9.5.4 Catering for Varying End User Expertise

UDOP technology is potentially very powerful and flexible, but brings with it both complexity and risk of misconfiguration to challenge novice users. It may therefore be necessary to provide the UDOP with the means to differentiate between different classes of users and vary both the power and complexity of the features offered and the degree of guidance provided. For example, apprentice users may be provided with limited ability to change a UDOP configuration with extensive guidance, whereas experienced users may be provided with a wider range of options for each of the steps in the pipeline and a more nuanced and complicated guidance. There might also be a top class of expert UDOP users who can undertake activities such as creating new data sources, processing and visualization types, in addition to the guidance required to use them.

Conclusions

The evolution of military requirements to support situational awareness (SA), shared situational awareness, and sensemaking within ever more diverse and complex situations suggests that existing COP concepts and implementations are no longer adequate or sufficient to meet the new needs. It is therefore imperative that alternative concepts, such as a UDOP, are explored with some urgency. The arguments put forward in this chapter cannot confirm that the concept of a UDOP as described is necessarily the answer. However, what can justifiably be argued is that a concept with many of the characteristics of a UDOP is most definitely required to replace what is currently in existence and is not fit for purpose.

A set of characteristics for a UDOP can be defined, and such characteristics can provide a means to identify to what degree any picturing solution might address current and future needs.

There are quite a number of issues still to resolve; for example, what an appropriate architecture and set of components should be to deliver an effective and comprehensive UDOP, especially within the context of infrastructure concepts, such as SOA. However, developing the technology in itself is insufficient, and thus UDOP concepts of use need to be developed alongside appropriate information sharing and assurance policies.

There are a number of UDOP implementations already available, some calling themselves a UDOP, and some not. However, it does appear that none of those currently in existence sufficiently meet the extent of the needs. This should not be surprising given the number of challenges detailed in this chapter.

Finally, one of the most significant omissions at the present time is that of an adequate degree of assessment and experimentation to determine what the real issues and benefits of various picturing capabilities are and might be. Therefore, it is difficult to state, with sufficient supporting evidence, what picturing capabilities and means of providing those capabilities (including UDOP) are better than alternatives. Those assessments, which have been conducted have tended to focus on specific issues, such as SA, in the context of an individual system of interest rather than assessing current picturing capabilities from a holistic point of view. Although this is an obvious approach to take, in order to constrain the scope of assessment and experimentation, it nevertheless also limits the value and applicability of the findings. The end result is that we do not have sufficient context to be able to adequately explore and understand what works well, and what does not, and determine the reasons why. A primary challenge is knowing what to measure and how, especially when an information fusion UDOP is being used in a distributed environment with multiple users.

owledgments

The UDOP concepts and ideas discussed in this chapter are a direct result of involvement of, and intellectual contributions from, many of the members of TTCP C3I Group Technical Panel 2. However, particular thanks are conveyed to Jason Moore from AFRL for his insights into a UDOP that were gained from both his work with ACES Viewer and the research ideas he has been exploring, such as accountable visualization.

rences

[1] "United States Intelligence Community Information Sharing Startegy," February 22, 2008, accessed at http://www.dni.gov/reports/IC_Information_Sharing_Strategy.pdf.

[2] "Single Information Environment (SIE), Architectural Intent 2010, Commonwealth of Australia, 2010, accessed at http://www.defence.gov.au/cio/_lib/doc/Single_Information_Envi-ronment.pdf.

[3] Rittel, H. W. J. and Webber, M. M. "Dilemmas in a General Theory of Planning," *Policy Sciences,* 4 (155-169), 1973.

[4] "The Comprehensive Approach," *Ministry of Defence (UK), Director General Joint Concepts and Doctrine, Joint Discussion Note 4/05*, January 2006, http://www.mod.uk/NR/rdonlyres/BEE7F0A4-C1DA-45F8-9FDC-7FBD25750EE3/0/dcdc21_jdn4_05.pdf

[5] Yang, C, and Blasch, E., "Fusion of Tracks with Road Constraints," *J. of. Advances in Information Fusion*, Vol. 3, No. 1, 14-32, June 2008.

[6] http://www.opengroup.org/soa/source-book/soa_refarch/intro.htm

[7] Chen, G., E. Blasch, D. Shen, H. Chen, and K. Pham, "Services Oriented Architecture (SOA) based Persistent ISR Simulation System," *Proc. of SPIE*, Vol. 7694, April 2010.

[8] Boukhelifa, N., Roberts, J., and Rodgers, P., "A Coordination Model for Exploratory Multi-View Visualization," *International Conference on Coordinated and Multiple Views in Exploratory Visualization*, 2003.

[9] Blasch, E..P., P. Valin, and E. Bossé, "Measures of Effectiveness for High-Level Fusion," *Int. Conf. on Info Fusion*, 2010.

10] Karwowski, W., M. Haas, and G. Salvendy, "A Review and Reappraisal of Adaptive Human-Computer Interfaces in Complex Control Systems," AFRL-HE-WP-TR-2006-0123, Washington, D. C., US Government Printing Offce, 2006.

11] Court, G., "Validating The Nec Benefts Chain," Int. Command and Control Research and Tech. Symp. (ICCRTS), 2006.

12] UK Ministry of Defence, *The Acquisition Handbook*, Edition 6, October 2005.

13] Alberts, D., and Hayes, R., "Power to the Edge: Command and Control in the Information Age," Washington, D.C., Command and Control Research Programme, 2003

14] Roam, D., *The Back of the Napkin: Solving Problems and Selling Ideas with Pictures*, Penguin Group, March 2008.

15] Nakamura, J. and Csikszentmihayli, M. , "The Concept of Flow," Ch. 7 in *The Handbook of Positive Psychology*, Oxford University Press, 2002.

16] Bowman, E., Pattison, T., and Gouin, D., "Intense Collaboration: Human and Technical Requirements for Agile C2," *Int. Command and Control Research and Tech. Symp. (IC-CRTS)*, 2009.

17] UK Ministry of Defence, "MOD Architectural Framework: Concepts and Doctrine Community of Interest Deskbook," *MODAF-M10-013*, July 2005.

18] *US Headquarters Department of the Army, Field Manual* No 3-0, June 2001.

19] Mulgund, S., and Landsman, S., "User Defned Operating Pictures for Tailored Situation Awareness," *Int. Command and Control Research and Tech. Symp. (ICCRTS)*, 2007.

20] "Interactive Visualization of National Airspace Data in 4D (IV4D)," *Aerospace Computing Inc., Final Technical Report*, AFRL-RI-RS-TR-2010-156, August 2010.

21] Bouchard, A. and Vernik, R., "Characterization and Showcasing of Network Visualization Approaches for Command and Control," *Visualizing Network Information*, Meeting Proceedings RTO-MP-IST-063, 2006.

22] Bouchard, A, Lapinski, L, Lavoie, J. and Roy, J., “Evaluation of Information Visualization Approaches for an Enhanced Recognized Maritime Picture,” *Proc. SPIE*, Vol. 6945 , 2008.

23] Zheng, J., Wang, P., Patton, E., Lebo, T., Luciano, J., and McGuiness, D., “A Semantically-Enabled Provenance-Aware Water Quality Portal,” *Proc. of EIM*, 2011.

24] Clark, A. J., Holliday, P., Clark, J., Mears, J., Chau, R., Nielsen, L., Chau, M. Eisenberg, H., and Eckersley, T.,”UDOP: A Collaborative System for Geospatial Data,” Ch14 in *Spatially Enabling Society, Research, Emerging Trends and Critical Assessment*, R. Abbas et. al., (Eds), Leuven University Press, 2010. Available on the Internet: http://t-sciences.com/pdf/white_paper/UDOP-A_Collaborative_System_for_Geospatial_Data.pdf.

25] Augeri, C., and Sousan, B., “The MightyMoRiver Project–Tracking Dynamic Events,” Proc. of the Knowledge Discovery/Modeling & Simulation (KDMS) Workshop, Knowledge Discovery and Data Mining (KDD) Conf., 2011.

26] Ahearn, S. C., Campbell, J. S., Skupin, A. and Villagran, J-C., “UDOP: Final Review and Recommendations,” 2010. Available on the Internet: http://t-sciences.com/pdf/white_pa-per/UDOP-Final_Review_and_Recommendations.pdf.

---

1. In our treatment of the UDOP concept, we consciously define the acronym as user-defined *operating* picture as opposed to user-defined *operational* picture to reflect the need to support a diverse set of users not limited to one particular level of command or only military employment.

2. It is recognized that these challenges apply almost universally across the Command and Control (C2) enterprise; but they nonetheless generate specific concerns for picturing capabilities, which is the focus of this chapter.

3. Includes training, equipment, personnel, information, doctrine, organization, infrastructure and logistics concerns.

4. Note that we don’t necessarily believe or agree that “UDOP” is an optimal term for representing the appropriate concept, but it does appear to be the one which is most commonly used.

5. That is the ability to undertake interactive explorations where one can change the value of a parameter in one view and all linked views also change in response to the same parameter change.

6. *US FM 3-0*, June 2001, “The operational level of war is the level at which campaigns and major operations are conducted and sustained to accomplish strategic objectives within theaters or areas of operations (AOs).”

7. IMAGO is not an acronym, but instead is a term from biology that was used to reflect the transformative nature of the scientific work performed. In entomology, imago is the final and fully developed adult stage of an insect after it has completed metamorphosis.

CHAPTER 10

User Information Fusion Decision Making Analysis with the C-OODA Model

Erik P. Blasch (United States), Richard Breton (Canada), and Éloi Bossé (Canada)

For pragmatic information fusion system design and analysis, the user, commander, or operator/analyst needs information in a timely manner to conduct actionable intelligence. With the development of complex information fusion systems, the user still provides valuable inputs to the information fusion system in contextual reasoning and situation understanding. In this chapter, we describe the cognitive observe-orient-decide-act (C-OODA) model as a method of user and team analysis in the context of the Data Fusion Information Group (DFIG) information fusion model. From the DFIG model—as an update to the Joint Directors of the Lab (JDL) model—we look at Level 5 Fusion of user refinement in the context of timely decision making. Using control theory, we present an example of a user timeliness assessment in an information fusion decision making model analysis. We model the information input delays in reaching a decision and the action output delays in executing the decision. The C-OODA comparisons to the DFIG model support systems evaluation and analysis as well as coordinating the time interval of interaction between the machine processing (e.g., information fusion) and user processing (e.g., perception and reasoning).

Introduction

Models (e.g., control models) can represent a system to determine what is happening, the parameters of interest, and methods for prediction. Models that incorporate man and machine systems are useful for determining who should interact with the system, what interfaces should be designed, where in the process should the user interact, which control actions to perform, and how to better the system [1]. System design analysis, while general for all systems, is important to the information fusion (IF) community [2]. Information fusion systems (IFSs) seek to reduce the enormous amount of data into actionable intelligence for user's to act upon [3]. Numerous literature contributions of process modeling of IFSs have been conducted to clarify user importance (i.e., role) in IFSs design.

One of the premier models for user decision-making is the extended OODA model (Boyd), as shown in Figure 10.1.

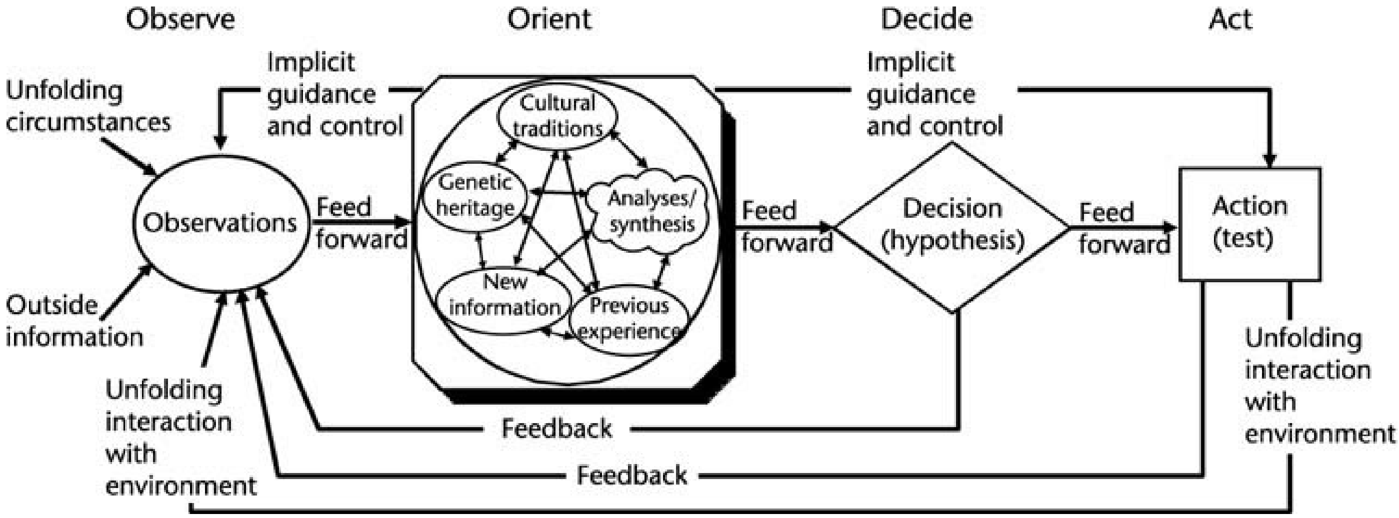


Figure 10.1 The extended OODA loop. (From: [4])

The observe-orient-decide-act (OODA) loop model has been widely used to represent decision-making (DM) in military environments, as shown in Chapter 2. However, the classical or extended versions of the OODA loop suffered from a lack of details to sufficiently support the design of systems. For instance, the model depicted in Figure 10.1 proposes a more detailed version of the orient process to the detriment of the others. Key to the developments and instantiations of the OODA models include application relevant decision-making based on context, time of analysis, and uncertainty analysis.

In most military documents, the OODA loop is often referred to as a simple representation of control processes. Because of the simplicity of its representation, the loop offers few details to describe how a human makes a decision in complex and dynamic command and control (C2) environments. This lack of details led to the development of several OODA model versions for different applications. The developments include the modular M-OODA [5], team T-OODA [6–8], and cognitive C-OODA [9, 10]. For each OODA loop model proposed, there are strong parallels between the OODA model structure and the information fusion models. A comparison of the OODA model to the other information fusion models was analyzed in relation to the Omnibus model [11].

The DFIG model [12] was a proposed update to the JDL model as per meeting in 2004 [2, 13]. In terms of the DFIG information fusion model, *observe* is Level 1 fusion of object assessment, and *orient* is Level 2 fusion of situation assessment. Both observe and orient have also been referred to as situation awareness. *Decide* is Level 5 fusion of user refinement and *act* is Level 4 fusion of process refinement. Act and decide are considered decision-making functions.

In military environments, executing the OODA loop faster than the adversary is central to gain advantage in the confrontation [4]. Consequently, the timing between the user and the information system is critical. The objective of this chapter is to use the cognitive version of the OODA loop, C-OODA, to develop a control process model to coordinate the timeliness of DM between the user and the information fusion evidential reasoning machine. There are two advantages of the C-OODA. It provides high level details on the cognitive processes involved in complex and dynamic DM performed in C2 environments and it includes in each of its modules a criteria-based (e.g., time and uncertainty) control process that is central to our objective of simulating the timeliness of DM between the user and the information fusion system.

We begin in Section 10.2 by exploring user-machine decision-making IF models with a summary in Table 10.1. Section 10.3 overviews the developments of the COODA model. Section 10.4 provides a notional simulation to characterize the user response times in a cognitive IF decision-making task and Section 10.5 concludes the analysis.

Table 10.1 Comparisons of Decision Making Model

Activity	*DFIG Model*	*Omnibus Model*	*OODA*	*C-OODA*
Command execution	Level 6	Resource tasking	Act	Action implementation
Decision making	Level 5	User control	Decide	Recall
Sensor management	Level 4	Decision making		Evaluate
Impact assessment	Level 3	Context processing	Orient	Projection
Situation assessment	Level 2	Pattern processing		Comprehension
Object assessment	Level 1	Feature processing		Feature matching
Signal/info processing	Level 0	Signal processing	Observe	Perception
Data acquisition/ registration		Sensing		Data gathering

Decision Making Models

10.2.1 DFIG and OODA Loop

Clearly, a mapping between the DFIG model (presented in Chapter 2) and the OODA loop cyclic process can be made as shown in Figure 10.2. The traditional information fusion data processing functions (i.e., estimation) include, observe and orient, which compose situation awareness. Numerous efforts have sought to model the evidence accumulation in providing a situational analysis including perception of situations [14], presentation of object assessments [15], descriptions of situations [16], architectures of situational awareness [17], and user assessment of situations [18]. In duality, the information fusion action functions (i.e., control) include decide and act as the decision making processes. Examples of information fusion action analysis includes decision making [19], user refinement [20], and sensor management [21].

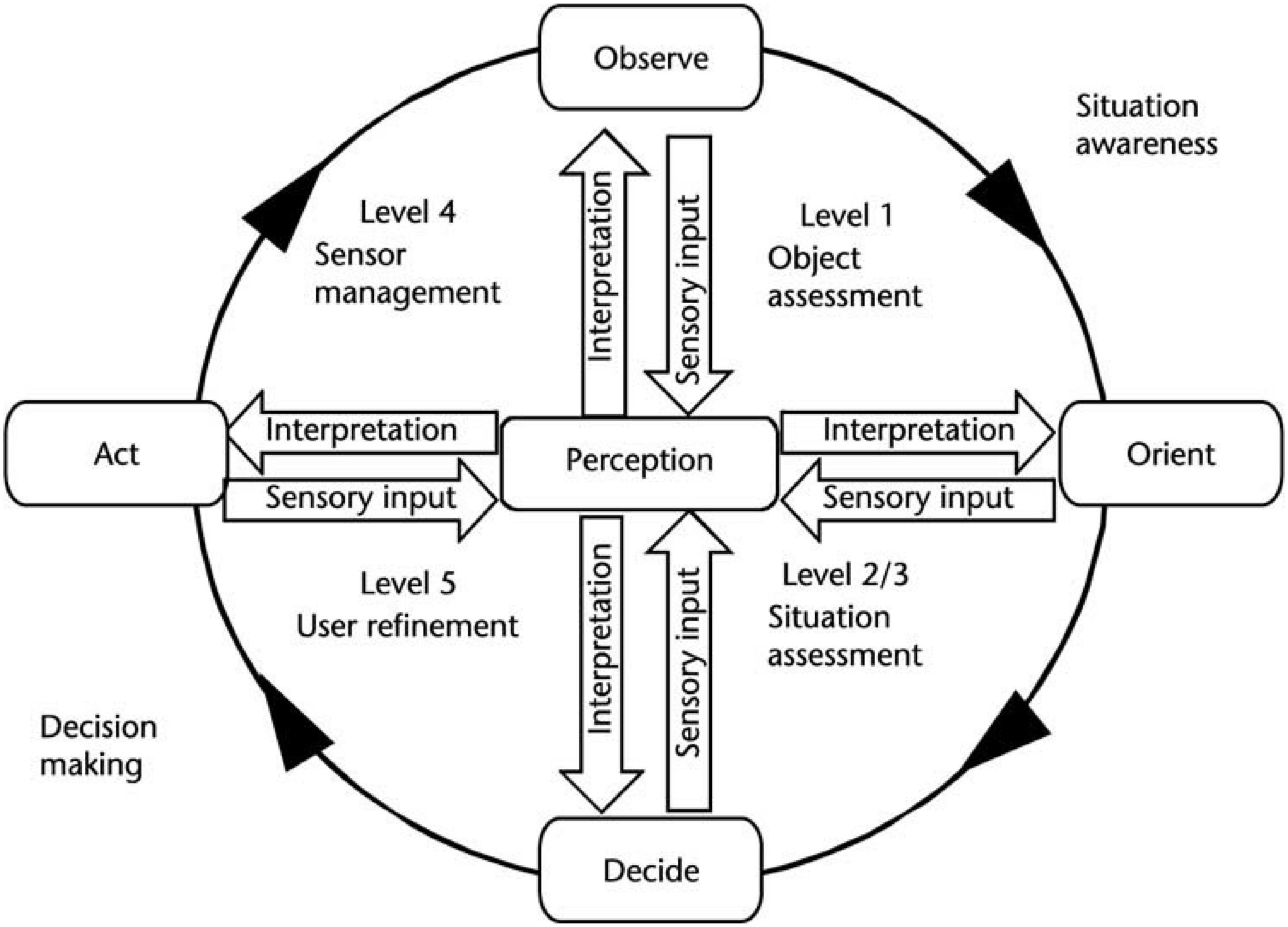


Figure 10.2 The OODA in relation to the DFIG model.

The OODA phases are:

- *Observe*: A user/organization interacts with the environment, typically by controlling sensors, querying information needs, and assimilating observations from a display;
- *Orient*: A user/organization distills information from data to determine situational understanding through assessment of the environment to determine a coherent state of affairs;
- *Decide*: User engages situational knowledge derived from orientation to prioritize and select plans/results;
- *Act*: The user/organization engages in a process plan that satisfies current needs.

The OODA and DFIG models have similar properties in trying to capture the decision process. When the user must reason over an enormous amount of data for a contextual situation, cognitive analysis is required for mission success. Cognitive models include the developments from physiology [22], decision support [23, 24], and automation [25], to high-level information fusion [26]. Table 10.1 relates the relevant information fusion decision making models.

10.2.2 Multiplayer OODA

Figure 10.3 shows the time windows associated with multiple DMs of a user and adversary from which differing response times are associated with action. As an example of differing action cycles, Figure 10.3 shows the case for the need for rapid decision making. On the left, if the user (denoted as “us”), reacts to immediate threats, the adversary the ability has (denoted as “them”) already been able to attack. If the proactive strategy is used and sensed information details anticipated events, the user could infer potential threats. Finally, on the far right, if the OODA loop (sensing and processing combined with behavior analysis) provides the user with a priori information. In this scenario, it would allow the user to act very quickly to prevent actions from occurring. To model the OODA cycle, a queuing analysis was used to determine the prevention and proactive protection and results are in [27].

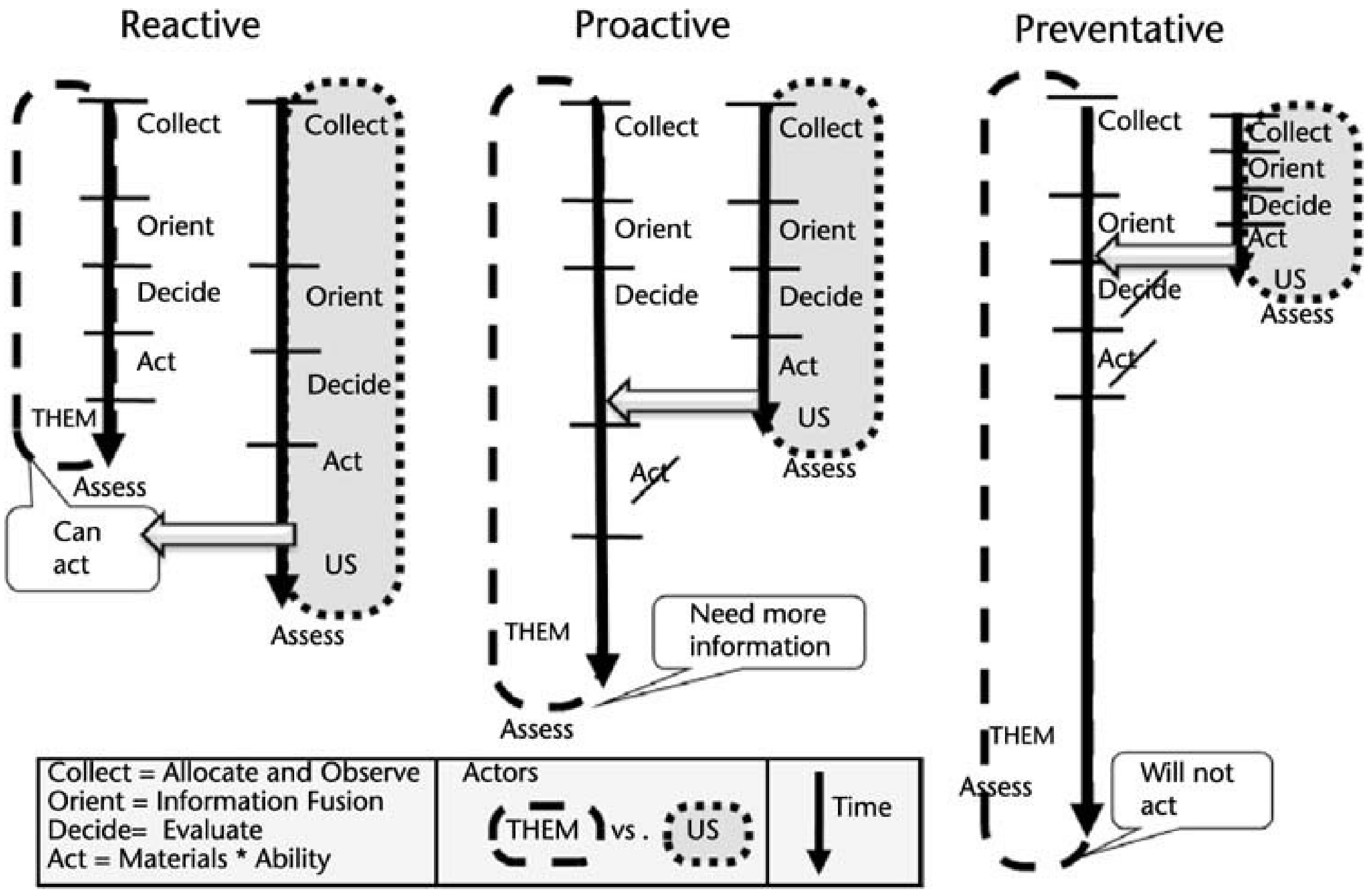


Figure 10.3 Progression from reactive to preventative event chains with OODA process loops [7].

The Cognitive OODA Loop

Repeated attempts at modeling C2 reflect the importance of such modeling for understanding C2, which contribute to the design of support tools and training efforts aimed at an improvement of C2.

Describing the C2 cycle usually involves descriptive models which divide the task into processes [28], such as the stimulus, hypothesis, option, response (SHOR) model. Mayk and Rubin [29] provide a systematic analysis of C2 descriptive models. Comparatively, prescriptive models are theory-driven and are used to develop software for decision support.

To develop the C-OODA, some classical and well-accepted models have been used to increase the cognitive granularity of the OODA loop. For instance, Breton [6] used the situation awareness (SA) model proposed by Endsley [30] to model the observe and orient phase and the recognition-primed decision (RPD) making model of Klein [31] for the decide phases.

10.3.1 Situation Assessment Models

Level 2, Situation assessment, is the estimation and prediction of relations among entities, to include force structure and force relations and communications which require adequate user inputs to define entities. The human in the loop (HIL) of a semi-automated system must support perception such as Endsley's model [30] of perception, comprehension, and projection (as shown in Chapter 2). To understand how the human uses the situation context to refine the SA, Breton [5] uses the RPD model [31]. The RPD model develops the user decision making capability based on the current situation and past experiences. The RPD model shows the goals of the user and the cues that are important. The RPD model allows us to capture the reduction in reaction time and increase in accuracy for the cases in which the user cues the IFS and when the IFS cues the human. According to Breton [5], some parallels between the processes included in the RPD and SA models can be made as shown in Figure 10.4.

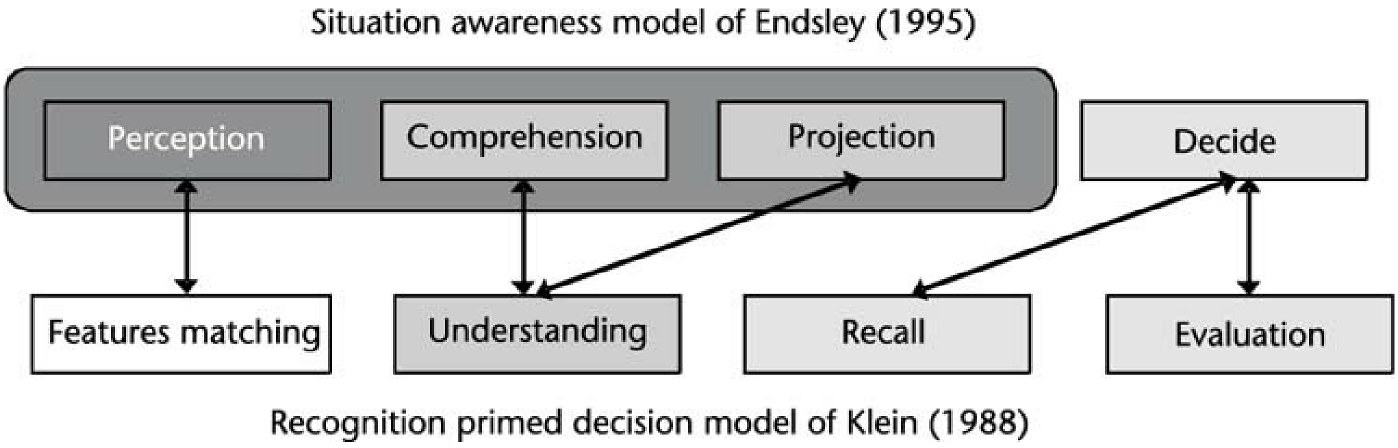


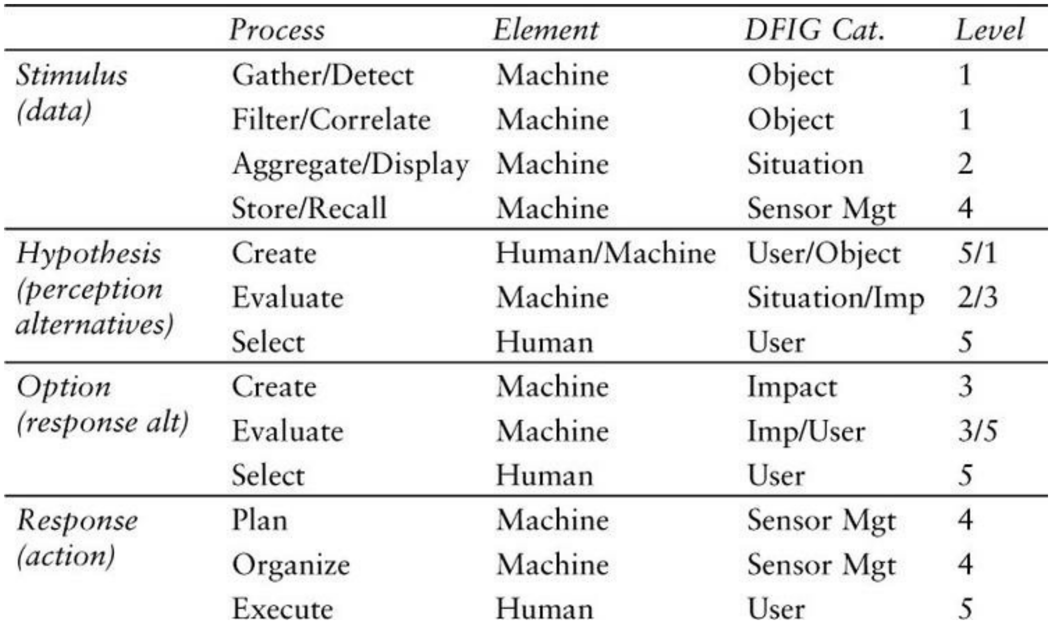
Figure 10.4 Comparison of SA and RPD models. [6].

The user must present the priority of information needs to the IFS where the information priority is related to the information desired. The user must have the ability to choose or select the objects of interest and the processes from which the raw data is converted to the fused data. One of the issues in the processing of fused information is related to ability to understand the information origin or pedigree. To utilize the information priority list, Blasch [32] used the SHOR model to detail IF functions.

10.3.2 SHOR Model for Action

To determine how to model the user cognitive decision/action cycle, we can use information from the judgment and decision making community. [33] Simon [34] looked at the analysis as a function of intelligence, design, and choice. Huber [35] listed the process as a five-step function, including (1) problem identification, (2) problem definition, (3) problem diagnosis, (4) generate alternatives, and (5) evaluate and selection. Finally, Wohl [36] developed the SHOR model (Table 10.2) for military tactical decision making. In the C-OODA model, we are interested in how the user can constrain (i.e., problem diagnosis) the data as well as be cued by the IFS for decision and action. If we adapt the SHOR model for fusion systems, we have a representation for the hypothesis space for C-OODA comprehension and evaluation.

Table 10.2 The SHOR Analysis of the User-fusion System



	Process	Element	DFIG Cat.	Level
Stimulus (data)	Gather/Detect	Machine	Object	1
	Filter/Correlate	Machine	Object	1
	Aggregate/Display	Machine	Situation	2
	Store/Recall	Machine	Sensor Mgt	4
Hypothesis (perception alternatives)	Create	Human/Machine	User/Object	5/1
	Evaluate	Machine	Situation/Imp	2/3
	Select	Human	User	5
Option (response alt)	Create	Machine	Impact	3
	Evaluate	Machine	Imp/User	3/5
	Select	Human	User	5
Response (action)	Plan	Machine	Sensor Mgt	4
	Organize	Machine	Sensor Mgt	4
	Execute	Human	User	5

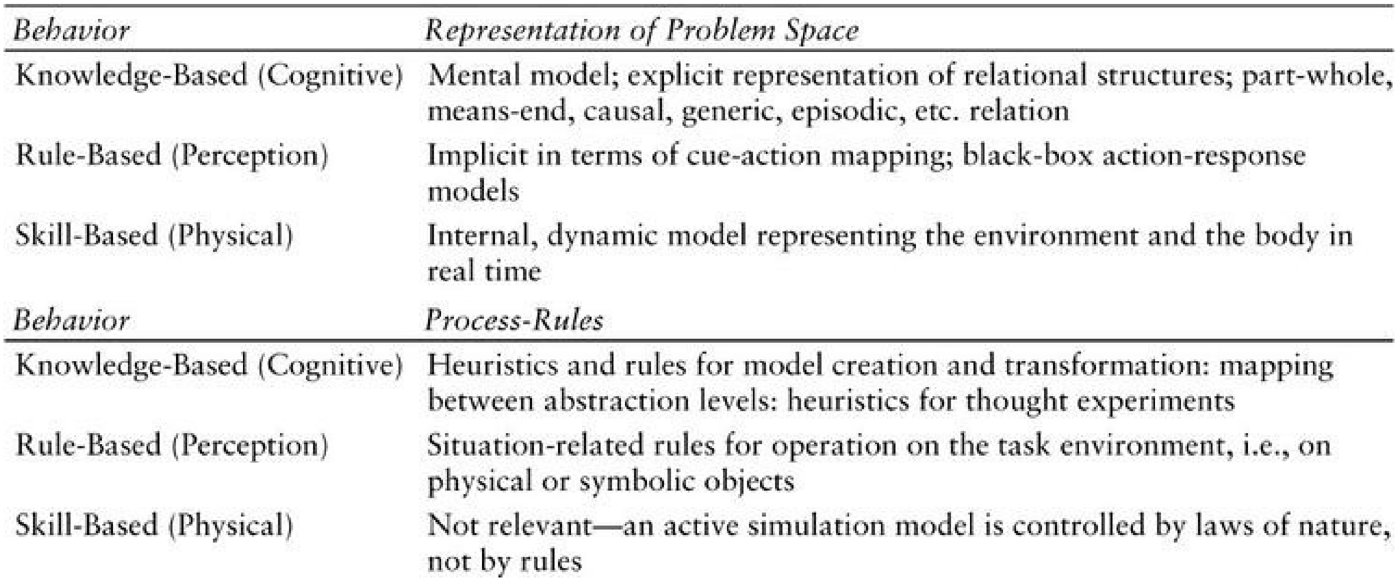
Far right column is the levels in the DFIG model.

We see that the user plays a large part in the process of determining information fusion system (IFS) actions. The SHOR model can be developed further to include the processing levels of the IFS so as to determine the levels of interaction between the user and the IFS. From Table 10.2, we see that many interactions between the IFS occur between Levels 2/3, and 5. Such an example is determining relevant cues from the environment for decision making as per the C-OODA.

10.3.3 The Skills-Rules-Knowledge Model

In Rasmussen's model [37], user goals are determined from the decision desired. To achieve the correct goal, planning of actions and situation identification is performed at the knowledge level. Once a situation or task is learned, rules can be instantiated as to the recognition of features to be associated from one situation to the next. Such a case is when a human is proactive to receive data inputs for pre-established rules of behavior. Once the rules are in place, the user can utilize automatic actions to data inputs to allow for faster response time performance. The depiction of the Rasmussen's levels is shown in Table 10.3.

Table 10.3 Behavior Representation and Process-Rules



Behavior	*Representation of Problem Space*
Knowledge-Based (Cognitive)	Mental model; explicit representation of relational structures; part-whole, means-end, causal, generic, episodic, etc. relation
Rule-Based (Perception)	Implicit in terms of cue-action mapping; black-box action-response models
Skill-Based (Physical)	Internal, dynamic model representing the environment and the body in real time
Behavior	*Process-Rules*
Knowledge-Based (Cognitive)	Heuristics and rules for model creation and transformation: mapping between abstraction levels: heuristics for thought experiments
Rule-Based (Perception)	Situation-related rules for operation on the task environment, i.e., on physical or symbolic objects
Skill-Based (Physical)	Not relevant—an active simulation model is controlled by laws of nature, not by rules

To analyze the C-OODA domain-relevant actions, we could employ a cognitive work analysis (CWA) [38] to determine the amount of actions needed over a specified time period. Suchman [39] developed a concept for situated action, which can be used for known situations, however when the situation is unknown, the user needs to integrate information for situation understanding (SU). The Modular OODA (M-OODA) provides a decomposition of control processes for SU to action.

10.3.4 The Modular OODA (M-OODA)

The M-OODA incorporates explicit control and flow modular components with the current understanding of military C2 as shown in Figure 10.5 [5]. A module operates as a simple control system with inputs, outputs, and processing times that are not sequential but rather iterative between modules. The control flow follows from a state diagram at each of the modules and can be analyzed as a whole or separately. The M-OODA model modifies the OODA loop based on the following principles:

1. It adopts a modular approach in which each process of the OODA loop is represented as a generic module structured around three components: process, state and, control;
2. It incorporates explicit control elements within and across modules enabling a bidirectional data/information flow and feedback between modules;
3. It provides a basic architecture for modeling a variety of team decisionmaking in with the OODA loop.

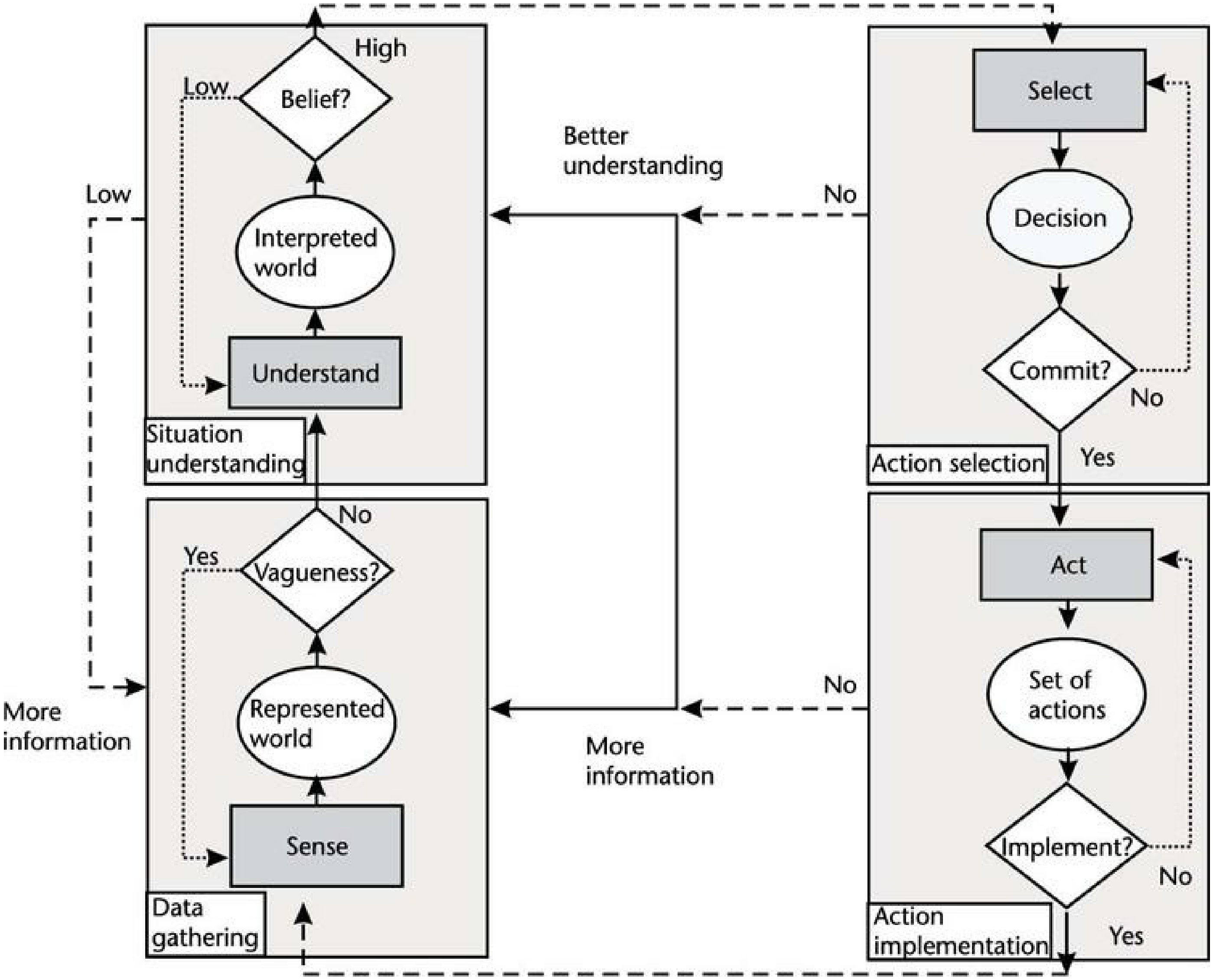
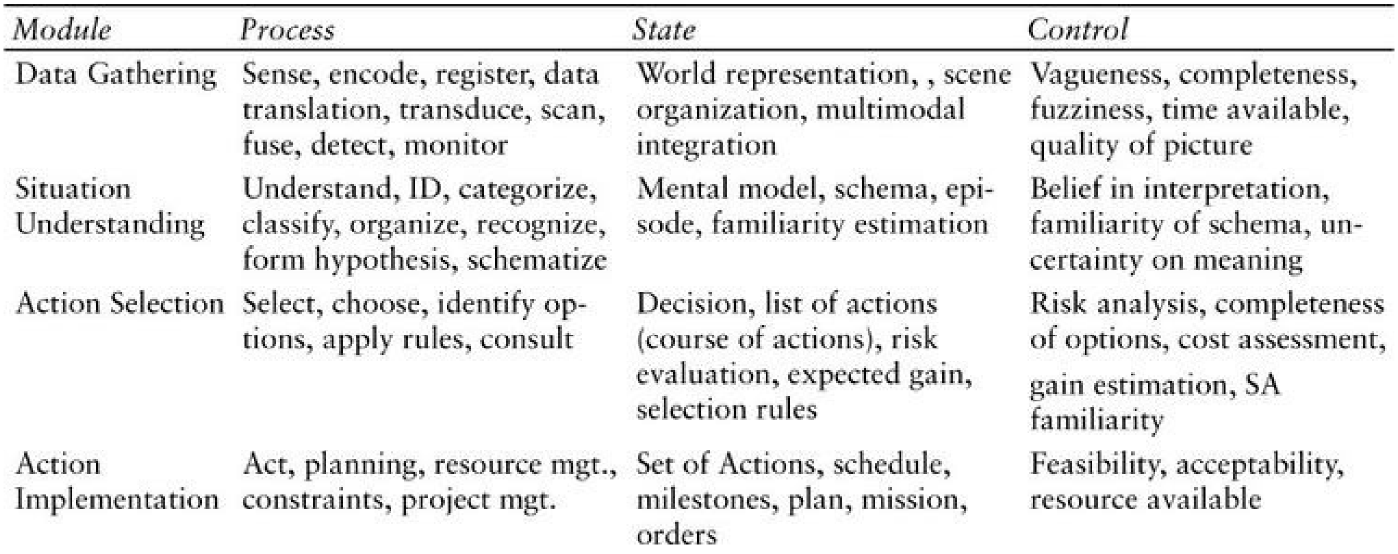


Figure 10.5 M-OODA Control modeling.[5].

To utilize the OODA concept in a C2 system, elements of the modules include: data gathering (observe), situational understanding (orient), action selection (decide), and action implementation (act), shown in Table 10.4.

Table 10.4 Specifications M-OODA Components [5]



Module	*Process*	*State*	*Control*
Data Gathering	Sense, encode, register, data translation, transduce, scan, fuse, detect, monitor	World representation, , scene organization, multimodal integration	Vagueness, completeness, fuzziness, time available, quality of picture
Situation Understanding	Understand, ID, categorize, classify, organize, recognize, form hypothesis, schematize	Mental model, schema, episode, familiarity estimation	Belief in interpretation, familiarity of schema, uncertainty on meaning
Action Selection	Select, choose, identify options, apply rules, consult	Decision, list of actions (course of actions), risk evaluation, expected gain, selection rules	Risk analysis, completeness of options, cost assessment, gain estimation, SA familiarity
Action Implementation	Act, planning, resource mgt., constraints, project mgt.	Set of Actions, schedule, milestones, plan, mission, orders	Feasibility, acceptability, resource available

10.3.5 The Cognitive Process Included in the C-OODA

From the M-OODA architecture, the C-OODA has been developed by Breton [5]. The objective of the C-OODA is to increase the level of granularity of the OODA loop by formulating a detailed cognitively valid representation of the C2 decision cycle. The C-OODA [10] also provides more details on the control component of the M-OODA. According to Breton, the control is based on the time available in the situation and the level of uncertainty in the situation. For instance, if the level of uncertainty is high, but there is no time left for further processing, the cognitive processing of the C-OODA is stopped. Figure 10.6 presents the different types of states resulting from the different processes included in the C-OODA.

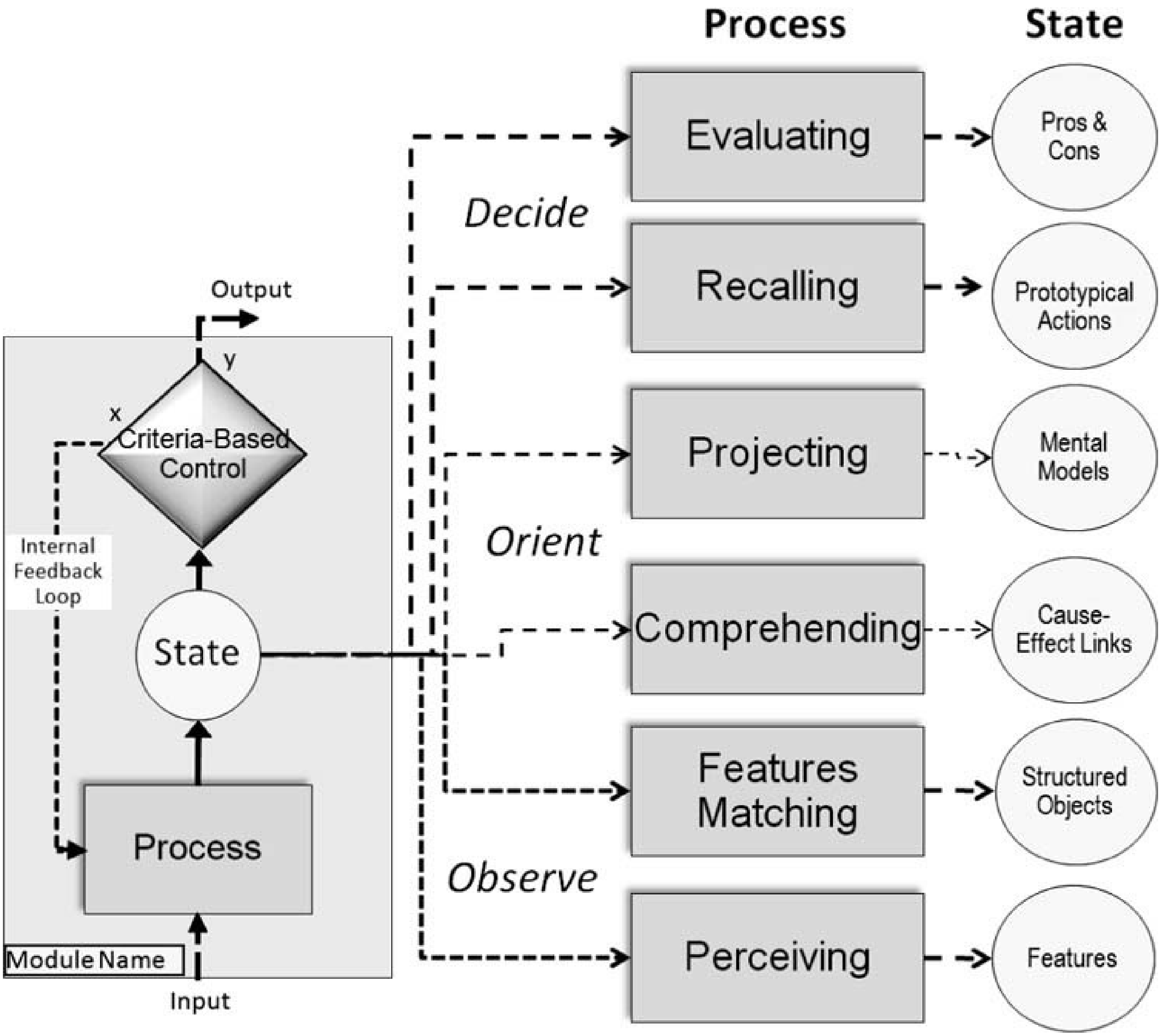


Figure 10.6 C-OODA Control states [11].

The C-OODA model divides up the cognitive decision-making cycle. Other approaches to modeling user cognitive decision making include categories of neglect, consult, rely, and interact [40]. In each case, the user evaluates the output. The five basic user refinement functions of—planning, organizing, coordinating, directing, and controlling—relate to the cognitive action by varying the time required for each C-OODA process activity.

Simulation

In this simulation, we assess the C-OODA for decision making based on the time to make a response. In decision making analysis, there are three types of responses (knowledge, skills, rules), which are cognitive reasoning, rule-based perceptual actions, and physical executions that alter the estimated timeliness of action.

Control theory is a popular method to analyze mechanical, biological, and psychological systems [41]. There are many algebraic models that empirically collect data from human-factor studies and perform a regression analysis to parameterize the effects. In this case, we are concerned about the timeliness to afford a user with a decision threshold from which control models are a candidate model. Control models assess state estimation accuracy or stability (time to converge). In the COODA model, the human is engaged in an iterative process with the objective of reducing information uncertainty. Such a process is stopped when the time required to iterate is higher than the time available or when the information certainty reaches a specific threshold. Using control theory, we develop a model for time responses for an input delay, action, and output delay as shown in Table 10.5.

Table 10.5 Specifications of Time Delays in the C-OODA

C-OODA	*Input Delay*	*Action*	*Output Delay*
Perception/ Feature Match	Get data	Exponential	Data Processing
Comprehension/ Projection	Organizing data	Ramp	Multiple processes
Recall/ Evaluate	Selection of option	Step	Query Selection
Act	Physical Action	Impulse (Immediate)	Request sent

We model the user DM process as a series of linear time invariant (LTI) control operations with feedback, represented in state space as:

$$\begin{aligned} \dot{x}(t) &= \mathrm{A}x(t) + \mathrm{B}Ku(t) \\ y(t) &= \mathrm{C}x(t) + \mathrm{D}Ku(t) \end{aligned} \tag{10.1}$$

where **A** is the state matrix, **B** is the input matrix, **C** is the output matrix, **D** is the feedforward matrix, and *K* is a constant as shown in Figure 10.7.

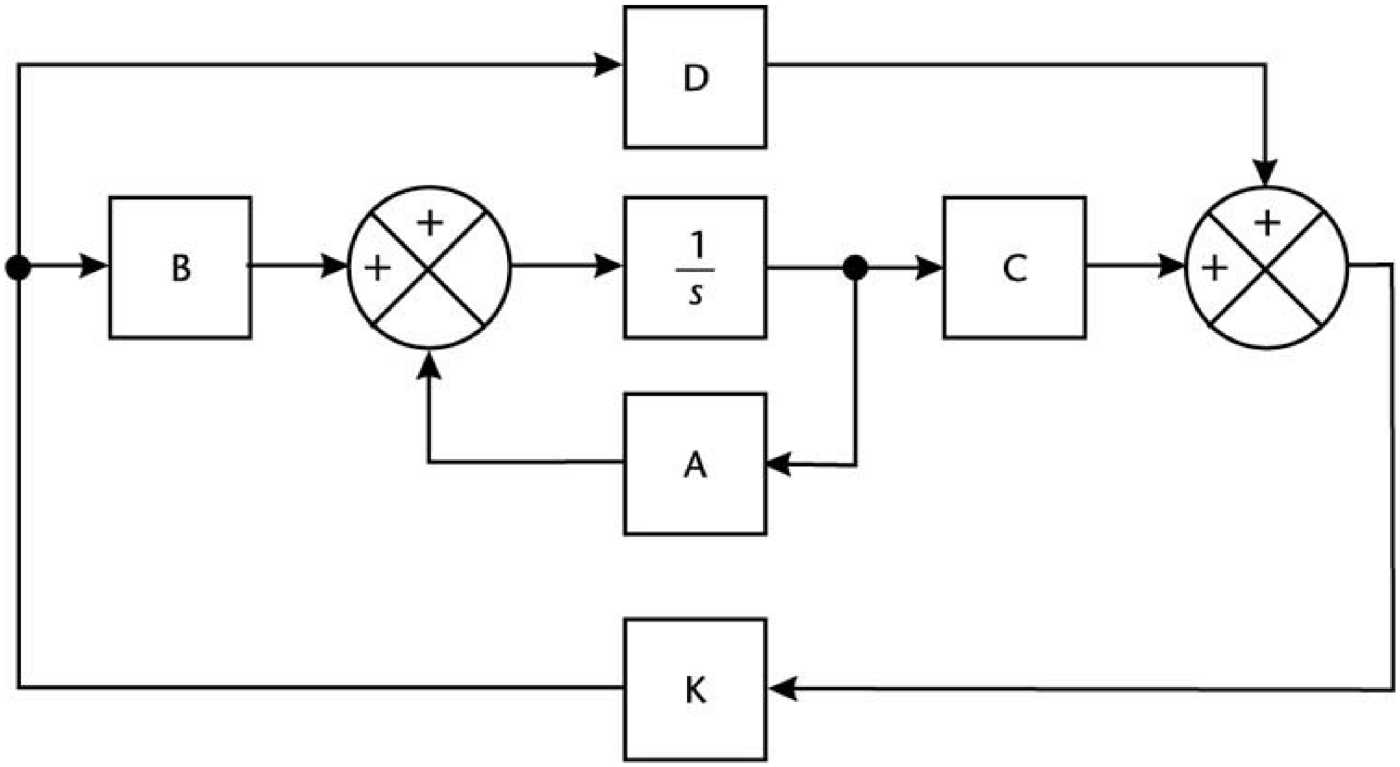


Figure 10.7 State space C-OODA control module.

To model a user function, we utilize a first order system with an exponential response as presented in Figure 10.8.

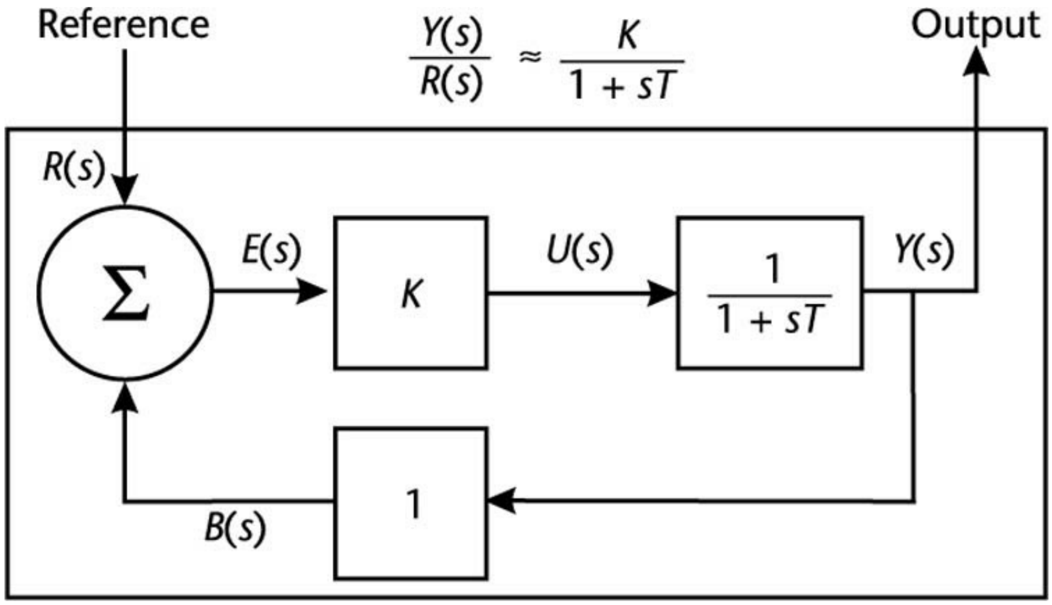


Figure 10.8 First order control model with time delay.

Using the Laplace notation, *s*, the transfer function has a typical exponential time response:

$$b(s) = \frac{e^{-TD_i s}}{s+1} \tag{10.2}$$

We simulate the deadtime for an *input time delay* (TD_i) for a decision *i*, as related to the user achieving a control decision. Likewise, in the action selection requires time as modeled as an *output time delay* (TO_i). The updated state-space representation is:

$$\dot{x}(t) = -\mathbf{A}x(t) + \mathbf{B}u(t - TD_i) \tag{10.3}$$

$$y(t) = \mathbf{C}x(t - TO_i) + Du(t) \tag{10.4}$$

To determine the estimation parameters of **A** and **B**, as well as the output analysis of **C** and **D**, we model the importance of the information processing as related to the functions in the C-OODA. We note that the transfer function response delays can vary over users and domains which might be difficult to get exact numbers, however, as per human-factor studies; we could get notional times to determine the bottlenecks. For example, Level 1 fusion, orient, or comprehension/projecting requires the most time in analysis (input delay), has the largest impact (amplitude) in the decision making, and takes the most time to provide a set of prioritized actions (output delay). The final step of action selection requires the least amount of delay and amplitude as most other options have been removed to produce a single parameter control loop. In the simulations, we used the MATLAB functions:

```
sys1 = ss (A, B, C, D, 'InputDelay', TDi)

sys2 = ss (A, B, C, D, 'InputDelay', TDo)
```

To detail C-OODA modeling, we describe the system time response over the interval that a decision could be made. It is similar to a probability distribution model for the timeliness of action as represented by the exponential decay. We vary the input and output (I/O) time delays for each component separately, as shown in Figure 10.9, to model the contributions to the overall time response. Figures 10.9–10.11 are notional results and serve as a modeling example for real world collections.

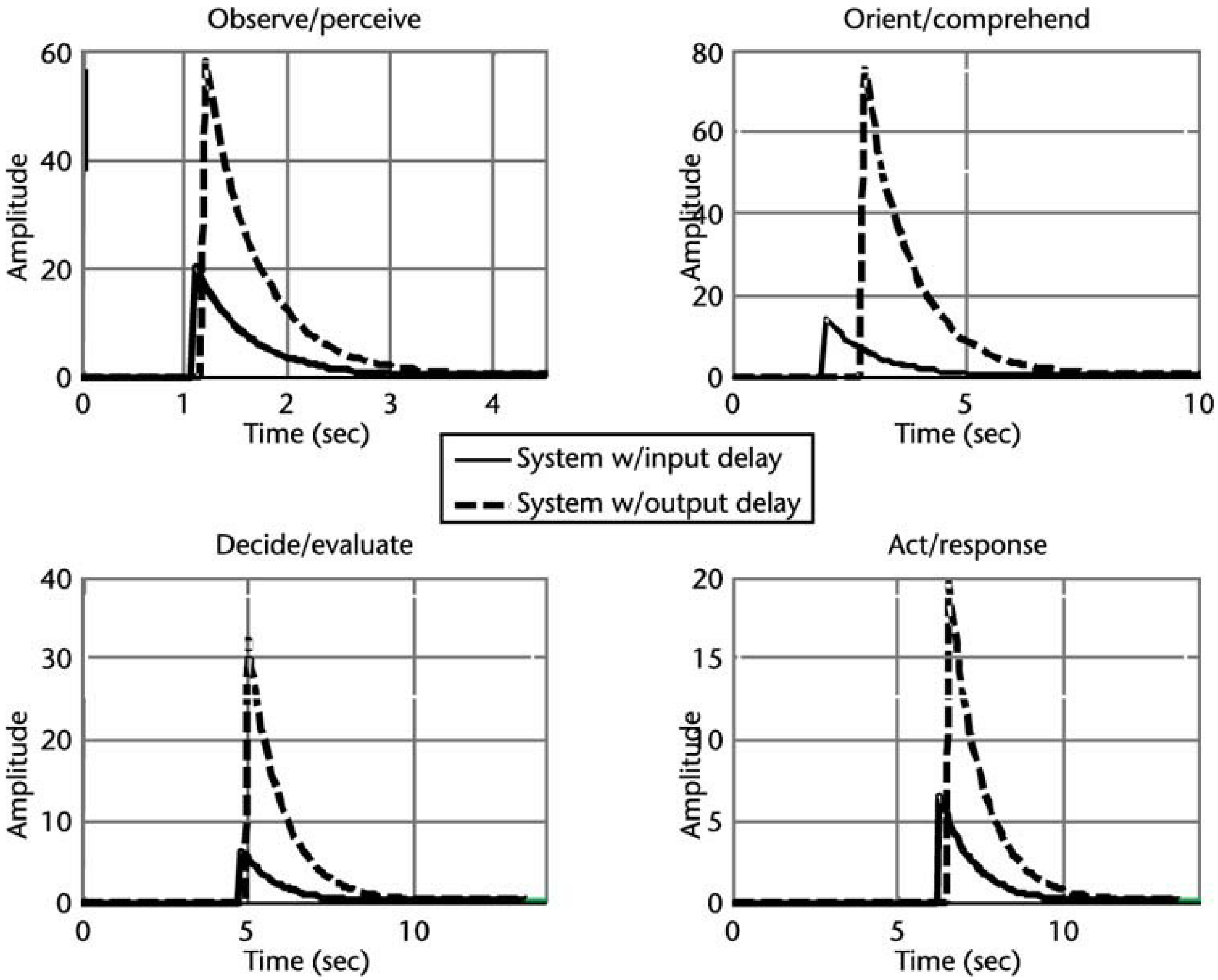


Figure 10.9 Input/output delay models for the C-OODA.

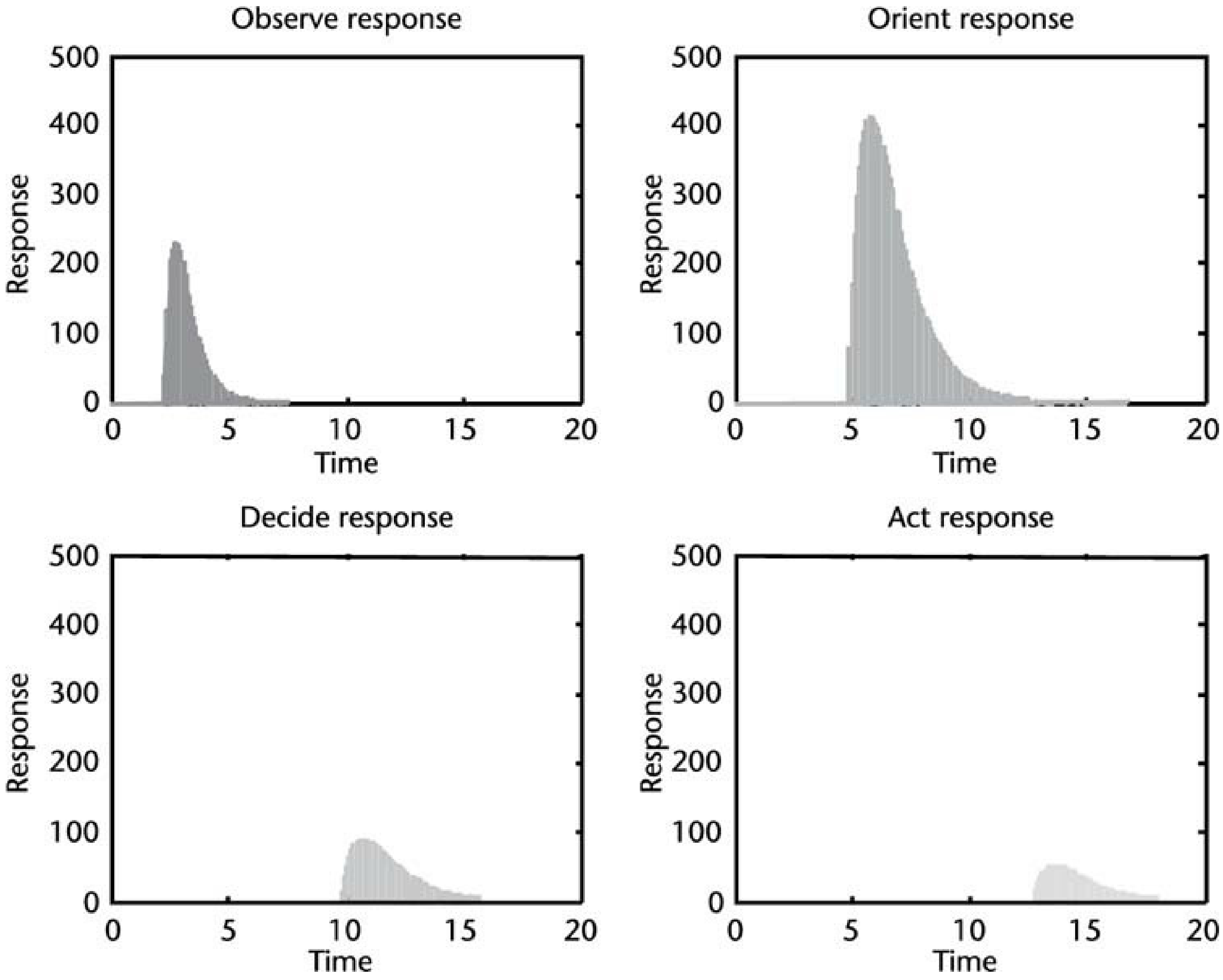


Figure 10.10 Delay responses in C-OODA decision fusion.

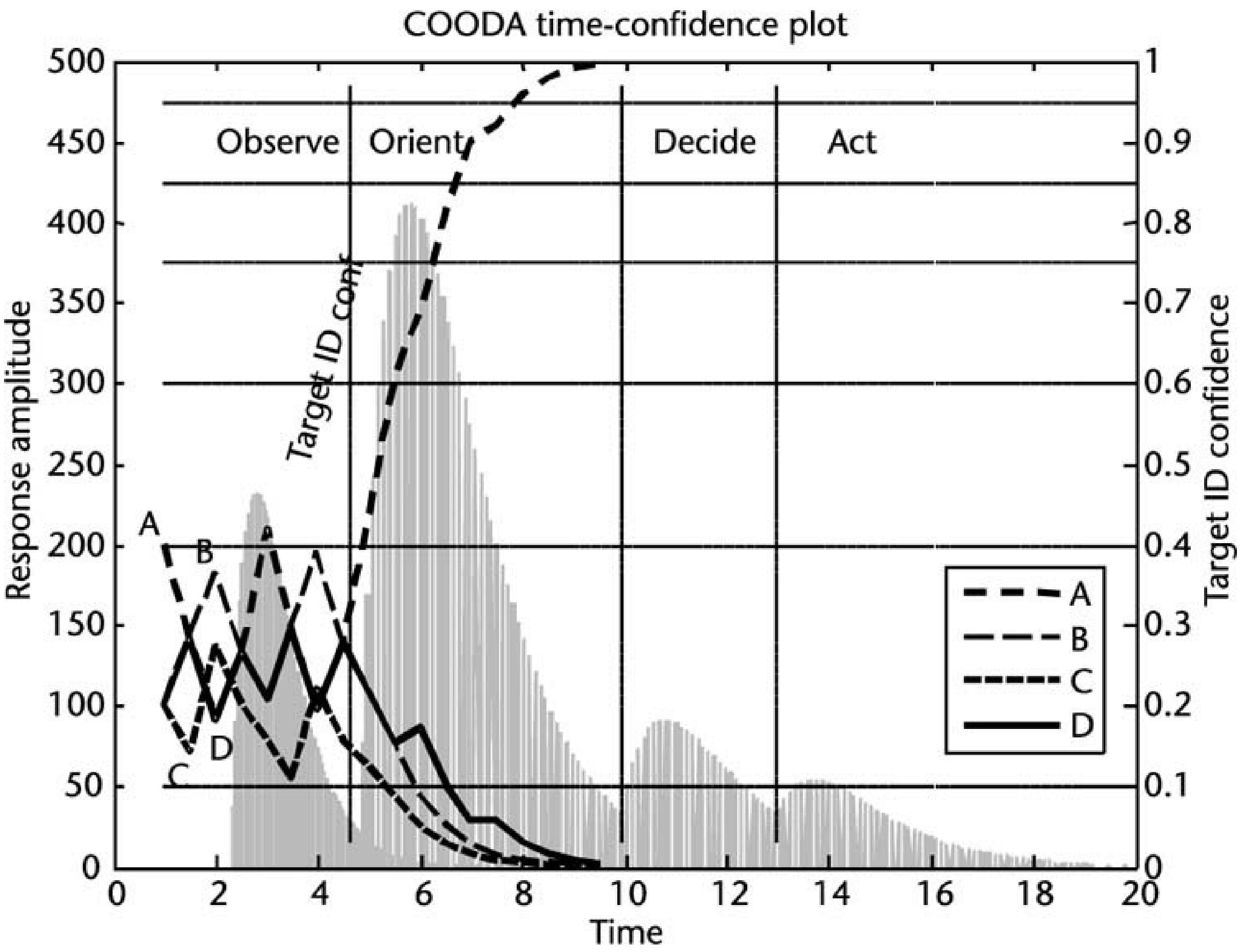


Figure 10.11 Overall time response.

Figure 10.9 shows a time response function including the I/O delays used to reach a decision and the ability to evaluate and execute the decision at each COODA stage. Figure 10.10 details the overall system response as a timeliness analysis of the C-OODA process. An example was simulated in an object recognition task where the C-OODA involves mainly features matching and comprehension (second observe process and first orient process). The time to estimate the target classification is based on feature fusion for decision making requiring 9s from observation to target fused classification presentation and user assessment. The user looks at the display and determines what actions to take, such as tasking another sensor to follow the target. The action decision of which sensor and where to point the sensor requires 13-17s.

The methodology and analysis can be used in IFSs evaluations. Using the COODA as descriptive model, we can decompose the timeliness of data-to-decisionto- action over the various processes. Determining which processes are most time intensive can aid in future IF system enhancements to augment the user's needs for actionable intelligence. In this example, the peaks in Figure 10.11 suggest higher time response for matching perceived environmental features and comprehending those features in order to recognize the object.

Discussion and Conclusions

In this chapter, we overviewed the recent developments in information fusion DM and action selection models using the cognitive OODA (C-OODA) model. Many developments have progressed from the original Boyd model in DM in the cognitive, psychological, biological, and perceptual literature to instantiate the role of the user in information fusion system design. We advocate the use of the C-OODA which offers a high level of cognitive granularity and detailed criteria-based control modules that include both time and uncertainty as factors in cognitive processing. The C-OODA descriptive Command and Control (C2) model affords a control processing model to coordinate the timeliness of DM between the user and the information fusion evidential reasoning and state estimation machine. The chapter looks at a timeliness assessment from data gathering and perception to decision making and action. Future research would evaluate the modeling developments (of which control theory is one choice) of data throughput, estimation accuracy, and evidential confidence from human-factor operational studies to develop pragmatic control interactions between the user and the machine.

rences

[1] Blasch, E., “Sensor, User, Mission (SUM) Resource Management and Their Interaction with Level 2/3 Fusion,” *International Conference on Information Fusion*, 2010.

[2] Blasch, E., I. Kadar, J. J. Salerno, M. M. Kokar, S. Das, G. M. Powell, , D. D. Corkill, and E. H. Ruspini, “Issues and Challenges of Knowledge Representation and Reasoning Methods in Situation Assessment (Level 2 Fusion),” *Proceedings of SPIE* 6235, 2006.

[3] Fabian, W. Jr., and E. P. Blasch, “Information Architecture for Actionable Information Production, (A Process for Bridging Fusion and ISR Management),” *National Symposium Sen Data Fusion*, 2002.

[4] Fadok, D. S., J. Boyd, and J. Warden, *Air Power’s Quest for Strategic Paralysis*, Maxwell Air Force Base AL: Air University Press, (AD–A291621), 1995.

[5] Breton, R., and R. Rousseau, “The M-OODA: Modeling of the OODA Loop as a Modular Functional System,” Def. Res. and Dev. CA-Valcartier, *DRDC TM 2008-130*, May 2008.

[6] Breton, R., and R. Rousseau, “Modeling Approach for Team Decision-Making,” *Defence R&D Canada–Valcartier Technical Report*, DRDC Valcartier TR 2003-368, March 2003.

[7] Breton, R., and R. Rousseau, “The Analysis of Team Decision Making Architectures,” in M. J. Cook, J. Noyes, and Y. Masakowski (eds.), *Human Factors of Decision Making in Complex Systems*, United Kingdom: Ashgate Publishing Ltd., 2006.

[8] Rousseau, R., and R. Breton, “Cost/Benefit Issues in Team Decision Making in Relation with Team Structure,” *Proceedings NATO TG23 Team Effectiveness group Human in Command Workshop*, 2005.

[9] Breton, R., and R. Rousseau, “The C-OODA: A Cognitive Version of the OODA Loop to Represent C2 Activities,” *Proceedings of the10th International Command and Control Research and Technology Symposium*, 2005.

10] Breton, R., “The Modeling of Three Levels of Cognitive Xontrols with the Cognitive-OODA Loop Framework.” Def. Res. and Dev. CA-Valcartier, *DRDC TR 2008-111*, September 2008.

11] Bedworth, M., and J. Obrien, “The Omnibus Model: A New Model of Data Fusion,” *International Conference on Information Fusion*, 1999.

12] Blasch, E. P., and S. B. Plano, “JDL Level 5 Fusion Model ‘User Refinement’ Issues and Applications in Group Tracking,” *Proceedings of SPIE*, Vol. 4729, 2002.

13] Blasch, E., I. Kadar, K. Hintz, J. Bierman, C-Y. Chong, and S. Das, “Resource Management Coordination with Level 2/3 Fusion,” *IEEE Aerospace and Electronic Systems Society Magazine*, March 2008, Vol. 23, No. 3, pp. 32–46.

14] Kadar, I., “Data Fusion by Perceptual Reasoning and Prediction,” *Proc. Tri-Ser. Data Fusion Sym.* JHU, 1987.

15] Blasch,E. P., “Assembling a Distributed Fused Information-based Human-Computer Cognitive Decision Making Tool,” *IEEE Aerospace and Electronic Systems Magazine*, Vol. 15, No. 5, May 2000, pp. 11–17.

16] Lambert, D. A., “Situations for Situation Awareness,” *International Conference on Information Fusion*, 2001.

17] Salerno, J. J., M. Hinman, and D. Boulware, “Building a Framework for Situational Awareness,” *International Conference on Information Fusion*, 2004.

18] Blasch, E., “Level 5 (User Refinement) Issues Supporting Information Fusion Management,” *International Conference on Information Fusion*, 2006.

19] Blasch, E., “Information Fusion for Decision Making,” in NATO ASI: *Data Fusion for Situation Monitoring, Incident Detection, Alert and Response Management*, 2004.

20] Blasch, E., “User Refinement in Information Fusion,” *Handbook of Multisensor Data Fusion: Theory and Practice, Second Edition*, M. E. Liggins, D. Hall, and J. Llinas (eds.), CRC Press, 2008.

21] Kahler, B., and E. Blasch, ”Sensor Management Fusion Using Operating Conditions,” *IEEE National Aerospace and Electronics Conference (NAECON)*, July 2008.

22] Blasch, E. P., and J. Gainey Jr., “Physio-Associative Temporal Sensor Integration,” *Proceedings of the SPIE*, Vol. 3390, 1998.

23] Paradis, S., R. Breton, and J. Roy, “Data Fusion in Support of Dynamic Human Decision Making,” *International Conference on Information Fusion*, 1999.

24] Breton, R., R. Rousseau, and W. L. Price, “The Integration of Human Factors in the Design Process: A TRIAD Approach,” *Defence Research Establishment Valcartier*, TM 2001-002, 2001.

25] Breton, R., and E. Bossé, “The Cognitive Costs and Benefits of Automation,” *NATO RTOHFM Symposium: The Role of Humans in Intelligent and Automated Systems*, October 2002.

26] Perlovsky, L. I., “Cognitive High Level Information Fusion,” *Information Sciences*, Vol. 177, 2007, pp. 2099–2118.

27] Blasch, E., “Proactive Decision Fusion for Site Security,” *International Conference on Information Fusion*, 2005.

28] Lawson, J. S., “Command and Control as a Process,” *IEEE Control Systems Magazine*, March 1981, pp. 5–12.

29] Mayk, I., and I. Rubin, “Paradigms for understanding C3, Anyone?,” in *Science of Command and Control: Coping with Uncertainty*, S. E. Johnson and A. H. Levis (eds.), London: AFCEA International Press, 1981, pp. 48–62.

30] Endsley, M. R., “Toward a Theory of Situation Awareness in Dynamic Systems,” *Human Factors Journal*, Vol. 37, No. 1, March 1995, pp. 32–64.

31] Klein, G. A.,“A Recognition-Primed Decision (RPD) Model of Rapid Decision Making,” in *Decision Making in Action: Models and Methods*, Klein, G., J. Orasanu, R. Calderwood, and C. E. Zsambok (eds.), Norwood, NJ: Ablex, 1993, pp. 138–147.

32] Blasch, E., “Situation Impact and User Refinement,” *Proceedings of the SPIE*, Vol. 5096, 2003.

33] Andriole, S., and L. Adelman, *Cognitive Systems Eng. for User-Computer Interface Design, Prototyping, and Evaluation*, Lawrence Erlbaum, 1995.

34] Simon, H., *The New Science of Management Decision*, New York: Harper Brothers, 1960.

35] Huber, G. P., *Managerial Decision Making*, Glenview, IL: Scott Foresman, 1980.

36] Wohl, J. G., “Force Management Decision Requirements for Air Force Tactical Command and Control,” *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 11, 1981, pp. 618–639.

37] Rasmussen, J., A. M. Pejtersen, and L. P. Goodstein, *Cognitive Systems Engineering*, New York: Wiley, 1994.

38] Vicente, K. J., *Cognitive Work Analysis*, *Toward Safe, Productive and Healthy Computerbased Work*, Lawrence Erlbaum, 1999.

39] Suchman, L. A., *Plans and Situated Actions: The Problem of Human-machine Communication*, Cambridge University Press. 1987.

40] Blasch, E., and S. Plano, “Level 5: User Refinement to Aid the Fusion Process,” *Proceedings of SPIE*, Vol. 5099, 2003.

41] Jagacinski, R. J., and J. M. Flach, *Control Theory for Humans*, Mayhaw, NJ: Lawrence Erlbaum Associates, 2003.

Part IV
Scenario-Based Design

CHAPTER 11

Scenario-Based Design for Situation Analysis

Elisa Shahbazian, Luc Pigeon, Éloi Bossé, and James Kraft (Canada)

The design of a distributed Command and Control (C2) system, which provides high-level information fusion (HLIF) enabled decision support, requires the consideration of multiple interrelated factors and constraints which do not lend themselves to simple analyses. Such systems are very complex, with challenges concerning numerous aspects of the system, including legal, jurisdictional, information sharing and exchange, information availability and quality, user roles and hierarchy, and application of both hardware and software technologies. There are numerous stake holders and no one has an understanding of or access to the entire knowledge in order to be able to resolve any of these challenges by themselves. There are too many unknowns for a traditional top-down design methodology to be feasible to use. Because scenarios are task-based and descriptive, it was judged that the scenario-based design (SBD) approach needs to be explored for HLIF enabled systems where the human is part of the system. By being descriptive, the scenarios preserve the human-support requirements. Therefore, this chapter proposes an SBD methodology for the design of an HLIF enabled decision support. The SBD methodology is illustrated on two stressful scenarios detailed in Chapter 12.

Introduction

A Canadian contribution to the collaborative work undertaken between Australia, Canada, the United Kingdom, and the United States on higher-level information fusion through the Technical Cooperation Program (TTCP) panel on information fusion, is a scenario called "Military Strike," in Atlantis detailed in [1]. In the context of this scenario, there are multiple distributed and adversarial stakeholders who use data/information fusion to establish their situational awareness. The data/ information to be fused is generally imperfect. It can be uncertain, incomplete, imprecise, inconsistent, ambiguous, or some combination of these, due to limited sensor coverage, report ambiguity, report conflicts, or inaccuracies in measured data, as well as due to information sharing constraints as a result of jurisdictional and other security constraints. It can come from various distributed sources that do not always include the pedigree information.

Generally it is anticipated that in C2, both the Level 1 data/information fusion enabled tactical picture compilation and the HLIF enabled decision support capabilities can enhance the C2 performance. However, the design and implementation of such capabilities within the C2 is a very complex undertaking, with challenges concerning numerous aspects of the system, including legal, jurisdictional, information sharing and exchange, information availability and quality, user roles and hierarchy, application of technologies both hardware and software (e.g., sensors, decision support, and human factors). There are numerous stake holders, and no stake holder alone has an understanding of, nor access to, the entire knowledge in order to be able to resolve the numerous challenges by themselves. There are too many unknowns for a traditional top-down design methodology to be feasible to use, and there is no established methodology for validation of the design choices as well as human systems integration considerations.

Because scenarios are task-based and descriptive, it was judged that the SBD approach needs to be explored for HLIF enabled systems where the human is part of the system, since SBD has been used during the conceptual design of large information systems [2]. While there is a great deal of literature describing the design of commercial information systems, there is very little published on the use of this methodology for the design of large military systems providing technology-enabled decision support.

This chapter summarizes the results of the findings in the literature on the application of SBD methodology for system design. It proposes a tailored SBD methodology based on these findings to address the challenges of the design of military C2, and describes the process of the application of this tailored methodology for the design of HLIF enabled decision support the Military Strike in Atlantis scenario [1]. This chapter describes an iterative approach to maturing the design and development of fusion enabled military C2 in partnership with the technical staff, user, and Subject Matter Expert (SME) personnel within a scenario and the prototyping environment that evolves and matures through iteration.

Findings on SBD Methodology

To be able to understand why an SBD methodology has been proposed and how it has been applied to system/software design, a literature survey was conducted. The SBD methodology was initially proposed in mid-to-late 1980s to ensure that human factors aspects be addressed in software design [3]. With software systems becoming more and more complex, many additional benefits of SBD methodology in software design have been observed [4]. It provides a good explanation of how SBD helps understand the various constraints in the design, such as technological, psychological, and legal constraints. Figure 11.1 depicts how and why SBD methodology is beneficial as explained in [4].

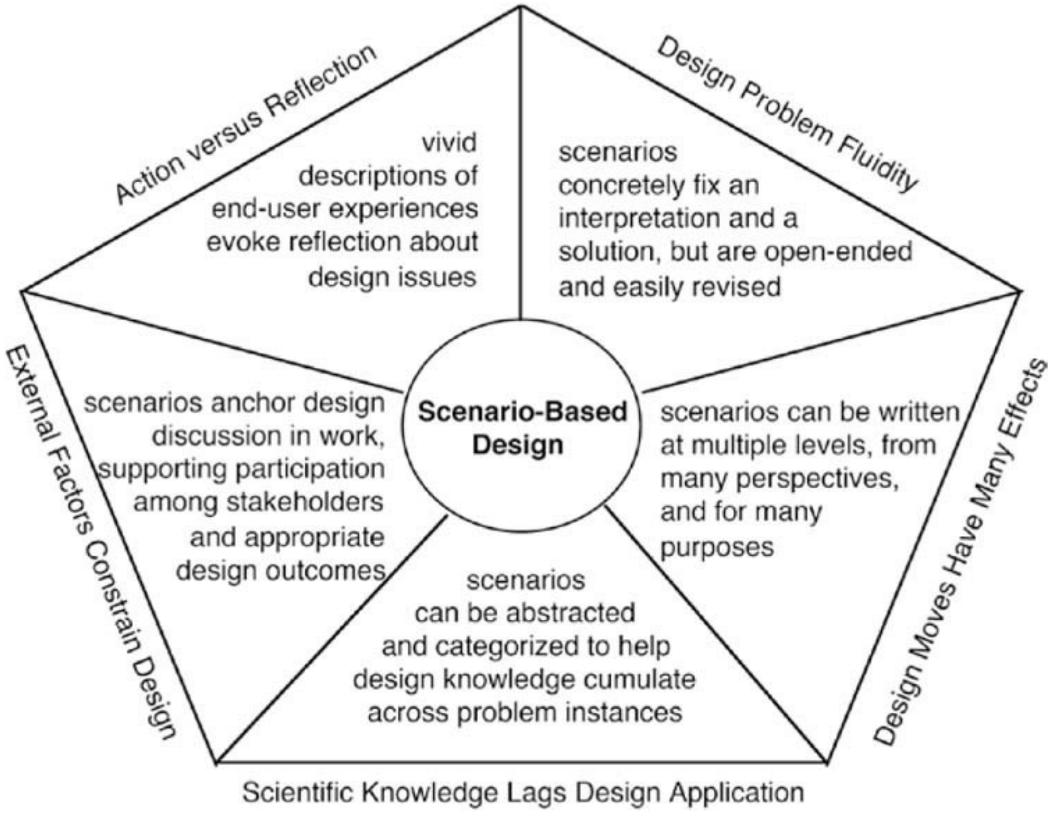


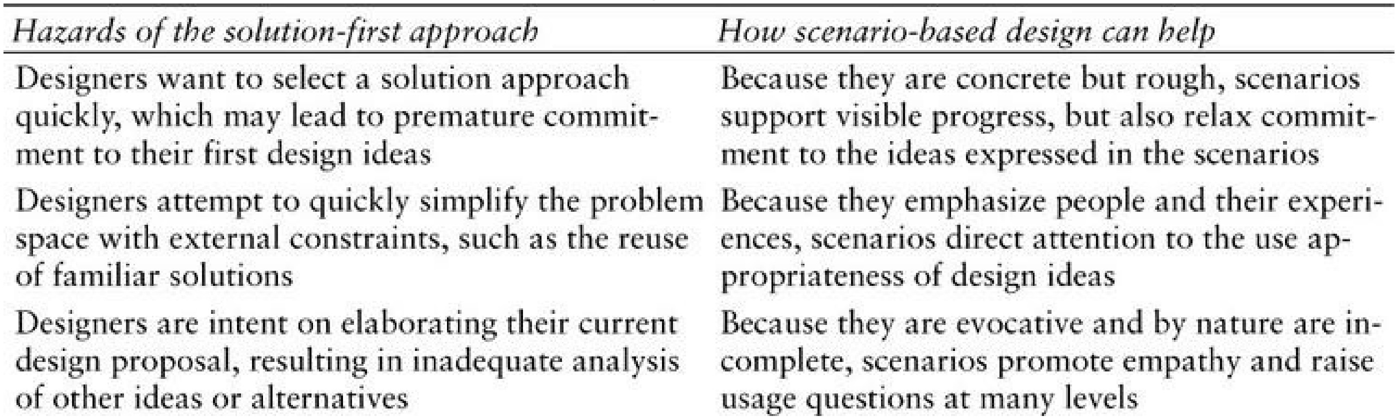
Figure 11.1 Benefits of scenario-based design.

Carroll and Rosson [3–7] have been the main proponents of the SBD methodology, and here are a list of very pertinent observations excerpted from their work:

- SBD helps understand the various constraints in the design, (e.g., technolgical, psychological, legal). A scenario is a concrete design proposal that a designer can evaluate and develop, but is also rough in that it can be easily altered and allows many details to be deferred [4].

- The software designers don’t work from hierarchical decompositions; instead, they interleave top-down and bottom-up development with partial and interim solutions that play no concrete role in the ultimate solution [5].
- The minimalist view of design is proposed as a process of iterative development and goal discovery, with a strong and concrete starting point, and a coherent design vision that may (perhaps must) entrain design dilemmas and trade-offs [5].
- Scenarios can be seen as a vocabulary that can span both descriptions of existing tasks, incorporating existing technology, and of future tasks, incorporating envisioned technology [6].
- A scenario envisions a concrete design solution, but it can be couched at many levels of detail. Carroll, et. al., specify a possible design hypotheses by specifying the tasks users can carry out, but without committing to lower-level details describing how the tasks will be carried out, or how the system will present the functionality for the tasks [7]. Scenarios overcome hazards of the solution-first approach as shown in Table 11.1 [7].

Table 11.1 How SBD Addresses Solution First Design Issues



Hazards of the solution-first approach	*How scenario-based design can help*
Designers want to select a solution approach quickly, which may lead to premature commitment to their first design ideas	Because they are concrete but rough, scenarios support visible progress, but also relax commitment to the ideas expressed in the scenarios
Designers attempt to quickly simplify the problem space with external constraints, such as the reuse of familiar solutions	Because they emphasize people and their experiences, scenarios direct attention to the use appropriateness of design ideas
Designers are intent on elaborating their current design proposal, resulting in inadequate analysis of other ideas or alternatives	Because they are evocative and by nature are incomplete, scenarios promote empathy and raise usage questions at many levels

The SBD framework developed in [7] and shown in Figure 11.2 was judged to be a very good baseline framework for application of the SBD methodology for the design and development of data/information fusion enabled military C2. One of the great strengths of scenario-based methods is that they support of a diverse range of system development life cycle.

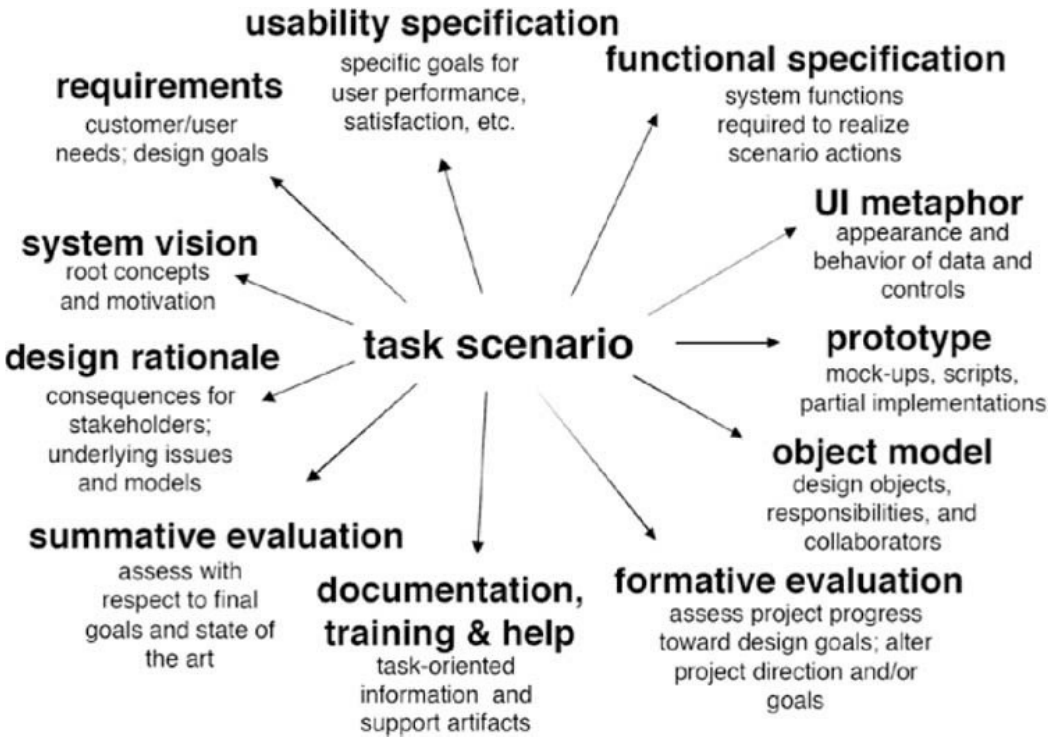


Figure 11.2 An overview of the SBD framework (e.g., user interface (UI) metaphor).

Among the reviewed literature, most of the applications where an SBD has been applied were much simpler than data/information fusion enabled military C2, except for a few, such as [8]. These applications are still not as complex as military C2; however, they demonstrated successful application of the methodology, and [8] specifically provides explicit detailed descriptions of the documentation describing the various phases of design decomposition during the design step in Figure 11.1. Previous examples are helpful for appreciation of the scope of the work in design of complex systems. The section below discusses the proposed SBD framework to address the complexity of data/information fusion enabled military C2.

11.2.1 The Proposed SBD Framework for Military C2

The proposed SBD framework should support the design of a very large system consisting of multiple, interconnected, distributed decision making inter-jurisdictional nodes. Starting from its very early stage of development, where many aspects of the system functionality, communication and information contents, and exchange requirements are not sufficiently defined or understood. These aspects are iteratively evolving through the system development life cycle, building an understanding of requirements, functionality, communication and information contents, and exchange protocols and capabilities. SBD should allow for piecewise design and analysis of selected subfunctionality and incremental evolution. SBD should allow for nonparticipatory analyses, as well as participatory design at various stages of development. SBD should also provide for the possibility of incremental evolution of the scenario, as required to support the evolution of the design. There are too many unknowns for a traditional top-down design methodology to be feasible to use. While the SBD framework described in Figure 11.1 would allow for iterative usability analysis of the design at various levels, it does not provide for the iterative evolution of the problem scenario.

The artifacts of design activity will constitute narrative descriptions of scenarios at various levels of system design/decomposition and prototypes of various level of fidelity depending on the specific aspect being investigated.

At different levels of the system design maturity, different teams of experts should collaborate to analyze the scenarios, and then build and exercise the prototypes with appropriate personnel for the specific phase of the system development. When analyzing high level requirements, subject matter experts knowledgeable about the current legacy systems should work with designers and technology experts, while when analysing application of a HLIF method in a specific decision support capability, operational staff should also participate to analyze the user interface (UI) of the decision support tools. The development should be collaboration between system/software engineers, technology experts, human factors experts and subject matter experts, who will be of different domain at different levels of the development life cycle.

At different phases of the design, experts in different domain should take a leading role. The system/software engineer and the technology expert should play the leading role in the system development. Considering the size and complexity of the targeted system, a human factors expert and a subject matter expert (different at various stages of the lifecycle) must be always involved in the design; however, since this is an iterative process which allows for partial design, discussions, analyses, redesign, and further experimentation, expert participation need not be continuous. At different levels of design, appropriate teams of technology, subject matter experts, and operations personnel iteratively will brainstorm, propose prototype redesigns, new scenario development, modification to previous design approaches, and scenarios. Group sizes could be as small as two or three individuals including a subject matter expert; however; in various stages of the system life cycle, larger groups of operational personnel could be required.

Considering the complexity of the system and its numerous challenges that need to be understood and solutions developed, the artefacts of design activity will be continuously built on and iteratively analyzed.

As mentioned above we propose to tailor Rosson's and Carroll's framework [7], shown in Figure 11.1. In this methodology, high-level problem scenarios are analyzed and decomposed into activity scenarios, information scenarios, and interaction scenarios. The design process allows for analyzing the technology applications and constraints, and subsequently leads to a prototype and evaluation activities. We propose to document the Rosson and Carroll methodology with the type of documentation described by Bardram [9] for the design phase, tailored to ensure that the information and interaction aspects of the scenarios are addressed and documented appropriately. On a high level, Rosson's and Carroll's methodology appears to address many aspects of a complex system design. However, the main issue that is not addressed is the fact that there will be a need for a back-up loop, as shown in Figure 11.3. A back-up loop is needed because it is unrealistic to assume that the problem scenario can be fully described from the beginning in and large systems often contain multiple distributed decision makers, multiple jurisdictions, and numerous challenges involving information quality, availability, sharing, exchange, doctrine, and legal constraints. This is in addition, technology readiness, availability, feasibility and trust issues, introduce additional elements that are likely not be possible to define in advance.

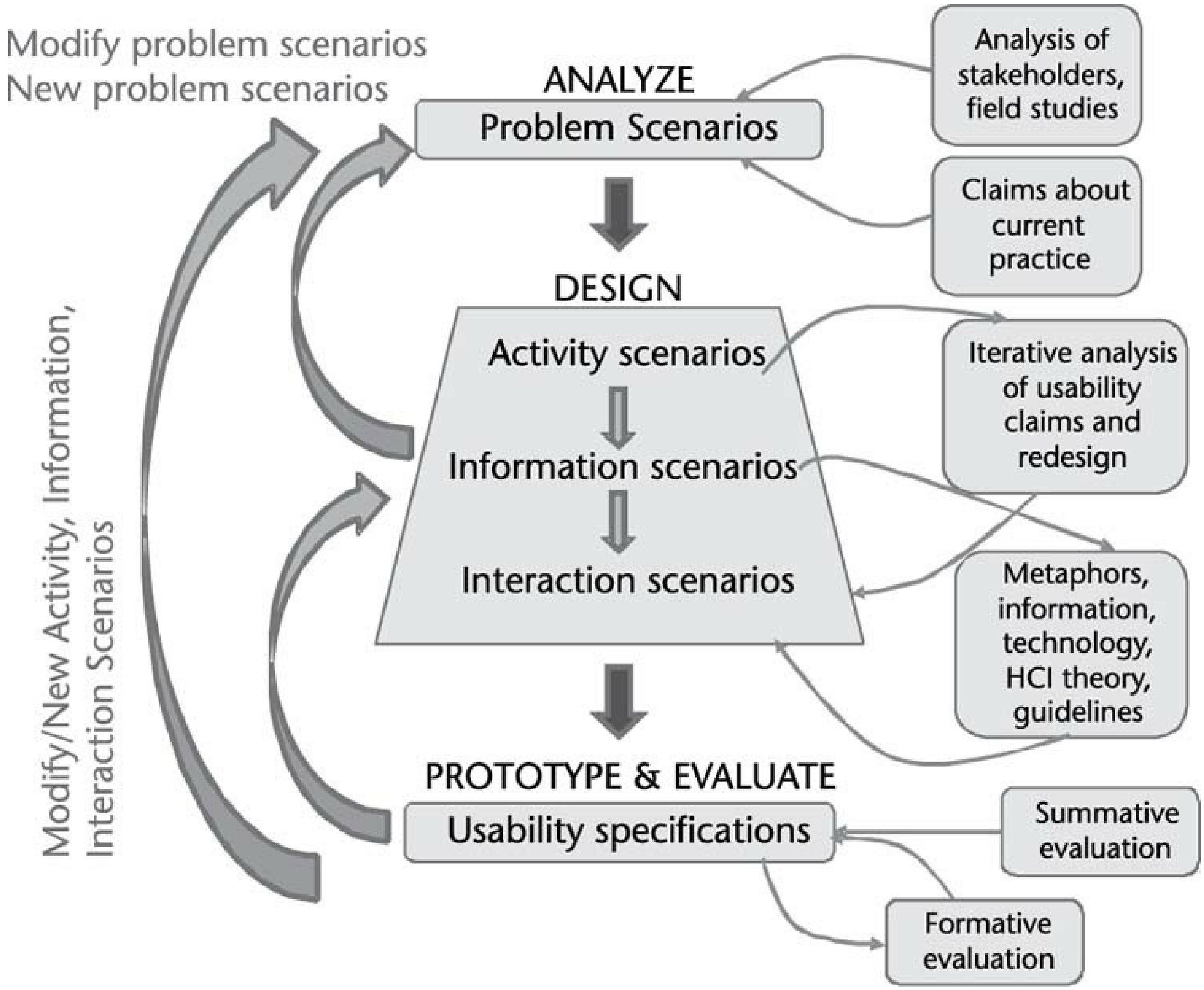


Figure 11.3 Proposed SBD framework.

The problem scenarios will continuously evolve. New problem scenarios describing specific activities identifying new challenging situations are likely to be required as a result of the design phase, as well as through demonstration and discussion of the prototypes with the subject matter experts. Furthermore, through demonstration and discussion of the prototypes with the subject matter experts the activity, information, and interaction scenarios will also evolve. The evolution will identify new activities, requirements for information processing and sharing, technology solutions, and tools and methods.

The benefits produced as an outcome of application of such a framework will be numerous, including:

- Building an understanding of current practice and challenges that need to be overcome in collaboration with the stake holders and subject matter experts;
- Identifying requirements for work activities, functionality, communication, information contents and exchange protocols, capabilities, and operator functionality to help overcome the challenges;
- Identifying technology needs and constraints, and development and demonstration of technology enabled solutions to the requirements above in partnership with stake holders and subject matter experts;
- The establishment of an iterative methodology to incrementally design a complex distributed system, including the technological solutions, which has been corroborated by the stake holders and subject matter experts;
- The selection of the system/software architecture which would facilitate the iterative prototyping and evaluation activities. For example, some papers reviewed recommended agent based architectures to facilitate easy and independent iterations.

These results, both the methodology as well as validated technological solutions can be directly used in a targeted system development activity. The SBD methodology should be used throughout the whole life cycle of a large distributed system development. However, it can also be used for requirements analysis or to analyze technological solutions for enhancements to overcome some problems in specific areas of a system.

Prototype development of various levels of fidelity depending on the life cycle will be used. In the design activities, Unified Modeling Language (UML) could be used to specify, visualize, modify, construct, and document some artifacts of activity, information and interaction scenarios. The proposed SBD framework is now being demonstrated on the Military Strike in Atlantis Scenario (Chapter 12).

11.2.2 Specifics of the Military Strike in Atlantis Vignette

The Military Strike Vignette takes place in a fictitious country called Atlantis, continent located in the North Atlantic Ocean, between Europe and Greenland[1] and detailed in Chapter 12. Atlantis is composed of six countries: Blueland, Orangeland, Redland, Brownland, Whiteland, and Greyland.

The vignette has been designed to take place amid very complex historical and political situations and, in order to complicate the assessment of the situation, military operations occur amid commercial and peace-time activities. It is centred on a two-ship convoy being simultaneously attacked by submarine and aircraft off the Atlantis west coast. Two surprised attacks occur while other activities are taking place. These activities include commercial air traffic, maritime traffic, observation of whale migration, and other military operations, such as air patrol surveillance and missile deployment. Moreover, one of the opponent countries has planned to take control of the Celtic Straits on that day. The vignette is a description of specific military activities within a dynamic context where various crises are currently ongoing between countries.

Scenario-Based Design Process Based on Atlantis Problem Scenario

This section describes a process that breaks down a scenario into activities and actors to determine user requirements. The SBD process diagram in Figure 11.4 shows the following procedure:

1. Decompose the problem scenario to identify the activities and the actors;
2. Describe the actions taken by the actors and system to accomplish each activity;
3. Extract system functional requirements, performance requirements, and business rules;
4. Describe the information entities, attributes and information interrelationships for each activity. This provides logical data models which can be used to create physical data models;
5. Design, build, test and implement a prototype, which may focus on subset activities.

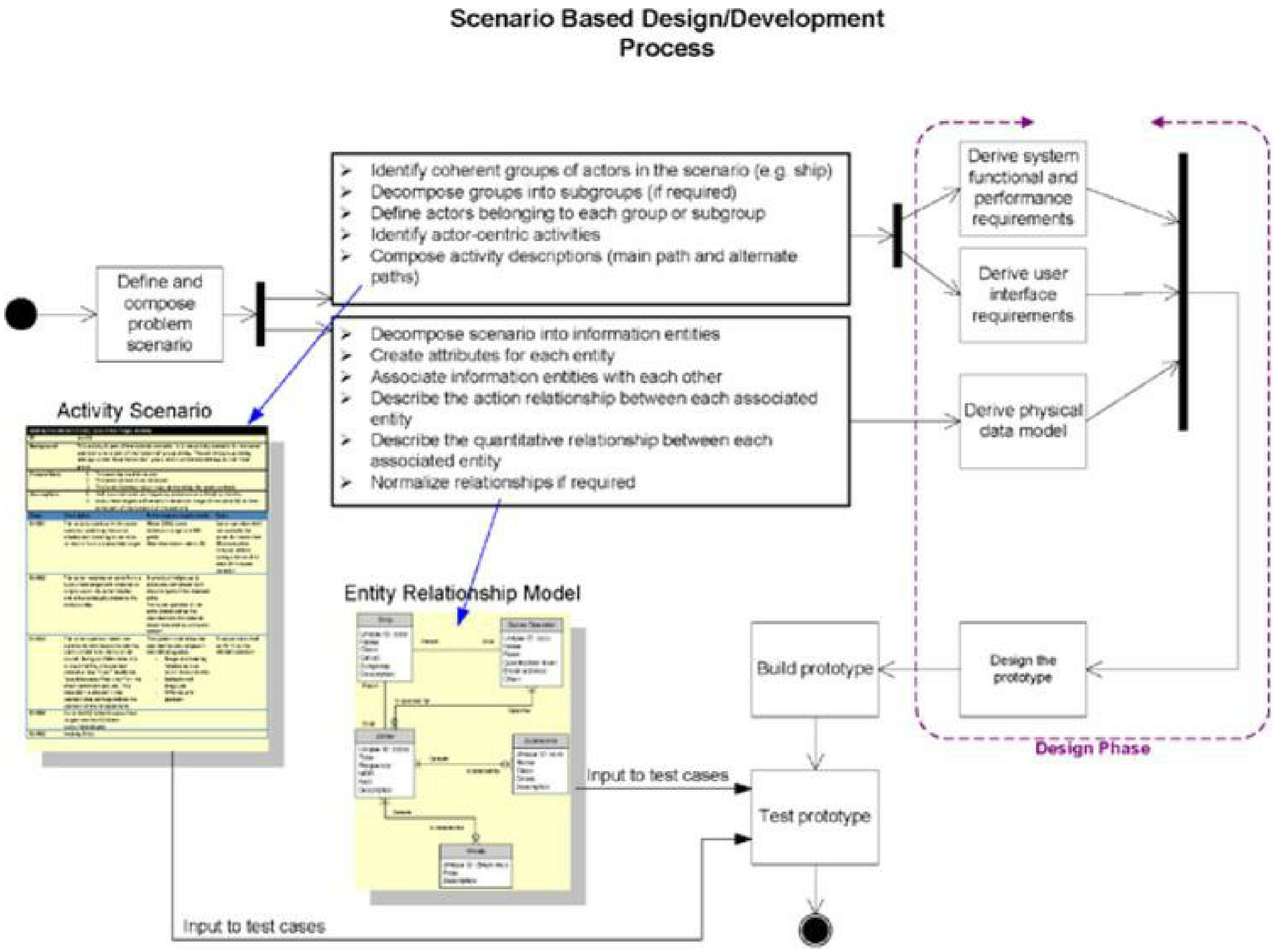
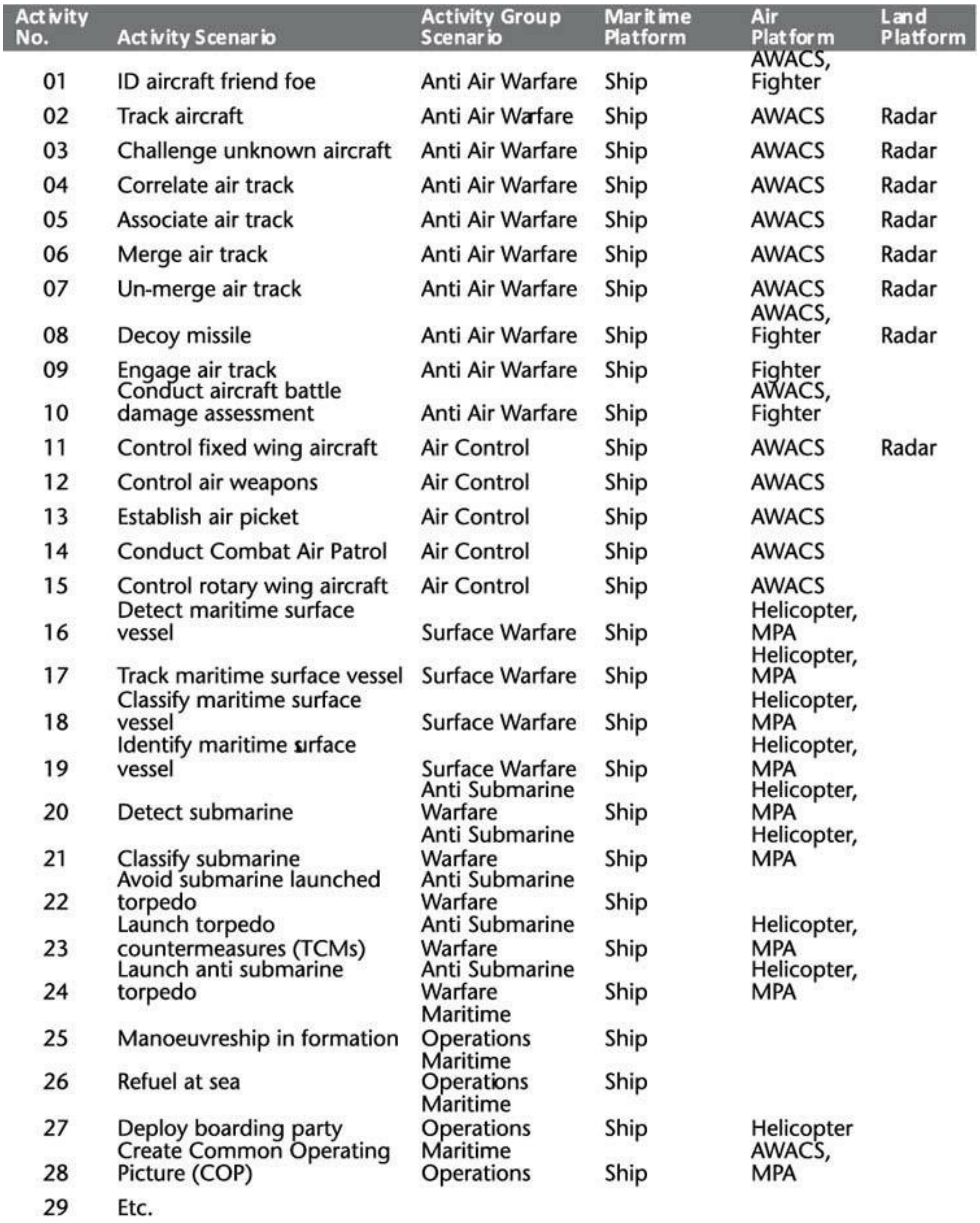


Figure 11.4 The example SBD process diagram.

Activities are validated using prototypes to identify any aspects in the vignette that are not sufficiently well explained to substantiate the activities. Figure 11.5 shows a decomposition of the activities from the Atlantis scenario vignette. This is a partial list of activities, as the list can evolve throughout the SBD process analysis.



Activity No.	Activity Scenario	Activity Group Scenario	Maritime Platform	Air Platform	Land Platform
01	ID aircraft friend foe	Anti Air Warfare	Ship	AWACS, Fighter	
02	Track aircraft	Anti Air Warfare	Ship	AWACS	Radar
03	Challenge unknown aircraft	Anti Air Warfare	Ship	AWACS	Radar
04	Correlate air track	Anti Air Warfare	Ship	AWACS	Radar
05	Associate air track	Anti Air Warfare	Ship	AWACS	Radar
06	Merge air track	Anti Air Warfare	Ship	AWACS	Radar
07	Un-merge air track	Anti Air Warfare	Ship	AWACS	Radar
08	Decoy missile	Anti Air Warfare	Ship	AWACS, Fighter	Radar
09	Engage air track	Anti Air Warfare	Ship	Fighter	
10	Conduct aircraft battle damage assessment	Anti Air Warfare	Ship	AWACS, Fighter	
11	Control fixed wing aircraft	Air Control	Ship	AWACS	Radar
12	Control air weapons	Air Control	Ship	AWACS	
13	Establish air picket	Air Control	Ship	AWACS	
14	Conduct Combat Air Patrol	Air Control	Ship	AWACS	
15	Control rotary wing aircraft	Air Control	Ship	AWACS	
16	Detect maritime surface vessel	Surface Warfare	Ship	Helicopter, MPA	
17	Track maritime surface vessel	Surface Warfare	Ship	Helicopter, MPA	
18	Classify maritime surface vessel	Surface Warfare	Ship	Helicopter, MPA	
19	Identify maritime surface vessel	Surface Warfare	Ship	Helicopter, MPA	
20	Detect submarine	Anti Submarine Warfare	Ship	Helicopter, MPA	
21	Classify submarine	Anti Submarine Warfare	Ship	Helicopter, MPA	
22	Avoid submarine launched torpedo	Anti Submarine Warfare	Ship		
23	Launch torpedo countermeasures (TCMs)	Anti Submarine Warfare	Ship	Helicopter, MPA	
24	Launch anti submarine torpedo	Anti Submarine Warfare	Ship	Helicopter, MPA	
25	Manoeuvreship in formation	Maritime Operations	Ship		
26	Refuel at sea	Maritime Operations	Ship		
27	Deploy boarding party	Maritime Operations	Ship	Helicopter	
28	Create Common Operating Picture (COP)	Maritime Operations	Ship	AWACS, MPA	
29	Etc.				

Figure 11.5 Atlantis scenario activity decomposition.

The next step, illustrated through Figures 11.6 to 11.8, is to define the actors. Figure 11.6 shows the high level activities of a warship, which is a member of a blue force.

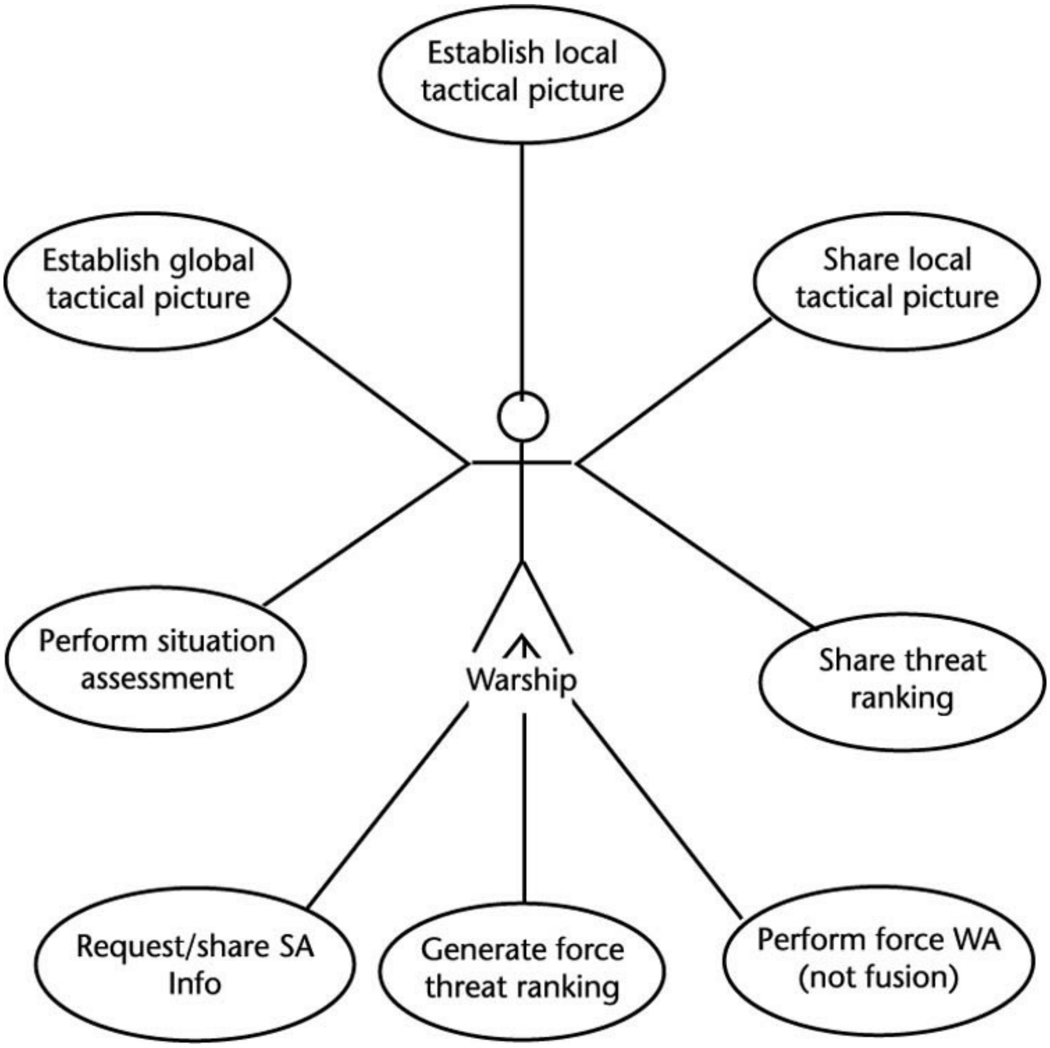


Figure 11.6 Warship activities.

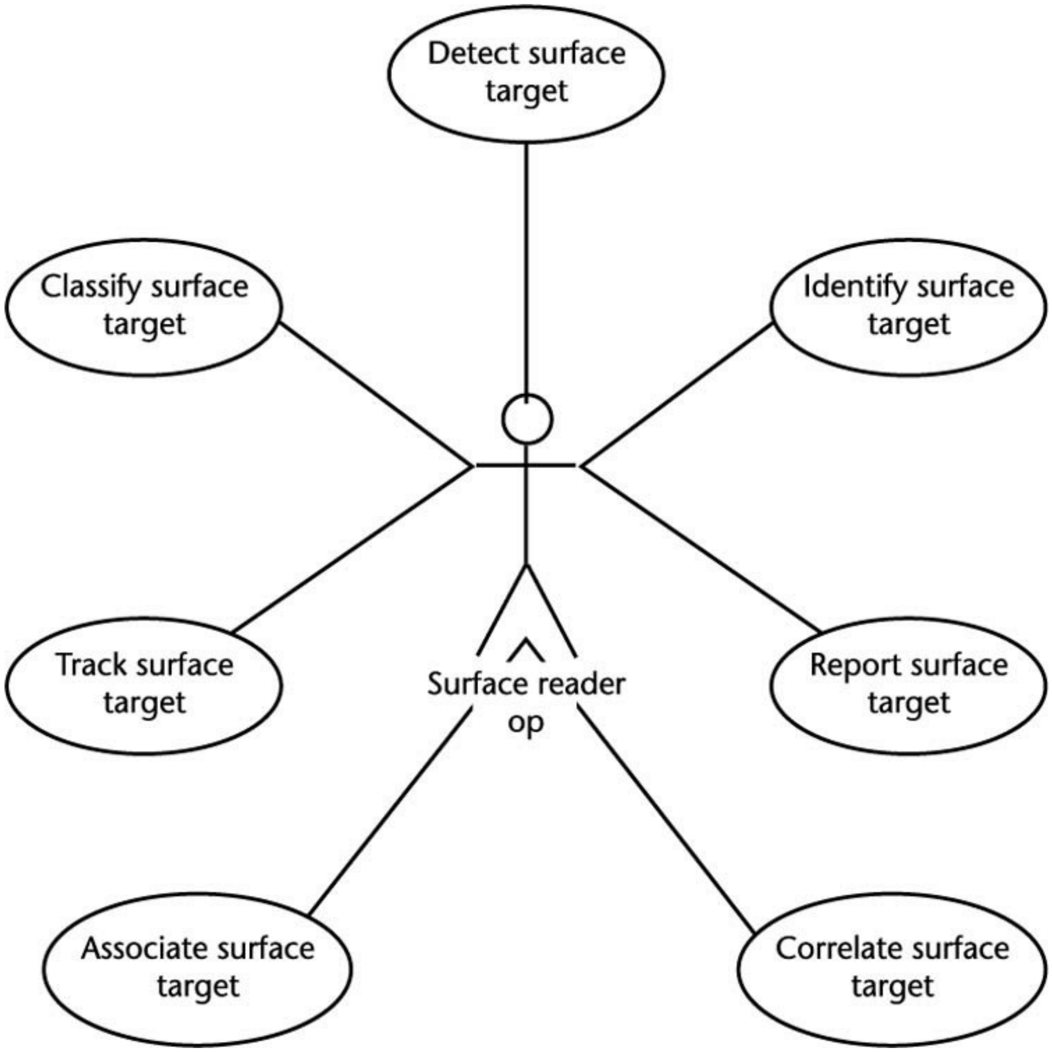


Figure 11.7 Example of an actor-centric activity list.

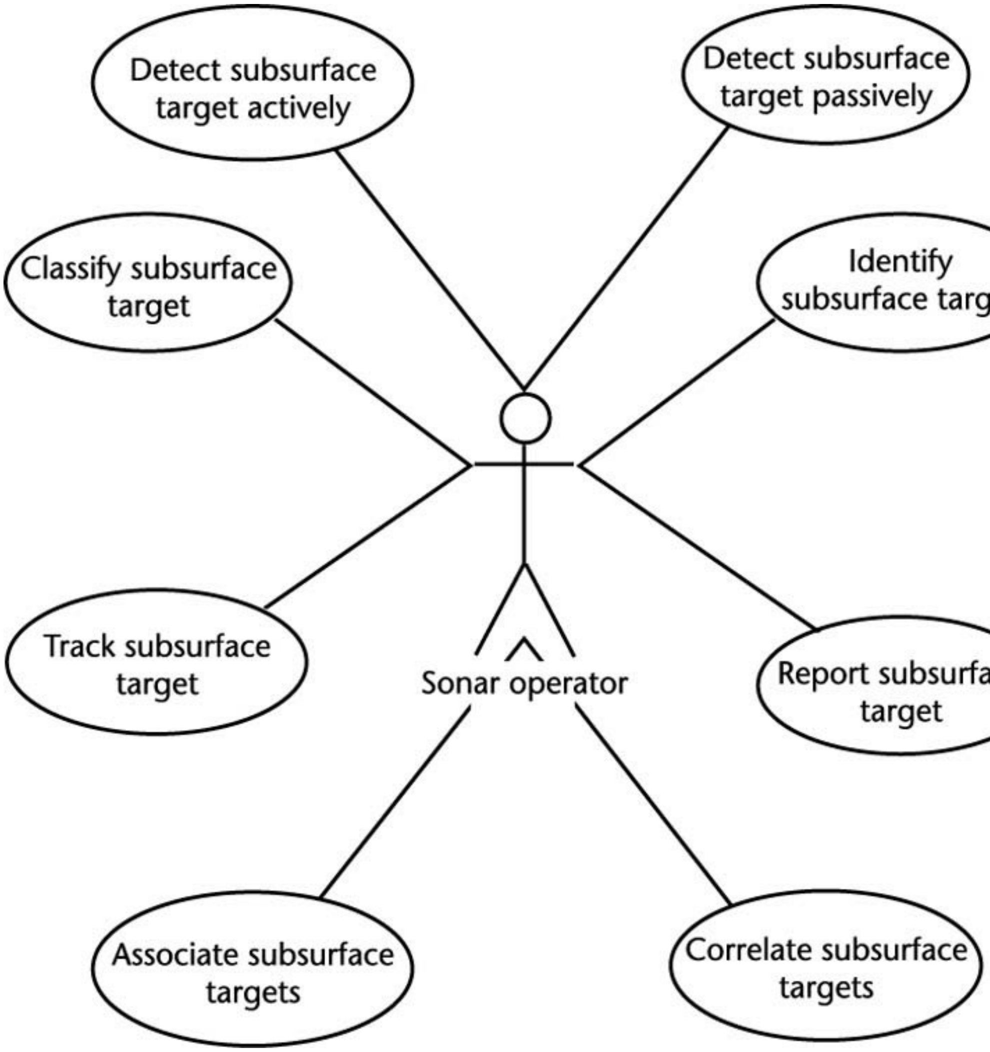


Figure 11.8 Example showing actor-centric activities.

Figure 11.7 shows the surface radar operator, who is a child entity of the radar operator group of actors, which is a child group of the warship group, which is a child entity of the blue force ship entity group, which is a child of the ship entity group.

The detect, classify, identify (ID), and track activities in Figure 11.8 are subactivities of establish local and global tactical picture activities of Figure 11.7. Similarly, correlate and associate surface targets activities are subactivities of perform situation assessment activity. Figure 11.8 shows the sonar operator, which is a child of the warship entity group, which is a child of the blue force ship entity group, which is a child of the ship entity group. These activities are subactivities of warship activities as well.

These are just example activity decompositions. However, one of the major the benefits of the SBD is that at any stage of the design analysis it is not necessary to produce a complete decomposition. We can focus on a subset specific set of activities, examine their behavior, interactions, performance, and user interface, and then proceed with analysis of another activity at any time during the analysis.

By combining the activities and the actors, we can show a dialogue between actors and systems that permit extraction of different types of requirements. These descriptions are accompanied by entity relationship diagrams, shown in Figures 11.9 and 11.10. Additionally, a swimlane diagram is included for the challenge unknown aircraft activity, which is shown in Figure 11.11.

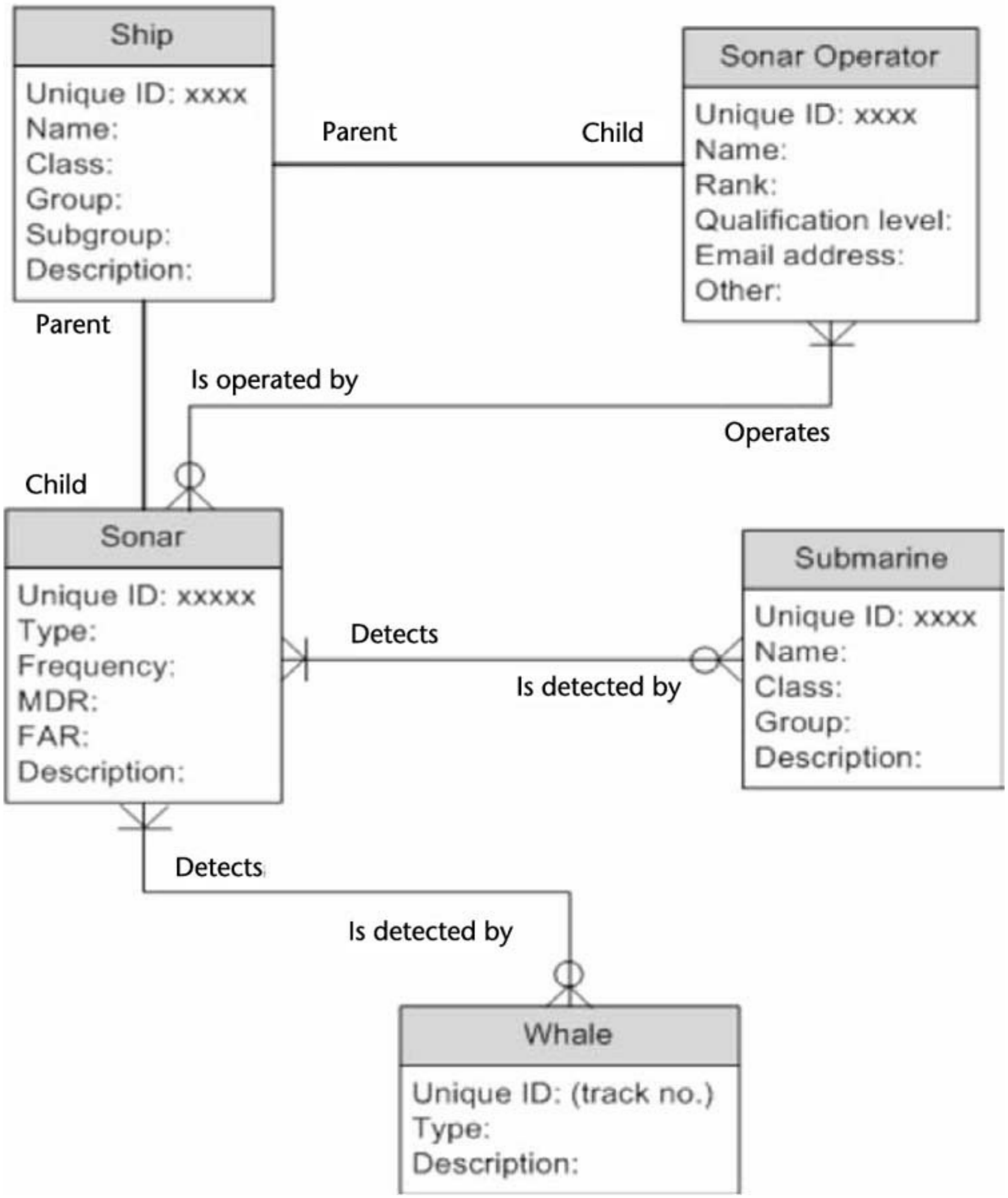


Figure 11.9 Entity relationship diagram related to the “detect” subsurface target activity scenario.

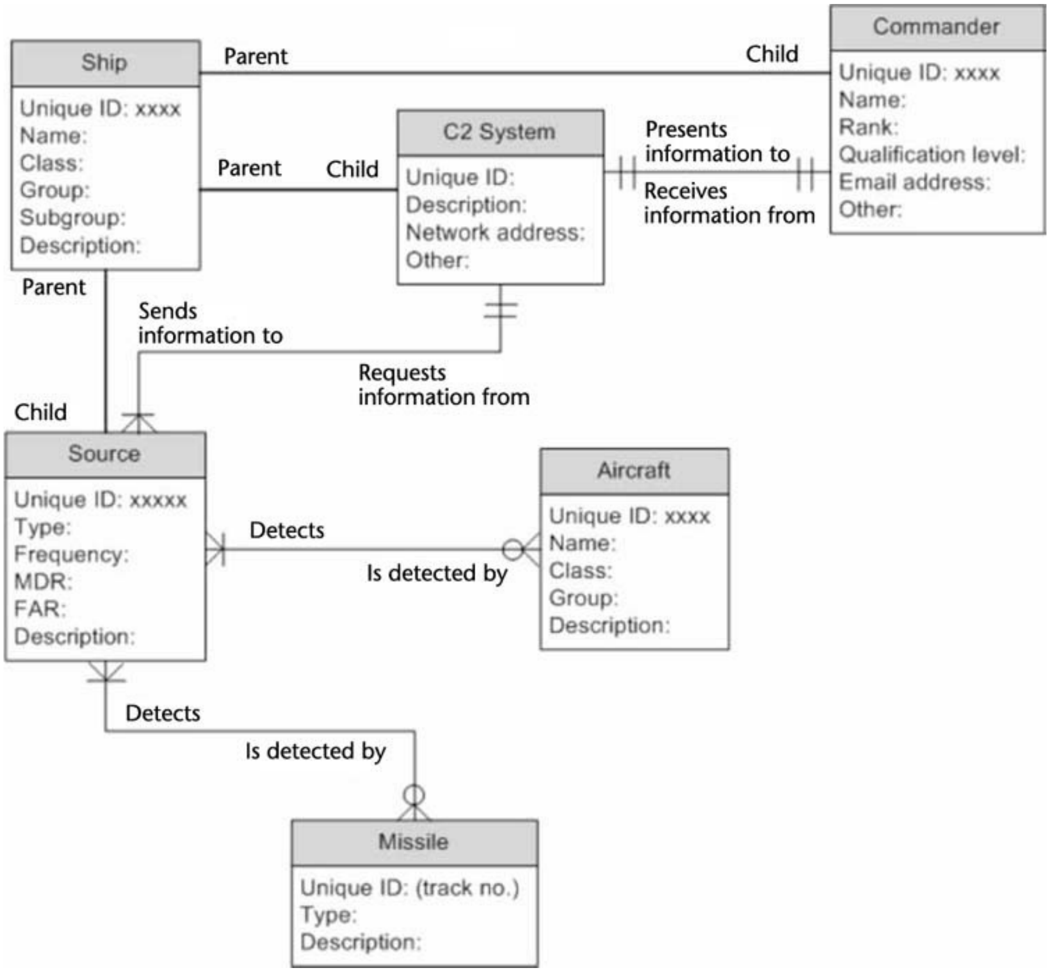


Figure 11.10 Entity relationship diagram related to the Act-03 challenge unknown aircraft activity.

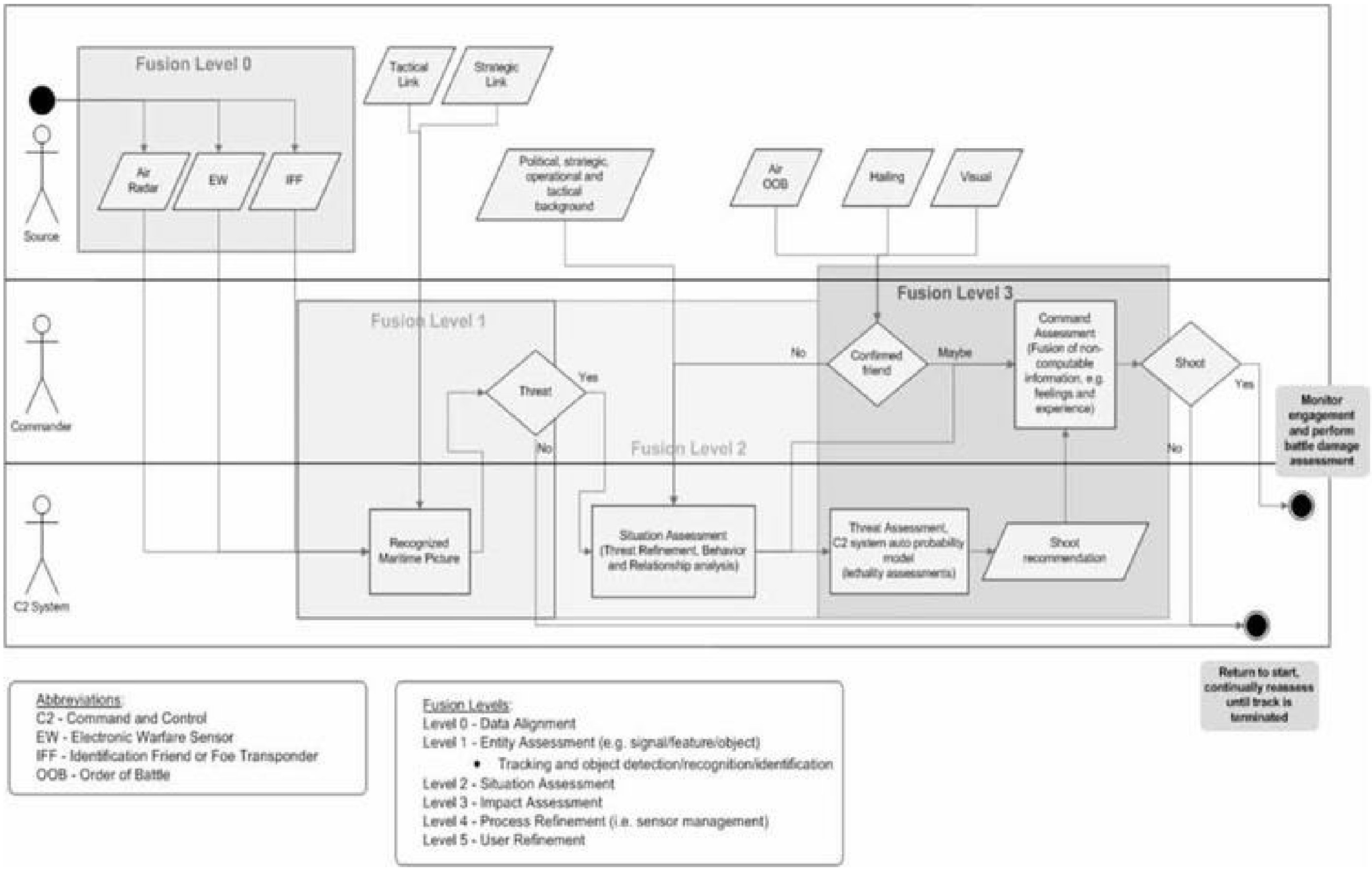


Figure 11.11 Activity process diagram for Act-03 "challenge unknown aircraft."

These entity relationship diagrams are useful to analyze input and output information requirements and characteristics. Examples of useful considerations that such a diagram can deliver to support analysis (e.g., for the entity sonar operator) includes:

- What type of information is required from the C2 concerning the tactical situation (this includes information quality, pedigree and mission characteristics for the sonar operator to be able to assess and associate the data received from the sonar subsystems)?
- What are specific data characteristics of submarine and whale, helping distinguish between the two? Are there any remote sources of information that will enable the user to tell the difference between the submarine and the whale?
- What Level 1 fusion methods are available for the user, given the characteristics of data that can be accessed?
- What HLIF methods (association, maneuver detection, clustering, etc.) could help refine the identification/classification of entities (such as a submarine or a whale)?
- What should be the user interface for providing these Level 1 fusion and HLIF enabled decision support methods to the operator?
- What is the degree of automation for providing each of these decision support

tools? How much information regarding the methods used by the automated tools should be available to the operator? Is this mission dependent and if yes, what are the conditions?

- How is the information about the underwater detections and situation assessment shared between task force units?

These considerations need to be deliberated on in any design methodology, such as top-down waterfall, or SBD. However, the most significant benefit of an SBD is that the answers need neither be final nor unique. There could be multiple technological solutions for each of the above questions. Depending on the technological solution, different assumptions regarding data collection and dissemination between entities will be made, leading to changes in the scenario.

The design activity is more complex, and there are even more considerations to deliberate on. There would be a set of questions similar to those listed for the detect subsurface target activity scenario, such as considerations for input and output data characteristics and decision support requirements for the establishment of situational awareness, as well as human systems integration approaches. In addition, there will be considerations regarding threat assessment, force operations, and resource management. The commander performs a much larger subset of warship activities (shown in Figure 11.7), and these will require a broader range of HLIF enabled decision support.

The Activity Process Diagram for Act-03 (challenge unknown aircraft), shown in Figure 11.11, provides more detail regarding the information flow and specific areas where Level 1 fusion and HLIF enabled decision support would be required in the C2 system to visualize various possibilities and technological solutions.

In the SBD framework, these requirements are viewed as hypotheses, and the analysis is performed in collaboration between technical personnel and a subject matter expert (SME). SMEs can be of various backgrounds and levels of maturity. Initially, when working on a complex scenario, such as Military Strike in Atlantis in a futuristic setting, SMEs need to be military staffs who have participated in allied force operations in littoral warfare.

Given that HLIF is a new and not yet sufficiently validated technology, and considering the futuristic setting of the problem scenario, there are not only technology application issues to be resolved, but also operational and doctrinal issues that (as shown in Figure 11.1) may not even be defined yet. In the proposed SBD framework, the designing collaboration team will make assumptions/hypotheses regarding operational doctrinal as well as technological issues, including:

- Information sharing protocol and contents between coalition task force participants;

- Command structure of the task force;
- The identification of HLIF methods that are pertinent for decision support in each step of an activity (there could be more than one appropriate method for each type of decision making);
- The analysis of information input and output requirements for these methods;
- The analyses of the constraints, geographical, political, etc. that will impact the performance of these methods;
- The analysis of feasibility of each method, considering the information quality and availability issues;
- The analysis of how these fusion enabled capabilities are provided to the operators, including the extent of the automation;
- The analysis of other criteria, such as complexity or cost versus performance trade-offs.

As mentioned above, there could be more than one alternative hypothesis for each issue, and these will be prototyped and tested. During the design phase of the prototype, the analysis of the hypotheses made regarding the command structure, information sharing content, and protocol—as well as the specific information needs of specific technology solutions selected—are very likely to lead to observations that the initial problem scenario (the vignette) is insufficiently defined. The design teams (both SME and technical) will augment the vignette to be able to develop the simulation environment to observe the proposed solution. The vignette will continue to evolve through the iterations of prototyping phases, and this iterative process will lead not only to a better understanding of technology solutions, but also to doctrinal and operational recommendations.

Looking at the incomplete list of activities, (not shown in this chapter) it is apparent that the overall process of decision support development for the Military Strike in Atlantis vignette is a very lengthly process using any design methodology. The main advantage of the SBD is that it enables the design team to investigate a small manageable part of this large project, as many groups in the fusion community have done and continue to do. However, unlike the general practice of just focusing on a specific small solution development, it provides an overall framework for tackling the overall large system design program in small iterative subtasks, keeping the overall system view together.

One interesting observation in the SBD framework is that the prototype, including the simulation and stimulation functionality for evaluating the performance of various technology solutions in various activity scenarios, need not be of same fidelity. This allows for shorter prototyping cycles in certain phases of the project. When observing the behaviors of HLIF functionality, one could make assumptions regarding the model for information quality characteristics onboard (the output from the Level 1 fusion processes) and remote information (coalition participating unit shared information (e.g., datalinks)), and proceed with prototyping of HLIF functionality in selected activity scenarios. We anticipate that most of HLIF enabled decision support validation with SMEs, operational staff as well as human factors specialists for operational and level of automation considerations could be done using such low fidelity simulations.

In the overall process of the decision support system development, there would be many validation perspectives, including algorithmic, technical performance, cost and complexity trade-offs, as well as further human factors studies. The level of fidelity at each phase should be selected according to the nature of the validation under investigation.

Conclusion

This chapter describes the analyses performed to evaluate the application of a scenario-based design (SBD) methodology for the design of the HLIF-enabled decision support (DS) capability using the Military Strike in Atlantis vignette as an example scenario.

The methodology proposed by Carroll and Rosson [3–7] has been tailored to account for a much more complex and uncertain task of designing a distributed military command and control, where we employ HILF-enabled decision support in a small subset SBD analysis.

These analyses lead to the conclusion that a complete design of HLIF enabled decision support in this (or another such complex) scenario can be a very lengthy process. The SBD framework allows decomposing this very large and complex problem into smaller, more manageable subprojects.

The SBD methodology will allow for a subset activities to be prototyped and validated in many iterations with various choices of HLIF methods. SMEs and operators will collaborate in order to gain valuable insight into trade-offs for choosing how such HLIF enabled functionality performs, how it should be used, and what information requirements exist to ensure optimal performance.

The SBD methodology will lead to iterative vignette enhancements to address the operational and technology applications constraints. SBD will also help identify aspects of information sharing, doctrine, and business rules [10] specifications that are required to ensure that HLIF enabled functionalities are appropriately designed and implemented.

rences

[1] Blanchette, M., “Military Strikes in Atlantis—A Baseline Scenario for Coalition Situation Analysis,” The Technical Cooperation Panel, *Technical Report TR-C3I-TP1-1-2005*, 2005.

[2] Hertzum, M. “Making Use of Scenarios: A Field Study of Conceptual Design,” *International Journal of Human-Computer Studies*, Vol. 58, Nos. 1–6, 2003, pp. 215–239.

[3] Rosson, M. B., and J. M. Carroll, “Paradox of the Active User,” in *Interfacing Thought: Cognitive Aspects of Human-computer Interaction*, Carroll, J.M. (ed.), Cambridge, MA: MIT Press, 1987.

[4] Carroll, J. M., “Five Reasons for Scenario-based Design,” *Proceedings of the 32nd Hawaii International Conference on System Sciences*, 1999.

[5] Carroll, J.M., “Reconstructing Minimalism,” *Minimalism beyond the Nurnberg Funnel*, Cambridge, MA: MIT Press, 1997.

[6] Carroll, J. M., M. B. Rosson, G. Chin Jr., and J. Koenemann, “Requirements Development in Scenario-based Design,” *IEEE Transactions on Software Engineering*, Vol. 24, No. 12. December 1998.

[7] Rosson, M. B., and J. M. Carroll, “Scenario Based Design,” *The Human-Computer Interaction Handbook*, 2002, pp. 1032–1050.

[8] Amyot, D., “Group Communication Server: A Scenario-Based Design Exercise,” University of Ottawa, 1998,. http://lotos.site.uottawa.ca/ucm/pub/UCM/VirLibGcs98/gcsJune1998.pdf.

[9] Bardram, J. E., “Scenario-Based Design of Cooperative Systems Re-designing a Hospital Information System in Denmark,” *Group Decision and Negotiation*, Vol. 9, 2000, pp. 237–250.

10] Blasch, E., P. Valin, É. Bossé, M. Nilsson, J. Van Laere, and E. Shahbazian, “Implications of Culture: User Roles in Information Fusion for Enhanced Situational Understanding,” *Int. Conf. on Info Fusion*, 2009.

CHAPTER 12

A Coalition Approach to High-Level Information Fusion

Dale A. Lambert (Australia), Steven Wark (Australia), Éloi Bossé (Canada), Luc Pigeon (Canada), Clinton Blackman (United Kingdom), and Michael Hinman (United States[1])

This chapter is concerned with the theory and practical implementation of high-level information fusion (HLIF). It relates to a collaborative program of work in HLIF that has been conducted between Australia, Canada, the United States, and the United Kingdom through The Technical Cooperation Program (TTCP) panel on information fusion. In this chapter, we bring together ideas presented in previous chapters, overview the Coalition Distributed Information Fusion Testbed (CDIFT), describe three vignettes, and explain methods for combining low-level information fusion (LLIF) object assessment with HLIF situation and threat assessment, user refinement, and sensor management.

Introduction

This chapter presents contributions from the four aforementioned nations, showing how the work has been integrated through a common scenario, which the panel proposes to subsequently offer as a benchmarking scenario to the higher-level fusion community. We outline developments in multination collaboration efforts for a common scenario, models and tools, and a demonstration of different nation's products running on the Coalition Distributed Information Fusion Testbed (CDIFT) of Chapter 6.

12.1.1 Vision

The remit of the TTCP panel on information fusion is to pomote collaborative research and experimentation between the member nations in the area of Information Fusion; however, the TTCP panel has embraced a broader vision. Currently, there is a tendency for data fusion proponents to belong to either a low-level or high-level information fusion, with the HLIF operating more as an aggregation of ideas than as a community. The demand for data fusion at all levels, meanwhile, increases at a considerable pace as the effects of the Information Age propagate. In response, the TTCP panel on Information Fusion is seeking to coordinate capabilities and build a strong sense of community.

12.1.2 Content

The TTCP panel on Information Fusion has established a sense of community by collaborating on both theoretical and practical pursuits. The publication of the first book by the TTCP panel (Bossè, Roy, and Wark)[1] reflects the theoretical collaboration. This chapter reports on practical collaboration.

Two ingredients have been essential for practical collaboration. The first has been the development of a common scenario as a context for collaboration, which allows participants to map their problems and solutions into a common environment. The second has been the construction of the CDIFT, which allows products from the contributing nations to be exercised concurrently.

Section 12.2 outlines the common scenario while Section 12.3 presents the CDIFT architecture. The remainder of the chapter highlights some of the contributions from the different nations. Section 12.4 summarizes Canadian scenario platforms, sensor models, trackers, and early HLIF capabilities. In Section 12.5, aspects of the U.S. Air Force Research Laboratory Fusion2+ product are presented. This is followed in Section 12.6 by an overview of a United Kingdom clustering approach. Section 12.7 offers some remarks about the Australian higher-level fusion implementations, followed by the Australian HLIF display technology in Section 12.8. The subsequent two sections

(Sections 12.9 and 12.10) highlight vignettes in urban operations and coalition search and rescue, respectively, that are being developed for inclusion into the scenario.

Scenario

This section presents a very short description of a vignette [2] designed to stimulate and test the high-level information fusion concepts and algorithms being studied by the TTCP panel. We adapted a scenario (context) called *Atlantis* in which vignettes can be developed to represent various defence and security problems, such as military strikes, combat search and rescue (CSAR), urban operations, cyber security, and harbor security.

Atlantis is a fictitious continent located in the North Atlantic Ocean, between Europe and Greenland. For the purpose of this scenario, the land areas of Iceland, Ireland, and the United Kingdom—Shetland excepted—do not exist. The shape of the continent looks like the continental United States (USCON) rotated by 90º with one third of its size. As shown in Figure 12.1, Atlantis is composed of six countries: Blueland, Orangeland, Redland, Brownland, Whiteland, and Greyland. The historical background to the crisis is summarized in [2]. Briefly, in the nineteen century, most of Atlantis—Greyland excluded—formed the Radobecan Empire until the end of World War I, when it was divided into the five current countries. The change of the borders, in addition to the different social, political and economic conditions, have brought disagreements between the countries.

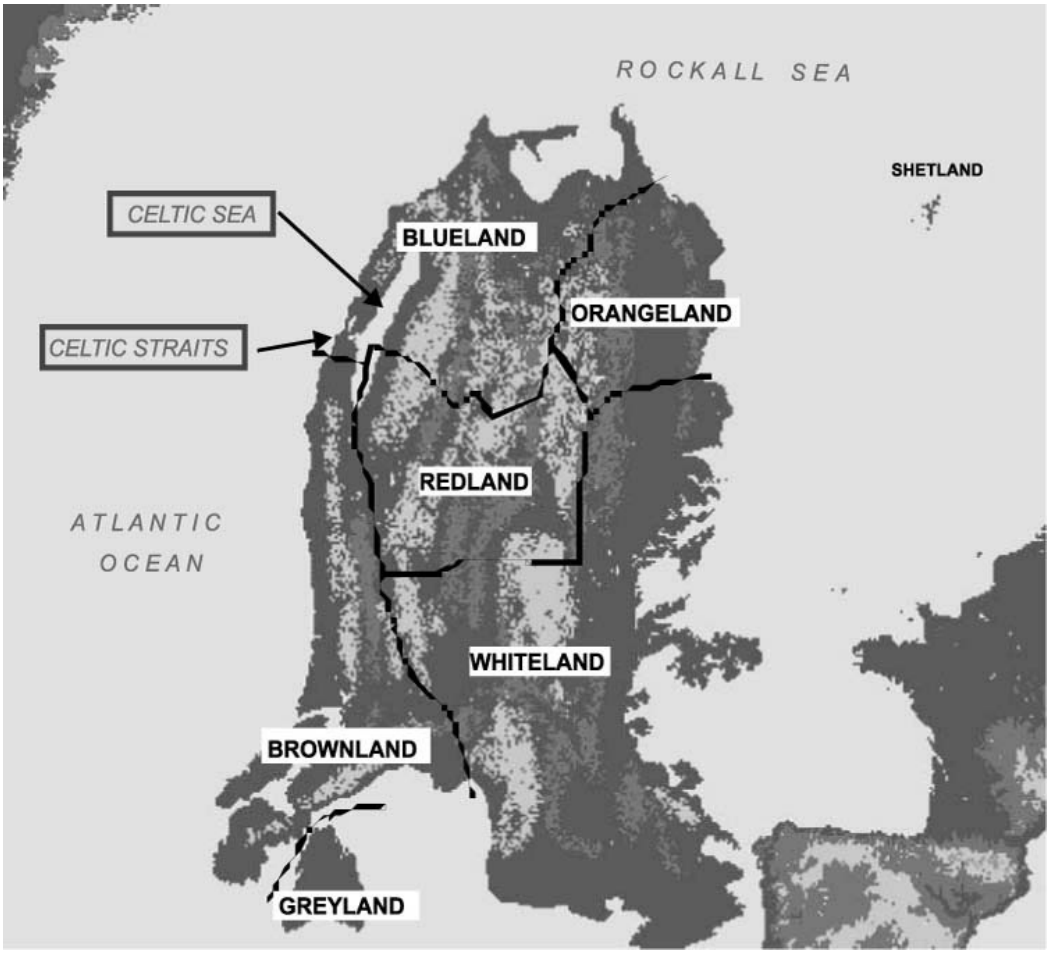


Figure 12.1 Location of the Atlantis countries.

One of the vignettes that has been developed is centered on a two-ship convoy being simultaneously attacked by submarine and aircraft off the Atlantis west coast. In order to complicate the assessment of the situation, the two surprise attacks occur while other activities are taking place. Those activities include commercial air traffic, maritime traffic, observation of whale migration, and various military operations, such as air surveillance patrolling and missile deployment. Moreover, one of the opponent countries has planned to take control of the Celtic Straits on that day. This vignette called "Military Strikes in Atlantis" describes specific military activities within an evolving context where various crises happened between the countries, as illustrated in Figure 12.2.

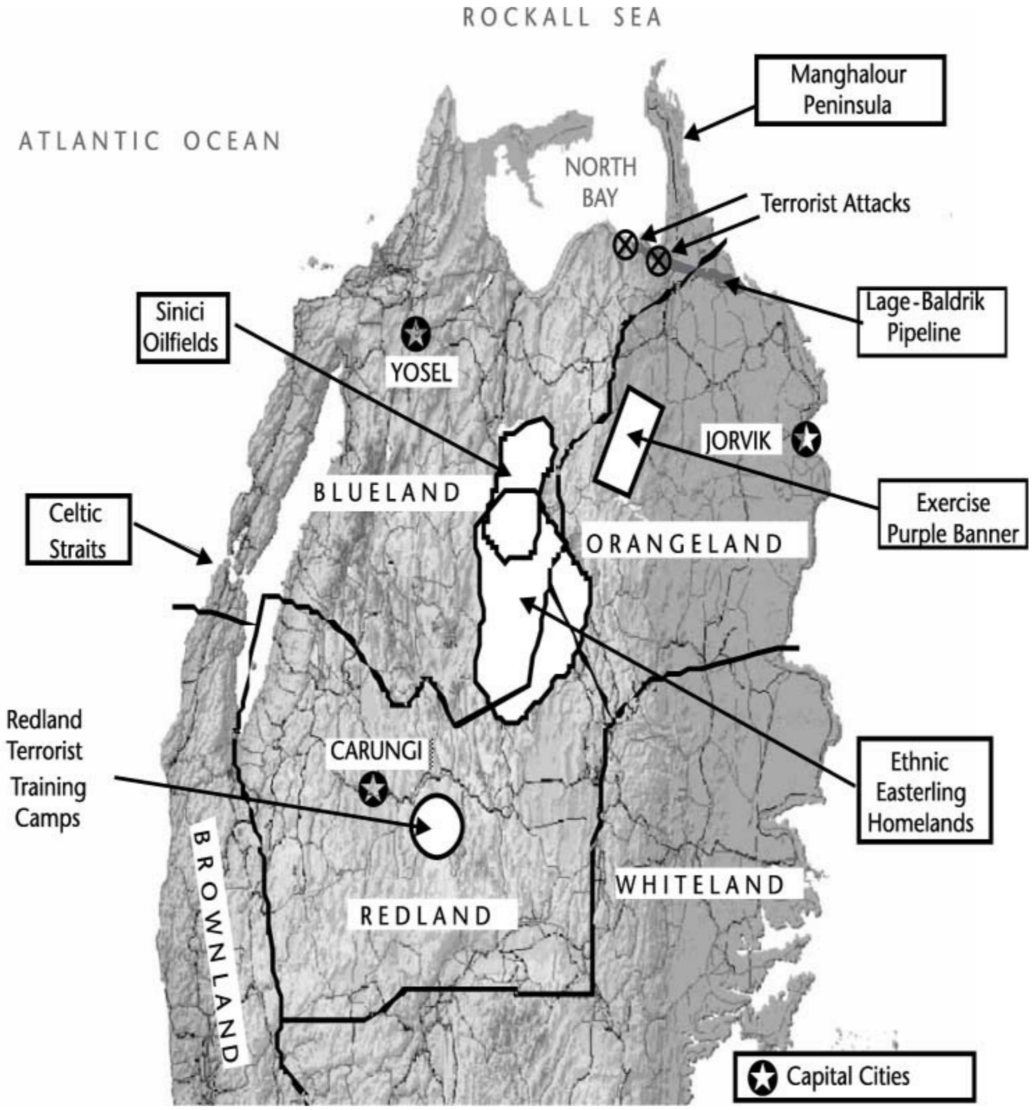


Figure 12.2 Location of the main conflicts prior to the current crisis.

The current crisis advanced when two mysterious explosions occurred in the Manghalour Peninsula. The first explosion hit one oil storage tank in Lage, and the second one destroyed a section of the Lage-Baldrik pipeline. Orangeland widely argued that Blueland could no longer protect the oil reserves that are crucial to them from terrorist attacks. A week later, Orangeland forces invaded the Manghalour Peninsula, taking control of its airspace and moving forward surface-to-air missiles (SAMs). They promptly neutralized Blueland's air defence systems around the North Bay and took control of the Bay's entrance, confining several of Blueland's vessels of war, especially fast patrol boat (FPB) and mine counter measure (MCM) vessels, near Breivik.

The UN Security Council (UNSC) progressively issued a set of resolutions to solve the crisis in northern Atlantis. The first resolution orders an immediate ceasefire and the withdrawal of Orangeland forces from Blueland. Further to, the refusal of Orangeland to comply with that resolution, a second resolution was approved requesting all states to prevent any trade with Orangeland. The UN Security Council announced a third resolution imposing an embargo on Orangeland and requested the Alliance Council for military assistance to restore international peace and security in northern Atlantis. Two North America countries decided to participate by sending a task group to enforce that resolution. The task group of ships, composed of 1 × Wasp class LHD, 1 × Burke class DDG, 1 × Iroquois class DDG, and 2 × Halifax class FFG, have left their home base toward the North Atlantic Ocean and Rockall Sea.

The vignette is composed of various actions or activities occurring in the same time frame. Those components include the following:

Redland Warships: description of the operations carried out by the Redland Navy in support of the planned air attack of the Celtic Straits. Those operations include the dispatching of vessels in the Atlantic Ocean off the Celtic Straits, the boarding of merchant ships going to Blueland seaports in the Celtic Sea, and the escort of merchant ships from Redland seaports through the Celtic Straits.
Alliance Convoy: description of the convoy composed of the cargo and Alliance Task Group leaving North America for Atlantis.
Commercial Air Corridors: description of the air corridors used by commercial airplanes for the transoceanic flights between Europe and North America and the domestic flights in the Atlantis continent.
Blueland Ground-Based Radars: description of the radar network used by Blueland Air Force for air surveillance and air traffic control.
Maritime Routes: description of the commercial maritime routes usually taken by the merchant ships going to or coming from the Celtic Straits.
Whale Migration: description of the platforms watching the humpback whales swimming off the Atlantis western coast towards the North Atlantic Ocean.
Missile Deployment: description of the missiles previously deployed by Redland in support of the air attack of the Celtic Straits.
Redland Airborne Surveillance: description of the air platform used by Redland to detect and track operations carried out beyond its border.
Submarine Attack: description of the attack of the convoy by an Orangeland submarine.
Air Attack of the Convoy: description of the attack of the convoy by two Redland fighters.
Air Attack of the Celtic Straits: description of the fighters sent by Blueland to counter-attack the Redland fighters and the attack performed by Redland to take control of the Celtic Straits. This second attack involves air, naval, and ground platforms.

CDIFT

The CDIFT is an initiative of the TTCP panel to establish a distributed, heterogeneous coalition environment to support development and evaluation of information fusion technologies and applications.

The CDIFT incorporates a synchronous simulation layer using high-level architecture (HLA) to support ground truth and sensor modeling, and an asynchronous notional Joint Task Force Information Grid (JTFIG) to represent real-time access to coalition information sources and sensor feeds. The U.S. Joint Battlespace Infosphere (JBI) is used as the primary medium for information exchange between coalition systems via a publish/subscribe model, as shown in Figure 12.3. Other technologies are also exploited, such as web services and the Australian open-source Avis[2] architecture as an agent messaging layer. Applications access these layers as their information requirements dictate.

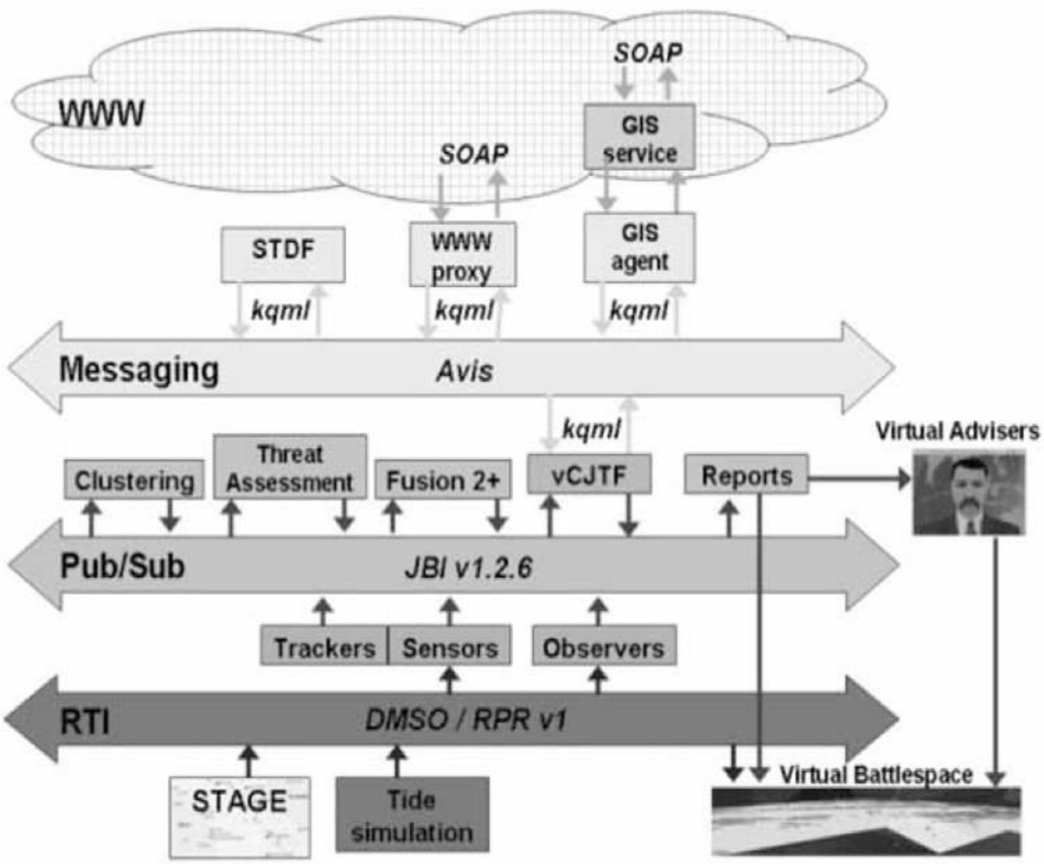


Figure 12.3 Architecture employed in the CDIFT provides a distributed, heterogenous coalition environment.

The functional components of the CDIFT follow:

1. Ground-truth generation is done using STAGE simulation software[3], which can

inject ground-truth into the scenario in real time, or be saved as data files for later playback.

2. Sensor modeling and tracking provides sensor and track reports that are published onto the JTFIG and are available to any application that subscribes to these reports. Both simplistic and realistic high-fidelity sensor and tracker models are available on the CDIFT.
3. Information fusion applications and services subscribe to the sensor and tracker reports and publish complete or partial information fusion products back on the JTFIG so as to be available to other applications.
4. Point-to-point communication between applications, subcomponents, or agents when processing information utilizes the agent messaging layer. This provides a level of abstraction that allows brokers to locate agents with the required capabilities, and multiple agents to negotiate an appropriate solution as the system complexity increases. The CDIFT is exposed as a web service using a web proxy. Agents register their capabilities with the web proxy, which then tasks them to satisfy web service requests.
5. User interfaces are provided by individual applications, and also by report subscriber services on the JTFIG. These can be deployed wherever they are needed, and are configured to only subscribe to reports that are of interest to the particular site, role, or context. The way the information is presented to the users is also configurable to support different roles, contexts, and user preferences.

Platforms, Sensor Models, and Trackers

The numerous agents of the vignette components listed in Section 12.2, such as sensors, platforms, weapons, and organizations, are exercised through high-level information fusion applications on the CDIFT infrastructure of Section 12.3, to help tell a story about what is occurring in Atlantis.

Detailed simulations are being used to represent sensors, such as radars, and platforms. We attempt to simulate Level 1 information fusion object assessments, such as tracks and identifications, since the object of our investigations is highlevel information fusion. Below are short excerpts of the way each component of the vignette is being simulated. All details are described in [2]. For the purpose of illustration only, a handful of examples are outlined here.

12.4.1 Redland Warships

On their way to carry out naval exercises in the Atlantic Ocean, Redland vessels of war —1 × FFG (Kotor class) and 2 × PCF (Rafael class: ex-Yugoslavia Koncar class)— have reported detecting submarine activities and found moored mines off the Celtic Straits. Redland claimed that the straits were no longer safe for the transit of merchant ships because of the likely presence of mines or submarines, and blamed Blueland for its conflict with Orangeland as being the main cause of the hazard in the straits. For that reason, Redland sent 2 × MCM (Birt class: ex-Yugoslavia Klanac class) and 1 × SSK (Elwood class: ex-NL Walrus class) ships to search for mines and submarines.

12.4.2 Convoy

A cargo ship carrying ammunition has just left a North American seaport in the direction of Bercelport (Celtic Sea) in Blueland, while the task group is going to the Rockall Sea (which is north of Atlantis). In order to protect the cargo from being boarded by Redland vessels, Blueland has requested that the task group escort it until they meet a Blueland Navy ship (1 × Descubierta class FFG) off the Celtic Straits. The frigate has deployed its towed array sonar and its CH-124 (Sea King Helicopter) ready to detect and hunt submarines. The frigate sails at 15 knots, 1 NM ahead of the cargo. The helicopter is equipped with ping sonar and 2 x Mk-32 torpedoes.

12.4.3 Commercial Air Corridors

Two types of air routes are considered in this vignette: the North Atlantic air routes used for commercial flights between North America and Europe, and the Continental Atlantis air routes used for air service between Atlantis airports. To make connections between European and North American eastern cities, airplanes have to fly over the northern Atlantic Ocean, and, consequently, the North Sea and the Atlantis continent, whereas European and North American western cities are usually linked through the Arctic. Figure 12.4 shows typical air routes used by airplanes to cross the northern Atlantic Ocean.

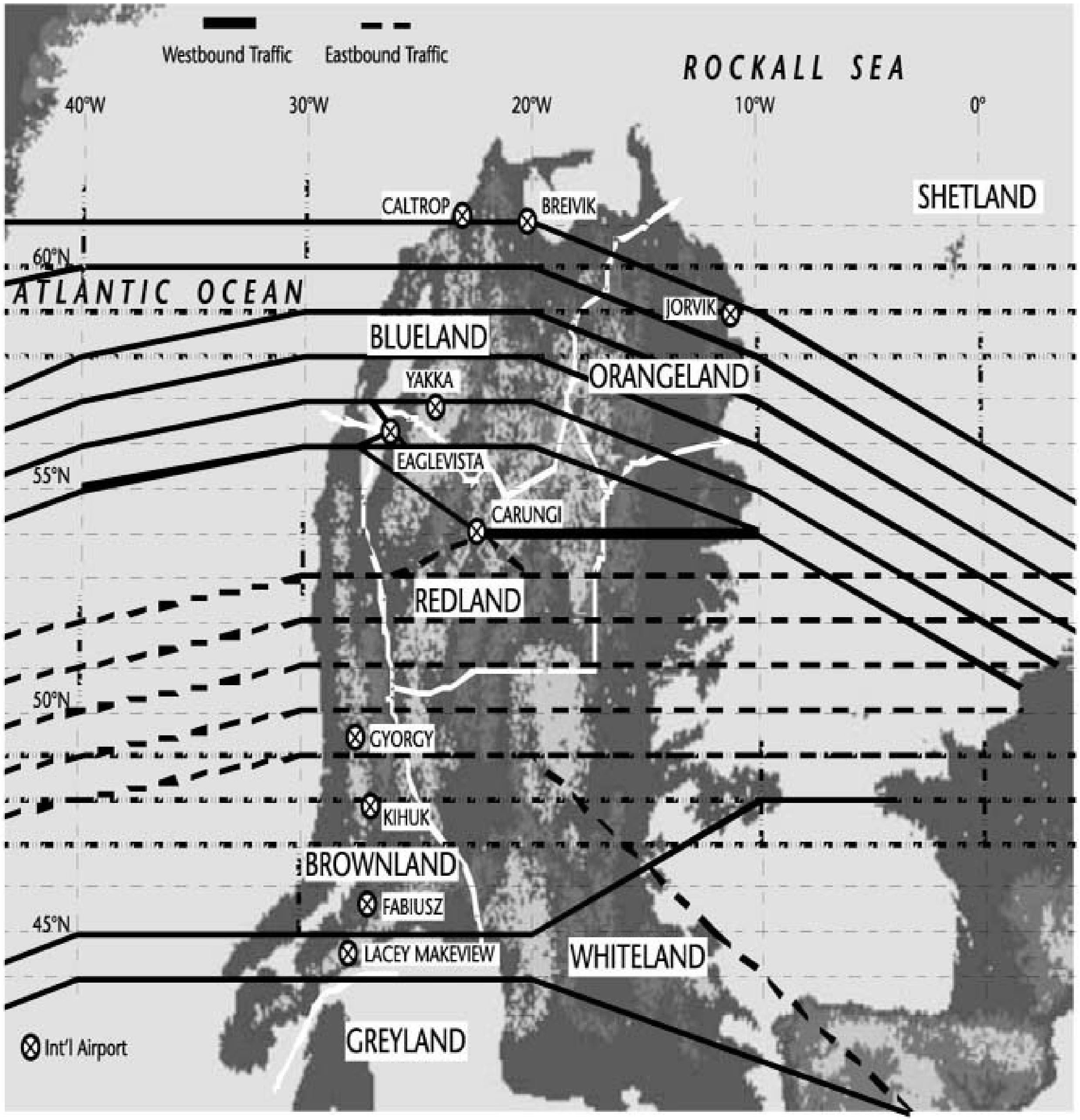


Figure 12.4 Typical routes of the North Atlantic airspace.

12.4.4 Blueland Ground-Based Radars

Blueland airspace is divided into three Air Defence Sectors: North-East, North-West and South, as per Figure 12.5. The Operations Centre of each sector controls all flying activities (military, commercial and civilian) in the sector. The South Sector, which is in charge of the Celtic Straits area, is bounded to the south by the Brownland and Redland borders, to the west by the Atlantic Ocean, and to the north by the following segment points: 58º30'N 27º00'W, 58º30'N 25º00'W, 58º00'N 21º00'W, and the junction of the borders with Redland and Orangeland (57º00'N 18º00'W). The Operation Center of the South Sector is located at Nellis air force base (AFB) (57º06'N 22º22'W).

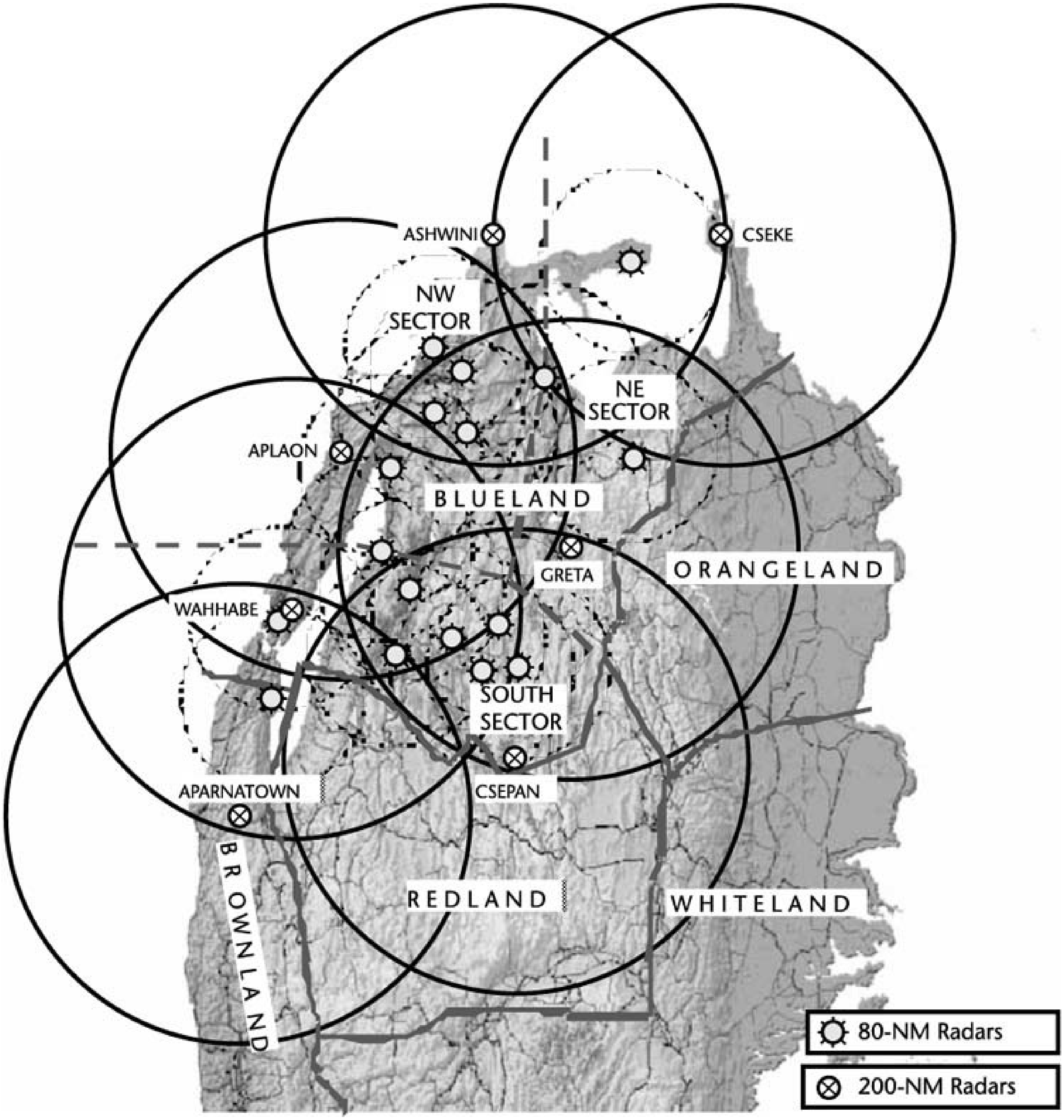
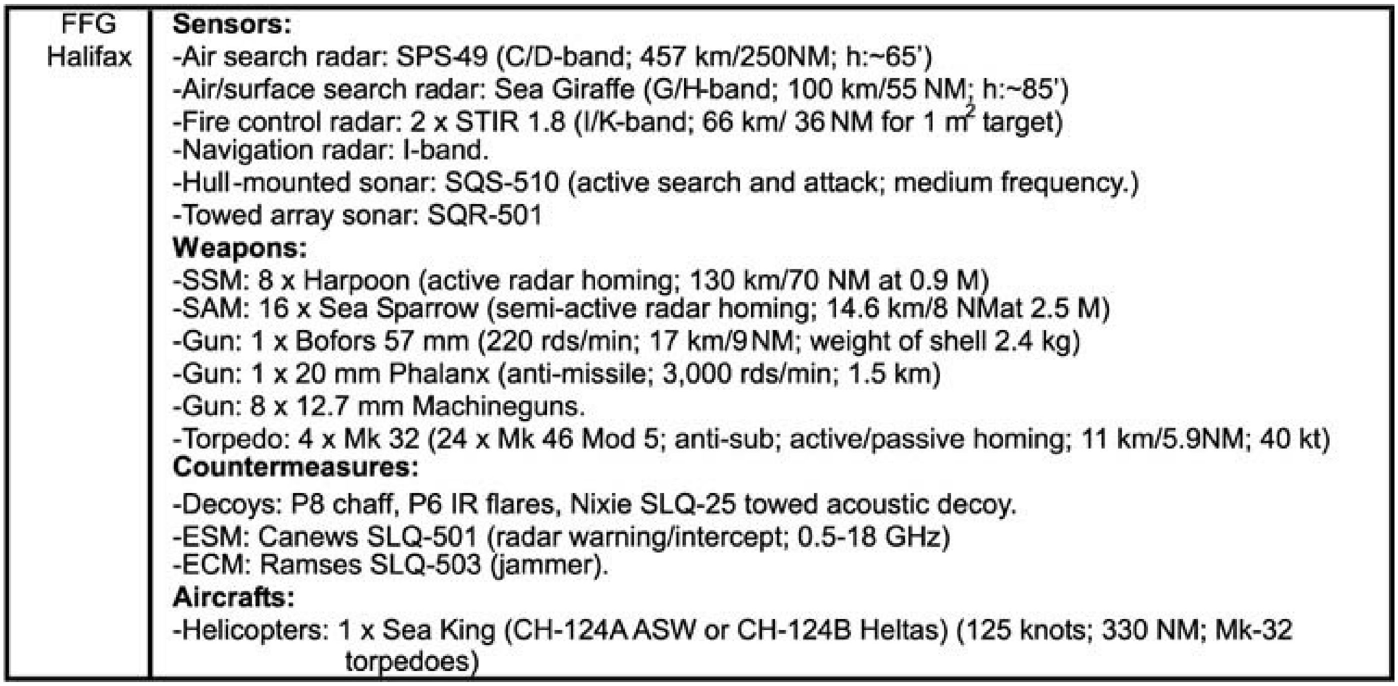


Figure 12.5 Location and coverage of the Blueland long-range and short-range radars.

12.4.5 Events and Order of Battle ORBAT

A similar description applies for other components of the vignette [2]. The exact time sequence of all events is in the simulation environment of CDIFT. In addition, each specific entity, such as a frigate, is simulated according to order of battle (ORBAT) information like that featured in Figure 12.6.



FFG Halifax	**Sensors:** -Air search radar: SPS-49 (C/D-band; 457 km/250NM; h:~65') -Air/surface search radar: Sea Giraffe (G/H-band; 100 km/55 NM; h:~85') -Fire control radar: 2 x STIR 1.8 (I/K-band; 66 km/ 36 NM for 1 m^2 target) -Navigation radar: I-band. -Hull-mounted sonar: SQS-510 (active search and attack; medium frequency.) -Towed array sonar: SQR-501 **Weapons:** -SSM: 8 x Harpoon (active radar homing; 130 km/70 NM at 0.9 M) -SAM: 16 x Sea Sparrow (semi-active radar homing; 14.6 km/8 NMat 2.5 M) -Gun: 1 x Bofors 57 mm (220 rds/min; 17 km/9 NM; weight of shell 2.4 kg) -Gun: 1 x 20 mm Phalanx (anti-missile; 3,000 rds/min; 1.5 km) -Gun: 8 x 12.7 mm Machineguns. -Torpedo: 4 x Mk 32 (24 x Mk 46 Mod 5; anti-sub; active/passive homing; 11 km/5.9NM; 40 kt) **Countermeasures:** -Decoys: P8 chaff, P6 IR flares, Nixie SLQ-25 towed acoustic decoy. -ESM: Canews SLQ-501 (radar warning/intercept; 0.5-18 GHz) -ECM: Ramses SLQ-503 (jammer). **Aircrafts:** -Helicopters: 1 x Sea King (CH-124A ASW or CH-124B Heltas) (125 knots; 330 NM; Mk-32 torpedoes)

Figure 12.6 Sample from the scenario ORBAT.

Fusion 2+

The U.S. Air Force Research Laboratory (AFRL) contribution to the CDIFT is the Joint Battlespace Infosphere (JBI) that is being utilized as the publish/subscribe environment by the other nations, along with the Fusion 2+ Testbed, which provides software tools for processing the text reports and analyzing them to determine the current situation. The Fusion 2+ Testbed has been developed to support information fusion research and development. As first discussed in [3] and [4], the Fusion 2+ Testbed is comprised of the following components: data collection, document parsing, and model analysis.

Initially, the analyst will specify a region of interest and identify specific items of interest, which will be used to develop or modify models. These models will then be used to drive the data collection. The data collection is performed via an AFRL developed meta-search engine that can simultaneously query and retrieve documents from multiple sources. These documents are then distributed for parsing. Natural language extractors are utilized to parse free text messages and/or documents. Formatted messages, such as Tactical Reports (TACREPs), are processed by the Generic Intelligence Processor (GIP) [5]. Model analysis tools then utilize the evidence provided to ascertain if any segment of a model is unfolding. The model analysis tool leverages graph theory, and searches the input graph, which is generated based on the evidence for any subset of the target graphs, developed by the analyst. Graph matches that exceed a specified threshold are provided as potential alerts to the analyst.

Introducing the Fusion 2+ Testbed into CDIFT required two modifications. The first is the development of synthetic open-source documents based on the North Atlantis scenario vignette that will be used as the inputs to the Fusion 2+ software. These documents are either in the form of TACREPs or news reports. The other modification entails the development of new models for the Model Analysis tool. Once again, these models are specified by the analyst based on their experience with the area of interest.

Indicators of Collective Behaviour

The United Kingdom's Indicators of Collective Behaviour (ICB) algorithm exemplifies the potential that CDIFT offers as an environment for evaluating fusion algorithms. ICB was not designed to be an operational tool, but rather to be used as a training aid for future commanders to reveal the sorts of future capability that would be enabled by fusion. For this role, the ICB algorithm was developed for DSTL's Wargame Infrastructure and Simulation Environment (WISE) facility, which is a formation level, land-based wargame used as an operational analysis model [6].

WISE can operate both as a wargame, in which military players command virtual forces interacting in a simulated environment, and as a testbed which operates without any human interaction. In a typical wargame, WISE might portray a complex warfighting environment with hundreds of maneuvering entities under the control of "Red" and "Blue" commanders, each of whom responds to his/ her perception of the situation, which in turn, is generated by the simulated intelligence surveillance, target acquisition and reconnaissance (ISTAR) entities under his/her control. In general, each commander's view of the battlefield is imprecise because there is a stochastic element in WISE's sensor models, and incomplete because of terrain, weather, and sensor range effects. The application of ICB offsets some of these effects, leading to improved situational awareness and a measurably improved military outcome.

12.6.1 Indicators of Collective Behaviour Algorithm

The proposition underlying the indicators of collective behaviour (ICB) algorithm is that maneuvering entities that are close together, moving at similar speeds, and/or moving in similar directions are likely to be acting collectively, and as such, might be posing a threat to a headquarters (HQ) or other deployed asset upon which they appear to be advancing. The algorithm has been fully described in [7], and it has three elements: identification of candidate clusters, assessing confidence, and inferring intent.

12.6.2 Identifying Candidate Clusters

Assuming a total of N battlespace entities have been detected and tracked (either in the CDIFT or in the WISE wargame) and that their positions, speeds, and headings are known. A fully-connected bidirectional weighted graph of these entities may then be drawn up in which the edge weights (EW_{ij}) represent dimensionless assessments of separation, speed, and heading:

$$EW_{i,j} = \left(\alpha \times \frac{d_{i,j}}{\max\limits_{i\neq j\in N}\{d_{i,j}\}}\right) + \left(\beta \times \frac{|\Delta s_{i,j}|}{\max\limits_{i\neq j\in N}\{\Delta s_{i,j}\}}\right) + \left(\gamma \times \frac{|\Delta h_{i,j}|}{\max\limits_{i\neq j\in N}\{\Delta h_{i,j}\}}\right) \tag{12.1}$$

where $d_{i,j}$ is the distance between entities i and j. Δs_{ij} and Δh_{ij} are differences in their speed and heading, and α, β and γ are constants which can be determined dynamically at run time to give greater or lesser emphasis to separation, speed, and heading criteria. Once the graph has been assembled, the Minimum Spanning Tree (MST) is calculated using Dijkstra's method, and this is subsequently broken down into forests using a Gaussian Parzen Window estimator to identify edges which should be cut. The result is a set of candidate clusters (Figure 12.7).

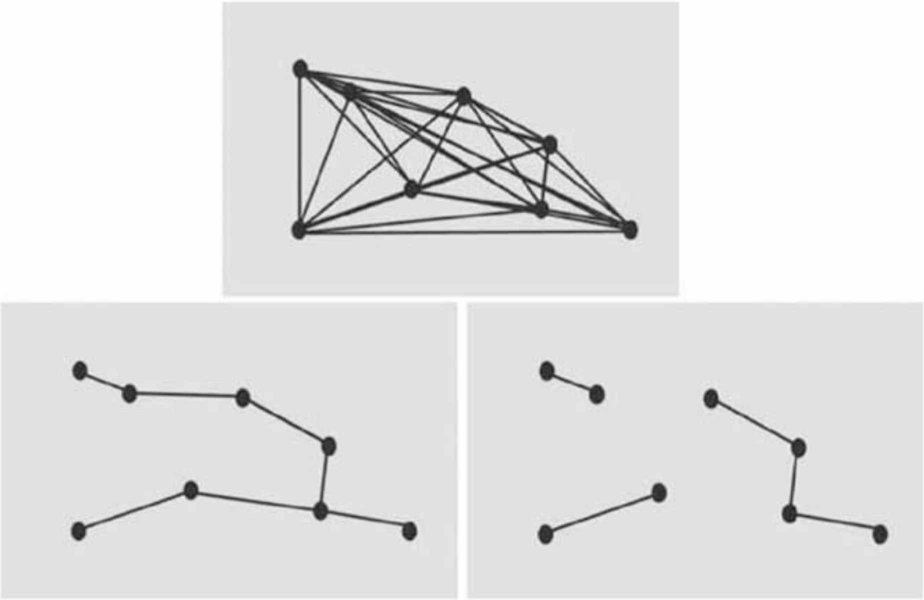


Figure 12.7 Example of an 8-node graph showing the minimum spanning tree and subsequent allocation into candidate clusters.

12.6.3 Assessing Confidence

Once candidate clusters have been identified, they can be monitored. In an operational system, an analysis of cluster parameters, over time would probably be the best way to assign measures of confidence in the coherence of each cluster. For WISE and CDIFT, a simpler approach has been adopted, based on the instantaneous similarity of speed and heading between group members. The basic idea here is that a candidate group, whose members display a small standard deviation in speed or heading, is more likely to be genuine than one with a large standard deviation. Normalizing by range and taking due care to avoid numerical artifacts associated with the apparent discontinuity between 0ºand 360º in heading gives a measure of object assoication *Confidence* defined as follows:

$$Confidence_C = \left(\frac{\beta \times \sigma(s_{i,j})}{\max\{s_{i,j}\} - \min\{s_{i,j}\}} + \frac{\gamma \times \sigma(h_{i,j})}{\max\{h_{i,j}\} - \min\{h_{i,j}\}} \right)^{-1} \tag{12.2}$$

Different confidence values may be used by commanders in deciding the priority order in which apparent threats will be addressed.

12.6.4 Inferring Intent

The point of CDIFT is that a variety of different information sources, including kinematic criteria and natural language reports, may be fused in order to infer intent. Within the ICB algorithm, an initial indicator of intent is provided by a simple analysis of the overall direction that a group is taking in relation to known targets, in this case the locations of important Blue entities, such as HQs. A pragmatic approach for this has been developed that is quick and easy to compute, based on the angular difference in bearing, θ, between a target and the mean heading of a group. Shown in Figure 12.8, the “Target Commitment Function,” $\tau(\theta)$, is defined as $\tau(\theta)=0.5[1+\cos^{2n+1}(\theta)]$, where θ is the difference in heading between that of designated target and the mean heading of the cluster and n is a free parameter, designed to give an appropriately shaped functional form (in initial work, n was set to a value of 10).

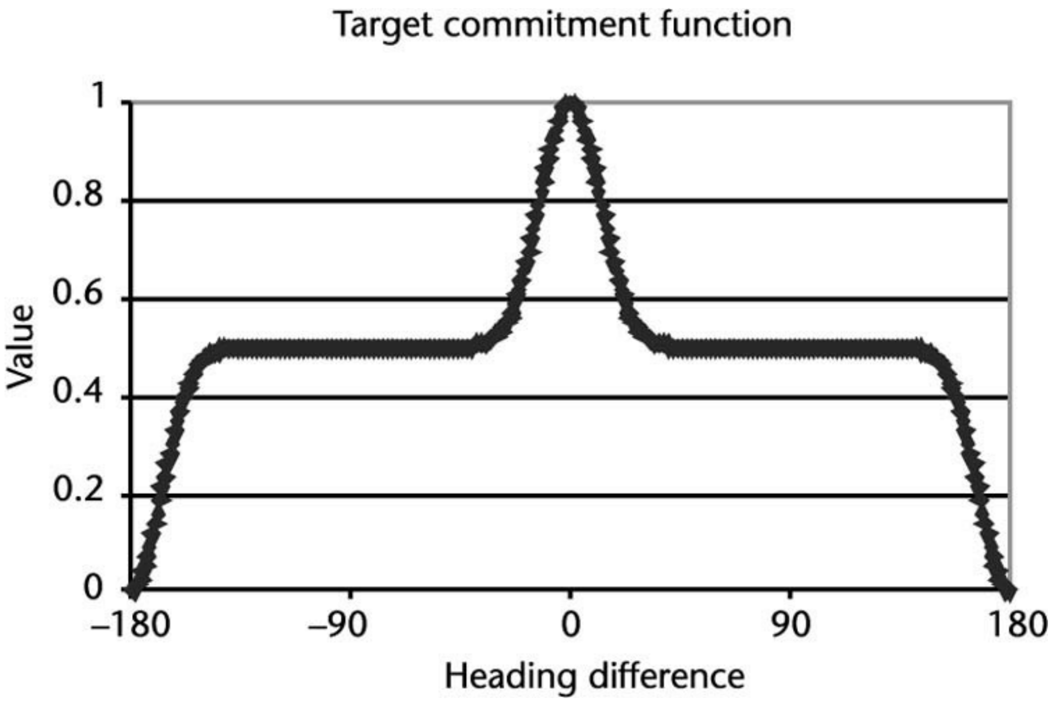


Figure 12.8 Target commitment function for n=10.

The advantage of this formulation is that τ lies between 0 and 1, and it is essentially tri-valued, with τ=1 indicating a cluster that is moving towards a target, τ=0 one that is moving away and τ=0.5 representing the undecided case (Figure 12.8). Exact values of τ may be used to help Blue commanders prioritize between different threats and make appropriate force allocations.

12.6.5 CDIFT Application

In the North Atlantis “Military Strikes” scenario, there are a number of entities that are genuinely moving together, such as the F16 fighters and shipping convoys. These are readily identified by ICB, giving confidence that more complex behaviors of other CDIFT vignettes will also be correctly discovered and, at the same time, providing a diverse set of situations within which the ICB can access confidence, and intent indicators, and other parameters.

The successful transition from a land-based training aid to a marine-based operational tool has demonstrated the flexibility of the CDIFT testbed as well as providing valuable opportunities for refining the ICB algorithm.

STDF Model

The Australian State Transition Data Fusion (STDF) Model [8–10] was developed as a unifying model across the JDL levels of fusion [11], as illustrated in Figure 12.9.

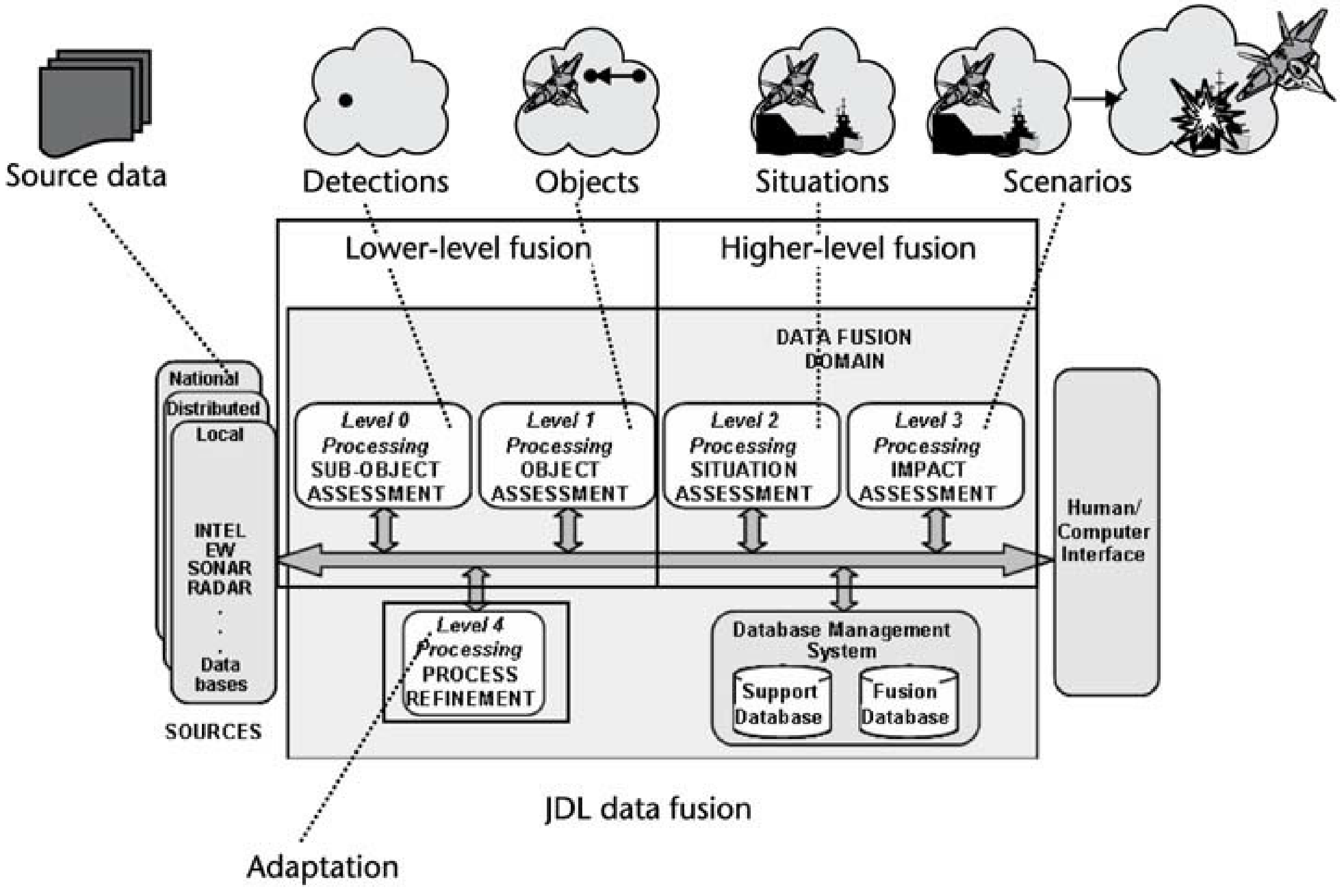


Figure 12.9 The JDL model of data fusion.

The STDF model is based on two premises:

1. At each level of fusion the world can be assessed in terms of states and transitions between those states. For example, Level 1 states are state vectors; Level 2 states are states of affairs; Level 3 states are scenario states; with objects, situations and scenarios being the sets of respective transited states over time.
2. At each level of fusion, the fusion process adheres to the same pattern of behavior, but the nature of the content changes. Figure 12.10 illustrates the common pattern of behavior. For example, registration is *coordinate* registration at Level 1; *semantic* registration at Level 2; and *situation* registration at Level 3.

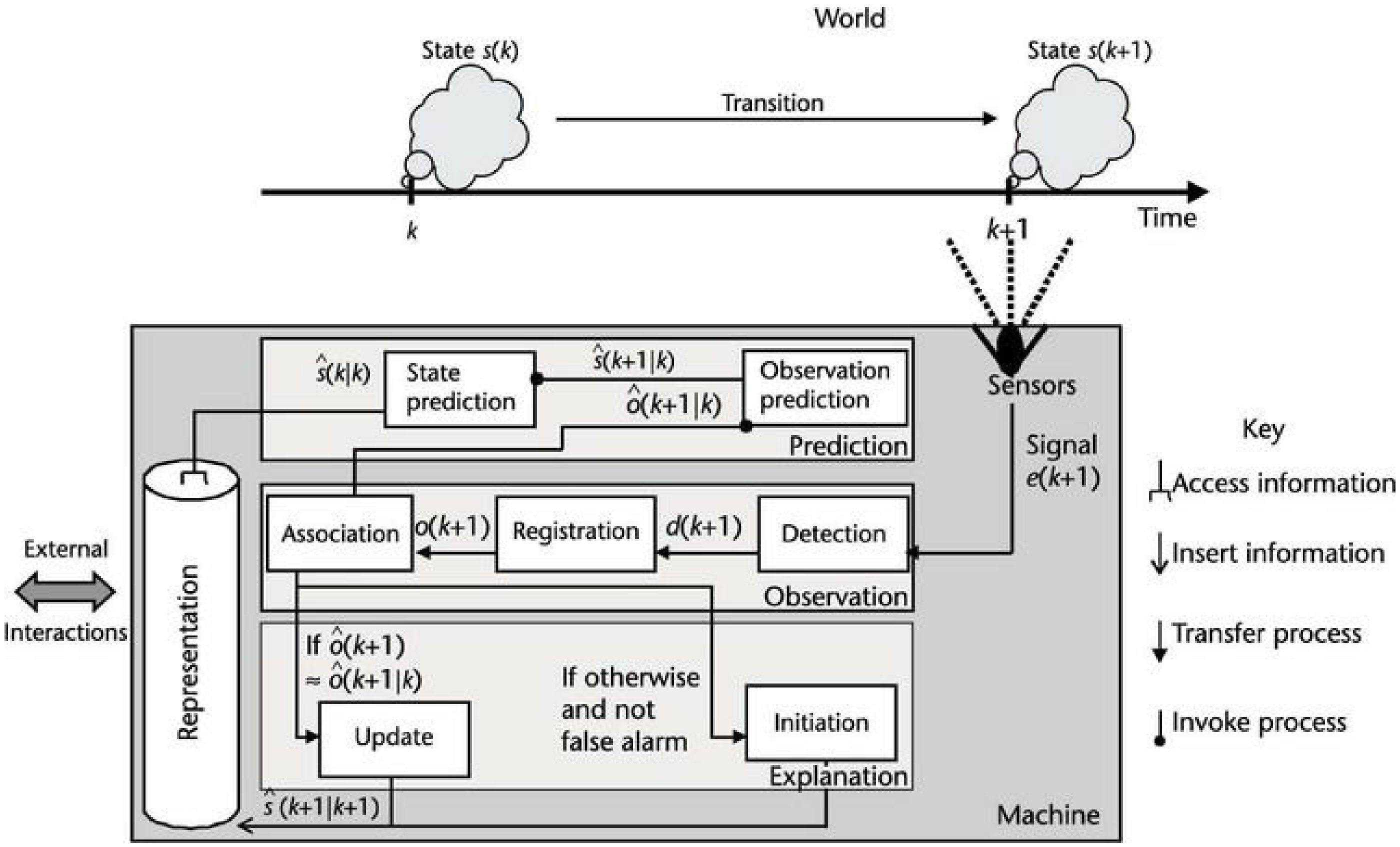


Figure 12.10 STDF model.

This section of the chapter focuses on implementation of the STDF model for higher-level information fusion in the context of the North Atlantis scenario.

12.7.1 State Representation

Figure 12.11 presents the STDF Model for Level 2 situation assessments.

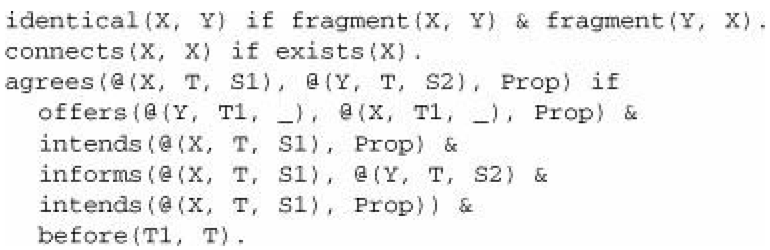
At Level 2, the world is understood in terms of situations as transitions between states of affairs, with states of affairs expressed as a set of statements about the world in a formal language associated with a formal logic. The inclusion of a formal logic allows the semantics of the interpretable symbols in the formal language to be defined or constrained. The choice of interpretable symbols in the formal language depends upon the nature of the domain under consideration.

For the military oriented North Atlantis vignette, the Mephisto framework [10, 12, 13] has been used to identify the basic interpretable symbols. Mephisto characterizes the world through five tiers and identifies concepts and associated interpretable symbols for each tier. Military events in the world typically involve all five tiers concurrently:

- Social: aligns, agrees, possesses, commands, and so forth.
- Cognitive: believes, expects, prefers, perceives, and so forth.

- Functional: senses, strikes, informs, moves, and so forth.
- Environmental: air, water, upland, outer space, and so forth.
- Metaphysical: exists, identical, before, connects, and so forth.

Logical constraints specify the meaning of each of these terms. Implementation of these logical constraints then constrains the machine's interpretation of those concepts. Some implemented examples of logical constraints include:



```
identical(X, Y) if fragment(X, Y) & fragment(Y, X).
connects(X, X) if exists(X).
agrees(@(X, T, S1), @(Y, T, S2), Prop) if
  offers(@(Y, T1, _), @(X, T1, _), Prop) &
  intends(@(X, T, S1), Prop) &
  informs(@(X, T, S1), @(Y, T, S2) &
  intends(@(X, T, S1), Prop)) &
  before(T1, T).
```

The semantic constraints facilitate both numerical calculation and abstract symbolic reasoning. The distance(C1, C2, D) predicate will calculate the great circle distance D between coordinates C1 and C2. Knowing before(T1, T2) and before(T2, T3) is sufficient to deduce before(T1, T3), even when numerical times for T1, T2 and T3 are not known.

12.7.2 Observation

At Level 2, observation comprises object assessment, semantic registration and propositional association (Figure 12.11). In the North Atlantis context, object assessments are the tracker outputs described in Section 12.4. These are typically vectors of the form:

<id, time, <x, y, v_x, v_y>, lat_ref, long_ref, P, type>

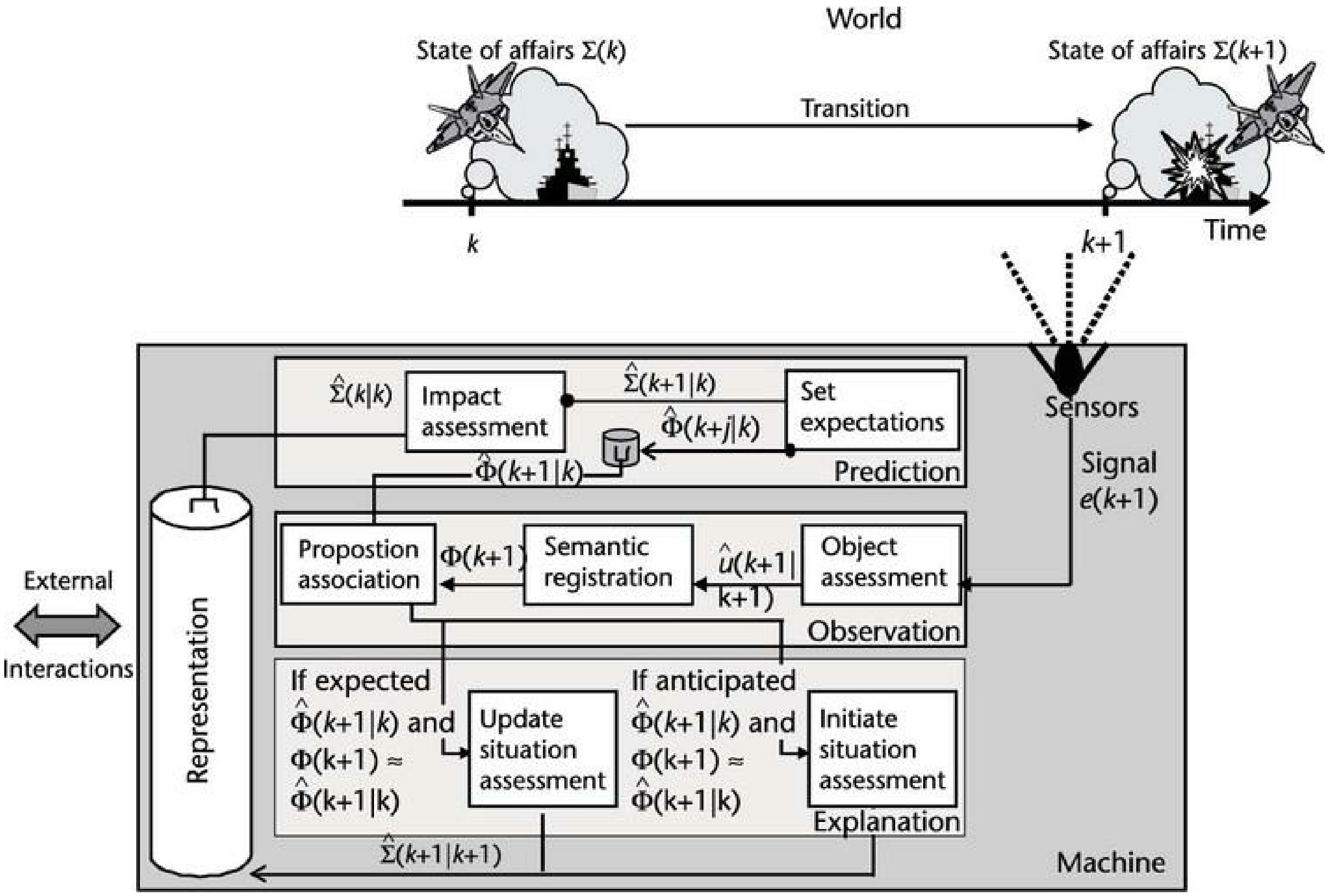
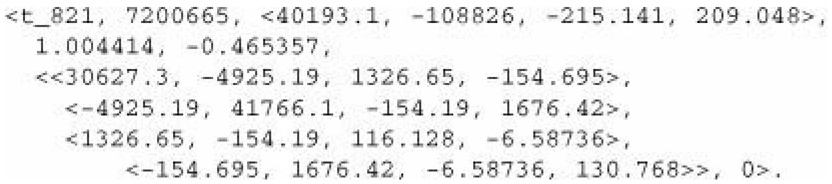


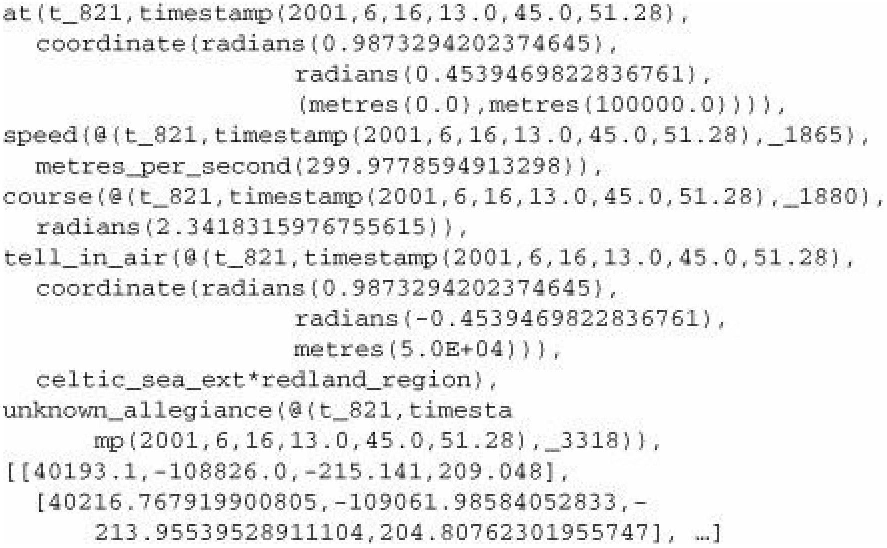
Figure 12.11 STDF model for situation assessments.

where: id is a unique identifier; time is a relative time; x is position in the x direction; y is position in the y direction; v_x is speed in the x direction; v_y is speed in the y direction; lat_ref and long_ref are the relative source reference coordinates; P is a covariance matrix; and type is a classification between air, surface, subsurface or unknown. The following is a sample object assessment vector:



```
<t_821, 7200665, <40193.1, -108826, -215.141, 209.048>,
  1.004414, -0.465357,
  <<30627.3, -4925.19, 1326.65, -154.695>,
    <-4925.19, 41766.1, -154.19, 1676.42>,
    <1326.65, -154.19, 116.128, -6.58736>,
        <-154.695, 1676.42, -6.58736, 130.768>>, 0>.
```

The semantic registration process accepts object assessments and translates them into formal sentences supported by the Mephisto framework. The following is the semantic registration output of the previous sample vector:



```
at(t_821,timestamp(2001,6,16,13.0,45.0,51.28),
  coordinate(radians(0.9873294202374645),
                   radians(0.4539469822836761),
                   (metres(0.0),metres(100000.0)))),
speed(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),_1865),
  metres_per_second(299.9778594913298)),
course(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),_1880),
  radians(2.3418315976755615)),
tell_in_air(@(t_821,timestamp(2001,6,16,13.0,45.0,51.28),
  coordinate(radians(0.9873294202374645),
                   radians(-0.4539469822836761),
                   metres(5.0E+04))),
  celtic_sea_ext*redland_region),
unknown_allegiance(@(t_821,timesta
      mp(2001,6,16,13.0,45.0,51.28),_3318)),
[[40193.1,-108826.0,-215.141,209.048],
  [40216.767919900805,-109061.98584052833,-
      213.95539528911104,204.80762301955747], …]
```

This registers the Level 1 object assessment in a form that can be reasoned with by Level 2 processes. If the user wishes to receive updates on any semantically registered information, then the natural language generation code can be applied. When applied to the previous example, it generates the following output:

> With some degree of uncertainty, at the time of 13.0 hours 45.0 minutes and 51.28 seconds on the 16th day of June 2001, t_821 is at location 56.56 degrees latitude, -26.0 degrees longitude, with an altitude between 0.0 metres and 100000.0 metres. It has a speed of 299.97 metres per second, has a course of 134.17 degrees, is of unknown allegiance, and is in the airspace over Redland's Celtic Sea.{end ext}

English accounts of semantic registrations have previously been disseminated to the users through virtual advisers (see Section 12.8).

Semantic registration is a value adding process that draws on considerable domain knowledge couched in Mephisto terms. Domain knowledge covers geography, both in abstract region connection calculus terms and through access to a Geographical Information System (GIS) system; the capability and disposition of available forces; the political alliances; and political intent.

Under the cognitive model outlined in Section 12.7.3 below, semantically registered observations are classified as perceptions. The propositional association challenge is to reconcile the incoming perceptions with expectations and anticipations formed through the prediction process. Propositional association superficially mirrors Level 1 data association in that it involves gating, prediction, scoring, and assignment. Gating is used to hash candidate perceptions into different clusters, but extends beyond data association gating by providing both a logical and regional basis for gating. The prediction process gathers the candidate expectations and anticipations. In low uncertainty contexts, the scoring and assignment steps reduce to unification based pattern matching. In higher uncertainty environments where multiple perception to expectation associations are possible, a scoring process is used to rate possible associations before the assignment process selects pairings, while possibly allowing multihypothesis options to proceed. Mathematics for this can be found in [9].

12.7.3 Prediction and Explanation

The higher-level STDF process is being implemented within the ATTITUDE TOO cognitive architecture, which has been written as a successor to the ATTITUDE cognitive architecture [14]. An ATTITUDE TOO agent's long-term memory comprises assertional memory consisting of semantic and epistemic long-term memory, and episodic longterm memory. The semantic memory holds the logical constraints discussed in Section 12.7.1. The epistemic memory contains the declarative domain knowledge alluded to in Section 12.7.2. Episodic memory consists of cognitive routines, each comprising a goal and a behavioral recipe for achieving that goal, expressed as a network of propositional attitude instructions. Semantic memory delivers meaning; epistemic memory delivers the "know that" domain knowledge; and episodic memory delivers the "know how" domain knowledge.

The prediction and explanation steps are primarily managed through cognitive routines, supplemented by the assertional memory. A simplified example of a cognitive routine for monitoring a ship traversing a sea lane is featured below:

```
   routine(traversing_sea_lane),
     ^(believe(i, entering_sea_lane),
       (Number_Missed_Updates is 0),
       (Max_Missed_Updates is 3),
      *( ^((Number_Missed_Updates =< Max_Missed_Updates),
             add_time(When, timeperiod(0.0,0.0,0.0,1.0), Expiry),

         line_segment(sea_lane(Lane), _, Terminal),
        +(^(expect(i,
                  one_of([ updated_along_sea_lane,
                             updated_across_sea_lane,
                      updated_stopped_on_sea_lane,
                      updated_enter_sea_lane_nexus]), by(Expiry)),
           +(^(not_believe(i, updated_enter_sea_lane_nexus),
                     intend(i, traversing_sea_lane_nexus,
      priority(0.9)),
                    disapprove),
               ^(not_believe(i, updated_along_sea_lane),
                    desire(i, updated_along_sea_lane)),
               ^(not_believe(i, updated_across_sea_lane)
                     desire(i, update_across_sea_lane)),
               ^(not_believe(i, updated_stopped_on_sea_lane),
                     desire(i, update_stopped_on_sea_lane_nexus)))),
      (Number_Missed_Updates
is Number_Missed_Updates + 1)),
      When is New_When)),
        +(^((Number_Missed_Updates > Max_Missed_Updates),
            believe(i, exited_sea_lane)),
          succeed)).
```

The example routine monitors transitions between entering_sea_lane, along_sea_lane, across_sea_lane, stopped_on_sea_lane, and enter_sea_lane_nexus states of affairs. The routine illustrates expectations being set for the last four of these states of affairs during execution, and these are reconciled with perceptions using the proposition association process discussed in Section 12.7.2. This particular example uses uncertainty to perform the association process, but does not carry the uncertainty into the routine and does not pursue a multi-hypothesis approach, though both can be done. The expectation signifies the prediction process while the beliefs represent the explanation process. In that respect the routine is particularly simple. Higher-order routines, such as predicting the likely final destination and time of arrival, draw on lower level routines like the one illustrated. Cognitive routines also typically deal with the interacting behavior of multiple objects, not just a single object as in the simple illustration.

Higher COP

The common operating picture (COP) is widely used to support situational awareness, providing a 2D dots on maps display that shows where entities are located in the battlespace relative to various geospatial features, such as the outputs of lowerlevel fusion. However, the COP does not support the comprehension and projection aspects of situational awareness— the outputs of higher-level fusion systems. The COP leaves the cognitive load on the user to interpret the picture displayed (Chapter 9). Furthermore, it provides no aid to achieving shared situational awareness for users in diverse roles and operating domains; how they interpret a common picture will be influenced by their individual context. While the COP may be sufficient for lower-level fusion, a context-sensitive higher COP (HiCOP) is needed to display the outputs of higher-level fusion systems.

Establishing and maintaining context is a fundamental requirement for information fusion [15]. Given the incomplete information provided by sensors, context is a necessary consideration for object detection and classification. Context is even more important when attempting to understand and act on the behavior of objects in the environment, as it depends on abstractions that cannot be directly measured. Failure to recognize context can inadvertently bias the interpretation of a situation with potentially tragic consequences[4].

Context is just as important for conveying information to the users of a Command and Control (C2) system. If context is not clearly established, users will need to reconstruct it from tacit knowledge and background information, and so their interpretation of a COP will depend on their current roles and situations. By clearly establishing context, users can rapidly assimilate the information provided, have a greater likelihood of correctly interpreting the situation, and share a common interpretation with other users.

Sensor fusion can be thought of as establishing the data about the entities in the environment, while higher-level fusion is about establishing the story behind the data. Thus, achieving situational awareness for the users of an information fusion system is akin to storytelling [16], which provides a compelling mechanism for describing complex and contextually sensitive relationships. Television news services provide a highly successful example of storytelling which incorporates multimedia to convey situation awareness about complex relationships in a local and global context. They establish context through the use of multimedia content such as imagery, video, and graphics, and use narrative to assist comprehension of a situation and its consequences.

The daily briefing in military command centers fulfills this role, but the scope and timeliness of these briefings is predicated on the production process. A better model would be to provide television-style multimedia briefings, or multimedia narrative, on demand for any situation as it develops. It is not practical to have a television production team assembling briefings on this scale, thus an automated system is needed. Storytelling is generally considered to be a uniquely human ability, but there is a growing body of work [17] that is demonstrating how real-time animated characters can provide training outcomes similar to humans.

Real-time animated characters, dubbed virtual advisers, shown in Figure 12.12, have been developed by Australia's Defence Science and Technology Organisation (DSTO) to act as automated storytellers that provide multimedia narrative on demand. The constrained format needed for military-style briefings provides a niche well-suited to the current technological limitations. Virtual advisers can interact with users using natural language and text-to-speech technologies, and can present multimedia content. Virtual advisers can provide additional context such as importance, confidence, and urgency through non-verbal cues. Appearance, facial expressions, gestures, behavior patterns, and voice prosody can all be used to provide contextual cues to the users. Incomplete information can be conveyed by selecting different multimedia elements to represent different levels of abstraction. Trust is another important factor that influences user engagement and confidence in the information presented [18]. It is important to manage the relationship between the users and the virtual adviser so that context can be conveyed without disrupting the users' trust in the virtual adviser.

Figure 12.12 One of DSTO's virtual advisers.

Virtual advisers provide a generic capability to provide multimedia narrative that can be rendered on demand. However, for military situation awareness a capability for displaying geospatial context is also required. A virtual battlespace capability provides this, as shown in Figures 12.13 and 12.14. It is a photorealistic 3-D geospatial representation of the battlespace, using real-time track feeds and digital terrain, imagery, and maps provided by web services. It can convey a richer and more up-to-date representation of the area of operations than a traditional map. Sensor envelopes, air corridors, and engagement zones can be displayed in 3-D. Multimedia annotations can be dynamically added to highlight objects and relationships of interest, and associate expansion information with the entities in the scene. This provides a rich capability for multimedia narrative when orchestrated with a virtual adviser.

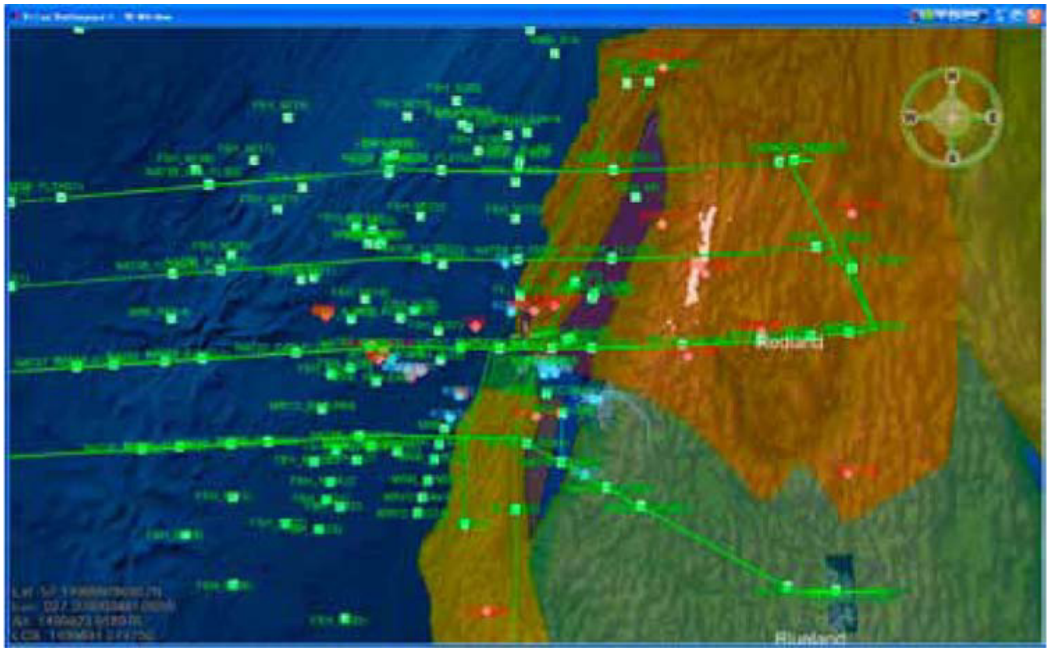


Figure 12.13 The Virtual Battlespace smoothly transitions to and from a dots-on-maps display to a virtual reality scene when appropriate.

Figure 12.14 The Virtual Battlespace provides a photorealistic 3-D geospatial display incorporating multimedia annotations.

Correction of errors, or content refinement after the fact is often difficult as more information becomes available. This can be facilitated by managing the users' trust in particular content. Untrustworthy characters can be used to present information where the reliability of the information is low, but the risk associated with ignoring it is high. Different media may also have implicit levels of trust associated with them; for example, text is considered a more reliable source than a virtual adviser [19]. By selecting media with lower associated levels of implicit trust, correction of the users' situation awareness when more reliable information is available can be facilitated. To avoid unwanted associations in the Virtual Battlespace, one approach to dealing with uncertain information is to use models that are not identified with any real system. For example, an unidentified submarine could be represented by a fictitious model that bears no resemblance to any existing submarine, as shown in Figure 12.15.

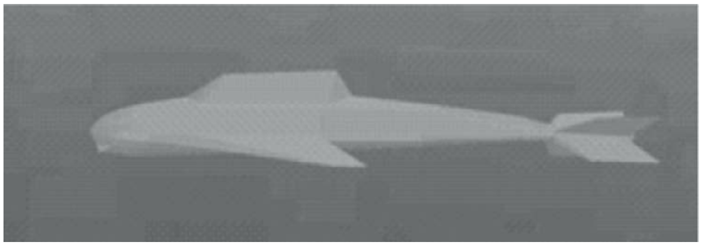


Figure 12.15 An unidentified submarine may be represented by a fictitious model to avoid unwanted associations.

The effective use of multimedia narrative as a higher COP depends on the ability of a computer system to generate and assemble multimedia components to convey a coherent, informative, and interesting message to the users. This requires, in some sense, the ability to emulate human storytelling expertise, and capture the director's art. This is a challenging requirement, but it is important to note that the aim of this work is only to provide some capability to enhance situation awareness. Basic templates for layout of media, and simple heuristics for media selection, could be used to assemble a basic multimedia presentation to aid the decision makers' comprehension of the situation and its consequences.

Further refinement of high-level information fusion and more intelligent multimedia selection and layout is required to realize an effective fully-automated system for a higher COP. The CDIFT provides an ideal environment in which to further develop and trial these capabilities, with virtual adviser and virtual battlespace technologies playing a significant role as situational awareness displays.

In the CDIFT, summary reports generated by fusion applications, in particular the virtual Commander Joint Task Force (vCJTF) agent, are published over the JTFIG. A number of report subscribers select only those reports relevant to the particular role of the site, priority and context. Based on the information content, context, and user preferences; these reports can be rendered in a number of ways: documented as a text output; presented by the virtual adviser; or displayed in the virtual battlespace, as shown in Figure 12.16. Currently, the content presented in this way included: simple summaries of the numbers of entities observed; the probability of association of maritime tracks with sea lanes; the probability of association of commercial aircraft based on air lanes; the operation of ground-based radars; clustering of tracks based on their kinematics; and prioritized threat assessments of air tracks. This capability is being extended and refined further as the CDIFT develops and more applications are incorporated.

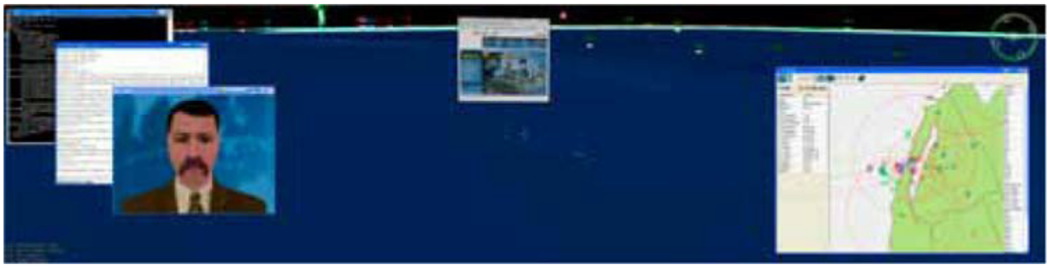


Figure 12.16 Example of how virtual advisers and virtual battlespace are used in the CDIFT.

Urban Operations

One of the best environments to envision higher-level fusion is the urban environment. Urban operations push toward the limits, combining army, navy, air, joint, and civil powers together, and ranging from the strategic to the tactical levels [20]. Global urbanization over the next 20 years will create an increasingly demanding operational environment for military forces to operate in. Moreover, these areas will take on importance too, with future opponents choosing to fight in urban areas to offset tactical advantages of more sophisticated forces. In many cases, simple avoidance of urban areas because of challenges created by a multidimensional battlespace, and the presence of noncombatants and complex infrastructures, is not an option. An army must be capable of fighting and winning throughout the range of conflict and in an environment where decisive action is required. Any force that cannot operate effectively in both urban and expanded battlespaces will be severely restricted in its future responsiveness [21]. Historical and contemporary operations implying urban environments are considered as complex, if not extreme, cases, such as Stalingrad, Hue, Mogadishu, Grozny, Bagdad, and Kandahar. They imply a wide range of resources from the whole spectrum of defense and civil services, plus NGO and media, and interactions at all levels of command. Even the study of counterinsurgency is closely linked with urban operations since they generally share the same theatre of operations.

The TTCP panel on information fusion is currently working on an extension of its basic scenario to include urban operations [22]. The first step has been to include—almost integrally—an existing scenario developed for a Kingston Staff College (Canada) exercise held in 2002 called Urban Challenge 2025 [21]. Afterwards, a NATO Research and Technology (RTO) study enriched it with tactical-level vignettes [23, 24]. To define vignette enrichments, military personnel and scientists wargamed the original Urban Challenge scenario in more specific contexts. These contexts were limited to three mission types: crisis response operation, defensive operation, and offensive operation. All of these developed and played from a C2 point of view.

There is a work in progress to include this enriched urban scenario within the panel-basic scenario story. This will most likely be translated into an ethnicbased urban conflict within Whiteland involving the descendents of Greyland immigrants. This group would require more autonomy for their province, and the conflict would occur within urban theaters.

New iterations of the urban vignettes are already forecasted. One proposal is to include two scenarios defined by the Joint Forces Command–Urban Operations Office (United States) [25]. The first is called “Counterinsurgency in Port Lewis, 2015–2021,” and covers a domestic counterinsurgency context resulting in bacterial meningitis and influenza outbreaks. The second is called “The Attack on Qabus 2027” and covers a foreign urban operation context.

Finally, parallel work is currently being performed to identify significant urban vignettes at the tactical level to act specifically as testbeds for HLIF discussions and concepts validation. Drafted vignettes are now characterized by the large spectrum of information fusion possibilities they offer (e.g., from signal recognition to multidimension situation assessment). In the next section, we highlight the CSAR vignette.

Combat Search and Rescue (CSAR)

The combat search and rescue (CSAR) mission vignette developed at DRDC Valcartier is being integrated is to CDIFT. This vignette was placed into fictitious exercise scenario, Final Lance-Atlantis, which was borrowed from the Canadian Forces Command and Staff College (CFCSC). The actions of each airborne tactical platform, as well as the elements of Combined Air Operations Center (CAOC) participating in the CSAR mission, were described in detail along the sequence of events that would occur during the conduct of a CSAR mission. A particular emphasis was placed on the methods, techniques and sensors used in compiling and exploiting both a recognized air picture (RAP) and recognized ground picture (RGP) using airborne assets. Although they are separate elements of the same mission, many of the RGP assets are airborne and are part of the RAP. Therefore, any movement of air assets to enhance the RGP would have a concomitant effect on the RAP. A synchronization matrix of friendly actions and communications, both directive and informative, was created. First, given the ideal situation with no enemy action and then reflecting the actions and decisions of three of the following unpredictable events:

a. Event 1: inability to locate the enemy ground positions due to cloud and terrain;
b. Event 2: enemy attack helicopters appear in the CSAR area;
c. Event 3: enemy SAM systems in the enemy rear area detects the CF-18 attack mission.

Although CSAR, like SAR, is considered a single mission, it is composed of two separate and distinct phases: the search phase and the rescue phase. In a peacetime SAR scenario, often the search phase is coincidental with the rescue phase. A SAR-capable aircraft will be dispatched to fly deliberate search patterns over the suspected area of concern. Once the lost person(s) are detected, like a floundering ship at sea, and the search aircraft is capable, like a SAR helicopter, the rescue takes place immediately. However, in the same example, if the search aircraft was fixedwing, then, with the search phase complete, the rescue mission would be launched separately.

The same concept holds true for the CSAR mission except that, due to the threat posed by conducting a rescue operation in enemy territory, the search and rescue phases of the mission are quite distinct and often are separated by days. There are numerous methods of searching for downed crews or personnel lost in enemy territory. As expected, the technique of flying a predictable search pattern over enemy territory would be fraught with risk. Combat search techniques would include:

- Location by electronic means using emergency locator transmitter (ELT) signals

emanating from downed aircraft. Those signals can be triangulated by satellites, like SARSAT, or by aircraft flying in friendly territory;
- Location by radio/secure radio transmissions from the downed crew(s) using escape and evasion radios. These transmissions can be voice, which could pass exact GPS location, or a homing signal for triangulation. Again these search techniques could all occur over friendly territory;
- Fly tactically supported reconnaissance assets (aircraft or UAVs) through the suspected target location to visually spot the crew(s). This is normally not done due to the risk of losing another asset and because the personnel on the ground have no way of knowing that the reconnaissance mission is friendly, and thus will likely be avoiding/evading detection.

On the second day following the commencement of the Alliance joint operations to secure Blueland territory and expel any coalition invasion forces, a U.K. Royal Air Force (RAF) Tornado call-sign HAWK27, conducting an electronic countermeasures and reconnaissance (ECR) mission, was shot down over the Celtic Straits by a SAM at 1608 hours. The crew did not report any radar activity, so it was believed that the missile was either an optically launched SA 8 or shoulder launched SA-14. Both systems had been reported in the area as part of the Coalition Airborne Regiment that invaded the Blueland portion of the Camrien Peninsula at the outset of hostilities.

The Tornado aircraft’s wingman reported that both aircrew had ejected safely and that the downed crew had reported no injuries via secure communication using their survival radio. Shortly after the location of the downed crew was known, a CH-124 Sea King helicopter from Wahhabe Airbase, with a crew of five, was sent to recover and evacuate the Tornado aircrew. At approximately 1800 hours, in the process of extraction of the downed Tornado crew, the Sea King crashed. The cause of the crash is attributed to mechanical failure. Two members of the Sea King crew sustained nonlife-threatening injuries that have limited their mobility on foot. The crash site, illustrated in Figure 12.17, and location of both crews is 5650N 2740W, which is approximately 60 nm north of the Brownland town of Amitava on the Camrien Peninsula.

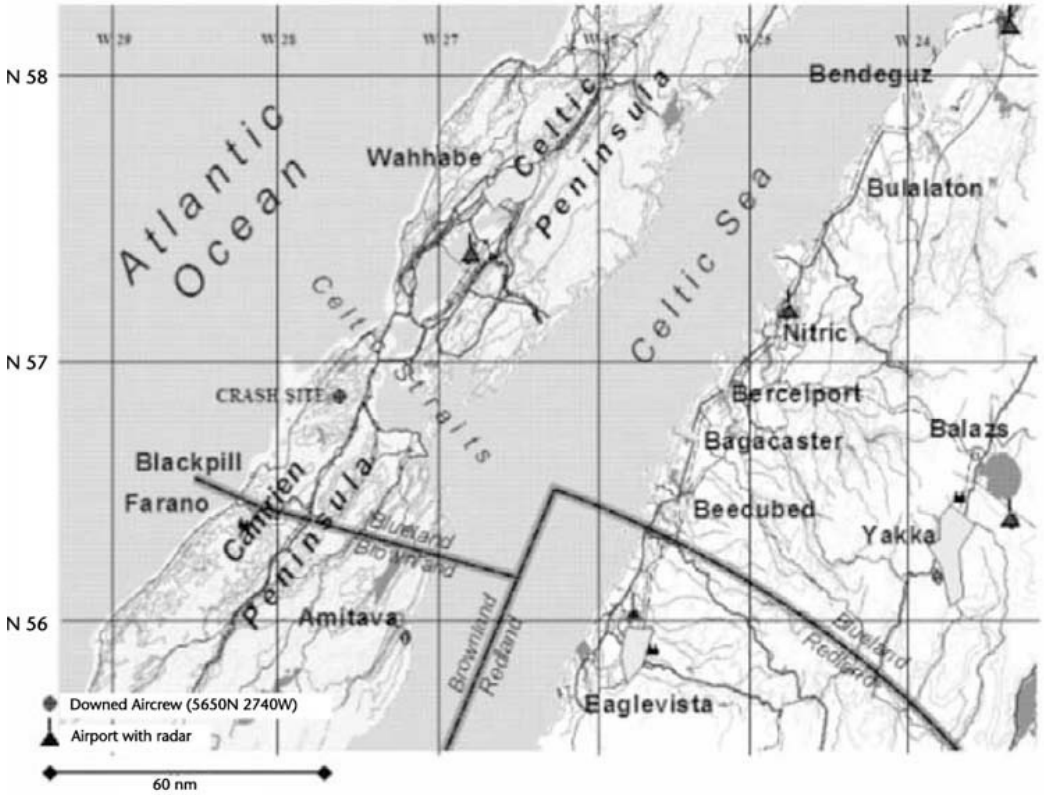


Figure 12.17 Tornado and Sea King crash site.

The situation was forwarded from the Combined Air Operations Centre (CAOC) to the Air Component Commander (ACC) for consideration in the Joint Force Commander's (JFC) force allocation for the upcoming planning period.

Finally, data sets describing the individual characteristics of each of the friendly and enemy airborne assets and ground-based SAM systems found in the CSAR scenario can be provided to assist in follow-on CDIFT simulation and/or ontological modeling.

Conclusion

This chapter outlines a collaborative program of work in high-level information fusion being conducted between Australia, Canada, the United States, and the United Kingdom under the auspices of a TTCP panel on information fusion. The chapter shows how work from the four nations has been integrated through a common scenario, which the panel proposes to offer as a benchmarking scenario for the high-level information fusion community.

rences

[1] Bossé, É., J. Roy, and S. Wark, *Concepts, Models, and Tools for Information Fusion*, Norwood, MA: Artech House, 2007.

[2] Blanchette, M., "Military Strikes in Atlantis—A Baseline Scenario for Coalition Situation Analysis," The Technical Cooperation Panel, *Technical Report TR-C3I-TP1-1-2005*, 2005.

[3] Salerno, J. J., M. Hinman, and D. Boulware, "Building A Framework for Situation Awareness," *International Conference on Information Fusion*, 2004.

[4] Salerno, J. J, M. Hinman, and D. Boulware, "A Situation Awareness Model Applied To Multiple Domains," *Proceedings of SPIE*, Vol. 5813, 2005.

[5] Gadz, W., A. Colby, and M. Seakan, "Information Extraction for Counterdrug Applications", In *Proceedings of the ONDCP International Technology Symposium*, 2003.

[6] Pearce, P. V., A. Robinson, and S. C. Wright, "The Wargame Infrastructure and Simulation Environment" (WISE). In *Knowledge-Based Intelligent Information and Engineering Systems International Conference*, V. Palade, R. J. Howlett, and L. C. Jain (eds.), and *Lecture Notes in Computer Science*, No. 2744, Oxford, pp. 714–722. Springer-Verlag, Berlin, 2003.

[7] Hossain, A., N. Walmsley, and P. Pearce, "A Minimum Spanning Tree Approach to Identifying Collective Behavior and Inferring Intent for Combat Models," *The International Command and Control Journal*, Vol. 2, No. 2, 2008, pp. 1–33.

[8] Lambert, D. A., "A Unification of Sensor and Higher-Level Fusion," *International Conference on Information Fusion*, 2006.

[9] Lambert, D. A., "STDF Model Based Maritime Situation Assessments," *International Conference on Information Fusion*, 2007.

10] Lambert, D. A., "A Blueprint for Higher-Level Fusion Systems," *Journal of Information Fusion*, Vol. 10, No. 1, 2009, pp. 6–24.

11] White, F. E., "A Model for Data Fusion," *National Symposium on Sensor and Data Fusion*, 1998.

12] Lambert, D A., "Grand Challenges of Information Fusion," *International Conference on Information Fusion*, 2003.

13] Lambert, D. A., and C. Nowak, "Mephisto I Towards a Formal Theory," *Proceedings of the 19th Joint Conference on Artificial Intelligence, The Australasian Ontology Workshop*, Hobart, Australia, 2006.

14] Lambert, D A., "Automating Cognitive Routines," *International Conference on Information Fusion*, 2003.

15] Steinberg, A.N., and G. Rogova, "Situation and Context in Data Fusion and Natural Language Understanding," *International Conference on Information Fusion*, 2008.

16] Wark, S., and D. A. Lambert, "Presenting the Story Behind the Data: Enhancing Situational Awareness Using Multimedia Narrative," in *IEEE MILCOM Conference*, 2007.

17] Gratch, J., et al., "Creating Interactive Virtual Humans: Some Assembly Required," *IEEE Intelligent Systems*, July/August 2002, pp. 54–63.

18] Cassell, J. and T. Bickmore, "External Manifestations of Trustworthiness in the Interface," *Communications of the ACM*, Vol. 12, 2000.

19] Smith, G., M. Dry, and M. Lee, "Trust and Confidence: Presenting Information Over a Network Using Text or a Speaking Head," *TTCP HUM TP7 Workshop on Aerospace Human Factors Issues in Network-Centric Warfare*, Salisbury, Wilts, United Kingdom, 2005.

20] Glenn, R. W., C. Paul, T. C. Helmus, and P. Steinberg,"People Make the City," *Executive Summary: Joint*

Urban Operations, Observations, and Insights from Afghanistan and Iraq, 2007.

21] “Future Army Experiment: Operations in the Urban Battlespace,” Directorate of Land Strategic Concepts, Fort Frontenac, Kingston (Ontario), Canada, 2002.

22] Pigeon, L., and É. Bossé, “Scenario Consolidation for C2 Urban Operations S&T Validation,” *Defence R&D Canada Valcartier Technical Report 2009-215*, 2010.

23] “Command and Control Challenges in Urban Operations: An Analysis and a System of Systems Study of Information Critical for the Commander in the Field,” *NATO RTO Technical Report TR-IST-046*, 2007.

24] Smeenk, B. “Scenario and Vignette Descriptions for RTO-IST 46 Command and Control in Urban Operations,” *TNO-FEL (NL) Memorandum*, 2005.

25] “Joint Urban operations, Joint Integrating Concept—Version 1,” United States Joint Forces Command, 2007.

---

1. Previously submitted for public approval, July 2009.

2. avis.sourceforge.net.

3. http://www.presagis.com/products_services/products/modeling-simulation/simulation/stage/.

4. For example, the NATO attack on the Djakovica refugee convoy, cited in the *Final Report to the Prosecutor by the Committee Established to Review the NATO Bombing Campaign Against the Federal Republic of Yugoslavia*, http://www.un.org/icty/pressreal/nato061300.htm.

CHAPTER 13

Operating Condition Scenario Modeling for Information Fusion Assessment

Erik P. Blasch (United States)

Real world operating conditions (OCs) influence sensor data, situational context, theoretical algorithms, and information representation. For example, the performance of a target detection, recognition, classification, and identification system (see Chapter 2) is based on the scenario of the collected information. OCs that affect data depend on sensor wavelength (e.g., electro-optical (EO), infrared (IR), and radar sensors), associated scenario phenomenology (e.g., target materials, weather, and lighting), and knowledge representation of the situation context. This chapter will discuss operating condition modeling for scenarios and how they affect automatic target classification (ATC) systems that provide situation awareness. The OCs are broken out into four categories: *target*, *environment*, *sensor*, and *ATC training*. These main categories will further contain subcategories with varying levels of influence. The purpose of this chapter is to develop a scenario-based OC distribution model for the real world that can be used to realistically represent the performance of multiple ATC systems and, ultimately, the decision made from the fused ATC results to support HLIF. An accurate OC theoretical model will greatly enhance the performance assessment of ATC systems by affording Bayesian conditioning in fusion performance analysis and aiding in the sensitivity assessment of fusion performance over different operational conditions.

Introduction

When considering the contextual scenario for an information fusion design, there is a need to understand the relationship between measured data and an object in the sensed scene. The measured data is greatly influenced by external circumstances, such as the object pose and state; proximity to environmental clutter factors (e.g., vegetation or weather conditions,), and sensor parameters (e.g., frequency, collection geometry, or resolution). These circumstances are collectively known as operating conditions or (OCs) [1, 2]. Developing a theoretical model of the world representation of OCs is a very challenging task due to the complex and dynamic environment. An understanding of the basic structure of automatic target classification (ATC) systems and the sensor technology is necessary to help reduce the number of conditions that need to be modeled to just the relevant factors, or key OCs.

13.1.1 Sensor-Based Classifier Operating Conditions

An ATC system contains one or more classifiers using collected data from one or more types of sensors such as radar, EO, or IR, and could potentially use context, human-in-the-loop, and sensor fusion [3]. A classifier contains the algorithms used in an automated process for object detection, classification, and identification. A classifier is typically trained on a group of desired objects to gain a statistical representation of the objects in the library. The algorithm then employs a detection scheme to find potential objects in a scene, clutter suppresses the scene, and compares detected objects not rejected as clutter to the library of objects for a potential matches, resulting in an identification or class label [4]. Although this process seems fairly straightforward, operating conditions affect the sensor data and the ATC assessment of the scene and can cause misidentification or rejection of a viable target test outcomes.

Object detection and identification by ATCs is influenced by the sensor performance which can be estimated at selected OC states. Key design parameters can be used to compute performance parameters for Synthetic Aperture Radar (SAR), EO, and IR sensors to determine the limits of a sensor's capability to detect and image target features. Parameters for radar such as the frequency, bandwidth, antenna size, and pulse repetition frequency (PRF) can be used to estimate resolution and signal-to-noise ratio (SNR) for selected ranges to targets [5]. For electro-optical sensors (EO/IR) key parameters such as frequency, sensor aperture, focal length, and transmission (through the sensor optics) can be used to estimate the noise equivalent power, noise equivalent temperature difference, and the minimum resolvable temperature [6].

A number of recent papers [1, 2] have identified, discussed, and categorized OCs into three categories: *target, environment*, and *sensor*, as well as the impact of OCs on

ATC performance. Many of these papers assessed the performance of ATC systems using measured datasets where some of the OCs were identified and varied. However, the potential OCs and even larger number of OC combinations meant that only a handful of conditions were varied in the trade space, providing only a glimpse into the impact of various operating states on ATC algorithms. Classifiers, typically discussed in the context of automatic target recognition (ATR) system development, lack a rigorous assessment in relation to varying contextual situations.

A significant portion of the SAR ATR problem tree is operating conditions [1], as shown in Table 13.1. For SAR ATRs, there are six target OCs identified; eight environment OCs discussed such as camouflage, concealment, and detection (CC&D); and ten sensing OCs are introduced. Work with forward-looking infrared (FLIR) ATR algorithms used measured data that was split into testing and training sets, identified as favorable and unfavorable [7, 8]. The precise nature of the OCs present in each FLIR dataset is not identified; however; the signature variability of the overall datasets are attributed to eight varying conditions. The evaluation of a hyperspectral imaging (HSI) system, which is comparable to visual camera systems, discussed the importance of describing the OC space when planning a data collection and pointed out the need to characterize the OC coverage and truth in the obtained data [9]. Four main HSI OC types are identified: target, sensor, environment, and interaction. There were seven target OCs listed, seven sensor OCs identified, and eight environment OCs. Finally, the five interaction OC examples are ground sampling distance, target aspect, number of pixels on target, shadowing, and obscuration.

Table 13.1 Example OCs for SAR, FLIR, and HSI Systems

OCS/Sensor	*SAR*	*FLIR*	*HSI*
Target	classes, configuration, articulation, damage, parts in motion, and version variant	classes,	vehicle type, version variants, size, manufacturer, surface material, surface variation, and engine heating
Environment	6 DOF pose, obscuration, layover, background, adjacency, CC&D, and weather	meteorological conditions, time of day, target locations, cluttered background, obscuration, and season.	atmospheric attenuation, atmospheric scattering, illumination, thermal history, precipitation history, background material type, foliage type, and foliage spacing
Sensing	depression, squint, signal, polarization, number of looks, resolution, anomalies, noise level, image formation artifacts, and strip versus spot mode	range to target	spectral range, spectral resolution, instantaneous field of view, viewing geometry, altitude, detector anomalies, and processing power

The relevant operating conditions described in previous work using SAR, FLIR, and HSI sensors are identified, some common factors and sensor specific OCs have emerged. The goal is to model the relevant OCs so that realistic OC distributions are produced which influence the ATC algorithms. OC model fidelity needs to adequately model reasonable bounds of what occurs in the real world. The impact to ATC decisions by OC interactions and the variability of individual OCs may be analyzed. A complete ATC system assessment of the fusion gain or loss [4] may be determined by combing ATC results over similar OCs, which can then help guide future ATC system developments, accommodations, and enhancements [2].

13.1.2 Scenario-Based Evaluation

To evaluate an ATC system, we address various performance metrics over *extended operating conditions* (EOCs). The performance metrics that satisfy objectives are location/classification accuracy, track life, and commander needs satisfaction. Since accuracy is dependent on sensor model fidelity, a functional analysis of improved target precision based on registered sensor data is needed. Track life results from persistent target classification. Commander needs satisfaction is based on the user, the scenario context, and an information needs decomposition.

The EOCs include a number of targets, a nominated area of interest (NAI), and sensor models. The NAI and high value targets of interest are known. Simulation models incorporate environment conditions, sensor exploitation capabilities, and target behaviors. Since an evaluation is to assess a real world performance capability, the most optimistic sensor assumptions are not used, but rather common sensor operating performance capabilities given the associated scenario details. To determine the performance over the EOCs, a design of experiments (DOE) formulates different scenarios for testing.

13.1.3 Design of Experiments for Scenarios

To effectively evaluate system performance, three types of parameters are assessed: (1) constants (e.g., NAI), (2) Monte Carlo variables (e.g., sensor variation), and (3) factors that change (e.g., number of targets) [10–12]. Table 13.2 shows a DOE analysis for target, sensor, and environment EOCs. Constant parameters are determined from unmodelable effects, issues that did not affect the outcome, or real world operating conditions. Constant target behavior includes emissions and onroad travel. Target factors include number of targets and confuser target clutter, while variable target behavior consists of target move-stop-move cycles, paths, and speed. The sensors and platforms are a result of the mission scenario. The sensor variables include exploitation accuracy from platform/sensor positions, while on/ off bias selection is an important sensor factor. The largest category of scenario OCs is from the environment which includes weather, NAI, facilities of target initiation/ termination, terrain/road conditions, and no-fly zones. For each scenario, a NAI is fixed and available terrain and weather data are provided.

Table 13.2 DOE of Test Scenarios

OC Category	*Parameter*	*Flat Terrain*	*Mountainous*
Targets	Number of Targets	6, 12, 25	1, 2, 5
	Moving confusers (dens./AOI)	Low (10), 50 , High (1,000)	Low (0), med (10), high (25)
	Routes move-stop-move	Variable	Variable
Sensors	Platform/sensor geometry	Variable	Variable
	Bias	On/off	On/Off
Environment	No-fly zones	Variable area locations	Variable area locations

Utilizing the static environmental conditions, different target densities, and sensor models are selected to test a variety of scenarios and evaluate system performance. Figure 13.1 represents the design space of different scenarios available for a given modernized integrated database, nominated area of interest, and air task order of mission objectives.

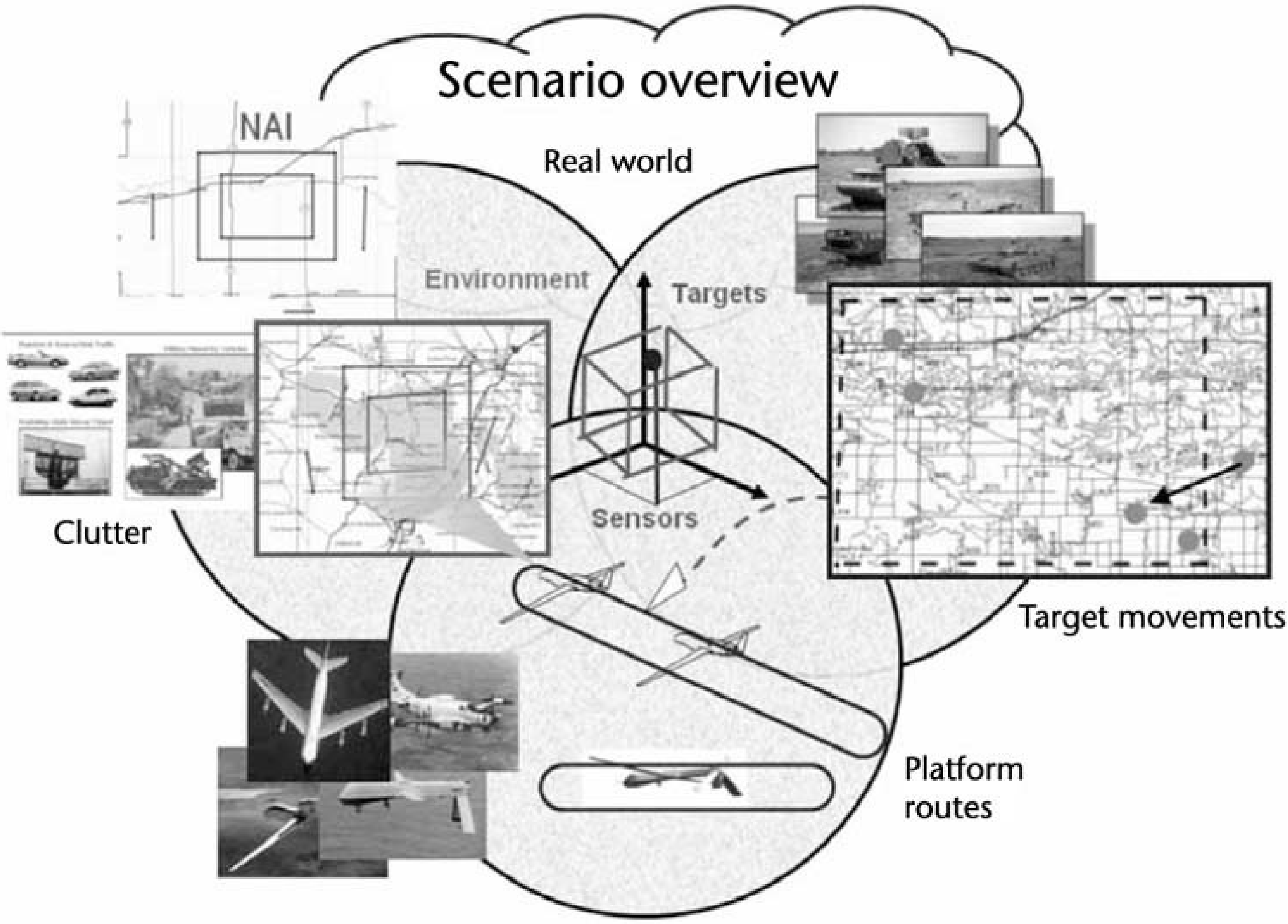


Figure 13.1 Scenario parameters.

After selecting the DOE scenario conditions, a simulator instantiates truth information, such as actual target behaviors and sensors measurements. Mean values and variance bounds associated with selected factors create a scenario which affords a longitudinal performance analysis based on ergodic assumptions. To develop scenario-based situation assessment testing of ATCs, an operation condition model is needed to robustly create real world parameter variations.

This chapter sets a baseline for ATC OC modeling for HLIF situational awareness. Section 13.2 describes model terminology. Section 13.3 details the influence of the OCs in a Bayes Net. Section 13.4 presents the OCs influencing the ATC fusion analysis. Section 13.5 discusses the use of the OCs on conditioning analysis for sensitivity and performance assessment. Section 13.6 summarizes the ideas presented and provides conclusions.

Operating Condition Model Terminology

Sensor specific operating condition sensitivities exist based on the OCs identified for SAR, FLIR, and HSI sensors. For example, the time of day and illumination of the nominated area of interest are listed as OCs for the FLIR and HSI sensors, respectively, but are not considered OCs for SAR sensors. However, the operating condition model will still need to produce time and illumination samples, even though all ATCs may not have sensitivity to the entire OC space modeled. The key to scenario design of information fusion representations is how OC sensitivities of different ATCs may be potentially exploited in the fusion process to enhance the decision of an ATC system. Some of the general terms used in describing OC models are discussed in the next section.

13.2.1 Direct Versus Indirect OCs

Conditions exist that may not directly impact the decision of an ATC or an ATC system for a given sensor, but the conditions influence other OCs in the modeling chain, which may be relevant to an ATC algorithm. These conditions are known as *indirect OCs*. Key OCs are factors that significantly influence a given sensor specific ATC algorithm. The key conditions that we choose to model are known as *direct OCs*. Some OCs have a dual role in the OC model by being directly relevant to an ATC and influencing one or more other conditions in the modeling chain.

Cloud cover is an example of an indirect OC (see Figure 13.10). Precipitation is identified as a relevant OC and is an example of a direct OC for EO systems. Precipitation cannot occur without cloud cover. The season of the year is an example of a dual OC because FLIR sensors have a direct sensitivity to the season. Season can impact the indirect cloud cover OC because locations in the world have a dry season and a rainy season, which means that the cloud cover OC will have a distribution based on the season of the year.

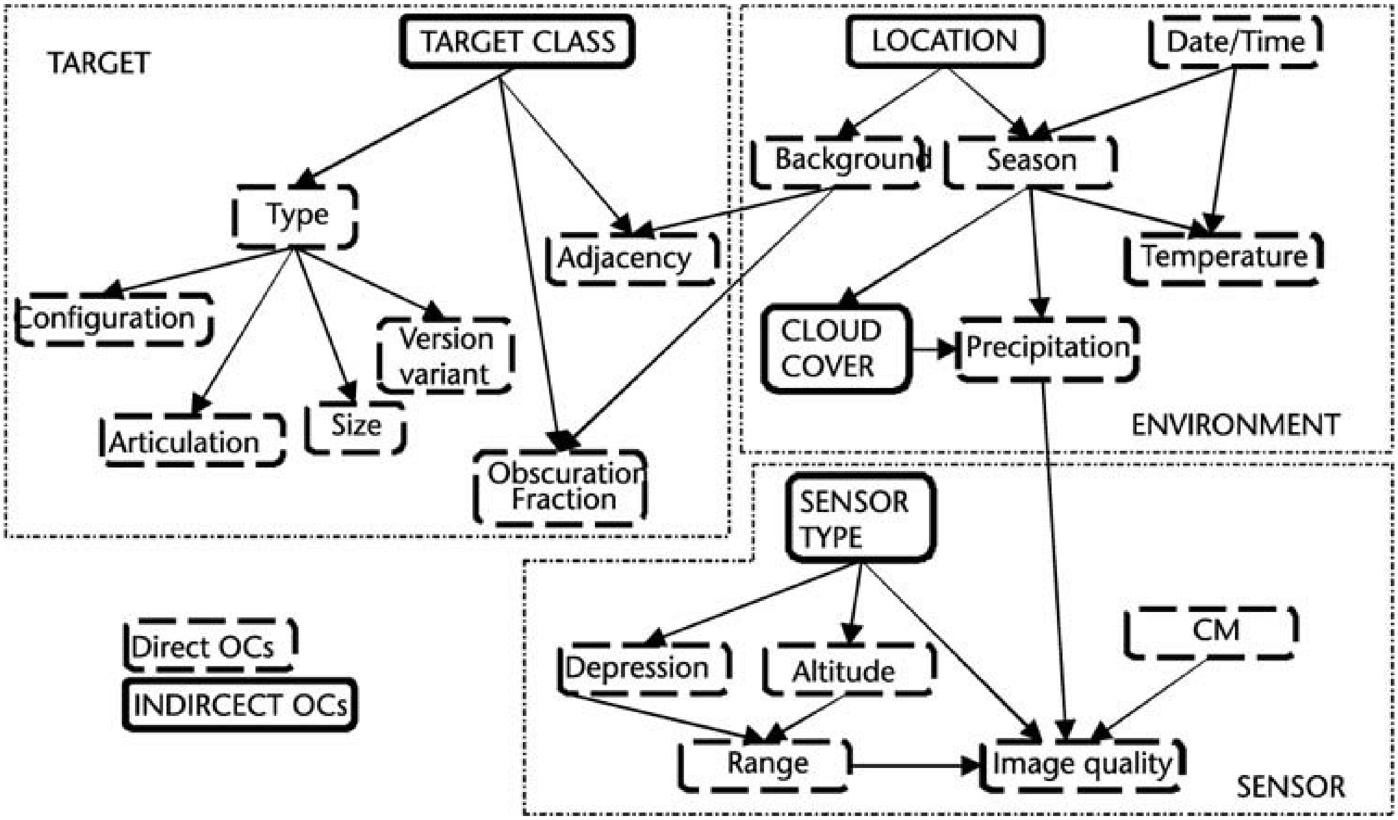


Figure 13.10 Operating condition model.

In summary, direct OCs are those OCs that the model is responsible for reporting as outputs and indirect OCs are those used only internally by the model.

13.2.2 Derived OCs

Some of the operating conditions are dependent on, or computed from, other OCs in the modeling chain. These are known as *derived OCs*. For example, range is a derived OC determined simply from the depression angle and sensor altitude (see Figure 13.10). A higher-fidelity OC model might incorporate platform velocity and signal time, to more accurately characterize the range to a stationary target imaged from a moving sensor.

13.2.3 Standard OCs Versus Extended OCs

The conditions present in data used to develop a statistical representation of an object of interest intended for a reference library are known as *standard operating conditions* (SOCs). Conditions that vary from the nominal training OCs are known as extended OCs (EOCs) [13, 14]. Whether an OC is standard or extended depends on the ATC consuming the OCs, so the OC model may contain both SOCs and EOCs.

It is widely known that ATC algorithms yield the best performance when the testing and training data have the same OC states. Performance degrades as the conditions vary further away from the OCs present in the training data. The vast majority of OCs encountered by an ATC system in the real world will most likely be EOCs; however, it is possible for some SOCs to be present. Since the OC model is only required to produce the ATC relevant conditions present in a scene, these OC states might be passed to a module (training or testing) in the ATC system, which is capable of determining if the OCs are SOCs, EOCs, or a combination. A disadvantage of this approach is that it requires modeling the many possible states for OCs. An alternative approach, as illustrated in Figure 13.2, is to model an OC as simply standard or extended from an OC model; however, all of the consumers of the OC models products would have to have the same standard OCs for the model to be completely accurate.

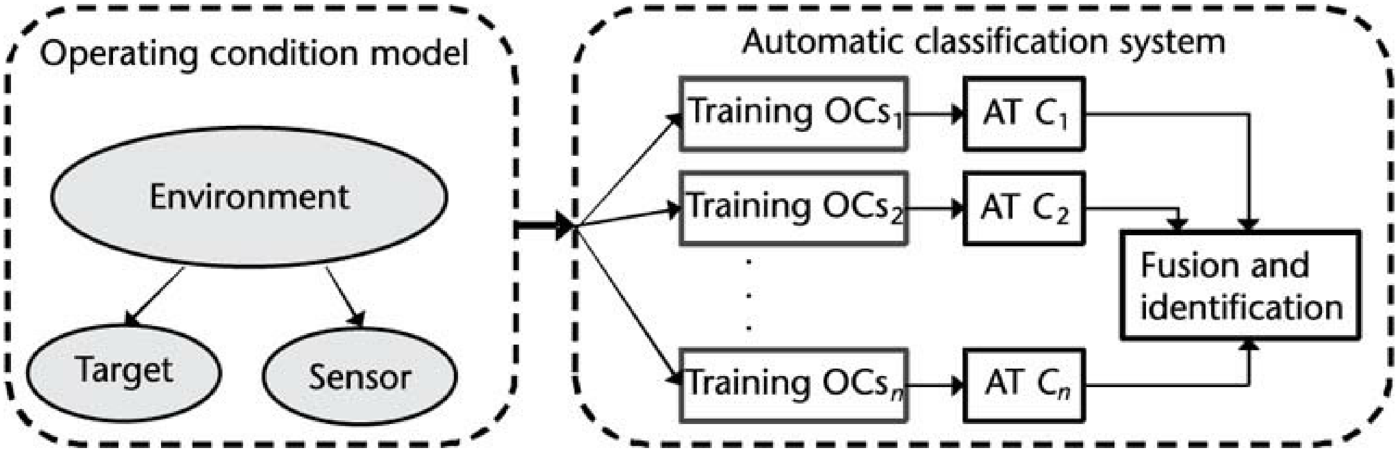


Figure 13.2 OC assessment by the automatic target classifier system.

Operating Condition Model Design

The operating condition model fidelity is a design consideration. The model complexity and processing time increase with greater fidelity, but the impact of the increased fidelity may be relatively small in the ATC fusion assessment process. A tradeoff between model complexity, processing time, and fidelity is necessary for a manageable OC model to be developed. The OC model needs to produce key OCs that reasonably compare to the conditions of the real world, for example, there is precipitation when it is cloudy during the rainy season. The OC model should generate realistic conditions within the realm of reasonable real world operations to increase model fidelity for input to ATC algorithms.

We wish to develop the real world model, but in practice there are many models. The selection of a specific model is essentially the same as selecting a particular scenario. We may have OCs models that allow us to scientifically vary some small set of OCs, without any real world considerations. We may have OC models that represent a narrow engagement of some specific type of target in a particular place, or that represent a particular campaign or larger engagement. These various models are not better or worse than each other; however, they must be selected to represent the conditions of interest for the system under test. A limitation of traditional information fusion system testing that they often do not include a an explicit statement of the test conditions. A principal benefit of OC models is that they allow a concise statement of the test conditions. To develop the OC model, Bayesian methods are used.

13.3.1 Bayes Model

The operating condition model is based on a Bayesian network (BN) approach [15]. Each direct or indirect OC in the model can be thought of as a node in the BN with a corresponding probability distribution. The distributions are sampled and, after propagating through the entire modeling chain, the OCs for a given instance of the scenario are produced.

A BN is an acyclic graphical probabilistic model that represents the knowledge of the domain. The BN nodes denote the variables and links denote dependencies between variables. In a static experiment, the variables determine the conditional independence values for the weights of the Bayes updates. Dynamic Bayes nets include time elements and have been used for knowledge management in situation analysis [16], activity recognition [17], and site classification [18].

13.3.2 Bayes Fusion from Real World (Scenario) Analysis

The Bayes net concept is used to determine the ATC decision-level fusion gain. The

ATC system probability of correct identification (P_{ID}) is computed as a function of the trade-space that includes OC model samples from the Real world $p(R)$, ATC results $p(A)$, Bayesian fusion results $p(B)$, and the contextual or a priori visible information $p(V)$. Bayesian networks determine the values associated with the different models $\{p(R), p(A), p(V), p(B)\}$, as shown in Figure 13.3. The key development is the incorporation of situation visual contextual information in conditioning the standard ATC system.

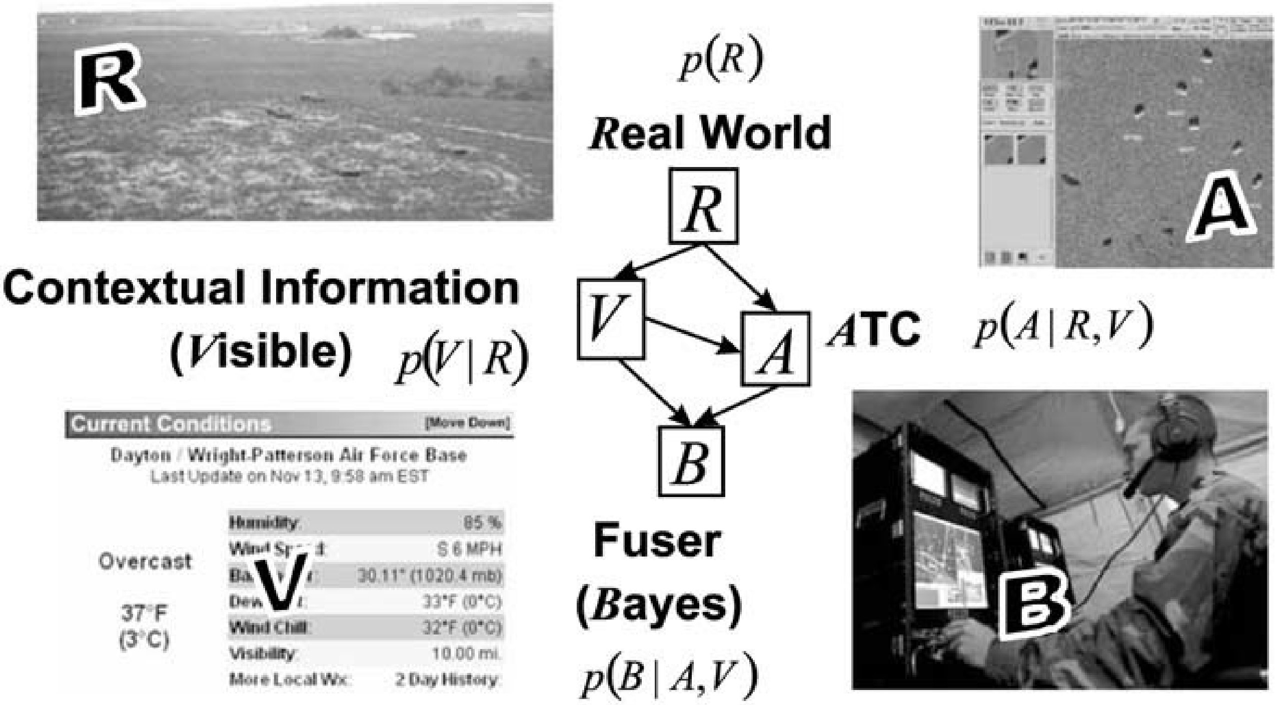


Figure 13.3 Bayesian network for an automatic target classification system.

The ATC system performance is given by

$$P_{ID} = \sum_{range(R)} p(B_R = r_T \mid R_T = r_T, R_C)\, p(R) \tag{13.1}$$

where B_R is real world Bayesian result, r_T is sampling of the real target type, R_T is the true target type, and R_C is real world operating conditions. The basic equation for $p(B|R)$:

$$p(B \mid R) = \sum_{range(A)} \sum_{range(V)} p(B \mid A, V)\, p(A \mid V, R)\, p(V \mid R) \tag{13.2}$$

Equation (13.2) derives the joint and conditional distributions [i.e., $p(B|R) = p(R|B)p(B)/p(R)$] from Bayes rule and from the removal of extra variables, known as marginalization, we have $p(B,R)=\sum_{range(A)}\sum_{range(V)} p(B,A,V,R)$, where $p(B, A, V, R) = p(B|A, V)\, p(A|V,R)\, p(V|R)\, p(R)$.

The samples produced from the OC model are one of the four basic inputs used in trade studies for Bayesian fusion of ATC decisions and for solving (13.2) to obtain fused ATC system-level performance. To determine the influences of the real-world probabilities, we need to address the four types of operating conditions.

Example Operating Conditions

The OCs and associated model described here is used to illustrate the general OC modeling concepts and processes. The direct operating conditions shown in Figure 13.4 collectively influence the data quality which in turn impacts the ATC system performance. The OC model developed for ATC fusion assessment is divided into the three major operating condition categories: target, sensor, and environment. The OC model assumes that only stationary surface ground vehicles are targets of interest and that only airborne sensor systems are present in the modeled world.

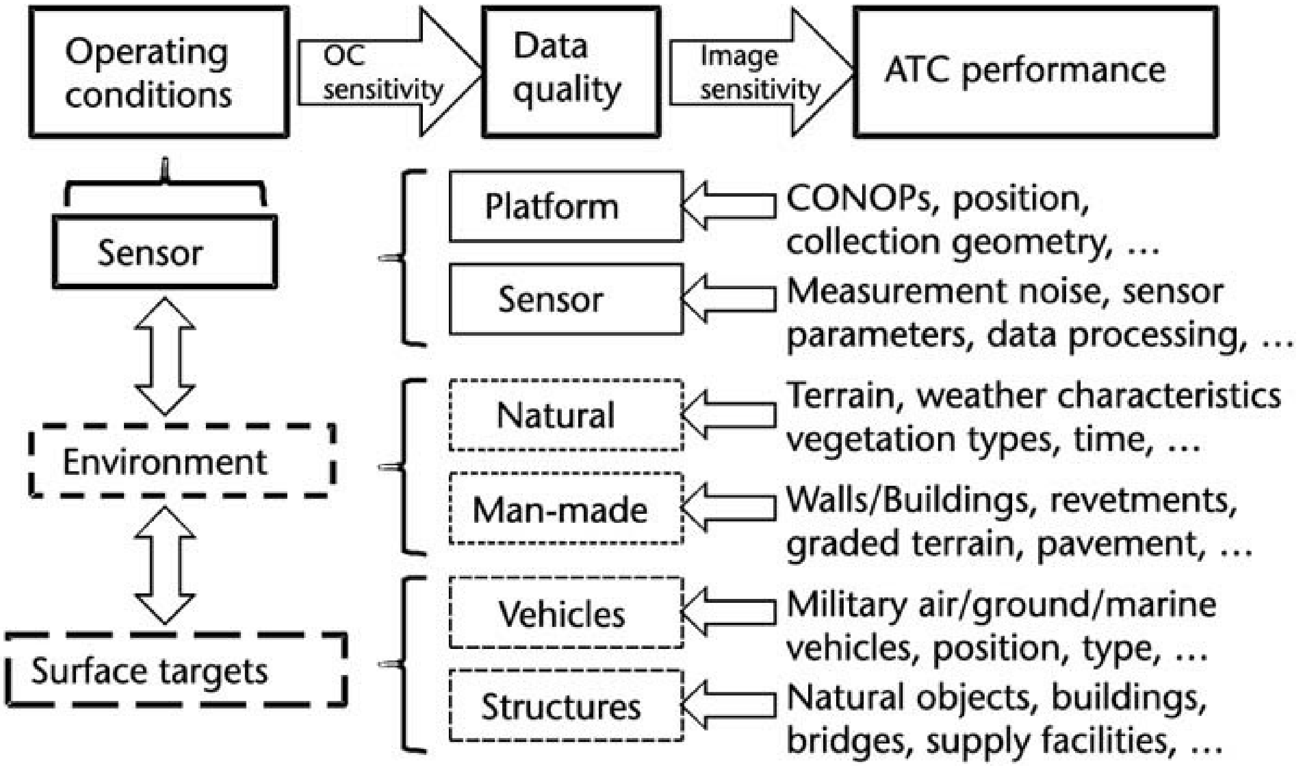
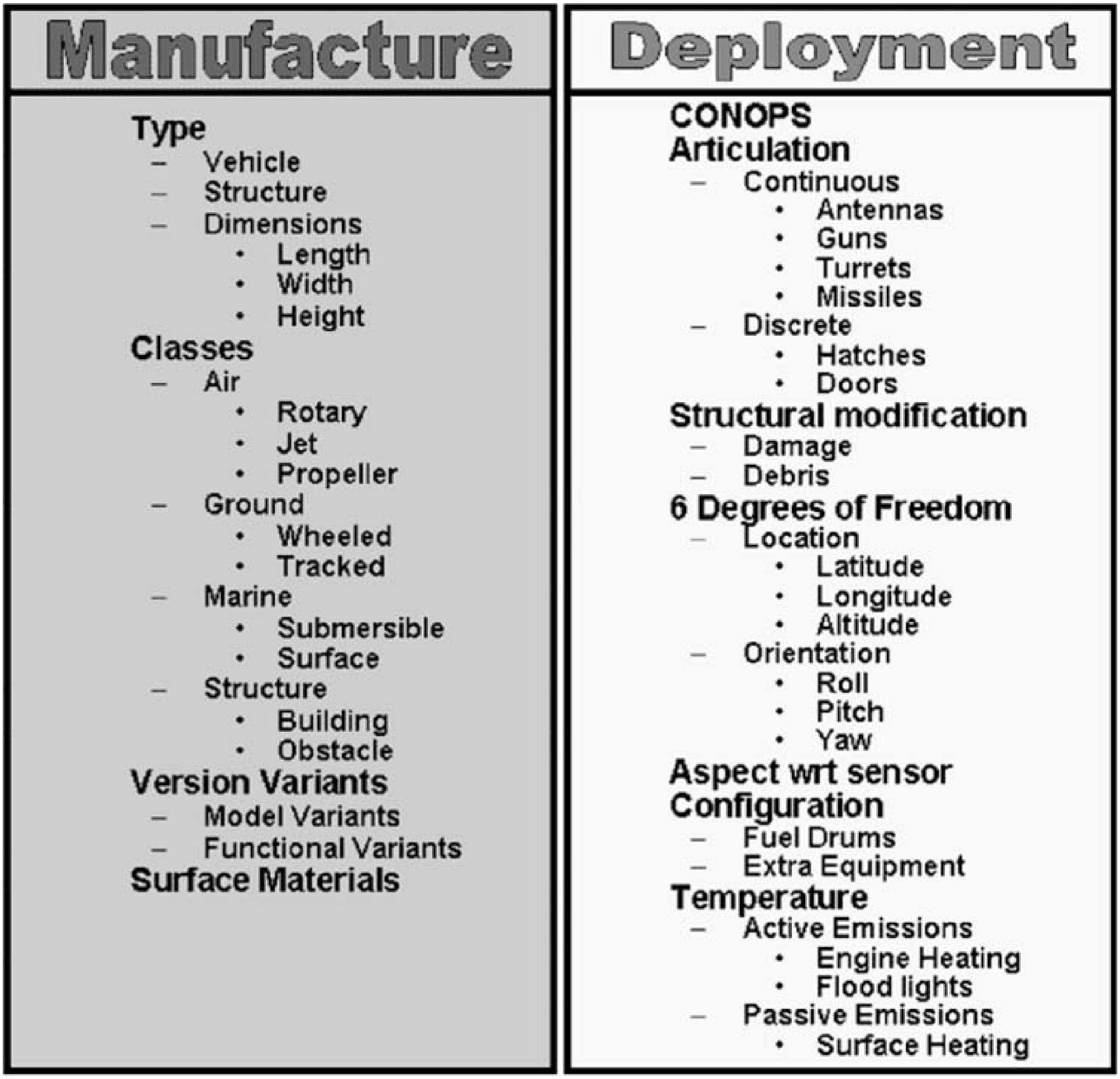


Figure 13.4 Three major OC categories.

The lists of potential operating conditions presented in the subsections that follow are the combined result of relevant work [1–8, 19–25]. The relevant operating conditions for radar and EO/IR based ATC systems were first gathered. Although an effort was made to include all of the possible conditions that may be relevant for ATC fusion assessment, it is likely that some OC sensitivities are not included in the lists.

13.4.1 Target OCs

A general list of potential target conditions that could be incorporated into the OC model are shown in Figure 13.5 and includes conditions which may be only relevant to a SAR, EO, or IR sensors. Our assumptions eliminated several conditions from consideration for inclusion in the OC model. The target classes were modified after removing air, marine, and structural targets from consideration so that classes of ground targets are examined. The relevant target OCs chosen for the OC model are discussed below.



Manufacture

Type
- Vehicle
- Structure
- Dimensions
  - Length
  - Width
  - Height

Classes
- Air
  - Rotary
  - Jet
  - Propeller
- Ground
  - Wheeled
  - Tracked
- Marine
  - Submersible
  - Surface
- Structure
  - Building
  - Obstacle

Version Variants
- Model Variants
- Functional Variants

Surface Materials

Deployment

CONOPS

Articulation
- Continuous
  - Antennas
  - Guns
  - Turrets
  - Missiles
- Discrete
  - Hatches
  - Doors

Structural modification
- Damage
- Debris

6 Degrees of Freedom
- Location
  - Latitude
  - Longitude
  - Altitude
- Orientation
  - Roll
  - Pitch
  - Yaw

Aspect wrt sensor

Configuration
- Fuel Drums
- Extra Equipment

Temperature
- Active Emissions
  - Engine Heating
  - Flood lights
- Passive Emissions
  - Surface Heating

Figure 13.5 General list of potential target OCs.

Only one indirect target operating condition is modeled, *target class* which is defined as a major grouping of a variety of similar vehicle types. The OC model contains eighteen classes: cars, buses, vans, trucks, trailers, construction equipment, agricultural equipment, main battle tanks, armored personnel carriers, infantry fighting vehicles, multiple launch rocket systems, self-propelled guns, radar, anti-aircraft artillery, surface-to-air missiles, surface-to-surface missiles, and another class for everything else, including clutter. Eight of the military vehicle classes were identified in recent literature [1]. Radars are used to track air targets and guide air defense systems; therefore, they were added to the list of military target classes along with surface-to-surface missiles (i.e., Scuds). Since the majority of ground vehicles in the world are civilian, an ATC system has a very high probability of encountering civilian vehicles. Therefore, seven additional classes of targets were added to the list to account for the presence of civilian vehicles. The target class contains both military and civilian vehicles in the OC model.

Figure 13.6 details an example of the target OC portion of the model. The eight target operating conditions of the proposed model contain seven direct OCs; target type, size, adjacency, obscuration fraction, target configuration, target articulation, and target version variant, as shown in Figure 13.6. The target type is a specific vehicle in a class, for example a DZ-98 road grader in the construction equipment class, an F-150 pickup truck in the civilian class, or a T-72 main battle tank in the military class. Each target type has an associated size dimension: length, width, and height. Each target type has a unique set of configuration, articulation, and version variant states associated with it. For example, a main battle tank has a turret to rotate and a gun to elevate, but an F-150 pickup truck doesn’t have these states as options. However, both the tank and truck have doors or hatches, which may be articulated (opened or closed). The direct OC configuration covers any alteration of components on a vehicle such as a truck with or without a cab. The same vehicle configured for a different purpose is a target variant, for example a passenger car and the same model as a police car. Adjacency is the distance from the target to the closest object in the scene. Obscuration fraction is the percentage of the vehicle that wasn’t imaged by the sensor, which depends on the collection geometry. The obscuration fraction can be significant if an object is between the sensor and the target. However, if the sensor images the same scene from a different geometry where the target is between the sensor and the object, then the obscuration is very small or nonexistent.

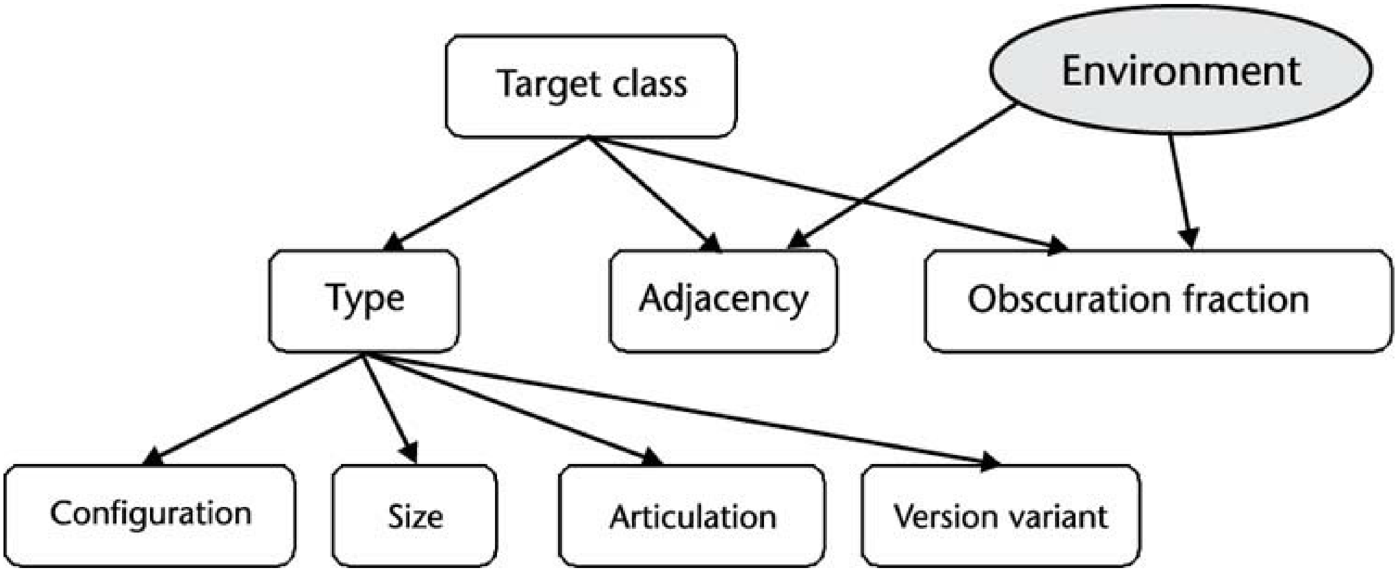


Figure 13.6 Target portion of the OC model.

The environment portion of the model influences some of the target OCs. For example, target adjacency and obscuration were studied for group tracking [26]. Target vehicles in a busy parking lot environment are more likely to experience adjacency than a single vehicle in an open field. An urban environment will contain many more opportunities for a target to be obscured to some degree as opposed to more rural or less densely built up areas.

13.4.2 Environmental OCs

A general list of potential environmental OCs addressing sensitivities of SAR, EO, and IR sensors for inclusion in the OC model is shown in Figure 13.7. Some of the environment OCs are influenced or derived from multiple categories which are illustrated by the overlapping of the meteorological, natural, and man-made lists. The model discussed in this chapter chose a subset of the environmental OCs.

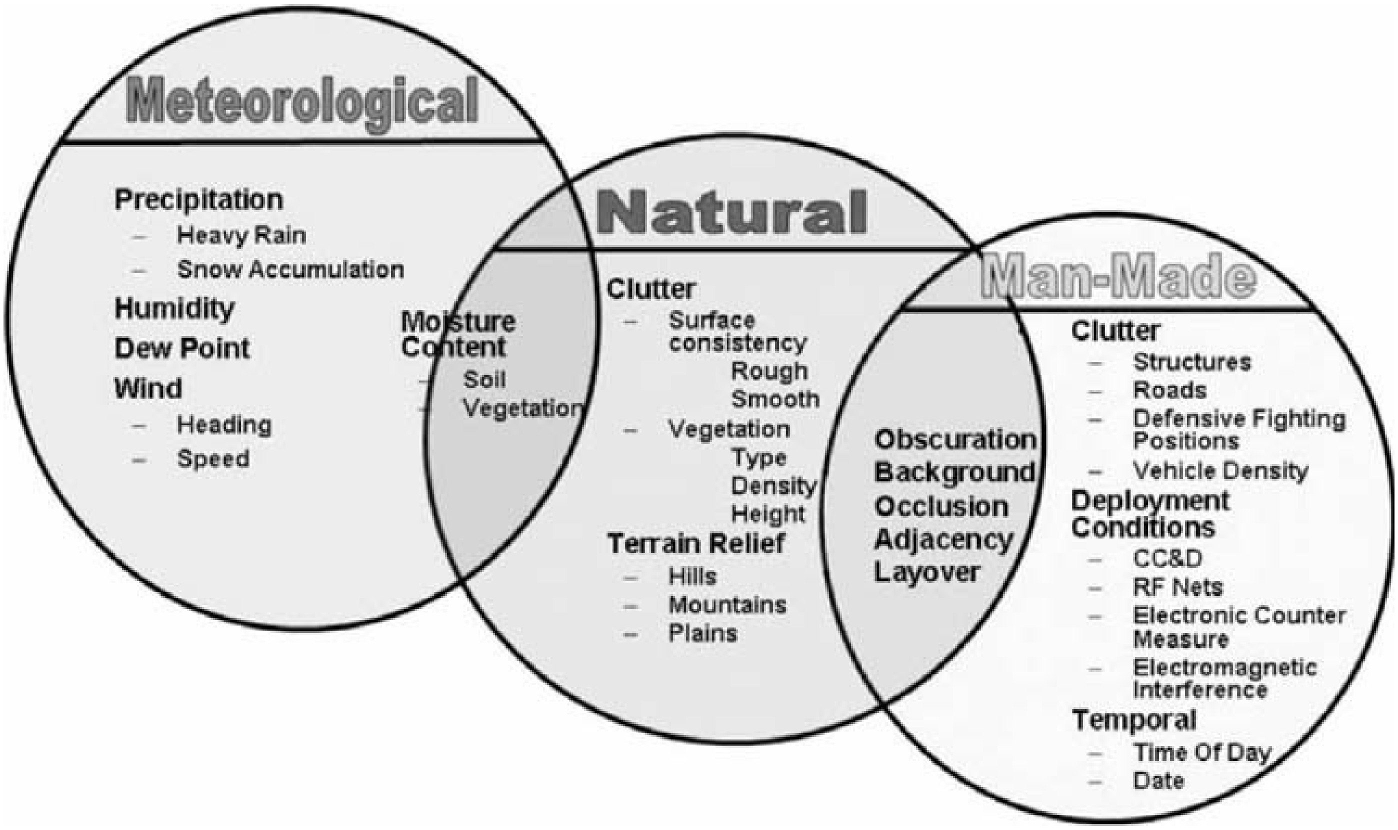


Figure 13.7 List of potential environmental OCs.

Two indirect OCs of location and cloud cover are part of the environmental OC model. Location is the point on the earth being imaged by the sensor and influences the type of background that can be encountered and, for a given time, influenced by what season of the year it may be. Cloud cover is used to help determine the presence of precipitation.

The five direct environment OCs modeled are: (1) season, which helps determine the level of cloud cover and amount of precipitation for any given location on the planet; (2) date/time of day, which helps determine the season and temperature distribution for a given location; (3) temperature; (4) background, or environment, which is the surface condition (such as asphalt, grass, forest, rock, or bare soil) of the scene being imaged by a sensor and is an input to the target OC portion of the model; and (5) precipitation, which is used as an input to the derived OC image quality.

13.4.3 Sensor OCs

In Figure 13.8, a general list of potential sensor specific OCs for SAR, EO, and IR sensor systems is presented. The overlap between platform specific conditions and the sensor conditions is due to the influence of platform motion [27] and position on the collection geometry and image quality [28]. For the purpose of analysis, a handful of sensor specific conditions were chosen for the OC model, such as the elements of an

EO/IR sensor performance model [29] for dynamic targets [30], HSI sensing [31], and a SAR performance model [32].

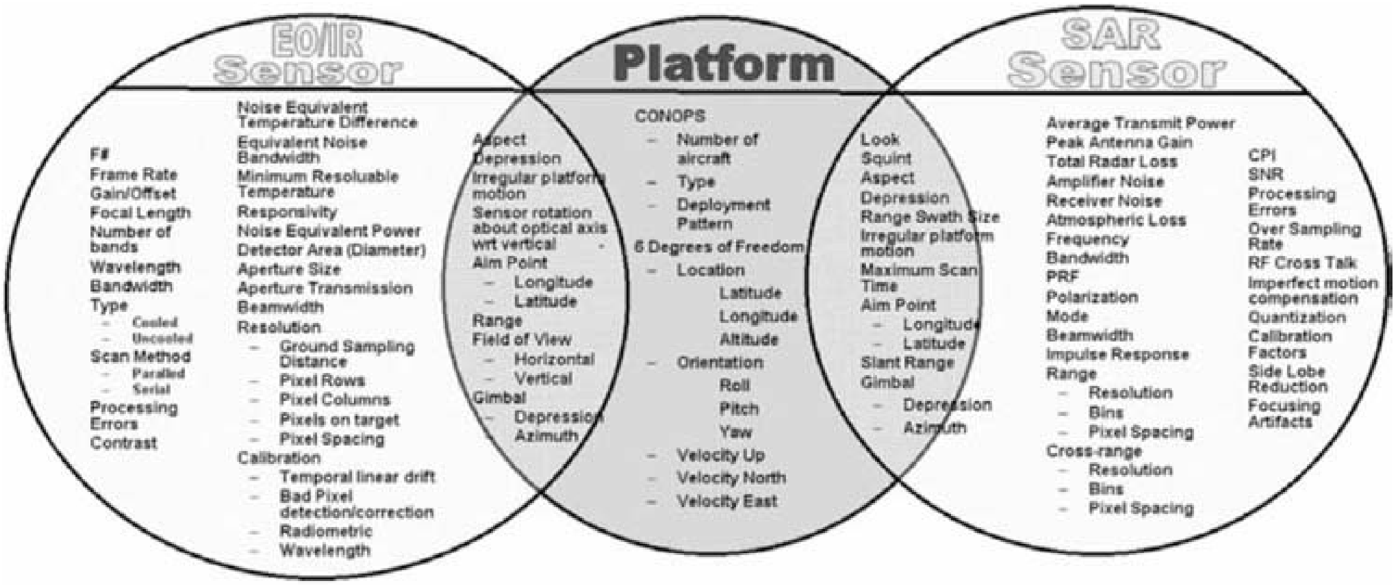


Figure 13.8 List of potential sensor OCs.

Sensor type is the only indirect OC of the sensor operating condition portion of the model and assumes that SAR, EO, and IR sensor data are all that is available for the ATC system. The sensor type influences other relevant OCs in the modeling chain. SAR sensors are generally more of a standoff system capable of imaging scenes and detecting targets at greater ranges than typical EO and IR sensor systems.

The five direct sensor operating conditions in the model are depression angle, altitude, range, counter measures, and image quality. The depression angle is the look angle from the sensor platform down to the area of interest. The altitude is the height of the sensor platform while imaging. Depression and altitude are then used to determine the range to the target vehicle which is a derived OC. The presence of counter measures, though generally rare for the anywhere-in-the-world OCs, will impact the image quality of the sensor data being affected. Regardless of sensor type, the range to target will influence the data quality and the ATC decision made from that data. A diagram of the sensor OC model is shown in Figure 13.9.

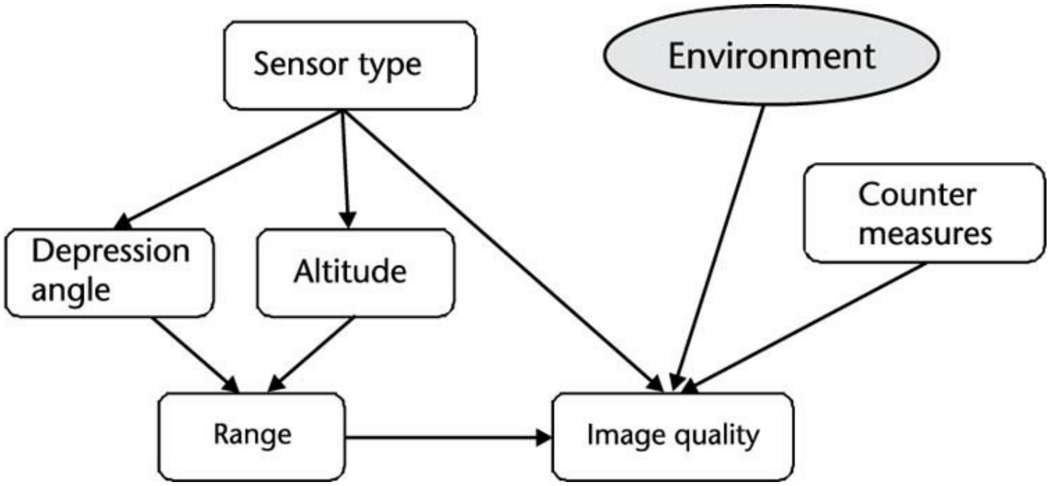


Figure 13.9 Sensor portion of OC model.

As with the target OC portion, the environment OC model, specifically precipitation, has an impact on image quality. The amount of influence the environment has is dependent upon the sensor type and range to target. If no precipitation is present, then the environmental impact is negligible. However for EO or IR sensors, the presence of precipitation is significant, and imagery under these conditions should probably not be used by an ATC system to make classification decisions.

13.4.4 ATC Training OCs

A fourth operating condition category is the ATC training conditions. The training OCs have an enormous impact on the decisions of ATC algorithms, as the data used to train the classifiers and the relevant scenarios impact the ability of the classifier to be robust to different OCs. We did not use the training OC in the DOE analysis, but is important for operational testing.

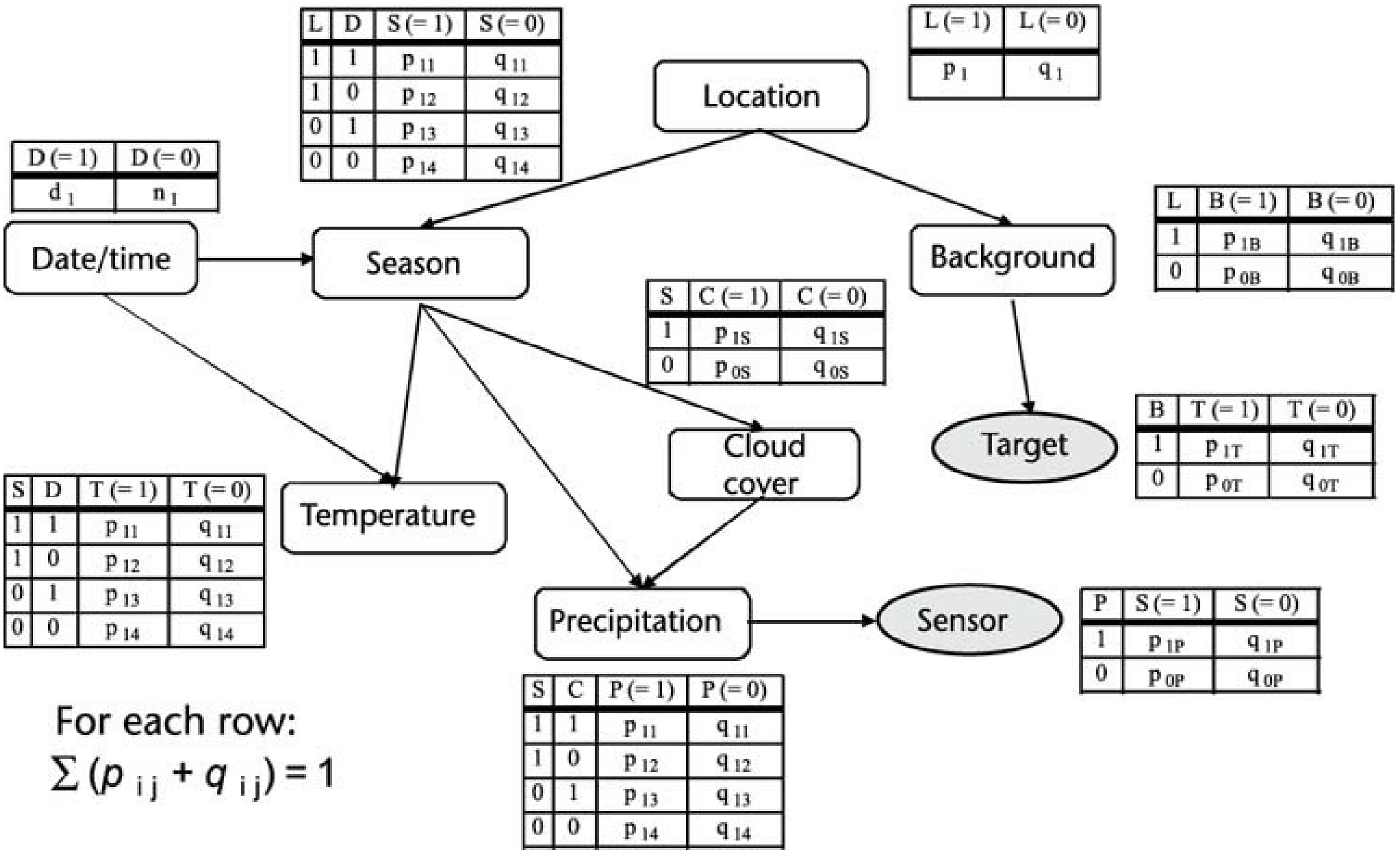
13.4.5 OC Model

The entire OC model described is shown in Figure 13.10, with the links between the three pieces of the model. The nodes containing the solid boxes (indicated with uppercase text) are indirect OCs, which are then used to influence the direct OCs in dashed boxes (indicated with lowercase text). The model discussed is relatively simple compared to the large variety of OCs that have been identified. The purpose or end use of the model clearly needs to be considered when selecting OCs to be incorporated into a model. The complexity and design of the OC model discussed in this chapter could vary significantly with the addition or alteration of just a few OCs in the modeling chain.

Conditioning on Operating Conditions

The use of OCs determines the causal conditioning associated with the Bayesian analysis. The a priori probability of events gained from modeling the sensor, target, environment, and ATC training OCs leads to a more thorough understanding of the a posteriori effects. One way to describe the analysis is much like filtering and learning, which compares the fusion output $P(B|R)$ versus $P(B)$ where $P(B|R)$ is based on the conditioning associated with likelihood of events (i.e., $p(B|A, V)\ p(A|V,R)\ p(V|R)$) from the ATCs visual (contextual) information.

Figure 13.11 details the delineation of a multitude of locations and backgrounds that lead to a more formalized understanding of environmental effects on target presence. Furthermore, the sensitivity of target presence due to background conditions would indicate the understanding of potential detection opportunities. For example, we see that the date and time (day and night) and location (Northern or Southern Hemisphere) determine the seasonal probability affecting the sensor model. In the chaining of the model, we determine the target detection probability as a function of the background and location (environment OCs), which leads to different ATC performance characterizations.



L	D	S(=1)	S(=0)
1	1	p_{11}	q_{11}
1	0	p_{12}	q_{12}
0	1	p_{13}	q_{13}
0	0	p_{14}	q_{14}

L(=1)	L(=0)
p_1	q_1

D(=1)	D(=0)
d_1	n_1

L	B(=1)	B(=0)
1	p_{1B}	q_{1B}
0	p_{0B}	q_{0B}

S	C(=1)	C(=0)
1	p_{1S}	q_{1S}
0	p_{0S}	q_{0S}

B	T(=1)	T(=0)
1	p_{1T}	q_{1T}
0	p_{0T}	q_{0T}

S	D	T(=1)	T(=0)
1	1	p_{11}	q_{11}
1	0	p_{12}	q_{12}
0	1	p_{13}	q_{13}
0	0	p_{14}	q_{14}

P	S(=1)	S(=0)
1	p_{1P}	q_{1P}
0	p_{0P}	q_{0P}

S	C	P(=1)	P(=0)
1	1	p_{11}	q_{11}
1	0	p_{12}	q_{12}
0	1	p_{13}	q_{13}
0	0	p_{14}	q_{14}

For each row:

$$\Sigma(p_{ij} + q_{ij}) = 1$$

Figure 13.11 Environment OC portion of BN model.

The Bayes net model is used to aid in the conditioning of a priori probability influences in the generalized decision-level fusion analysis and provides a minimial conidtioning possibility of target detections based on the known operating conditions.

Figure 13.12 shows the notional case in which the error associated without conditioning is higher than the decision-level fusion (DLF) results with conditioning. For example, if the ATC knows from a source of visual information that it is night, then it would weigh the fusion results from an IR system more than that of an EO output. As fusion results from contextual information [33] are accumulated (in this case, conditional evidence accumulation), the divergence between with and without conditioning grows. An alternative explanation is that the more specific the conditioning is to known operating conditions at the point of ATC or DLF, the higher the expected performance from the ATC or DLF algorithm [4]. Using the Dynamic Bayes net in a DLF, we simulate the use of the OC model for conditioning [Figure 13.12(a)] and the impacts on target detection, classification (i.e, type), and identification (i.e., allegiance) [Figure 13.12(b)]. As the number of OCs conditioned grows, target detection and classification increase with the target identity allegiance following the classification results. It is noted that the order of the OC conditioning would affect the slope of the curves in Figure 13.12.

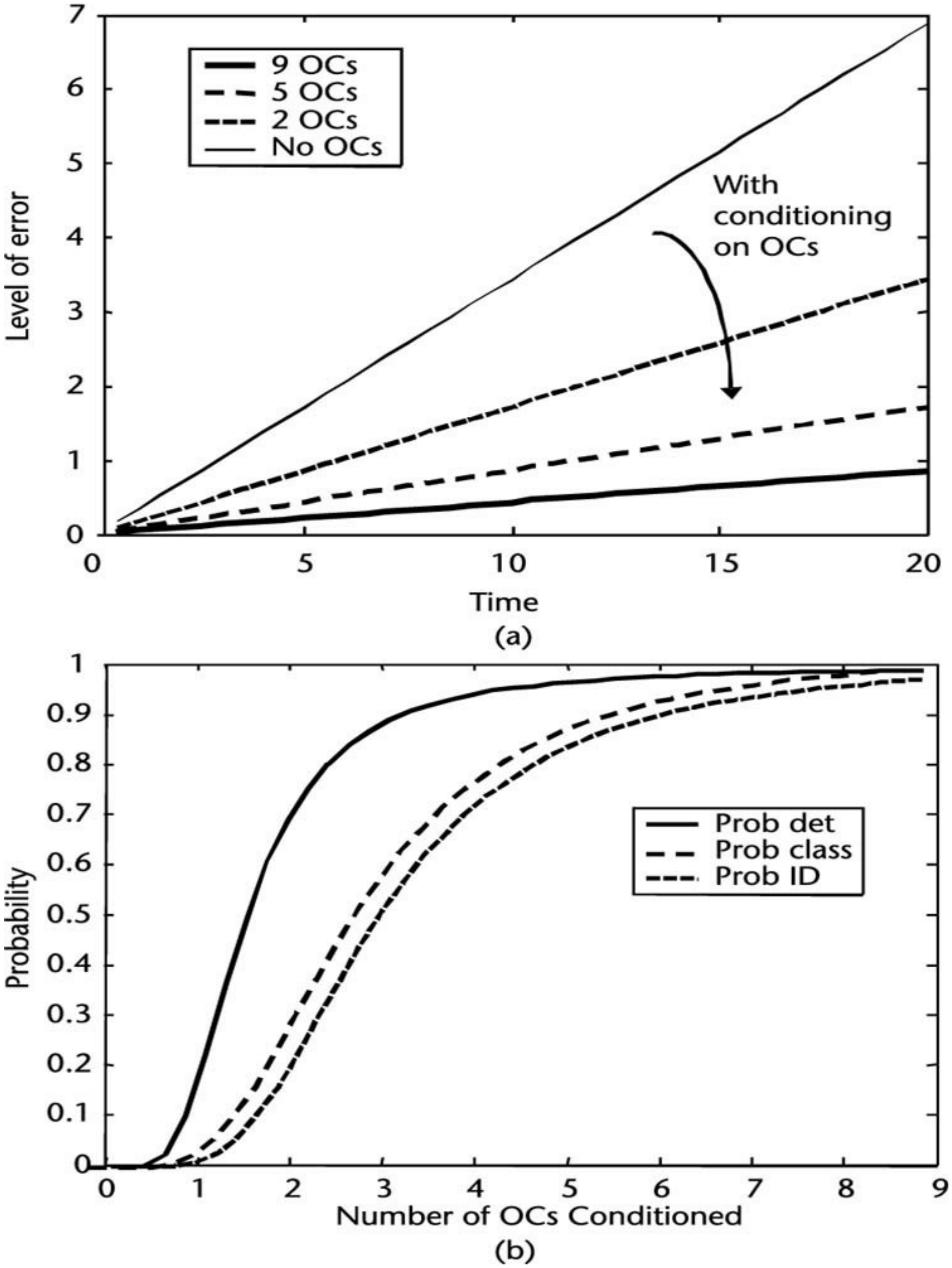


Figure 13.12 Impact of OC conditioning on target detection, classification and identification.

Conclusions

The design and development of an OC model for ATC system fusion assessment begins with an understanding of the basic structure of an ATC and the sensor phenomenology. OCs of sensors, targets, and environment have been extensively detailed in a target classification performance analysis to aid situational awareness. Since the OCs demonstrate the performance likelihood of an ATC to a given range of parameters, they can be used in a Bayes Net to better understand the causal links between OCs and decision-level fusion performance of a given scenario. The OC model produces probability distributions. The importance of conditioning the OCs produced by the model with those present in the ATC training library, or available to the fusion algorithm, reduces the amount of error in the ATC decision. Finally, some of the terminology used in OC modeling was refined for enhanced ATC fusion performance sensitivity analysis.

Standard and extended OCs were explained, along with the concept of direct, indirect, and derived OCs and how they relate to OC modeling. The OC model presented in this chapter is only one of numerous possible OC model designs with varying complexity. As the development of new sensor systems with better or different capabilities unfolds (such as feature-based classifiers [34]), new OC sensitivities will present themselves, other OCs will become less of a factor, and OC models will need to evolve to take these changes into account. A complete understanding of the relevant OCs and the ability to reasonably predict them will have huge payoffs in situation assessment. A high fidelity OC model would enable a more complete exploration of the sensitivity trade space over different scenarios of both ATC and fusion algorithms to the varying scenario parameters in the environment.

owledgments

Thanks go to Tim Ross and Bart Kahler for their contributions to the OC conditioning modeling.

rences

[1] Ross, T. D., J. J. Bradley, L. J. Hudson, and M. P. O'Connor, "SAR ATR—So What's The Problem? An MSTAR Perspective," *Proceedings of SPIE*, Vol. 3721, 1999.

[2] Kahler, B., E. Blasch, and L. C. Goodwon, "Operating Condition modeling for ATR Fusion Assessment," *Proceedings of SPIE*, Vol. 6571, 2007.

[3] Ross, T. D., and L. C. Goodwon, "Improved Automatic Target Recognition (ATR) Value Through Enhancements and Accommodations," *Proceedings of SPIE*,., Vol. 6237, 2006.

[4] Kahler, B., and E. Blasch, "Decision-Level Fusion Performance Improvement from Enhanced HRR Radar Clutter Suppression," *Journal of Advances in Information Fusion*, Vol. 6, No. 2, December 2011.

[5] Currie, A., "Synthetic Aperture Radar," *IEE Electronics & Communication Engineering Journal*, August 1991, pp. 159–170.

[6] Crawford, F. J., "Electro-Optical Sensors Overview," *IEEE AES Systems Magazine*, October 1998, pp. 17–24.

[7] Chan, L. A., S. Z. Der, and N. M. Nasrabadi, "Dualband FLIR Fusion for Automatic Target Recognition," *Information Fusion*, Vol. 4, 2003.

[8] Rizvi, S. A., and N. M. Nasrabadi, "Fusion of FLIR Automatic Target Recognition Algorithms," *Information Fusion*, Vol. 4, 2003.

[9] Thornburg, R., P. Maciejewski, R. Claypool, T. Horn, M. Jarratt, and L. Westerkamp, "Evaluation Methodology for Hyperspectral Automatic Target Cueing Systems," *Proceedings of SPIE*, Vol. 4725, 2002.

10] Waltz, E., and J. Llinas, *Multisensor and Data Fusion*. Norwood, MA: Artech House, 1990.

11] Blasch, E., M. Pribilski, B. Daughtery, B. Roscoe, and J. Gunsett, "Fusion Metrics for Dynamic Situation Analysis," *Proceedings of SPIE*, Vol. 5429, 2004.

12] Hanselman, P., C. Lawrence, E. Fortunano, B. Tenney, and E. Blasch, "Dynamic Tactical Targeting," *Proceedings of SPIE*, Vol. 5441, 2004.

13] Keydel, E. R., S. W. Lee, and J. T. Moore, "MSTAR Extended Operating Conditions: A Tutorial," *Proceedings of SPIE*, Vol. 2757, 1997.

14] Mossing, J. C., and T. D. Ross, "An Evaluation of SAR ATR Algorithm Performance Sensitivity to MSTAR Extended Operating Conditions," *Proceeding. of SPIE*, Vol. 3370, 1998.

15] Morgan, D., and T. D. Ross, "A Bayesian Framework for ATR Decision-Level Fusion Experiments," *Proceedings of SPIE*, Vol. 6571, 2007.

16] Costa, P. C. G., K-C. Chang, K. B. Laskey, and R. N. Carvalho, "A Multidisciplinary Approach to High Level Fusion for Predictive Situational Awareness," *International Conference on Information Fusion*, 2009.

17] Zeng, Z., and Q. Ji, "Knowledge Based Activity Recognition with Dynamic Bayesian Network," *European Conference on Computer Vision*, 2010.

18] Das, S., *High-Level Data Fusion*, Norwood, MA: Artech House, 2008.

19] Lauberts, A., M. Karlsson, F. Nasstrom, and R. Aljasmi, "Ground Target Classification Using Combined Radar and IR with Simulated Data," *Proceedings of the International Conference on Information Fusion*, 2004.

20] Chen, P., "HRR Profiling In GMTI Search Radar," *IEEE International Radar Conference*, 2000.

21] Grantham, J. W., and E. C. Meidunas, "Laser Radar in Adverse Weather," *Proceedings of SPIE*, Vol. 3380, 1998.

22] O'Kane, B., I. Biederman, and E. Cooper, "Modeling Parameters for Target Identification: A Critical Features Analysis," *Proceedings of the IRIS Passive Sensors Symposium*, 1996.

23] Sadjadi, F., and C. Chun, “Passive Polarimetric IR Target Classification,” *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 37, No. 2, 2001, pp. 740–751.

24] Ren, H., and C. Chang, “Automatic Spectral Target Recognition in Hyperspectral Imagery,” *IEEE Transactions on Aerospace And Electronic Systems*, Vol. 39, No. 4, 2003 pp. 1232–1248.

25] Boyd, R. W., *Radiometry and the Detection of Optical Radiation*, New York: Wiley, 1983.

26] Blasch, E. P., and T. Connare, “Improving Track Maintenance Through Group Tracking,” *Proceedings of the Workshop on Estimation, Tracking, and Fusion; A Tribute to Yaakov Bar-Shalom*, Monterey, CA, May 2001, pp. 360–371.

27] Ling, H., Y. Wu, E. Blasch, G. Chen, and L. Bai, “Evaluation of Visual Tracking in Extremely Low Frame Rate Wide Area Motion Imagery,” *International Conference on Information Fusion*, 2011.

28] Ling, H., L. Bai, E. Blasch, and X. Mei, “Robust Infrared Vehicle Tracking Across Target Change using L1 Regularization,” *International Conference on Information Fusion*, 2010.

29] Blasch, E., D. Pikas, and B. Kahler, “EO / IR ATR Performance Model to Support Fusion Experimentation,” *Proceedings of SPIE*, Vol. 6566, 2007.

30] Blasch, E., and B. Kahler, “Multi-Resolution EO/IR Tracking and Identification,” *International Conference on Information Fusion*, 2005.

31] Wang, T., Z. Zhu, and E. Blasch, “Bio-Inspired Adaptive Hyperspectral Imaging for Real-Time Target Tracking,” *IEEE Sensors Journal*, Vol. 10, No. 3, 2010, pp. 647–654.

32] Blasch, E., E. Lavely, and T. Ross, “Fidelity Metric for SAR Performance Modeling,” *Proc. of SPIE*, Vol. 5808, April 2005.

33] Blasch, E. P., and S. Plano, “Cognitive Fusion Analysis Based on Context,” *Proceedings of SPIE*, Vol. 5434, 2004.

34] Chen, Y., E. Blasch, G. Chen, T. Qian, and H. Chen, “Experimental Feature-Based SAR ATR Performance Evaluation Under Different Operational Conditions,” *Proceedings of SPIE*, Vol. 6968, April 2008.

Part V
Measures of Effectiveness

CHAPTER 14

A Toolbox for the Evaluation of Surveillance Strategies Based on Interpreted Systems

Patrick Maupin, Anne-Laure Jousselme, Hans Wehn, Snezana Mitrovic-Mimic, and Jens Happe[1] (Canada)

Introduction

Situation awareness (SAW) is essential to conduct decision-making activities. SAW is about the perception of the elements in the environment, the comprehension of their meaning, and the projection of their status in the near future [3]. Situation analysis (SAN) is defined as a process, the examination of a situation, its elements, and their relations, to provide and maintain a product, such as a state of situation awareness for the decision maker [4, 5].

The situation analysis process has to evaluate information with both knowledge and uncertainty. A formalization is necessary if one is interested in the reproducibility of results, complex space-time problem solving, and in a language to represent and reason about dynamic situations. In Chapter 4 [6], we proposed a formal model for situation analysis based on the interpreted systems semantics [7], a framework in which knowledge, information and uncertainty can be represented, combined, managed, reduced, increased, and updated.

The general problem of decision support can be split into two principal tasks: situation analysis and planning. These two tasks are not easily separable because the actions carried out according to the plan defined during the task of planning modify the state of the environment and of the agents. The act of planning modifies the information to which the agents have access (either their own or a collective epistemic state) and their situational awareness. Our approach to situation analysis considers that the situation is created by the execution of a joint protocol for the agents P in interaction with the environment's protocol P_e, the latter including a possible opponent. The interpreted systems semantics will be used as the single specification language for both planning and situation analysis [8].

In this chapter, we present a situation analysis toolbox (SAT), which implements the theoretical concepts put forth in our previous works [6, 8, 9] and was presented in Chapter 4. The general idea behind this toolbox is to build situations and analyze them. We use the formalism of the noncooperative dynamic game theory [10] to model the situation. In its most general form, it consists of a game between two teams having opposite goals, the players jointly seeking to maximize (or minimize) a function of distance between targets. The differential game theory of Isaacs [11], dating back to the 1960s, made it possible to model the pursuit-evasion game a simplified version of the problem of rendez-vous [12]. Isaacs' theory can be used to model many military problems by combining the traditional game theories of von Neuman and Morgenstein [13] (used especially in economy), variations calculus, and theory of optimal control. The interest to use such a framework for modeling the situations to be further analyzed is the existence of some analytically derivable optimal solutions, thus helping to measure the quality of solutions for simple cases. Another advantage of the game-theoretical framework is its great flexibility. Indeed, by slightly modifying the constraints of the basic game, a large range of situations can be easily modeled: collision avoidance, target tracking, or stowing.

SAT consists of five components, or subtoolboxes: the Discretization Toolbox, which allows an abstraction of a continuous environment into both visibility and navigation graphs; the Behavior Simulation Toolbox, which enriches the environment with containment probability maps; the State Generation Toolbox, which builds a transition state system based on joint strategies of some agents constrained by the context defined in the discretization toolbox and the behavior simulation toolbox; the State Searching Toolbox, which plays the role of a model checker and temporal logical formulas verification in the transition state system; and the Visualization Toolbox, which renders states and graphs on screen.

This chapter is organized as follows. In Section 14.2, basic notions of pursuitevasion games are given and the way situations are generated by motion strategies is detailed. Section 14.3 presents the situation analysis toolbox and details of its five components, together with their related theoretical concepts over a countersmuggling scenario situated in Howe Sound (located on Canada’s West Coast, northwest of Vancouver). Finally, Section 14.4 provides further potential use and applications of the SAT.

The reader is referred to Chapter 4 for the theoretical background on interpreted systems and for the formal definitions of situation, situation awareness, and situation analysis within this framework.

Situations Generated by Motion and Sensing Strategies

The games studied in this chapter are special cases of motion and sensing problems, which can be grouped into four categories [14]: sensor assignment, sensor placement, exploration, and pursuit-evasion. These four problems can be seen as special cases of a more general problem, involving single or multiple sensors, static or mobile sensors, and a searched target that is either punctual or distributed across the environment. The examples illustrating the functionalities of the SAT toolbox deal primarily with the resolution of pursuit-evasion (PE) games and sensor placement (SP), although we believe that the problems of exploration and sensor assignment can also be studied with minor modifications to the existing toolbox.

Local states encode physical terrain positions and hence the possible states of the agents can be stacked up on the map, allowing convenient illustrations. See Figure 14.4(b) for an example.

(a)

(b)

Figure 14.4 GIS map and associated visibility graph with unlimited range sensors over the set of reflex vertices. The agent’s knowledge of the environment is built through an exploration algorithm whose aim is to learn the environment while exploring the navigation graph. Several algorithms such as the one described in [26] and shown in Algorithm 1 are available.

14.2.1 Visibility-Based Pursuit-Evasion in Graphs

Problem solving situations, such as pursuit-evasion, are quite common, and come in various flavors. Depending on the scientific community, many synonyms are used to designate somewhat similar problems, such as art gallery problems in computational geometry [15, 16]; graph searching in computer science [17–19]; rendezvous problems in operations research [20–22]; and differential games in control theory [11].

A rough classification of pursuit evasion problems is based on the following variables: (1) whether the environment is continuous or discrete; (2) the pursuit success criterion is of *either* kind (being caught or not) or of degree (getting close to evader for more than n seconds); or (3) whether the problem involves a single pursuer (evaders) or multiple pursuers (evaders). Other agent-related variables distinguishing the pursuit-evasion problems involve sensing (whether agents' visibility is limited or not, uncertain or not); knowledge (whether the agents are aware of the environment's map, of their own or other agents' position, and whether there is or not an evader or pursuer in the environment); communication (whether multiple pursuer can communicate and thus coordinate the actions); steering ability (unequal agility for pursuers and evaders); motion capability (locally constrained, or jumping to other positions instantly); and memory (whether the agents remember their past moves).

We use the formalism of the noncooperative dynamic game theory [10] to model the situation. Pursuit-evasion (PE) games are a family of problems in which one group attempts to track down members of another group in an environment. The discrete formulation of the problem is also called *graph searching* and is due to Parsons [23]. Pursuer and evader agents are thus constrained to move within a graph whose nodes are possible positions (locations) and whose edges denote paths between two locations.

In its most general form, PE consists of a game between two teams having opposite goals, the players jointly seeking to maximize (or minimize) a function of distance.

We base our formulation on the work of Lavalle et al. on information spaces [24, 25]:

1. Let us consider an environment as a weighted (or labeled) graph $G = (V, \varepsilon)$, where V is the set of the vertices (or nodes), and ε is the set of edges, $\varepsilon = V \times V$. The weight d on the edge represents the distance between the two vertices connected by the edge.
2. Suppose there are n pursuers $p_1, p_2, \cdots, p_n$ and one evader p_e, each of which can be at any vertex of G. The positions of pursuer p_i, $i = 1, \cdots, n$ and evader e are v_i

and v_e respectively. So the state of system is $(v_1, \cdots, v_n, v_e)$, and the state space is $X = V^{n+1}$.

3. Each agent has a visibility sensor of sensing range ρ meaning that the agent can see a node if it is within its range. Let us denote by $\rho(v)$ the set of all the nodes that the agent can see at position v. The pursuers and the evader can have different visibility sensors with different sensing ranges. The observation space is the collection of subsets of V.
4. Capture occurs when a pursuer and the evader are at the same position (node) at the same time.
5. The speeds of pursuer p_i and evader are ω_i and ω_e respectively. For the sake of simplicity, we define the speed to be 1 for all the agents, meaning the agent moves from one node to another in one step, possibly accounting for edge's weight.
6. A basic action a_i for agent i is to move from one vertex to an adjacent vertex along an edge.
7. The initial state of the pursuers is either set to some desirable nodes or determined randomly. For the evader, it is set randomly.
8. The evader's strategy P_e which is unknown to the pursuers, is (see Figure 14.1(a)):
   a. If the pursuer is not visible, the evading agent stays on its actual position (reactive agent).
   b. If the evader sees one of the pursuers, it moves to maximize its reward function. If we define the reward function as the distance between the closest pursuer and the evader $d(p_i,e)$, the strategy for the evader is to maximize the distance between itself and any pursuer. If there are many such nodes, the evader moves to minimal index node.
9. The pursuers' strategy P_p is (see Figure 14.1(b)):
   a. If the evader is not visible to the pursuers, the pursuers move based on a depth first search algorithm.
   b. If the pursuers can see the evader, they move to minimize their cost function $d(p_i e)$.

```
case of
   if PosE=PosP do Λ
   if PosP∈Neighbours (PosE) do move(farthest node)
   else Λ
   observe
end case
```

```
case of
   if PosP=PosE do Λ
   if PosE∈Neighbours(PosP) do move(PosE)
   else move(any adjacent node)
   observe
end case
```

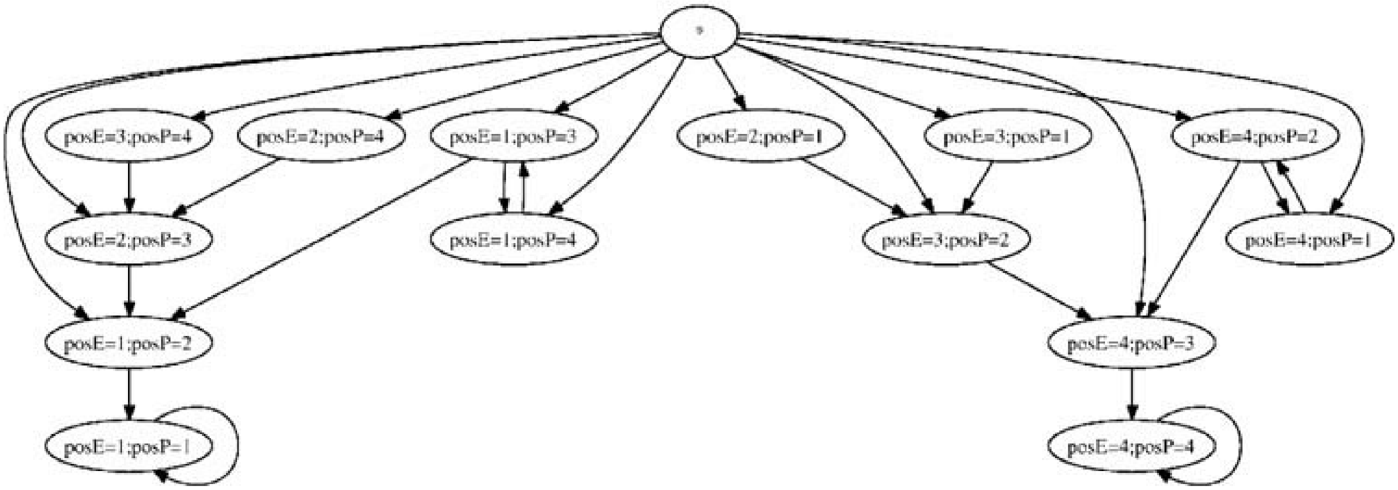


Figure 14.1 Strategies of two opponent players and corresponding state transition system in a simple case. Λ is the null action and does nothing.

Figure 14.1 shows examples of protocols for both pursuer P_p and evader P_e agents together with the corresponding state space. A very simple case consisting of only one pursuer, one evader, and four possible positions (four nodes in the navigation graph) is shown, since the state transition system becomes rapidly huge and consequently difficult to visualize. As it will be introduced in Section 14.3.2, neighbors are defined in two ways: (1) visibility graph and (2) navigation graph. In the latter case, a pursuer moving into a neighboring cell with an evader catches it, while in the former, it may not.

14.2.2 Sensor Placement Problem

From a formal point of view, the simplest version of the sensor placement problem can be seen as a motion and sensing problem involving multiple, static agents aiming at maximizing their sensing coverage of a given surface of interest. Most of the time, a joint objective for the modeler is to minimize the number of sensors used. On the other hand, in the simplest pursuit-evasion problem, the aim will be for a pursuer to minimize the distance to an evader and vice versa.

14.2.3 Exploration

The exploration problem is defined by Isler [14] as: "generate a trajectory for the robot so that the robot senses every point in an unknown environment as quickly as possible." An algorithm for coordinated exploration of an unknown environment applied to PE games has been proposed in [26] and is implemented in the SAT (see Algorithm 1). This approach allows the agents to acquire knowledge about the environment.

Algorithm 1: Exploration algorithm of Pellier and Fiorino [26]: Explore(*edb*, *p*, γ_e)

Require:

edb the depth exploration bound
p the current position of the pursuer

Ensure:

p the current position updated
γ_e the exploration path

```
i := 0; $\gamma_e := \varnothing$; $\gamma_e^* := \varnothing$; $N_n^* :=$ Unexplored($G_n$);
while $i < edb$ and $N_n^* = \varnothing$ do
    $N_n^* :=$ Unexplored($G_n$)
    for all $V_c \in N_n^*$ do
        $\gamma_e^* =$ ComputeMotion($G_n$, $p$, $V_c$);
        ExecuteMotion($\gamma_e^*$);
        Update($G_n$,VisibleVerticesFrom($V_c$));
        $\gamma_e := \gamma_e + \gamma_e^*$;
        MarkExplored($G_n$,$V_c$);
        $p := V_c$;
    end for
    $i$++;
end while
```

Situation Analysis Toolbox

The SAT software is intended to provide tools for generating situations and analyzing them. A key component of a situation is its context, which will be mainly represented here by the terrain on which the situation takes place. The area of interest is represented in the form of a polygon (i.e., boundary of the search area) with holes (which represent obstacles that the agents can neither see nor pass through). The software is divided into five subtoolboxes, which serve different functions: (1) discretization toolbox, (2) state generation toolbox, (3) state searching toolbox, (4) behavior simulation toolbox, and (5) visualization toolbox.

The high-level data flow diagram given in Figure 14.2 shows how these components are connected. First, the Discretization Toolbox provides an abstraction of the terrain into a visibility graph and a navigation graph, which will then serve as the basis for defining a part of the context γ and the set of possible global states S for the system. Another component of the context comes from the Behavior Simulation Toolbox, which provides agent containment probability maps of the area under consideration. The State Generation Toolbox inputs both the context $\gamma = (P_e, S_0, \tau, \Psi)$ and the strategies of the agents $P = [P_1, \ldots, P_n]$, and outputs a set of possible states. The situation thus built (i.e., the transition state system) can then be analyzed through the State Searching Toolbox, whose purpose is to answer logical queries of interest. The Visualization Toolbox provides graphical user interface for the SAT.

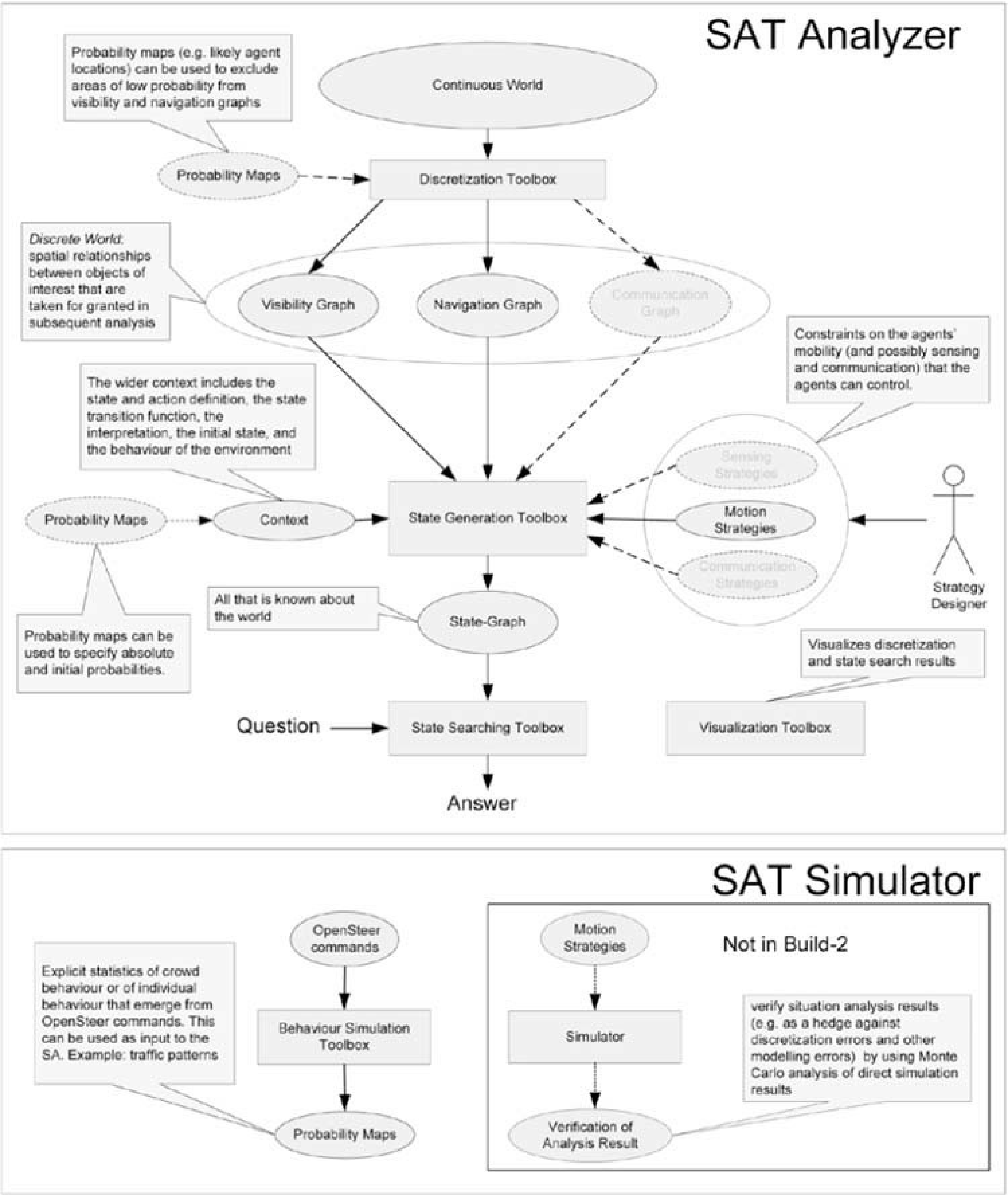


Figure 14.2 The Situation Analysis Toolbox.

14.3.1 Countersmuggling Vignette

We illustrate the SAT on a countersmuggling vignette shown in Figure 14.3. In this figure, several boats manned by smugglers (the hostile agents) operate in Howe Sound (which is close to Vancouver). They attempt to go from their hiding places to target locations where their illegal goods can be exchanged. These agents try to avoid obstacles such as islands and avoid colliding with each other. They also try to keep a distance away from the neutral agents such as ferries and pleasure crafts to avoid being seen and perhaps cause suspicion. More specifically, the neutral agents can be ferries or ships that follow a specific route, or boats that sail in any random direction. The locations of the home and target harbors are determined probabilistically.

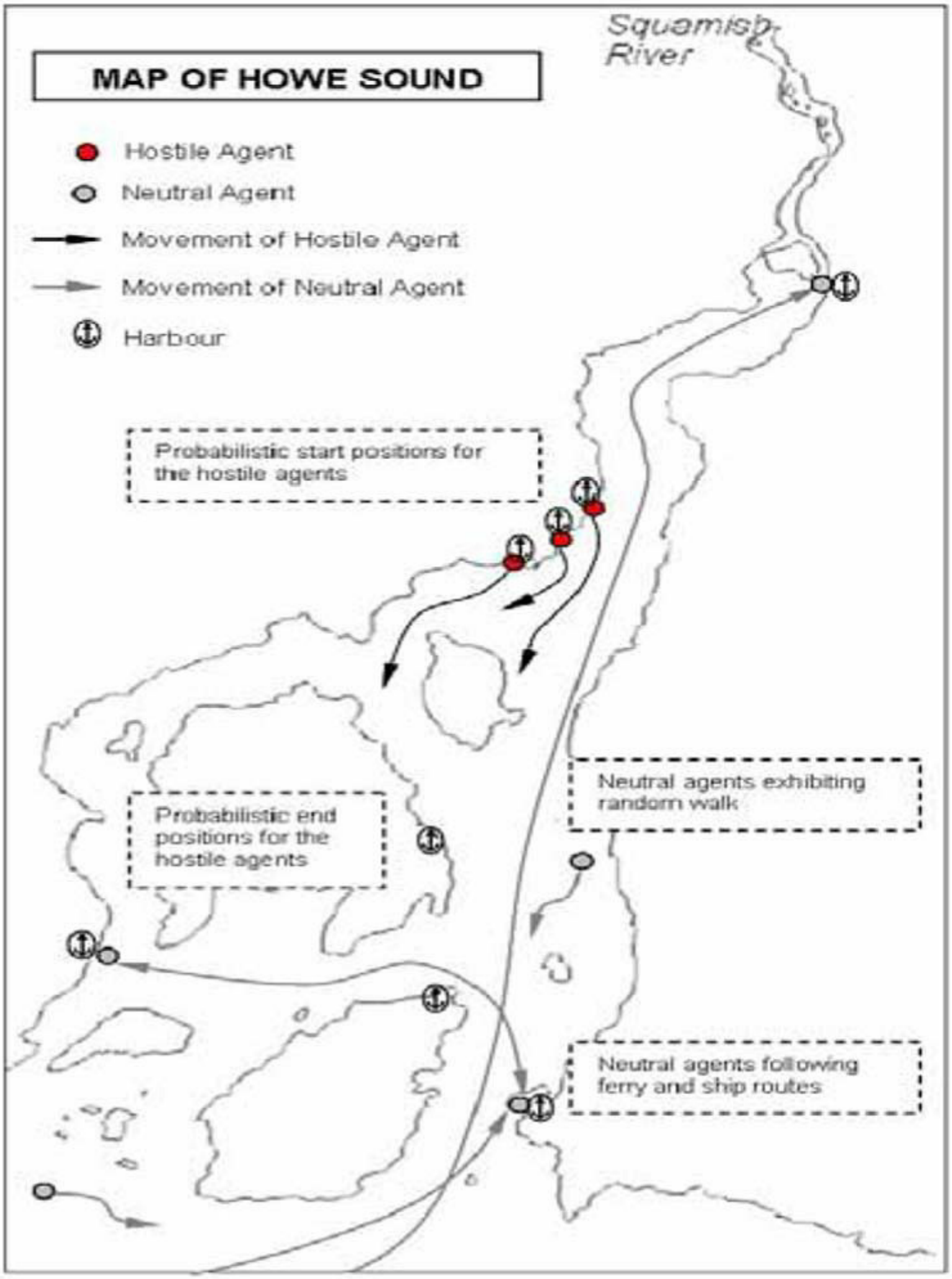


Figure 14.3 A smuggling operation has been reported in Howe Sound (northwest of Vancouver on Canada's West Coast). Can we guarantee that the smugglers will be detected?

14.3.2 The Discretization Toolbox

The main task of the Discretization Toolbox is to provide discrete representations of the real-world continuous search space geometry, suitable for situation analysis in the context of surveillance problems. The toolbox supports discrete representations in the form of either convex polygons (cells) or points, shown in Figure 14.4. Two main graphs are computed by the Discretization Toolbox: a visibility graph and a navigation graph. The visibility graph of a set of points Q in a polygonal environment is a graph $G_V(Q)$, whose nodes are the points in Q and in which two vertices are adjacent if they are visible from each other in the environment. An example for the countersmuggling scenario is shown in Figure 14.4. The Discretization Toolbox provides several options for selecting the set Q from among the points in the polygonal environment. The navigation graph G_N defines the transition function τ for the agents. In an agent's navigation graph, the vertices are either points or cells, and two cells/points are connected if an agent can move directly from the one to the other in one time step. As a rule, the navigation graph is a subgraph of the visualization graph as two adjacent points/cells are always visible from one another.

Figure 14.5 shows one sensor placement solution (Figure 14.5(b)) together with its corresponding discretization (Figure 14.5(a)) of the environment with a visibility range of 2 km. In these figures, all the islands (obstacles) are retained in the map.

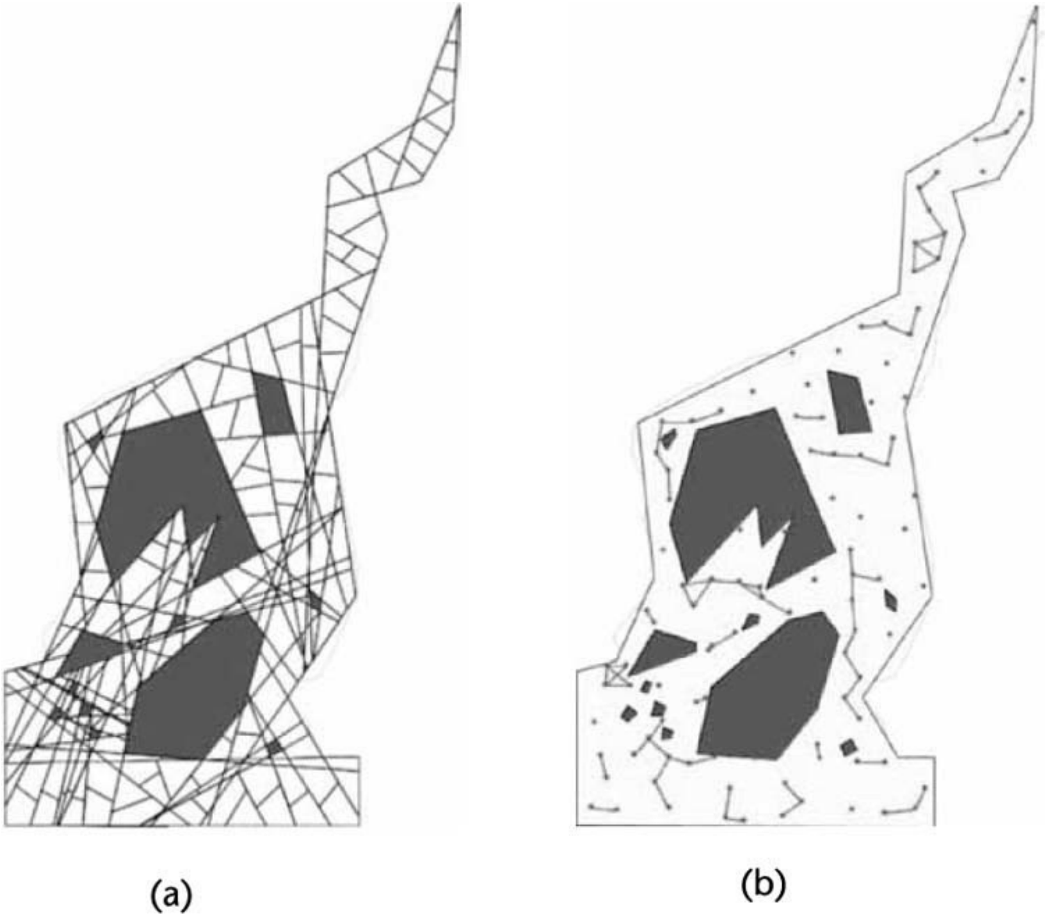
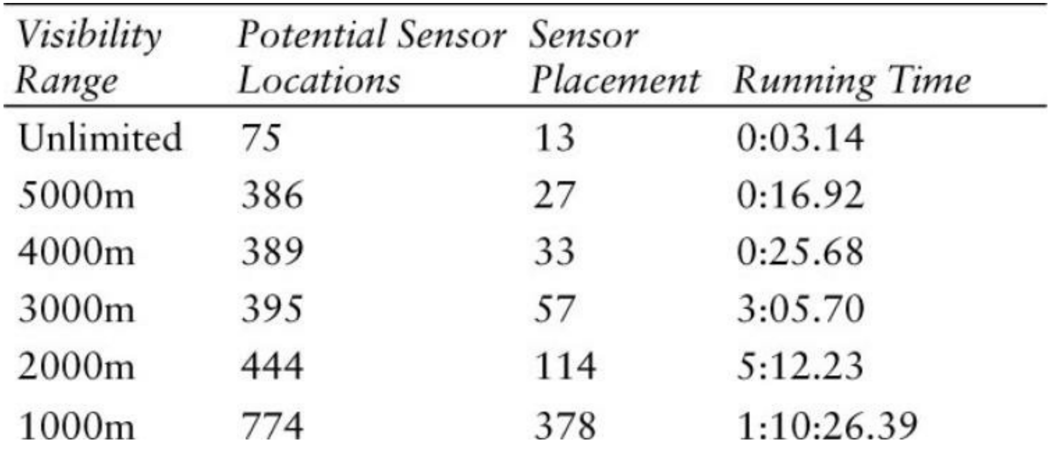


Figure 14.5 Cell decomposition of the environment and associated sensor placement solution for a visibility range of 2 km and a simplification of 1,000m.

The initial Howe Sound map has been simplified using a simplification parameter of 1000m, meaning that all the lines smaller than 1000m are removed. The simplification is used because the algorithm for positioning sensors with limited visibility range generates a search space decomposition as its initial step and decomposition can be computationally expensive.

Decreased sensor visibility range or deteriorating weather conditions may cause the increase in the number of sensors needed to monitor the area. Table 14.1 gives indications on the number of sensors required when the visibility range changes from 5km to 1km. The sensors are positioned until the 100% coverage is achieved[2].

Table 14.1 Sensor Placement for Different Visibility Ranges



Visibility Range	*Potential Sensor Locations*	*Sensor Placement*	*Running Time*
Unlimited	75	13	0:03.14
5000m	386	27	0:16.92
4000m	389	33	0:25.68
3000m	395	57	3:05.70
2000m	444	114	5:12.23
1000m	774	378	1:10:26.39

14.3.3 The State Generator

The State Generation Toolbox strives to be an implementation of the concepts and abstractions introduced in the first part of this chapter. It incorporates the concepts of state, agent, action, transition function, and strategy. It provides a state graph which represents an interpreted system $\mathcal{I}$ for the purpose of testing agent behavior strategies. This module takes as main inputs the abstraction of the terrain under the form of both the visibility and navigation graphs computed by the Discretization Toolbox.

Other inputs are the agents' local states definition l_i, which may include other parameters than the positions; the agents's protocols P_i; the context $\gamma = (P_e, S_0, \tau, \Psi)$; and the interpretation function π, which indicates whether some Boolean formulas ϕ are true in global states s. Given a state, the State Generation Toolbox outputs a set of successor states. Figure 14.6 is a display of the state space corresponding to a search in a PE game. Circles represent global states and are colored in light gray according the proportion of cleared cells; a black circle means that no state has been cleared, while a full light circle means that all the states have been cleared.

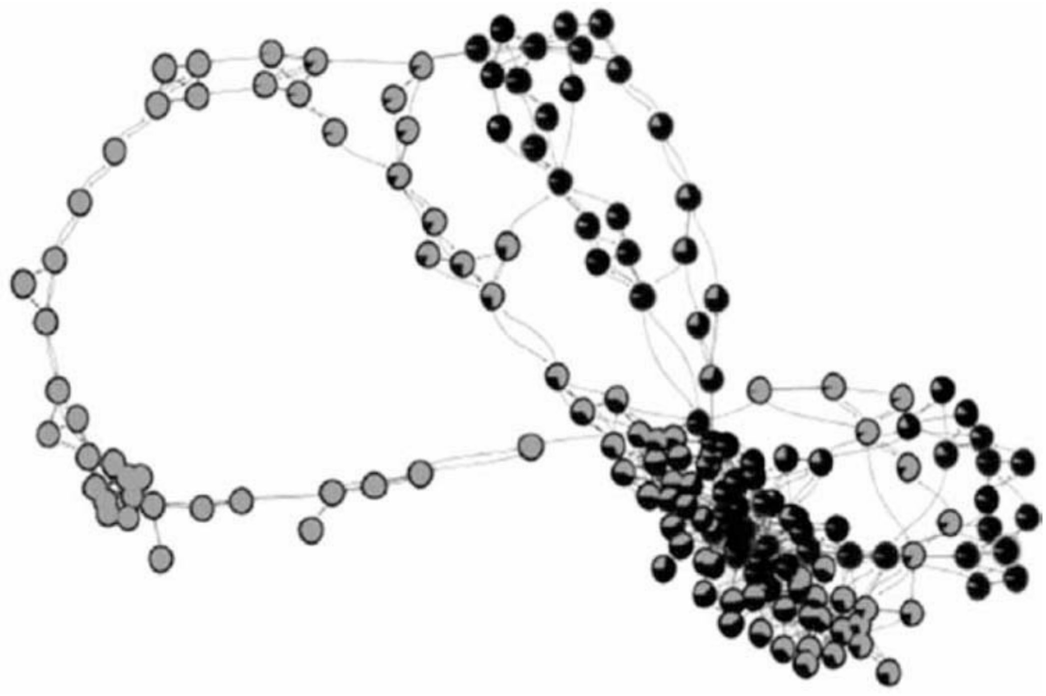
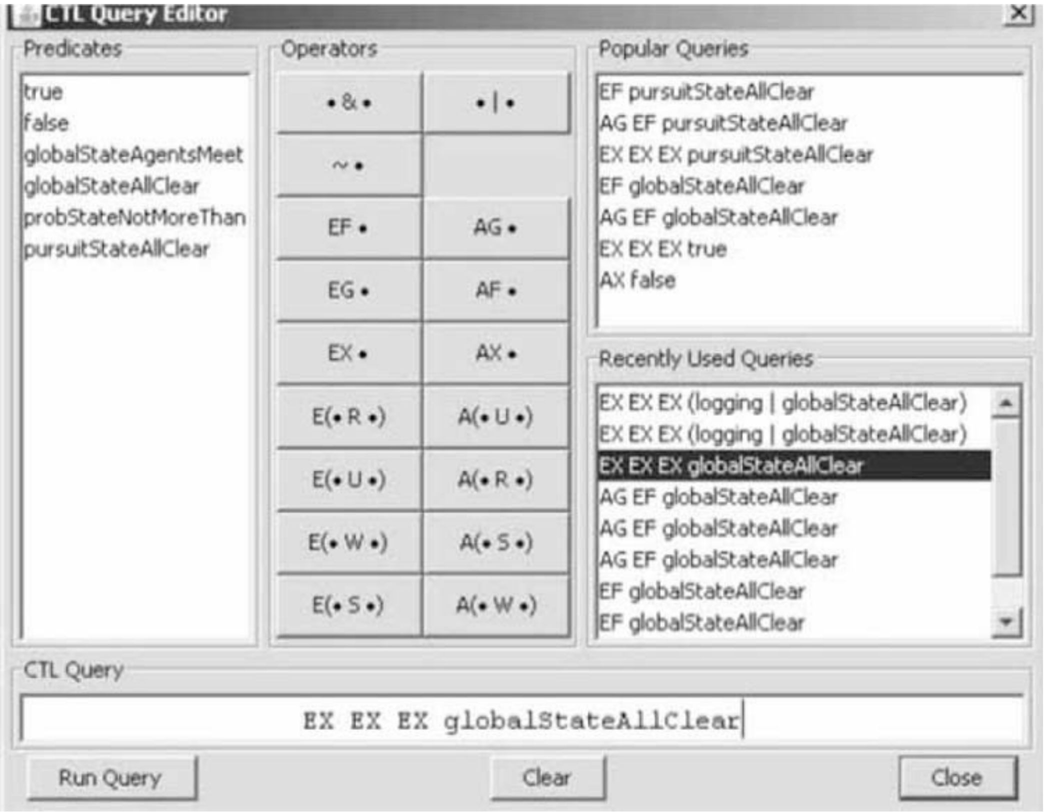


Figure 14.6 Example of display of the state space.



Popular Queries:

EF pursuitStateAllClear
AG EF pursuitStateAllClear
EX EX EX pursuitStateAllClear
EF globalStateAllClear
AG EF globalStateAllClear
EX EX EX true
AX false

Recently Used Queries:

EX EX EX (logging | globalStateAllClear)
EX EX EX (logging | globalStateAllClear)
EX EX EX globalStateAllClear
AG EF globalStateAllClear
AG EF globalStateAllClear
AG EF globalStateAllClear
EF globalStateAllClear
EF globalStateAllClear

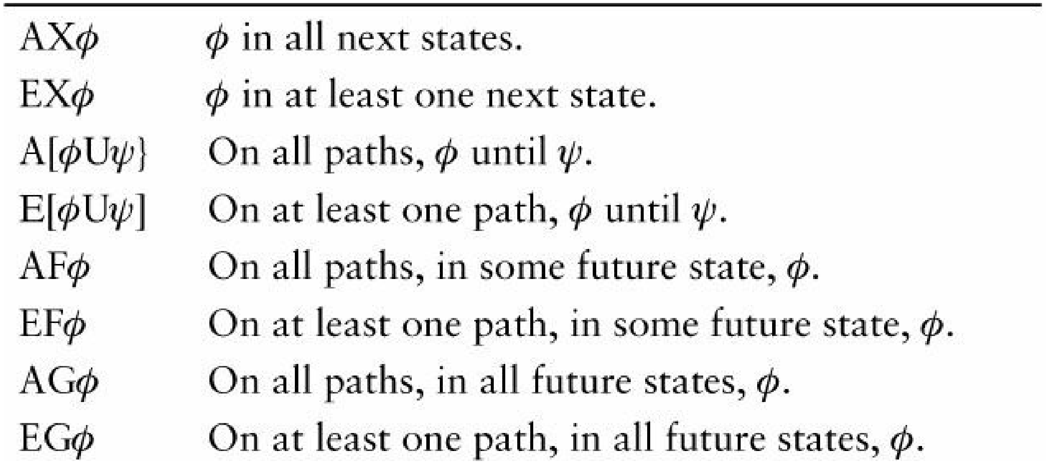
CTL Query: `EX EX EX globalStateAllClear`

Figure 14.7 CTL query Graphical User Interface.

14.3.4 The State Searching Toolbox

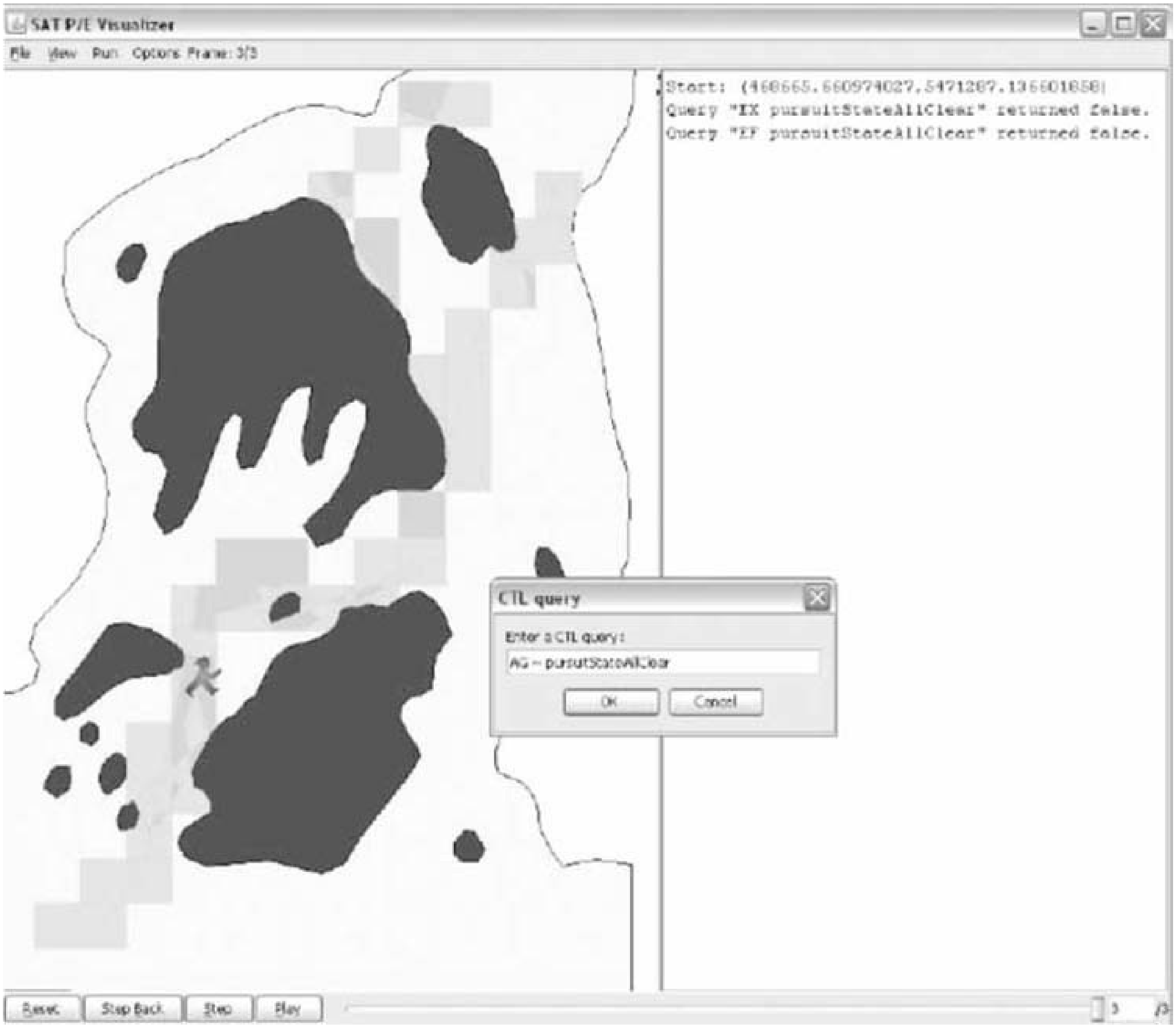
The main task of the State Searching Toolbox is to provide answers to formal queries by searching the state graph $\mathcal{G}$. This module takes as input the state transition function τ and a query ϕ, and outputs the query result. Although basic answers to queries on state transition systems are Boolean (either true or false), degrees can also be computed ranging from probably true to probably false. Also, additional statistical measures can be computed, such as the number of states where ϕ is true/false; the number of paths where ϕ is true/false; the length of path where ϕ is true/false; the length of path before encountering a counterexample; the average length of path; and the ratio of states/paths. The formal language of queries is computational tree logic (CTL), which is propositional logic over the states in a state graph, augmented by the list of operators shown in Table 14.2.

Table 14.2 CTL Operators



AXϕ	ϕ in all next states.
EXϕ	ϕ in at least one next state.
A[ϕUψ}	On all paths, ϕ until ψ.
E[ϕUψ]	On at least one path, ϕ until ψ.
AFϕ	On all paths, in some future state, ϕ.
EFϕ	On at least one path, in some future state, ϕ.
AGϕ	On all paths, in all future states, ϕ.
EGϕ	On at least one path, in all future states, ϕ.

The State Searching Toolbox plays the role of a model checker and thus serves as the situation analysis tool proper. Figure 14.8 shows an instance of the counter smuggling vignette. The previously computed restricted search space is discretized into cells. In a first step, cells have been selected through a threshold on the probability map (see Figure 14.10(b)). In a second step, a subset of the thresholded cells is further selected. These selected cells are sufficient to cover all thresholded cells using infinite-range, omnidirectional sensors. The user marks the start position of the pursuer, who is supposed to systematically search for evaders in the gallery. The predicate *pursuitStateAllClear* used in the query ascertains that (all cells in) a given state is clear of evaders—or equivalently, that any evader present in the search area has been caught. The results of past CTL queries are shown on the right-hand side of the screen. In this case, they return false because the topology of the area allows an evader to hide behind islands and avoid detection, no matter how the pursuer moves. Multiple pursuers would be necessary to completely sweep the area and guarantee detection.



```
Start: (468665.660974027,5471287.136601858|
Query "EX pursuitStateAllClear" returned false.
Query "EF pursuitStateAllClear" returned false.
```

Enter a CTL query: AG ~ pursuitStateAllClear

Figure 14.8 Executing queries using the Visualization Toolbox for the counter smuggling vignette.

14.3.5 The Behavior Simulation Toolbox

The main task of the Behavior Simulation Toolbox (see Figure 14.9) is to extract probability maps for agent containment from OpenSteer [3] multiagent simulations. The OpenSteer is a library implementing steering behaviors originally designed by Craig Reynolds [27] to model coordinated motion of animals, such as flying birds. Given the description of the scenario in Section 14.3.1, the Behavior Simulation Toolbox allows the generation of a map of the most likely locations for finding the smugglers. The main input to the Behavior Simulation Toolbox is a Geographic Information Systems (GIS) map corresponding to the area of interest. The output is a probability map, such as a set of polygonal cells marked up with their probability of agent containment, as seen in

Figure 14.10.

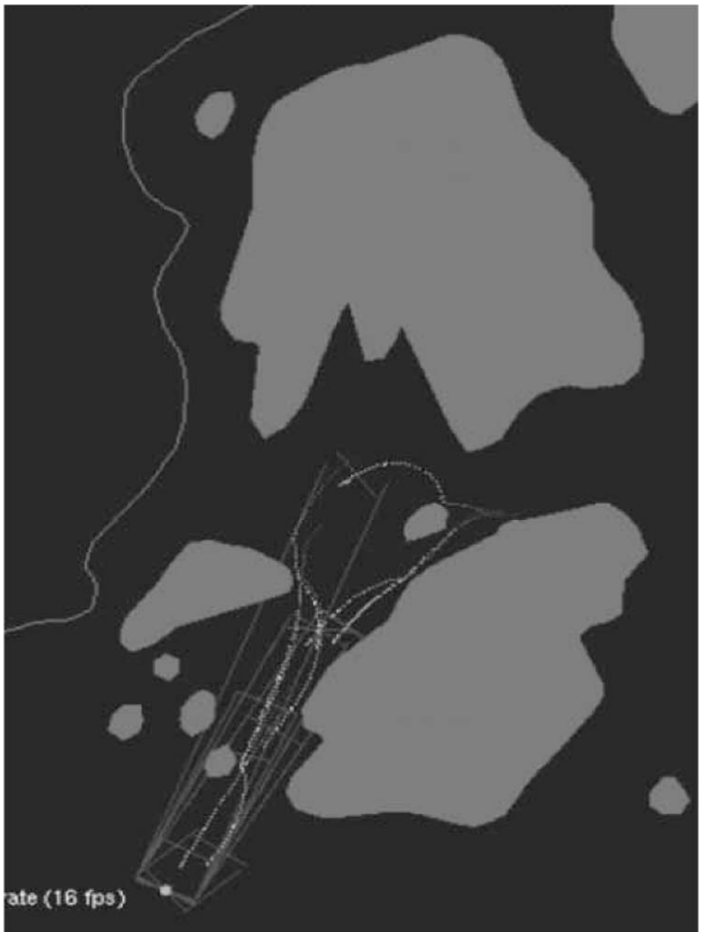


Figure 14.9 Screen shot of the Behavior Simulation Toolbox for the countersmuggling vignette.

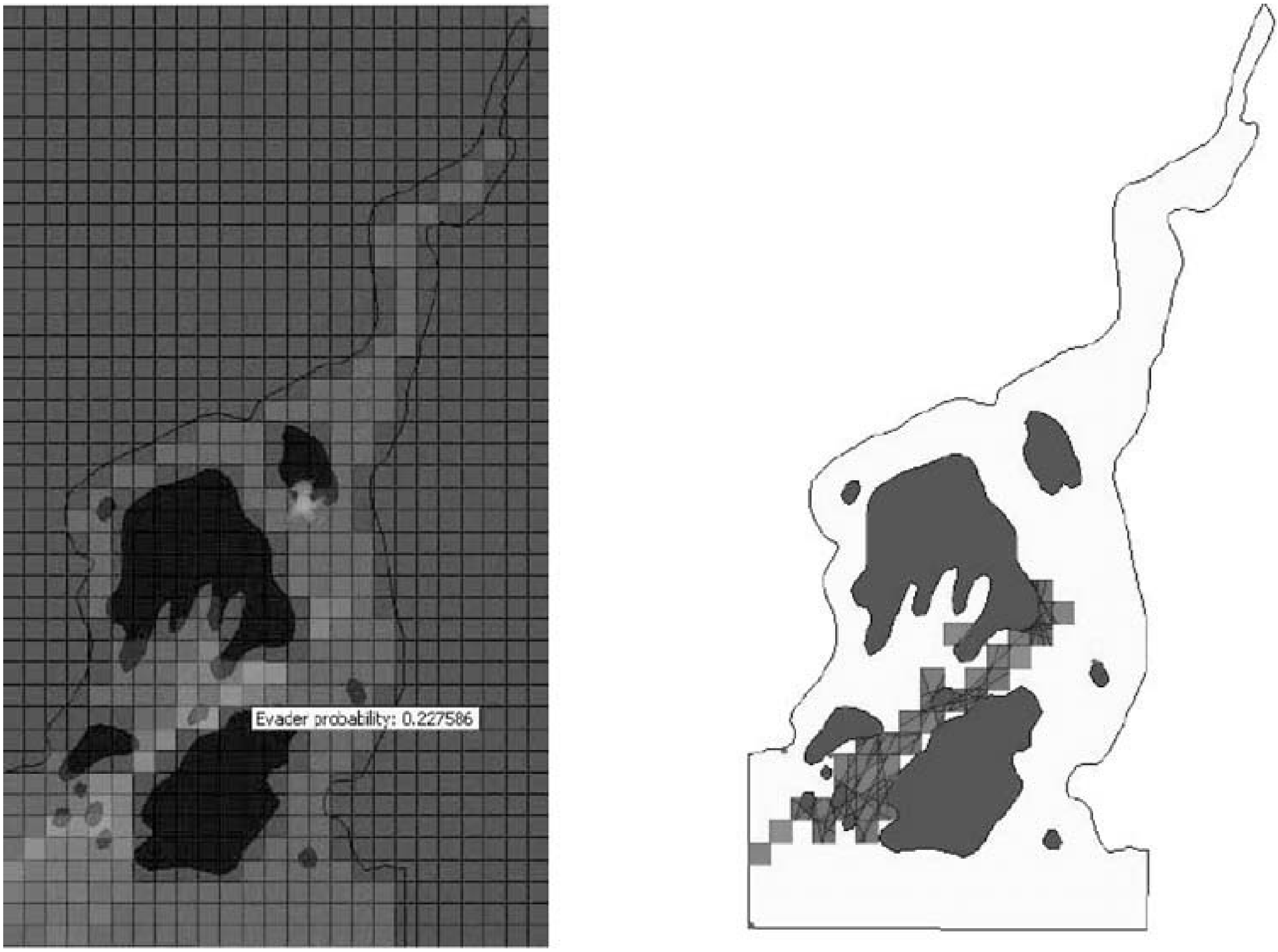


Figure 14.10 Probability map output by the Behavior Simulation Toolbox (http://opensteer.sourceforge.net).

14.3.6 The Visualization Toolbox

The Visualization Toolbox allows the user to run and visualize the discretization of a map into either cells or points (guards) [Figure 14.4(b)]; the input of CTL queries for the state search; the display of the results of the query; and the visualization of the graph search (Figure 14.8).

The state generator and the visualizer currently support the following problem types [14]: the pursuit-and-evade problem, the sensor placement problem, and the exploration problem.

Conclusions

We presented a Situation Analysis Toolbox (SAT) implementing formal notions of situation analysis based on epistemic transition state systems. Composed of five subtoolboxes, the SAT builds a situation based on an abstract version of the environment as a visibility graph and on the execution by a set of agents of a joint strategy derived from pursuit-evasion game theory. The context is further enriched with probability maps of presence of agents, built through modeling emerging behavior. Using the situational definitions and formalisms, SAT affords modeling, evaluation, and representation of dynamic environments.

owledgments

The authors want to thank Grace Lin for programming on the Behavior Simulation Toolbox.

rences

[1] Maupin, P., A-L. Jousselme, H. Wehn, S. Mitrovic-Minic, and J. Happe, “A Situation Analysis Toolbox: Application to Coastal and Offshore Surveillance,” *International conference on Information Fusion*, 2010.

[2] Maupin, P., A-L. Jousselme, H. Wehn, S. Mitrovic-Minic, and J. Happe, “A Situation Analysis Toolbox for Course of Action Evaluation,” *International Command and Control Research and Technology Symposium (ICCRTS)*, 2011.

[3] Endsley, M. R., and D. J. Garland, *Situation Awareness Analysis and Measurement*, Mahwah, NJ: Lawrence Erlbaum Associates, 2000.

[4] Roy, J., “From Data Fusion to Situation Analysis,” *International Conference on Information Fusion*, 2001.

[5] Steinberg, A. N., and C. L. Bowman, “Revisions to the JDL Data Fusion Model,” *Handbook of Multisensor Data Fusion*, D. L. Hall and J. Llinas (eds), The Electrical Engineering and Applied Signal Processing Series, Boca Raton, FL: CRC Press, 2001.

[6] Maupin, P., and A-L. Jousselme, “Interpreted Systems for Situation Analysis,” *International Conference on Information Fusion*, 2007.

[7] Fagin, R., J. Y. Halpern, Y. Moses, and M. Y. Vardi, *Reasoning About Knowledge*, Cambridge, MA: The MIT Press, 2003.

[8] Jousselme, A-L., P. Maupin, C. Garion, L. Cholvy, and C. Saurel, “Situation Awareness and Ability in Coalitions,” *International Conference on Information Fusion*, 2007.

[9] Maupin, P., and A-L. Jousselme, “A General Algebraic Framework for Situation Analysis,” *International Conference on Information Fusion*, 2005.

10] Basar, T., and G. J. Olsder, “Dynamic Noncooperative Game Theory,” *Mathematics in Science and Engineering*, Vol. 160, Academic Press, 1999.

11] Isaacs, R., *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*, Dover: Wiley and Sons, 1999.

12] Alpern, S., and S. Gal, *The Theory of Search Games and Rendezvous*, Kluwer Academic Publishers, 2002.

13] von Neuman, J., and O. Morgenstein, *Theory of Games and Economic* Behavior, Princeton, NJ: Princeton University Press, 1944.

14] Isler, I. V., *Algorithms for Distributed and Mobile Sensing*. Ph.D. thesis, University of Pennsylvania, 2004.

15] Chvàtal, V. A, “Combinatorial Theorem in Plane Geometry,” *Journal of Combinatorial Theory Series* B, 1975.

16] Bjorling-Sachs, I., and D. Souvaine, “An Efficient Algorithm for Guard Placement in Polygons with Holes,” *Discrete and Computational Geometry*, Vol. 13, No. 1, 1995, pp. 77–109.

17] Megiddo, N., S. L. Hakimi, M. R. Garey, D. S. Johnson, and C. H. Papadimitriou, “The Complexity of Searching a Graph,” *Journal of the ACM*, 1988.

18] Goldstein, A. S., and E. M. Reingold, “The Complexity of Pursuit on a Graph,” *Theoretical Computer Science*, 1995.

19] Gal, S., “Strategies for Searching Graphs,” *Graph Theory, Combinatorics, and Algorithms*, Springer, 2005.

20] Lim, W. S., “A Rendezvous-evasion Game on Discrete Locations with Joint Randomization,” *Advances in Applied Probability*, Vol. 29, 1997, pp. 1004–1117.

21] Alpern, S., and W. S. Lim, “The Symmetric Rendezvous-Evasion Game,” *SIAM Journal on Control and Optimization*, Volume 36 Issue 3, May 1998.

22] Alpern, S., “Rendezvous Search: A Personal Perspective,” *Operations Research*, Vol. 50, No. 5, 2002, pp. 772–795.

23] Parsons, T. D., *Theory and Applications of Graphs*, Springer-Verlag, 1976.

24] LaValle, S., *Planning Algorithms*, Cambridge University Press, 2006.

25] Guibas, L. J., J-C. Latombe, S. LaValle, D. Lin, and R. Motwani, “A Visibility-Based Pursuit- Evasion Problem,” *International Journal of Computational Geometry and Applications*, 1999.

26] Pellier, D., and H. Fiorino, “Coordinated Exploration of Unknown Labyrinthine Environments Applied to the Pursuit Evasion Problem,” *International Joint Conference on Autonomous Agents and Multiagent Systems*, New York: ACM Press, 2005.

27] Reynolds, C. W., “Steering Behaviors for Autonomous Characters,” *Proceedings of Game Developers Conference*, 1999.

---

1. The material presented in this chapter is a combination of what has been published in [1] and [2].

2. The provided running times are obtained by running the experiments on IBM ThinkPad, with Intel Core 2 Duo CPU T9600 at 2.80 GHz, and 1.98 GB RAM.

3. http://opensteer.sourcforge.net

CHAPTER 15

Measuring the Worthiness of Situation Assessment

Erik P. Blasch, John J. Salerno, and George P. Tadda (United States)

Situation assessment (SA) modeling has many instances of development, but there has yet to be a comprehensive set of metrics used for performance evaluation. The amount of data being presented and displayed to the analyst is overwhelming to a point where, in many cases, they are missing the salient or key activities of interest (AOI). Analysts are spending the majority of their time filtering through the data rather than performing analysis. To aid the user's situation awareness (SAW), we explore the nature of how we can rank various activities based on their impact and threat. We develop an information fusion SA reference model over SA metrics of confidence (precision, recall), accuracy, activities of interest, timeliness, and throughput to determine the intent and assessment of situation activities.

Introduction

Models can represent a system to determine what is happening, the parameters of interest, and methods for prediction which are important for high-level information fusion (IF) [1, 2]. One example is the proposed extension to the Joint Director of the Labs (JDL) model [3] which we call the information fusion situation assessment model, shown in Figure 15.1, which includes IF Levels: Level 0–data assessment (IFSA); Level 1–object assessment as explicit fusion; Level 2–situation assessment; Level 3–impact/threat assessment as tacit fusion; Level 4–process refinement; Level 5–user refinement; and Level 6–mission management.

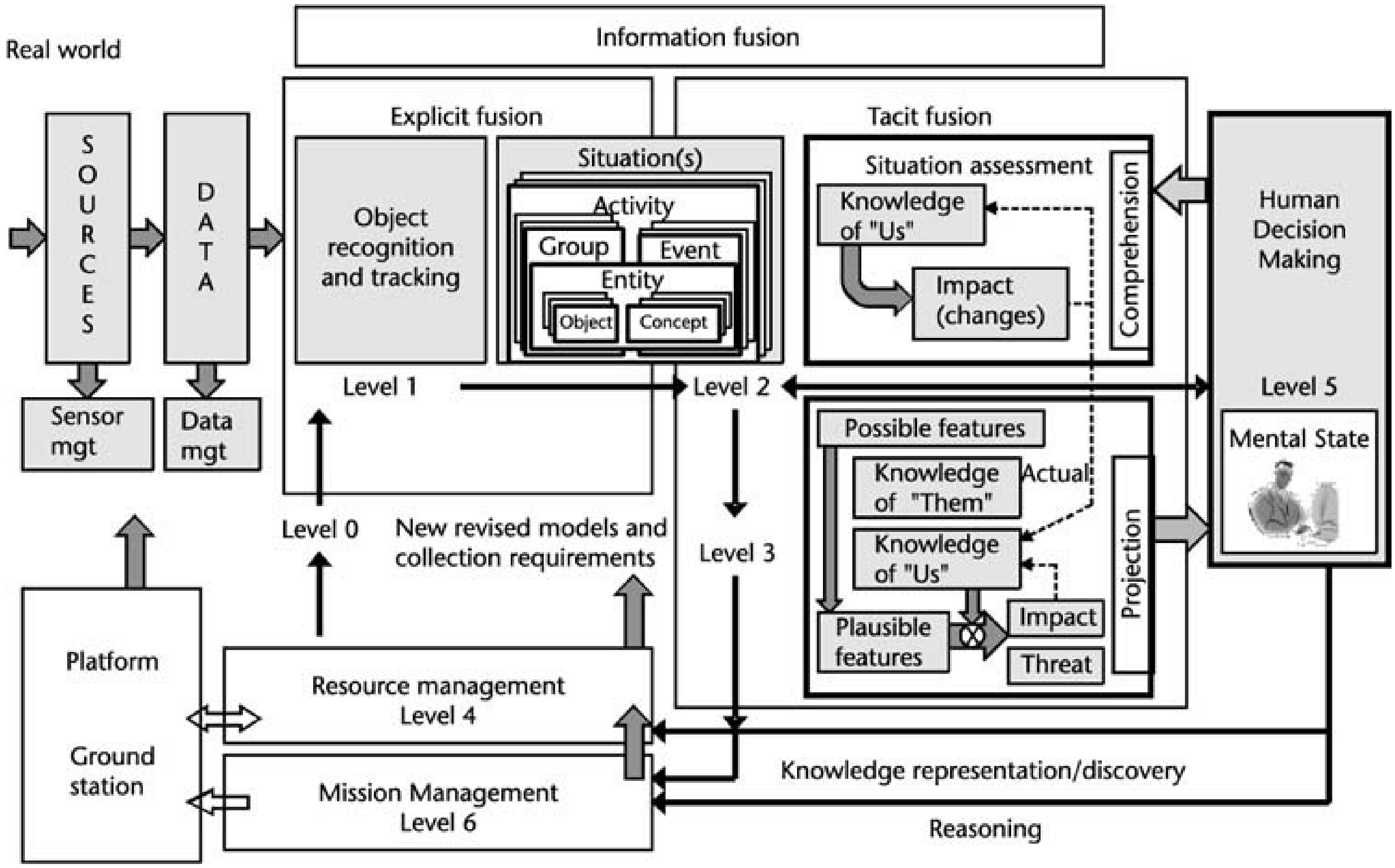


Figure 15.1 The information fusion situation assessment model.

SA involves deriving relations among entities and the aggregation of object states (i.e., classification and location). There is a need to develop SA models and metrics for interfaces, user control actions, and methods of situational analysis [4]. Information fusion systems (IFSs) seek to reduce the enormous amount of data into actionable intelligence for users to act upon [5]. Waltz [6] describes counter deception intelligence analyst information collection tool performance metrics of search efficiency, effort, accuracy, confidence, and intelligence report quality. These performance metrics lead to high-level information fusion effectiveness metrics of timeliness of product delivery; accuracy, relevance, breadth, and completeness of reports; and cost and investment in intelligence production [6]. Numerous literature contributions of process modeling of IFSs have been conducted to clarify SA importance in IFSs design [7]. There is a need to capture SA metrics for traditional physical situations (from sensor data), social situations (from news reports), as well as network situations (from network attacks, protection, and support [8]). In this chapter, we seek to describe SA measures of performance (MOPs) that support measures of effectiveness (MOEs) for determining the worthiness of SA analysis.

Threat (potential) and impact (damage) have been addressed by various researchers. As an example, Bossé, Roy, and Wark [9] define *situation assessment* (SA) as "a quantitative evaluation of the situation that has to do with the notions of judgment, appraisal, and relevance." Additionally, they define *threat assessment* (TA) as "an expression of intention to inflict evil, injury, or damage. The focus of threat analysis is to assess the likelihood of truly hostile actions and, if they were to occur, projected possible outcomes…" [9]. Finally, *impact assessment* (IA) is defined as "the force of impression of one thing on another; an impelling or compelling effect" [9]. Based on earlier work, Roy et al. [10] decompose threat/impact assessment into capability, opportunity and intent.

Blasch [11, 12] defines IA/TA as the "estimation and prediction of effects on situations of planned or estimated actions by the participants; to include interactions between action plans of multiple players (e.g., assessing threat actions to planned actions and mission requirements, decision making, and performance evaluation)." Additionally, he [13] relates the interactions between multiple players to determine the timeliness of actions.

Following the notion of the multiple players, Llinas et al. [3] defined Level 3 as "estimation of impacts (e.g., consequences of threat activities on one's own assets and goals)." As another example, Steinberg [14] states that "threat assessment involves assessing situations to determine whether detrimental events are likely to occur." He defines these notions of IA/TA:

- *Capability* involves an agent's physical means to perform an act;
- *Opportunity* involves spatio-temporal relationships between the agent and the situation elements to be acted upon;
- *Intent* involves the will to perform an act.

From the above definitions of SA/TA/IA, there is a need to instantiate the ability to measure the qualities of threats from situational information. We note two important concepts from the definitions. The first is that a person/agent is involved in the threat and a user is the defined recipient of the action. Second is that we need to detect the AOIs that can cause the threat. Efforts in SA modeling and analysis include the Information Fusion for Engine for Real-time Decision Making (INFERD) system [15], which was enhanced for multiple hypothesis cyber SA analysis [16] . Section 15.2 overviews the details of situation/threat assessment. Section 15.3 develops the metrics and Section 15.4 provides examples. Finally, conclusions are drawn in Section 15.5.

The Situation Assessment Concept

The current proposed model is shown in Figure 15.1 highlights the SA/IA/TA elements which are based on the multiple-player observe, orient, decide, act (OODA) loop of a user indentifying an event. The cognitive ODDA (C-OODA) from Chapter 10 decomposes the OODA loop for situation awareness (SAW), which correlates to the information fusion situation assessment (IFSA) model.

15.2.1 Situation Awareness Reference Model

In [17–22], Salerno et al. provide a set of definitions and a combined SAW reference model based on the JDL data fusion model [3] and Endsley's SAW model (shown in Figure 15.2) [23]. As part of [21], an analysis of projection was made to process threats and impacts; however, the threat projection is better described as *activities of interest* (AOI).

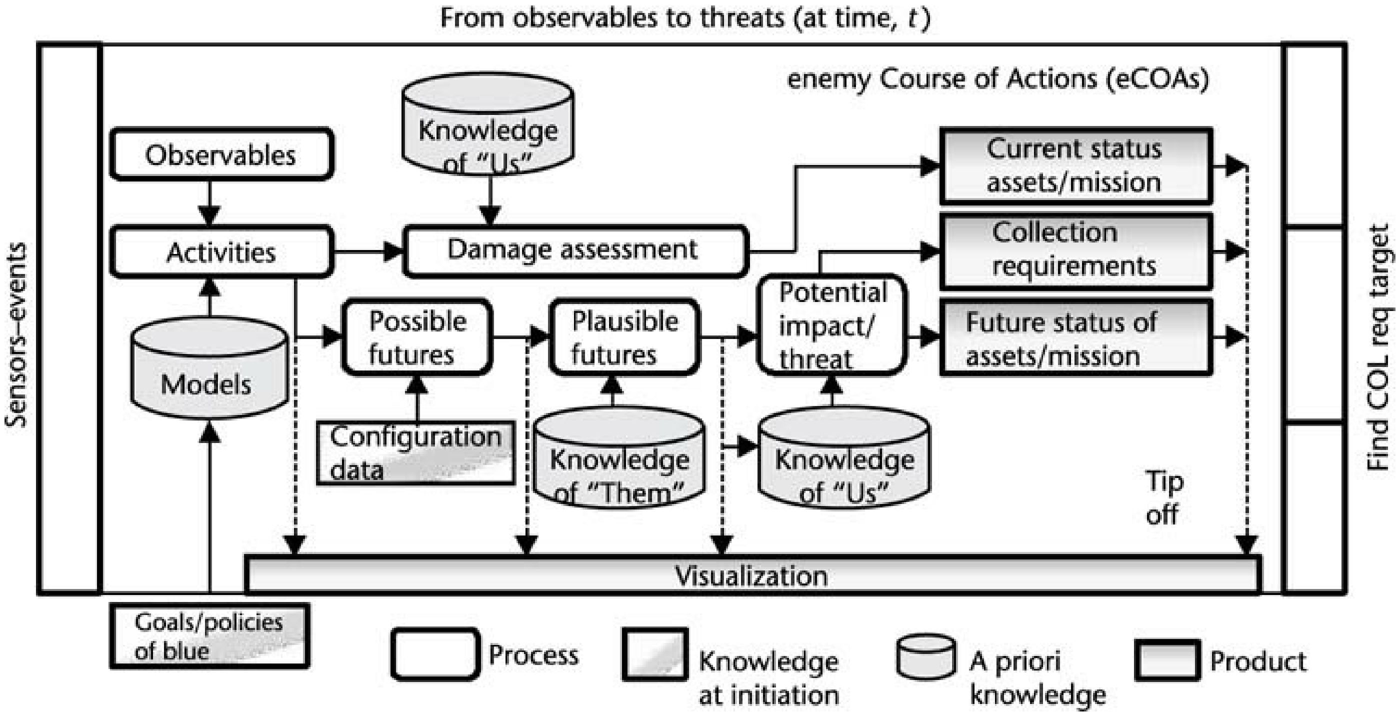


Figure 15.2 Situation awareness process model [7].

15.2.2 Activities of Interest Snapshot in Time

Elements of the IFSA model are highlighted in Figure 15.1 as the SA process over instances of time. Observables are the input to the SA process that provides a view of what is going on in the world (primitive elements of the environment). It is assumed that any attributes associated with the observables have been normalized, cleansed, and transformed into a form that can be used by the follow-on processes. The observables are cues into the activities that a decision maker needs or is interested in (and thus we refer to these as AOI) as a way to gain or maintain situation awareness.

The AOIs are based on goals, policies, or in general the "things" of interest. These AOIs can be stored and manipulated in such formats as graphs, Bayesian networks, Markov models, or any of the numerous modeling techniques. As observables enter the process, they are (1) categorized and correlated with a new stage or step within an existing, ongoing activity; (2) associated with no existing activity and hence become the start of a new activity; or (3) can be a trigger leading to the combination, merging, or removal of existing activities. The aggregation, correlation, and association of events are similar to tracking and identifying individual objects, as defined by Level 1 fusion [24]. Objects are no longer a physical entity like a target but an activity—a collection of events and observables where an observable is associated with an activity or step of an activity.

At any given time, say *t*, we have a set of ongoing activities (defined as the current situation). At this point, we are interested in analyzing the meaning of these activities which is considered to be SA (as shown in the IFSA model in Figure 15.1, and labeled "comprehension."). The overall objective of SA is to determine if any of the ongoing activities have an impact to us or if they can have (in the future) an impact to "us" (threat). Comprehension looks at the current activities and assesses the previous activities impact. Since these activities have already happened, we refer to this as *damage assessment* (DA). That is, have any of the identified activities caused a current impact and specifically have they caused harm that requires development of a recovery plan to resolve the condition(s) that the activities have generated? In order to accomplish damage assessment, one not only needs the current, known activities, but also what each activity means to "us" (i.e., does the given activity impact us in some way?). The data needed to perform DA analysis is part of what we call "knowledge of us" and identifies to the decision maker, whether there is a current impact to any of his/her capabilities or assets, and whether there is an impact in his/her ability to perform the mission.

A user may also be interested in a view of what the adversary or competitor is doing or may possibly do—which has generally been described as "getting inside the adversary's OODA loop." The sooner we understand what the adversary can/might do, the more options become available to the decision maker (as shown in the IFSA model in Figure 15.1, and labeled "projection"). The first step of projection is to take each AOI and project it forward based on the a priori knowledge provided as part of the model. Here we don't discuss time itself; we are not projecting the activities based on time, but rather the next step in the process. In some cases, it could take milliseconds to go from one stage to another, and in other cases it could be days or longer. The number of stages that we look forward is defined under configuration data. Based solely on the models themselves, we have projected each current activity one or more steps forward; however, these projected or possible future activities do not take into account whether they are plausible. In order to determine plausibility, we need to consider additional knowledge. We need both the knowledge of "them" (KOT) and knowledge of "us" (KOU). Specifically, we need to know whether the adversary has the capability, capacity, intent/goal, and have they exhibited similar behavior in the past. We also need to know whether they have the opportunity to accomplish the intent(s)/goal(s).

In many cases, opportunity is based on the vulnerabilities of "us" (provided as part of the KOU). Thus, starting with the list of possible futures, we use the KOT and KOU to constrain the possible into the plausible for each activity of interest. But again what do these plausible futures mean to me/us? To answer this question, we again use part of the KOU (importance of the assets/capabilities) to identify potential impacts and threats to meeting our objective(s). As an example, Table 15.1 lists features of cyber attacks over capability, opportunity, and intent.

Table 15.1 Features

Feature	*Description*	*Classification*
f_1	Hacker sophistication	Capability
f_2	Link vulnerability	Opportunity
f_3	Host data criticality	Opportunity
f_4	Hacker physical location	Intent

Information gathered from projection not only lists future potential impacts/threats, but the knowledge that we can determine future collection requirements. Based on each of the futures, key differentiating events can be identified that assist in determining which of the plausible futures may actually be unfolding. The key differentiating events and SA metrics can determine the collection requirements needed to increase the certainty in identifying whether a plausible future activity is occurring or will occur.

One of the risks in a reference model, such as the one in Figure 15.1, is that it can be perceived as a sequential flow of data or information rather than a descriptive model of components and ideas. To help clarify the sequential process, Figure 15.2 attempts to define a process flow and end products that are based on the concepts of the situation awareness reference model as its framework. A primary feature of the situation awareness process model is that it (1) defines components that can be implemented as automated computer applications or shared human/computer systems that can then be tied together within system architectures, and (2) describes the flow of information and data sources.

The SAW reference model provides a set of definitions that can serve as a reference for describing systems that aid with SA, while the SAW process model captures a process flow at a single point in time. Together, the two models provide a common set of definitions for situation awareness. The IFSA model provides a flow or thread of information through the system for a given time, *t*. The goal in presenting the reference model and process thread was to describe how one can identify significant AOIs of concern to oneself and to one's goals/objectives. In doing so, the hope is to focus the analyst's attention onto what is important, thus minimizing the current work overload and maximizing the decision maker's situation awareness. Products include plausible adversarial futures ranked based on threat (most "dangerous" and most "likely" and generally refer to as enemy courses of actions (eCOA)), a list of collection requirements, and possible alerts/tip-offs based on differentiating events and anticipated futures. Now we describe metrics to assess SA.

15.2.3 Data Information Ratio

In [25] we introduced a concept of the data to information ratio (DIR) which measures the amount of compression or reduction that can be achieved by aggregating events/objects into groups and activities. For HLIF analysis (as in relation to the SA metrics), the DIR supports analysis of the situation:

$$DIR = \frac{Number\ of\ Observations}{Number\ of\ Complex\ Entities} \tag{15.1}$$

As an example, consider a military force, composed of a number of geographically disperse units and containing many vehicles. If we were to track and display all of these objects, a typical display would be black from the target density. If we were able to group these objects into clusters and identify these clusters [26] as the units and display icons that represent them, the display would be much more understandable for track maintenance [27]. A second example comes from cyber security. An analyst is required to try to find a potential attack within thousands of alerts (pings, probes, etc.) However, if there was a capability that could aggregate alerts together into what we call *attack tracks*, an analyst could be reviewing hundreds of tracks rather than the thousands of observations/alerts. It should be noted that aggregation alone is not sufficient; we need to also provide a way to draw their attention to those that are important. The ranking or prioritization is done by assessing the threat (and thus the potential impact) that an ongoing activity may have on us, our assets, or to the mission. To enable a decision maker's overall situation awareness, we now detail data, information, or knowledge SA metrics to understand a given activity's impact or threat.

Metrics

How well does a system work? Once we have defined the system and its purpose, we can then develop metrics [28] to evaluate how well it performs and the worthiness of the metrics to provide SA. Such metrics were described in [25] and classified into four categories or dimensions: (1) confidence, (2) purity, (3) cost utility, and (4) timeliness, as shown in part of Table 15.2. Additionally, we add throughput and cost as key metrics in SA.

Table 15.2 Situation Assessment System Metrics

Dimension	*Metric*	*Definition*
Confidence	Precision	% Correct alerts
	Recall	Probability of detection
Accuracy/Purity	Misassignment rate	% Incorrect associations
	Evidence recall	% Found
Timeliness	Delay	Time between event and alert
Throughput	Data processed	Number of events
Cost	Cost utility	% cost savings

Confidence and accuracy/purity indicate how well the lower level algorithms work in combining the observations into activities and thus how well our system is doing in correctly identifying and tracking the activities of the overall situation. Cost/utility provides us an indication as to how well the assessment capabilities work, while the timeliness provides an overall system measure of performance (MOP) to provide the right information in a sufficient amount of time for the user to make a decision.

Confidence is a measure of how well the system detects the true activities and is typically reported as a probability. It is composed of three factors: (1) recall, (2) precision and (3) fragmentation. *Recall* measures the percentage of activities correctly identified by the Level 1 fusion system (correct detections) in relation to the number of "real" activities as defined in the ground truth (known activities) and is defined as:

$$Recall = \frac{Correct\ Detections}{Known\ Activities} \tag{15.2}$$

Precision is the percentage of activities correctly identified by the system (correct detections) in relation to the total number activities (detected activities) provided by the system:

$$Precision = \frac{Correct\ Detections}{Detected\ Activities} \tag{15.3}$$

Fragmentation is defined as the percentage of activities reported as multiple activities that should have been reported as a single activity (i.e., the number of results that identify the same known activity):

$$Fragmentation = \frac{Number\ of\ Identified\ Known\ Activities}{Detected\ Activities} \tag{15.4}$$

Accuracy/purity characterizes the quality of the proposed activities (i.e., were the observations correctly matched and connected to the right activity track?). To measure purity we define a misassignment rate as the percentage of evidence or observations that were incorrectly assigned to a given activity. Evidence recall is the percentage of evidence, or alerts, detected in relation to the total known events or observations. Based on our use, we found these metrics to provide value toward our understanding of capability evaluation.

Cost utility was modified from [25] and redefined as an *activities of interest* (AOI) score. There is an “opportunity cost” of the best alternative activity forgone when utilizing limited resources. We measure how effective the system is in ranking the AOIs in a predetermined order to highlight the opportunity cost. For example, in the cyber domain, we are interested in those activities (e.g., potential attacks) that were most detrimental to our operations based on either impact/threat or mission (where an “attack” score was highlighted based on “most likely” or “most dangerous”) [21].

Timeliness measures the ability of the system to respond within time requirements of a particular domain. For example, we are interested in how quickly the system can identify the activity or AOI. However, the timeliness value alone is meaningless unless we take into account whether there is sufficient time to take action.

Throughput is resolved based on timeliness, as the more time you allow for a decision, the more data that can be processed. Figure 15.3 shows how the given metrics are mapped against the IFSA model.

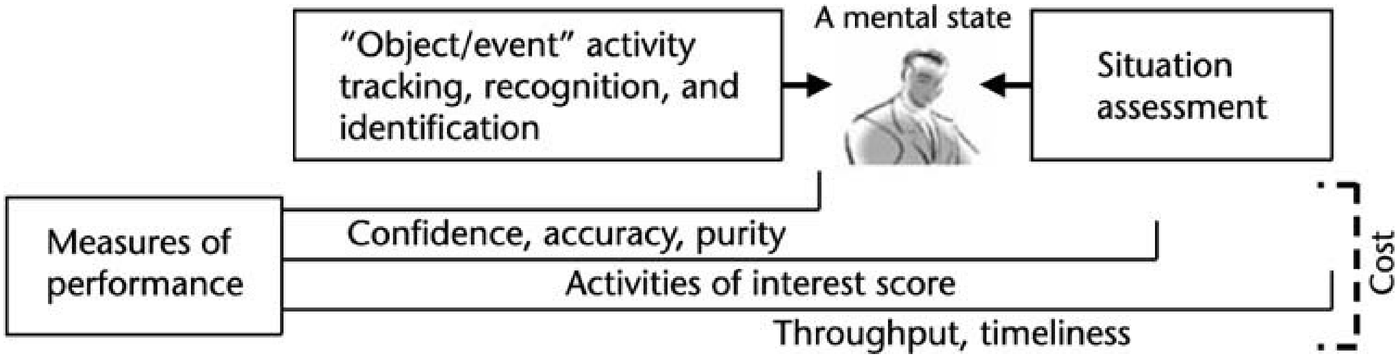


Figure 15.3 Metrics mapped to the IFSA reference model.

15.3.1 AOI Score

Focusing on the cost utility, the AOI score provides us insight into our assessment process. The objective of the AOI score is to measure how well the system has identified the activities that are most important to the user. The criteria for what is most important are dependent on the decision maker's or analyst's needs. In determining potential adversary courses of actions (COAs), the decision maker is interested in which COAs are most likely and which are most dangerous—each answer using different information as part of the analysis and providing the decision maker with a different perspective. For example ranking the "most likely," needs to take into account the adversary's capability/capacity (what can they do), intent (what is/are their goals) and opportunity. "Most dangerous" must not only take into account the information required to determine the most likely but also KOU vulnerabilities and an assessment as to whether the adversary will be successful, and to what extent he will incur damage. The AOI ranking can also change between the current analysis and the projection of future impacts/threats.

15.3.2 Measuring How Well We Are Doing

We can compute the AOI score as:

$$AOI\ Score = \frac{NAOI * NA - \sum_{i=1}^{NAOIR} P_i}{NAOI * NA - \sum_{i=1}^{NAOIR} i} \tag{15.5}$$

where

NAOI = Number of Activities of Interest in Ground Truth
NAOIR = Number of AOIs in Results
NA = Total Number of Activities in Ground Truth
P_i = Position of the i^{th} Activity of Interest

If any (or all) of the AOIs are not part of the results list or if their position is greater than the total number of activities in the ground truth, we set the position value (P_i) for those AOI(s) equal to the total number of activities in the ground truth (*NA*). By adding this condition, if there are no AOIs included/identified in the results list, the AOI score will equal 0. Whereas, if there is only a subset of the actual AOIs identified, the system will get credit for only those in the subset.

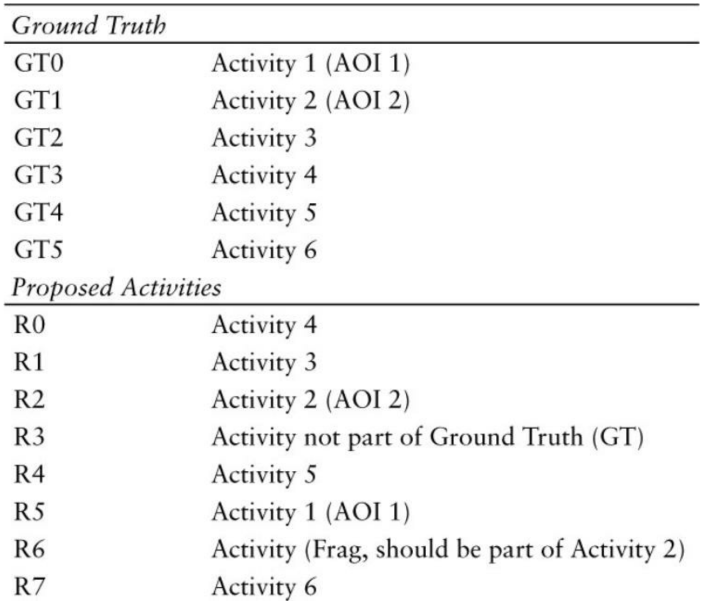
Example

We illustrate the basic concept through a cyber example. Individual observations are provided by such systems as Intrusion Detection Systems (IDSs), Netflow, Cisco Systems and System Logs, and aggregated into AOI tracks. The AOI tracks are outputs of a Level 1 Information Fusion capability, and can be simply displayed as an activities list (one form of output). If no other knowledge is available, the AOI list would be just that: a list of activities in no order with no indication of importance. However, in the real world, the number of activities can be very large (hundreds to thousands based on pruning or lack of). Some ordering is needed to place the most important ones that the analyst or decision maker should be concerned with at the top. In the cyber example, we would like to prioritize the activities based on their current or future impact/threat. So, how can we measure both what the additional knowledge provides and how well our ranking algorithms work?

15.4.1 Calculated Example with Few Activities

Let us consider a simple example [7]. Given the ground truth and the results provided as output by our Level 1/2 IF system (situation recognition and identification) without any further knowledge or assessment as shown in Table 15.3, we can compute a baseline. The baseline can then be used to determine the value added by additional knowledge and processing provided by situation assessment.

Table 15.3 Data for Example 1



Ground Truth	
GT0	Activity 1 (AOI 1)
GT1	Activity 2 (AOI 2)
GT2	Activity 3
GT3	Activity 4
GT4	Activity 5
GT5	Activity 6
Proposed Activities	
R0	Activity 4
R1	Activity 3
R2	Activity 2 (AOI 2)
R3	Activity not part of Ground Truth (GT)
R4	Activity 5
R5	Activity 1 (AOI 1)
R6	Activity (Frag, should be part of Activity 2)
R7	Activity 6

We first compute how well our Level 1/2 system has identified the activities that make up what we call the situation. Recall is (15.2) (6/8 = 0.75 or 75%). In this case, our system has identified more activities than what actually is occurring (one not even there, a false positive and the other is a fragment). Optimum recall would be 100%. Precision is (15.3) (6/6=1.0 or 100%). In this case the system has correctly identified all the activities in the ground truth. Fragmentation is (15.4) (1/8 = 0.125 or 12.5%). In summary, if we had a capability that provided the results as shown above, we would have a recall of 75%, a precision of 100% and a fragmentation of 12.5%. What do these numbers tell us? There were 25% more activities identified than there actually were, and, of the 25%, half of them (12.5%) were because they should have been associated with an existing activity (fragments). Let us now consider the question: Of the activities that have been identified are there any of them that we should be concerned with? Our next metric, AOI score, will indicate which AOIs are of concern.

Let us first compute a baseline AOI score where we have no further information about the activities. To compute the AOI score we have the number of activities of interest in the ground truth as 2 and the total number of activities in the ground truth as 6. The remaining 4, which we do not consider to be AOIs, are either activities that have no or minimal impact or threat to us. We next need to know the ordering the system identifies for those activities that are of interest (as identified by in the results). In our example, the first "important" activity is third in the proposed list, while the second important activity is in the sixth position (Table 15.3). We simply add these two values together for a value of 9. The final value we need to compute is just the geometric sum of the important activities which is simply two activities: 2+1=3. Substituting in the values into (15.5) we have a value of [(2)(6)−9]/[(2)(6)−3] = 3/9 or 0.33. This means that an analyst or user would have to consider roughly two-thirds of the activities before seeing the most important activities.

Now assume that we are provided additional knowledge about the activities of Table 15.4 after further analysis is provided by situation analysis. For example, if we were looking at a list of cyber activities, we might have more information on the computers (their connectivity, operating system, services, and applications) and what mission they perform. Using this knowledge, we can then determine how important the activity is based on impact or threat to the mission. Vulnerabilities also come in to play. For instance, is the system vulnerable to the attack being executed against the network, making that activity more important if the network is vulnerable and less important if not? To compute the additional information value in the ranking, we compute the AOI score a second time. Table 15.4 provides an example.

Table 15.4 Data for Example 2

Proposed Activities	
R0	Activity 4
R1	Activity 2 (AOI 2)
R2	Activity 1 (AOI 1)
R3	Activity 3
R4	Activity not part of Ground Truth (GT)
R5	Activity 5
R6	Activity (Frag, should be part of Activity 2)
R7	Activity 6

The only value that changes in our analysis is the sum of the positions of the activities of interest. The new value is 2 + 3 = 5, and the new value for the AOI score becomes ((2)(6)-5)/((2)(6)-3) = 7/9 or 0.78. One can easily see that if the most important activities are in positions 1 and 2, then the AOI score would be 1.0. Similar computations can be made for plausible futures and their impact/threat. Similar to SA, we need only ground truth for the given scenario. We note here that the AOI score does not distinguish between how important the AOIs are, just the fact that they are an AOI. Extensions to AOI score metric can take into account the value of the AOI, the contextual scenario, as well as correlated AOIs.

15.4.2 Simulated Example with Numerous Activities

After numerous Monte Carlo runs, of various activity and fragment changes, we determined the AOI scores associated with each sample. Since AOI is related to recall, we decided to use recall in a receiver operating characteristic (ROC) curve. Recall is also referred to as sensitivity, hit rate, or true positive rate. Using the precision, we calculated the false alarm rate and plotted the summary of results in Figure 15.4, for a single time event.

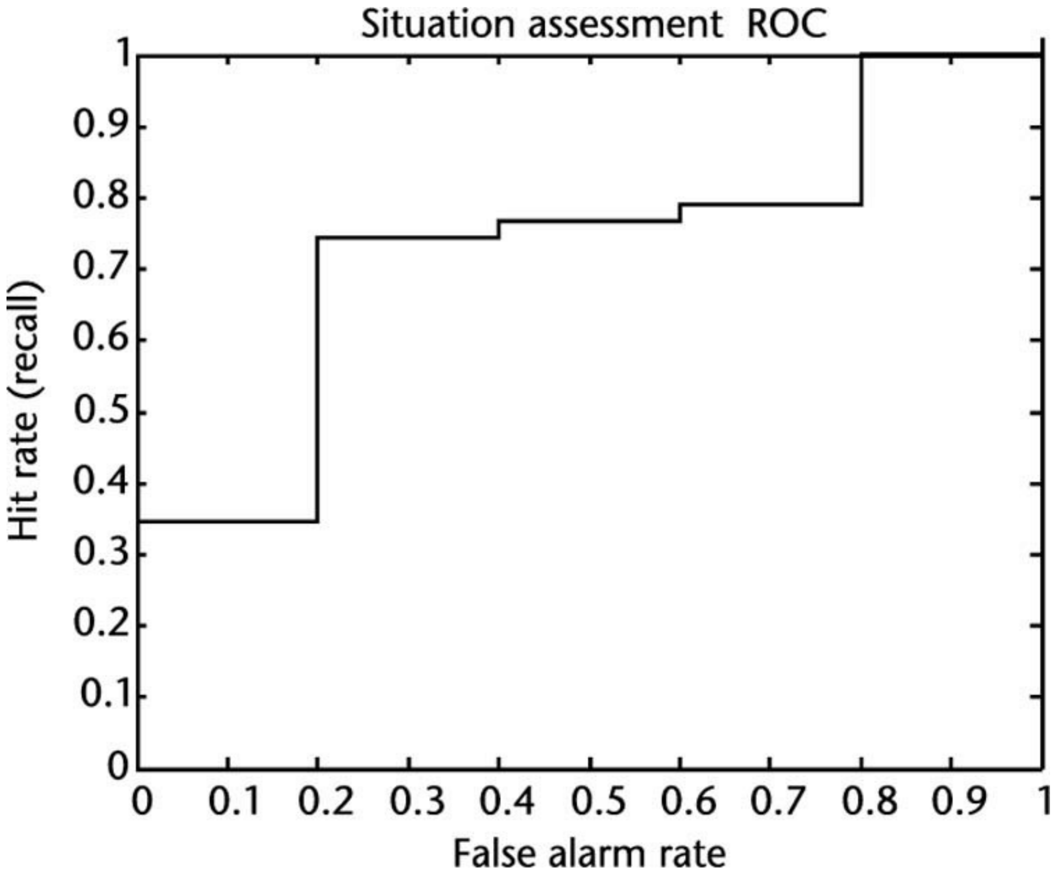


Figure 15.4 Situation assessment receiver operating characteristic curve.

While the ROC curve provides an overall assessment of the process, we were interested in the AOI and fragmentation scores. Given that the analysis is done for each time step, where the results varied, in Figure 15.5 we plotted the results as a function of normalized recall to compare ROCs, shown in Figure 15.6. We see that the fragmentation score did not correlate with the AOI or recall. However, with large increases in recall, there is a modest change in the AOI score.

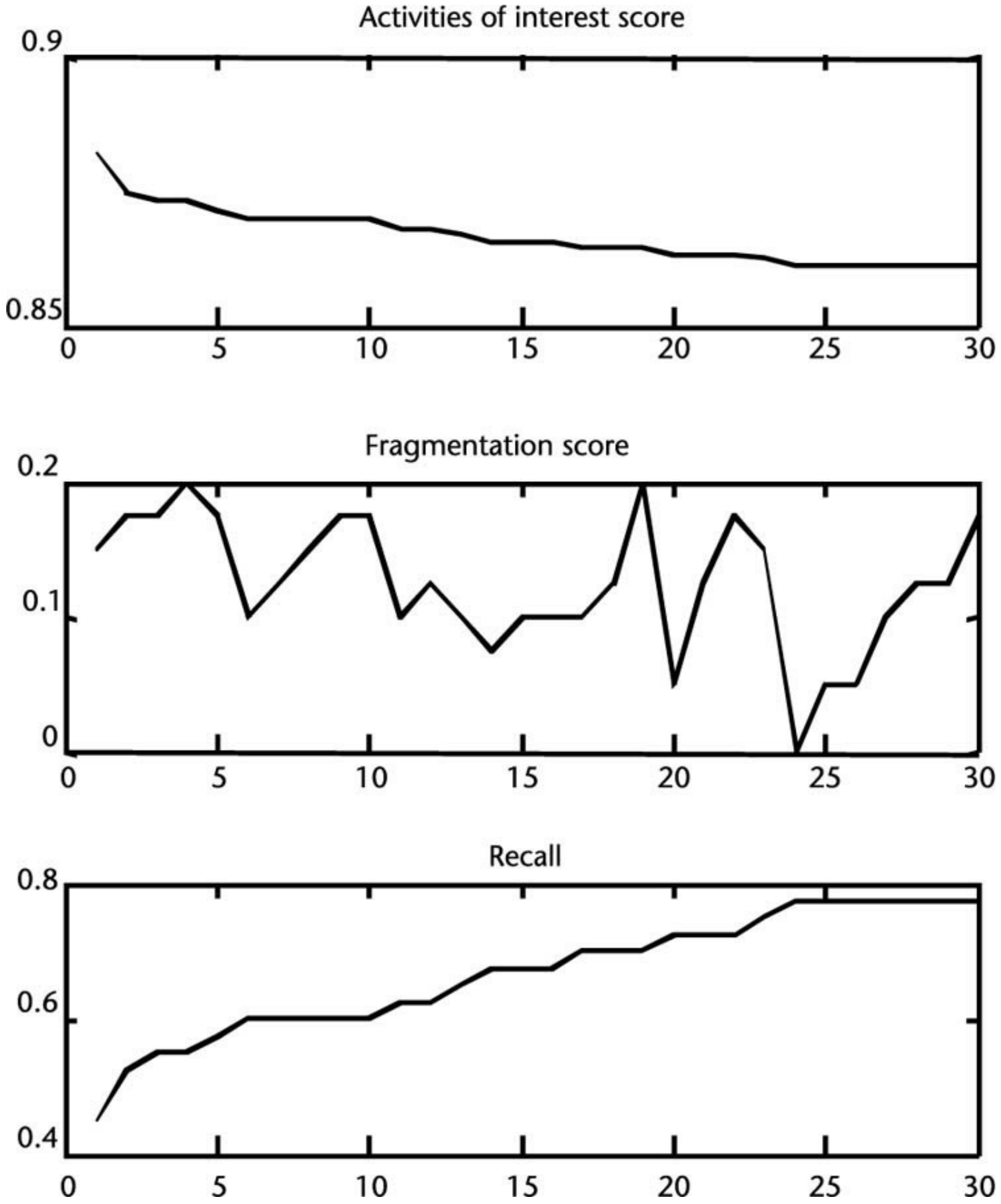


Figure 15.5 Sorted AOI, fragmentation, and recall results.

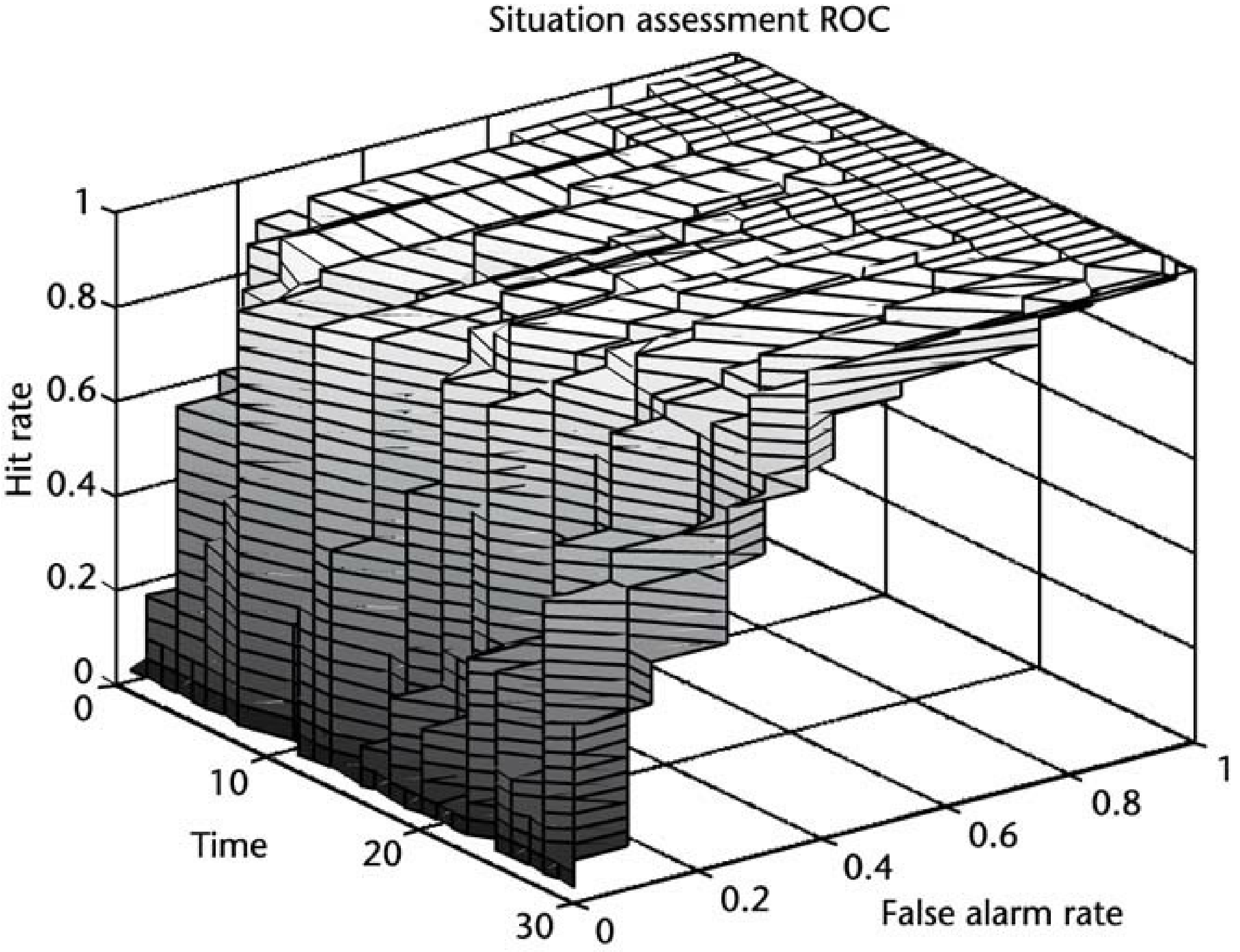


Figure 15.6 3-D sorted activities of interest situation assessment receiver operating characteristic curve.

Figure 15.6 demonstrates a 3-dimensional (3-D) plot of the ROC over time [29, 30]. We ordered the time results to correspond with the AOI score. The 3-D sorted SA ROC curve visualization demonstrates the ongoing analysis of the situation assessment. For example, when the fragmentation rate is low there is a low false-alarm rate (e.g., time step 20). Note that if there were no false alarms, then the plotting function did not start at zero, as a typical ROC curve.

Conclusions

In this chapter, we developed the metrics for situational assessment evaluation, including precision, recall, fragmentation, and an AOI score to determine the threat/impact assessment for a user situation awareness. We will further explore the scenario modeling, metrics, and visual analytics in future studies to aid a user in assessing, monitoring, and mitigating effects from an adversary conducting threatening activities. The activity of interest metrics can be used in measures of effectiveness for HLIF (as discussed in Chapter 16).

rences

[1] Blasch, E., J. Llinas, D. Lambert, P. Valin, S. Das, C-Y. Chong, M. M. Kokar, and E. Shahbazian, “High Level Information Fusion Developments, Issues, and Grand Challenges—Fusion 10 Panel Discussion,” *International Conference on Information Fusion*, 2010.

[2] Kessler, O., et. al., “Functional Description of the Data Fusion Process, Technology Report for the Office of Naval Technology Data Fusion Development Strategy,” Naval Air Development Center, Warminster, PA, November 1991.

[3] Llinas, J., C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, “Revisions and Extensions to the JDL Data Fusion Model II,” *International Conference on Information Fusion*, 2004.

[4] Blasch, E., P. Valin, E. Bosse, M. Nilsson, J. Van Laere, and E. Shahbazian, “Implication of Culture: User Roles in Information Fusion for Enhanced Situational Understanding,” *International Conference on Information Fusion*, 2009.

[5] Fabian, W. Jr., and E. P. Blasch, “Information Architecture for Actionable Information Production (A Process for Bridging Fusion and ISR Management),” *National Symposium on Sensor and Data Fusion*, 2002.

[6] Bennett, M., and E. Waltz, *Counterdeception Principles and Applications for National Security*, Norwood, MA: Artech House, 2007.

[7] Salerno, J. J., M. Sudit, S. J. Yang, G. P. Tadda, I. Kadar, and J. Holsopple, “Issues and Challenges in Higher Level Fusion: Threat/Impact Assessment and Intent Modeling (A Panel Summary),” *International Conference on Information Fusion*, 2010.

[8] Waltz, E., *Information Warfare: Principles and Operations*, Norwood, MA: Artech House, 1998.

[9] Bossé, E., J. Roy, and S. Wark, *Concepts, Models, and Tools for Information Fusion*, Norwood, MA: Artech House, 2007.

10] Roy, J., S. Paradis, and M. Allouche, “Threat Evaluation for Impact Assessment in Situational Analysis Systems,” *Proceedings of SPIE*, Vol. 4729, 2000.

11] Blasch, E., and S. Plano, “Level 5: User Refinement to Aid the Fusion Process,” *Proceedings of SPIE*, Vol. 5099, 2003.

12] Blasch, E., “Situation Impact and User Refinement,” *Proceedings of the SPIE* 5096, 2003.

13] Blasch, E., “Proactive Decision Fusion for Site Security,” *International Conference on Information Fusion*, 2005.

14] Steinberg, A. N., “Foundations of Situation and Threat Assessment,” in *The Handbook of Multisensor Data Fusion*, M. E. Liggins, et al. (eds.), London: CRC Press, 2009.

15] Sudit, M., M. Holender, A. Stotz, T. Rickard, and R. Yager, “INFERD and Entropy for Situational Awareness,” *Journal of Advances in Information Fusion*, Vol. 2, No. 1, June 2007.

16] Schwoegler, S., S. Blackman, J. Holsopple, and M. J. Hirsch, “On the Application of Multiple Hypothesis Tracking in the Cyber Domain,” *International Conference on Information Fusion*, 2011.

17] Salerno, J., G. Tadda, D. Boulware, M. Hinman, and S. Gorton, “Achieving Situation Awareness in a Cyber Environment,” *Proceedings of the Situation Management Workshop of MILCOM*, 2005.

18] Tadda, G., et al., “Realizing Situation Awareness within a Cyber Environment,” *Proceedings of the SPIE* Vol. 6242, 2006.

19] Salerno, J. J., “Resource management: A Necessary and Integral Component to Any Level 2/3 Fusion Capability,” *International Conference on Information Fusion*, 2006.

20] Tadda, G., “Measuring Performance of Cyber Situation Awareness Systems,” *International Conference on*

Information Fusion, 2008.

21] Salerno, J., “Measuring Situation Assessment Performance Through the Activities of Interest Score,” *International Conference on Information Fusion*, 2008.

22] Salerno, J., and G. Tadda, “Ranking Activities based on their Impact and Threat,” *International Conference on Information Fusion*, 2009.

23] Endsley, M. R., “Toward a Theory of Situation Awareness in Dynamic Systems,” *Human Factors Journal*, Volume 37, No.1, March 1995, pp. 32–64.

24] Blasch, E., *Derivation of A Belief Filter for High Range Resolution Radar Simultaneous Target Tracking and Identification*, Ph.D. dissertation, Wright State University, 1999.

25] Salerno, J. J., E. Blasch, M. Hinman, and D. Boulware, “Evaluating Algorithmic Techniques in Supporting Situation Awareness,” *Proceedings of SPIE*, April 2005.

26] Connare, T., E. Blasch, J. Schmitz, F. Salvatore, and F. Scarpino, “Group IMM Tracking Utilizing Track and Identification Fusion,” pp. 205–220, *Proceedings of the Workshop on Estimation, Tracking, and Fusion; A Tribute to Yaakov Bar-Shalom*, Monterey, CA, May 2001.

27] Blasch, E. P., and T. Connare, “Improving Track Maintenance Through Group Tracking,” *Proc. of the Workshop on Estimation, Tracking, and Fusion: A Tribute to Yaakov Bar Shalom*, Montery, CA, 360–37, May 2001.

28] Roscoe, B., M. Pribilski, and E. Blasch, “Evaluation Methodology for the DARPA Dynamic Tactical Targeting (DTT) Program,” *ATR Working Group Meeting*, 2004.

29] Alsing, S., E. P. Blasch, and R. Bauer, “Three-Dimensional Receiver Operator Characteristic (ROC) Trajectory Concepts for the Evaluation of Target Recognition Algorithms Faced with the Unknown Target Detection Problem,” *Proceedings of SPIE*, Vol. 3718, 1999.

30] Chang, C-I. “Multiparameter Receiver Operating Characteristic Analysis for Signal Detection and Classification,” *IEEE Sensors Journal*, Vol. 10, No. 3, 423–442, March 2010.

CHAPTER 16

Measures of Effectiveness for High-Level Information Fusion

Erik P. Blasch (United States), Pierre Valin (Canada), and Éloi Bossé (Canada)

Current advances in technology, sensor collection, data storage, and data distribution have afforded more complex, distributed, and operational information fusion systems (IFSs). IFSs notionally consist of low-level information fusion (data collection, registration, and association in time and space) and high-level information fusion (user coordination, situational awareness, and mission control). Low-level IFSs typically rely on standard metrics for evaluation such as timeliness, accuracy, and confidence. Given the broader use of IFSs, it is also important to look at highlevel fusion processes and determine a set of metrics to test IFSs, such as workload, throughput, and cost. Three types of measures (measures of performance (MOP), measures of effectiveness (MOE), and measures of merit (MOM)) are summarized. In this chapter, we seek to describe MOEs for high-level information fusion (HLIF) based on developments in quality of service (QOS) and information quality, or quality of information (QOI), that support the user and the machine, respectively. We define a HLIF MOE based on (1) information quality, (2) robustness, and (3) information gain. We demonstrate the HLIF MOE for a maritime domain situation awareness example.

Introduction

The objective of this chapter is to initiate a discussion on an appropriate set of effectiveness metrics for HLIF evaluation. We suggest that HLIF effectiveness has three parts: information gain, quality, and robustness, which we develop in the chapter.

The distinction between high-level and low-level information fusion (LLIF) has propagated from the 1980s discussions surrounding the needs and the relevant processes for information fusion. The Joint Director of Laboratories's (JDL) model and its subsequent revisions formed the high–low level distinction [1, 2].

Figure 16.1 overviews the LLIF/HLIF distinctions in which the user is an active part in the HLIF fusion process. From the numbering scheme (as discussed in Chapter 2), HLIF includes all levels beyond that necessary to track and identify objects. LLIF fusion, including the physical-based parameters, lends itself to quantifiable evaluation techniques. Current directions in measures of effectiveness (MOEs) for IFSs [3] need to address approaches beyond estimation (determining parameters from measured data) and fusion rules [4].

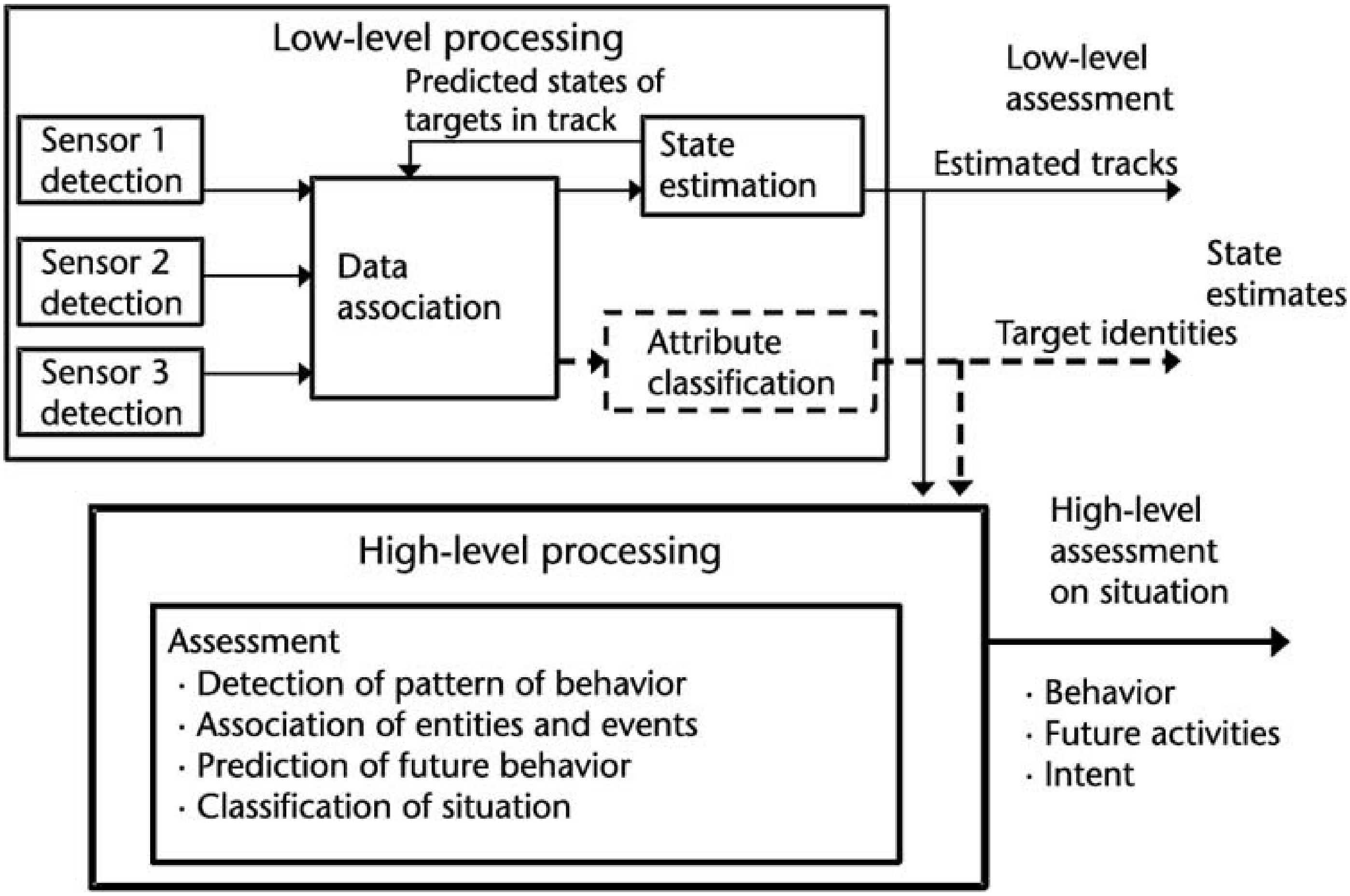


Figure 16.1 Distinctions between LLIF and HLIF.

Evaluating an information fusion system is not new. There is a large set of literature associated with MOPs, MOEs, and measures of force effectiveness (MOFEs) based in estimation. Two excellent summaries include Chapter 11 from Waltz and Llinas [5] and Chapter 20 from Llinas [6]. The compiled information from [5, 6] represent a comprehensive assessment of methods in estimation of MOP/MOEs and discussion of HLIF issues. However, addressing fusion process MOEs [7] as a system infers the need for additional discussion on HLIF performance metrics in real systems [8]. Systems perform well if they (1) support mission goals [9], (2) enhance operator work tasks [10], and (3) reduce uncertainty [11, 12].

Effectiveness implies that a system is capable of producing an effect. Many benefits of fusion include providing locations of events, extending coverage, and reducing ambiguity and false alarms [5]. The goal for the IFS is to support the users in their tasks, whether providing refined information, reducing time pressures, or determining completeness, accuracy, and quality in task completion. Effectiveness issues include:

- *Efficiency*: doing things in the most economical way (good input to output ratio);
- *Efficacy*: getting things done (i.e., meeting targets);
- *Effectiveness*: doing "right" things (i.e., setting right targets to achieve an overall goal (the *effect*)) [13].

Inherent in the definition of "effectiveness" is a level of performance needed to accomplish a goal. While a large number of ideas and metrics could be postulated; all having their merits and limitations, we focus on three:

1. *Information Gain*: valued added from which two pieces of information provide more content than individual pieces of information alone;
2. *Quality*: measures of performance that include accuracy, reduction in uncertainty, confidence, timeliness, throughput, and cost as well as credibility and reliability;
3. *Robustness*: consistent over domains of testing and application.

Together, these definitions form the basis of effectiveness in (1) presenting high quality data, (2) being derived from more than one source, and (3) being consistent and reliable over the situation. For different scenarios, the performance of an IFS needs to be robust over the object, sensor, and environment operating conditions. We postulate a general HLIF MOE as:

Effectiveness = InfoGain * Quality * Robustness

A taxonomy of other metrics could be expanded upon, in order to create an evaluation standard for MOEs. However, different environments would necessitate adaptive metrics tailored for the situation [14–16].

The rest of the chapter is as follows: Section 16.2 overviews the distinctions between low-level and HLIF with emphasis on background research in HLIF. Section 16.3 describes MOEs as derived from quality of information and quality of service contracts. Section 16.4 postulates a high-level MOE based on domains of information gain, quality, and robustness. Section 16.5 provides a maritime example. Section 16.6 includes a discussion and a conclusion.

Background

Since the 1980s, there has been a distinction between HLIF and LLIF. The distinction was made between traditional methods of control, estimation, and integration methods versus reasoning over relationships, entities, and global predictions. Such a distinction is made between physical processes and situational reasoning.

Currently, there is a need for systems to determine the intent observation [17, 18], threat prediction [19], situational awareness [20], situational analysis [21], situational prediction [22], force aggregation [23], cyber analysis [24], sensor placement [25], and political analysis [26]; all of which is part of a holistic interpretation of fusion systems [27] and events.

16.2.1 Low-Level Versus High-Level Information Fusion

The distinction between LLIF and HLIF, shown in Figure 16.1, highlights the fact that the fusion levels are developed as processes to support a user by translating data from sources into information for situation analysis [28] and display [29]. The distinction between LLIF and HLIF data fusion is based on the objective or subjective components that are available to the fusion system, and the community is developing tools for fusion assessment [30, 31]. While there is no formal definition of HLIF, it is referenced in the information fusion models and featured in articles [32] and texts [33]. It is suggested that the community determine a terminology that confers the meaning of the information fusion processes.

Low-level data and information fusion typically concerns objective and measureable quantities. Determining a target location and identity is an example of a measureable quantity [34, 35]. State estimation (objects, locations, and identities) and control can be achieved through LLIF concepts. Thus, a summary of LLIF is objective estimation (of objects) through observations.

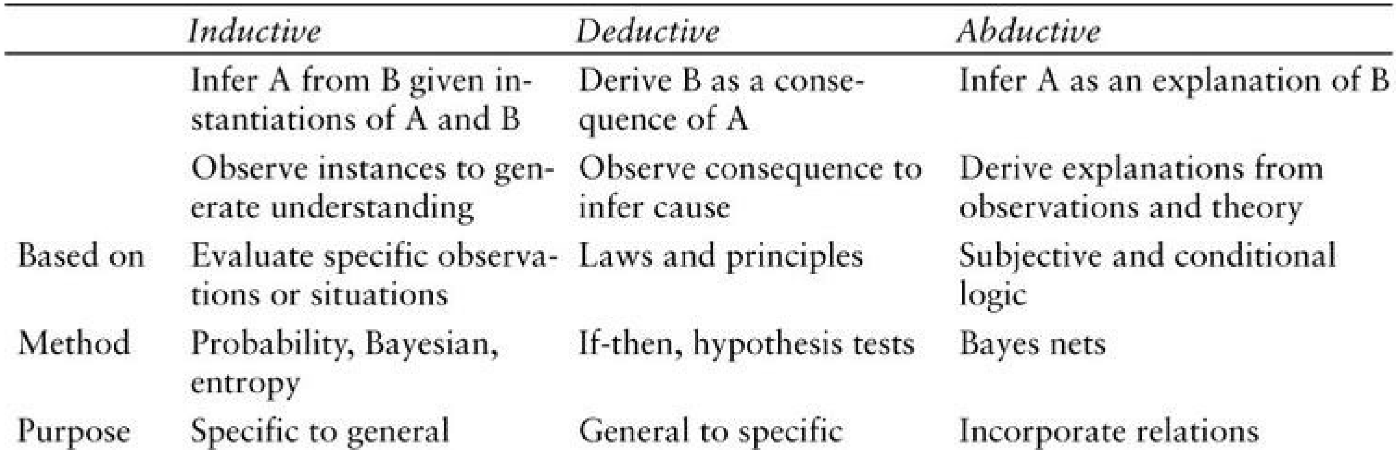
High-level information fusion involves the complex, command, and contextual information that is subjectively reasoned and analyzed to determine the situation, threat, and the operational usability of the information supplied to a user. Recent texts [31, 33] address techniques for HLIF.

Various processes can be both high- and low-level fusion, and can be constructed in a hierarchical fashion to support information and knowledge management [36] and decision support [37]. Such techniques of data mining, user reasoning, resource control, information management [38], and situation assessment have attributes of both high- and low-level fusion.

16.2.2 High-Level Information Fusion as a Form of Reasoning

Information fusion is growing as a technique of interest to the systems engineering community. Thus, it is important to arrive at a terminology that conveys the methods proposed. One distinction (out of many) to consider is the reasoning applied to the various levels for estimation and control. Table 16.1 lists some reasoning methods.

Table 16.1 Reasoning Methods



	Inductive	*Deductive*	*Abductive*
	Infer A from B given instantiations of A and B	Derive B as a consequence of A	Infer A as an explanation of B
	Observe instances to generate understanding	Observe consequence to infer cause	Derive explanations from observations and theory
Based on	Evaluate specific observations or situations	Laws and principles	Subjective and conditional logic
Method	Probability, Bayesian, entropy	If-then, hypothesis tests	Bayes nets
Purpose	Specific to general	General to specific	Incorporate relations

Deductive reasoning arrives at a specific conclusion based on generalizations of physical laws through such means as hypothesis testing. Inductive reasoning (through experience) takes events and makes generalizations. Abductive reasoning (e.g., Bayes nets) allows for an explanation of unrelated data. The type of reasoning affects system evaluation.

16.2.3 Information Fusion Systems Evaluation

IFSs include the technology, algorithms, and environment of operation (to include the people). Moving from LLIF to the abstract reasoning requires integrating the user into the analysis, such as for command and control. For a system to be operational, it needs to be verified (LLIF question) and validated (HLIF question), described briefly as:

Verification: “Am I measuring the IFS correctly?”
Validation: “Am I measuring the correct IFS?”

For instance, in an example of operational maintenance, the normalized timeliness metrics [39] are separated to verify individual machine performance as:

$$\text{Overall Equipment Effectiveness \%} = \text{Available \%} \times \text{Perf. Efficiency \%} \times \text{Quality Rate \%} \qquad (16.1) \qquad (16.1)$$

As per our definition, we bring to light the need for the information gain in establishing the timeliness, accuracy, and confidence associated with an IFS’s ability to help the user reason over data, make decisions, and act on the information. Evaluation looks at the quality of service, data, and information in a testing environment.

16.2.4 Quality of Service/Information Research

To address the MOEs at high- and low-level fusion, we need to look at the quality measures being developed over various domains and reasoning methods. Industry standard definitions, which will be utilized and discussed in Section 16.3, come from QOS [44] and information quality [47] standards.

Johnson and Chang [40] proposed QOI for data fusion in net-centric publish and subscribe architecture to update clients in a QOI paradigm rather than a QOS paradigm. They varied the message length in a QOI system versus fixed time metrics in a QOS system. To facilitate end users' needs in a net-centric environment, a QOI was used because of sensor-web enabled ontology development. They applied the QOI/QOS method to a target tracking example in which they generalized the end users' needs for QOI parameters from which the tracking system conferred the QOS capabilities over state and covariance information. Yu and Sycara [41] addressed the QOI in a distributed decision fusion system by learning the parameters. They applied the technique to determine the QOI information on target reliabilities, or better termed confidences, from a Dempster-Shafer method.

QOI is still an emerging topic as semantics are different for different users and systems. For example, QOI includes accuracy, timeliness, certainty, and integrity [42]. QOI integrity measures whether the data has not been manipulated as it impacts the shared situational awareness [43].

Closely related to QOI, is QOS as it relates to the information flow and availability. QOS has been well vetted in the communications literature [44] as throughput, delay, delay variation, error, and jitter. QOI/QOS requires comparisons of the following: usability versus usefulness; accuracy versus precision; verification versus validity. From these comparisons, we address information gain, quality, and robustness, respectively. As MOPs come from rigorous standard metrics to determine such things as accuracy, there is a need for pragmatic metrics to determine the validity of information aggregation for useful decision making. In the next section, we will look at the information quality literature that addresses issues of information service and information type to advance the discussion in HLIF metrics.

16.2.5 Metric Standardization

Standardization of HLIF performance measures are needed not for research by the fusion community, but rather in the testing, evaluation, and transition of the technology to operational settings. Much work has been completed in addressing various research measures as they pertain to the LLIF; however, formalizing a set of general metrics would aid a testing facility in HLIF end-user requirements. Determining the critical performance measures can be determined from four points of view:

- *Situational awareness*: users working with machines;
- *Situation displays*: machines working with humans;
- *Situation assessment*: users and machines in an enterprise;
- *Situational processing*: information fusion, data mining, sensor exploitation, and so forth; functions to afford human enhanced capabilities.

Information Fusion Quality Measures

To determine the contribution of any system, whether it is hardware or software, one has to test and evaluate the system. The evaluation can be conducted using either (1) simulated data/simulated users, (2) simulated data/real users, or (3) real data/real users. The interchange between the information fusion performance and the user interest is based on the quality of the data. Such standard information fusion QOS measures include timeliness, accuracy, confidence, throughput, and cost. Waltz and Llinas [5] listed timeliness, accuracy, and resolution. Others postulated QOI metrics; however, there is a large set of information quality (IQ) standards available.

Figure 16.2 illustrates an example of high-level needs as the user has many data bases available. Determining what is needed is as important as how good it is. The user requests decision-quality information at the correct time.

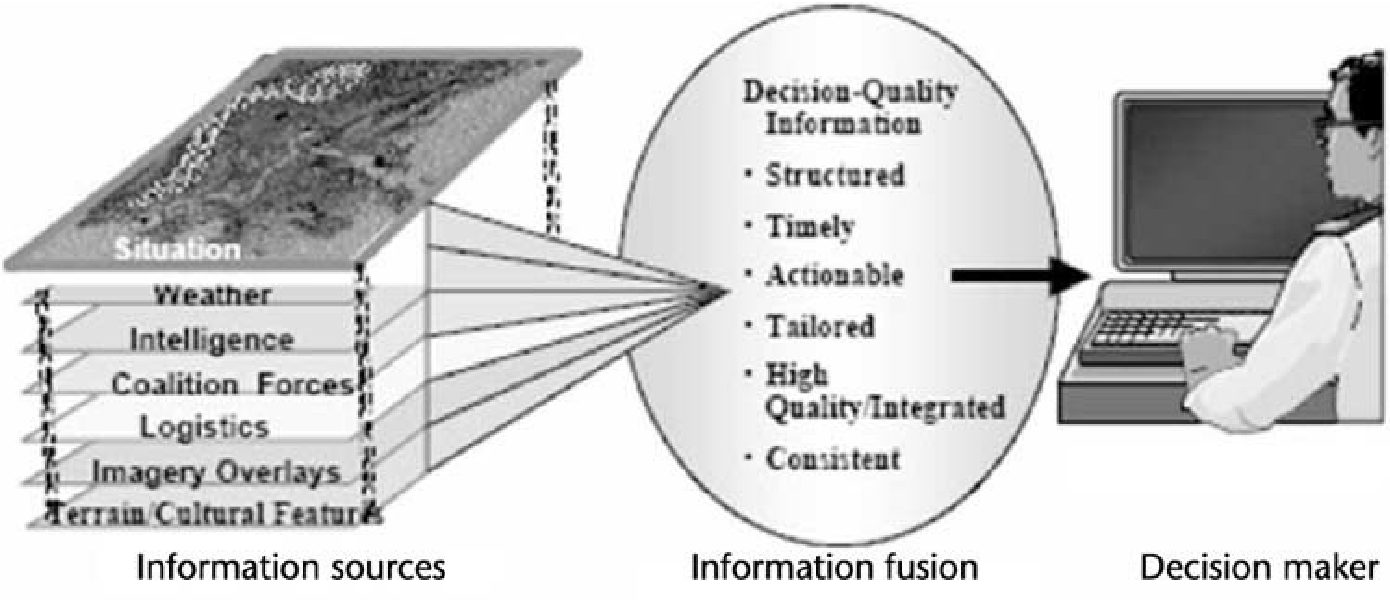


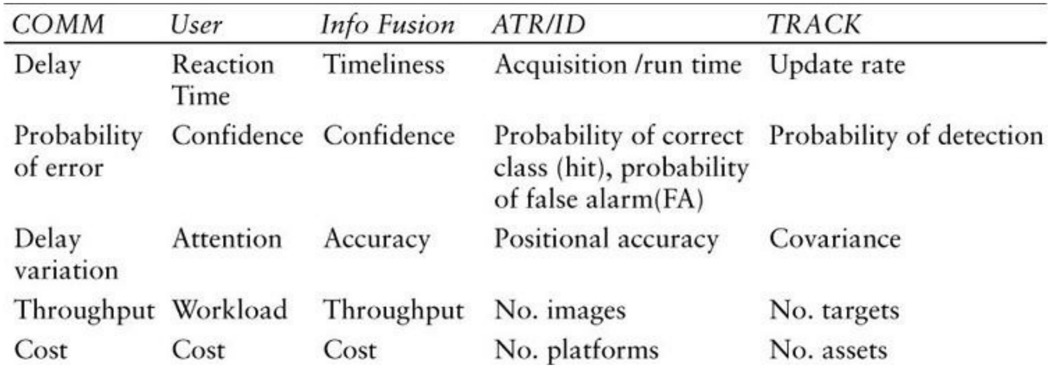
Figure 16.2 Information fusion decision quality [45].

Next, we develop a relation between QOS and IQ.

16.3.1 Quality of Service

Information fusion provides a service to the user. Such issues as efficiency and effectiveness are important in delivering actionable information. To afford interactions between future IF designs and users information needs, metrics are required. The metrics chosen include timeliness, accuracy, throughput, confidence, and cost. These metrics are similar to the standard QOS metrics in communication theory and human factors literature, as shown in Table 16.2 [36].

Table 16.2 Metrics for Various Disciplines



COMM	*User*	*Info Fusion*	*ATR/ID*	*TRACK*
Delay	Reaction Time	Timeliness	Acquisition /run time	Update rate
Probability of error	Confidence	Confidence	Probability of correct class (hit), probability of false alarm(FA)	Probability of detection
Delay variation	Attention	Accuracy	Positional accuracy	Covariance
Throughput	Workload	Throughput	No. images	No. targets
Cost	Cost	Cost	No. platforms	No. assets

In addition to the metrics that establish the core quality (accuracy/integrity) of information, there are issues surrounding information security and quality.

16.3.2 Information Quality

IQ can be described either from the data itself or from the application. IQ from data is based on the accuracy, reliability, and confidence associated with the data. Typically, these quality measures are based on a probabilistic uncertainty, reliability confidence, and ignorance, which is not to be confused with semantic uncertainty, reliability/availability in manufacturing, and incompleteness in possibilistic theory; respectively. Two groups developing IQ standards include the system management literature and the document retrieval literature for organizational effectiveness.

Additionally, IQ is based on the information suitability for a given task and can be subjective as relative to a specific user. For example, information quality assurance is the process to guarantee confidence that particular information meets some context-specific quality requirements. A list of dimensions or elements used in assessing subjective information quality is [46, 47]:

- *Intrinsic IQ*: Accuracy, objectivity, believability, and reputation;
- *Contextual IQ*: Relevancy, value-added, timeliness, completeness, and amount of information;
- *Representational IQ*: Interpretability, ease of understanding, concise representation, and consistent representation;
- *Accessibility IQ*: Accessibility, access security.

Additional measures include:

- *Authority*: expertise or recognized official status of a source, such as the reputation of the author and publisher;
- *Scope of coverage*: extent to which a source explores a topic. Consider time

periods, geography, or jurisdiction and coverage of related or narrower topics;

- *Composition and Organization*: ability of the information source to present its particular message in a coherent, logically sequential manner;
- *Objectivity*: the bias, or lack of bias, expressed when a writer interprets or analyzes facts. Consider the use of persuasive language, the source's presentation of other viewpoints, its reason for providing the information and advertising;
- *Integrity*: adherence to moral and ethical principles; soundness of moral character; the state of being whole, entire, or undiminished;
- *Comprehensiveness*:
  - Of large scope, inclusive: a comprehensive study;
  - Comprehending mentally: an extensive mental grasp;
  - Insurance: providing broad protection against loss.
- *Validity*: degree of truthfulness of the information;
- *Uniqueness*: originating point of the information, but also the manner in which it is presented and thus the perception which it conjures;
- *Timeliness*: current at the time of publication. Consider publication, creation and revision dates; and
- *Reproducibility*: (utilized primarily when referring to instructive information): means that documented methods are capable of being used on the same data set to achieve a consistent result.

IQ has many purposes such as verification and validation.

Verification is a quality assurance process that is used to evaluate whether or not a product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be in development, scale-up, or production, and is often an internal process. It provides assurance that the product will fit certain quality standards and can be a measure of confidence.

Validation is quality control process of establishing evidence (i.e., measuring and testing against the requirements) that a product, service, or system accomplishes its intended requirements. Validation often involves acceptance of fitness for purpose with end users and other product stakeholders.

Quality control and assurance can be derived from the MOM. MOMs are general goals for an IFS to obtain. Green et al. [48] together this list of MOMs that could be useful for a HLIF discussion (see Table 16.3). For instance, the user would advocate these desires from which a HLIF system could calculate.

Table 16.3 Desired Characteristics of MOM

Characteristics	*Definition*
Mission-oriented	Relates to force/system.
Discriminatory	Identifies real difference between alternatives.
Measurable	Can be computed or estimated.
Quantitative	Can be assigned numbers or ranked.
Realistic	Relates realistically to the C2 system and associated uncertainties.
Objective	Defined or derived, independent of subjective opinion (it is recognized that some measures cannot be objectively defined).
Appropriate	Relates to acceptable standards and analysis objectives.
Sensitive	Reflects changes in system variables.
Inclusive	Reflects those standards required by the analysis objectives.
Independent	Mutually exclusive with respect to other measures.
Simple	Easily understood by the user.

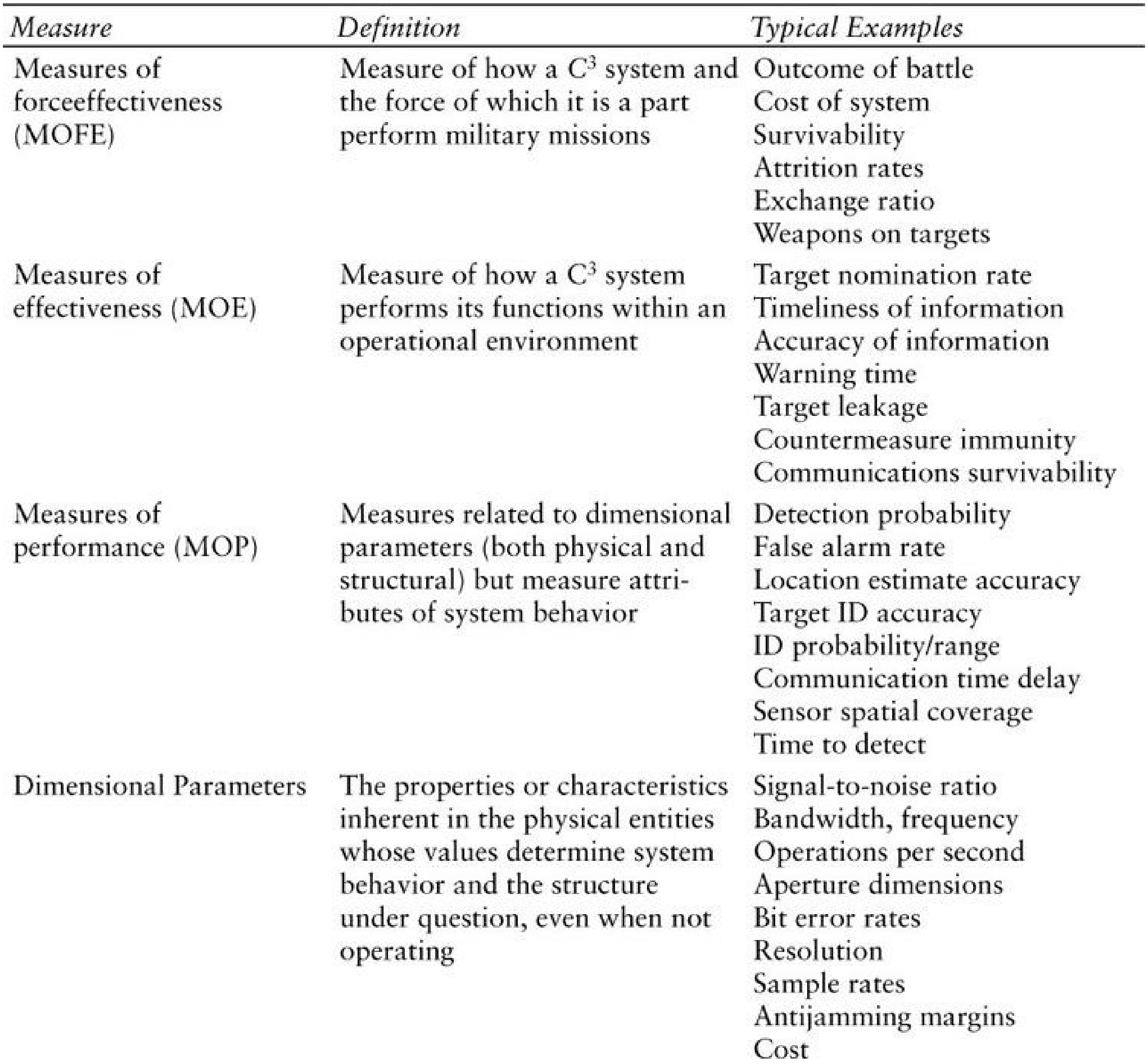
Information Fusion MOEs

MOEs derive from MOMs and typical HLIF metrics have been discussed in a military context. The fusion MOPs lead to MOEs to MOFEs. As the IF community is maturing, many new applications require a revisit the general definitions.

16.4.1 Low-Level Information Fusion MOEs

There are groups developing MOEs for LLIF tasks based on object data [49, 50] for military situations. Llinas [6] develops a mapping from LLIF to HLIF measures of military effectiveness. Table 16.4 lists the various metrics and definitions that are useful for HLIF discussion.

Table 16.4 Four Categories of Measures of Merit



Measure	*Definition*	*Typical Examples*
Measures of forceeffectiveness (MOFE)	Measure of how a C^3 system and the force of which it is a part perform military missions	Outcome of battle Cost of system Survivability Attrition rates Exchange ratio Weapons on targets
Measures of effectiveness (MOE)	Measure of how a C^3 system performs its functions within an operational environment	Target nomination rate Timeliness of information Accuracy of information Warning time Target leakage Countermeasure immunity Communications survivability
Measures of performance (MOP)	Measures related to dimensional parameters (both physical and structural) but measure attributes of system behavior	Detection probability False alarm rate Location estimate accuracy Target ID accuracy ID probability/range Communication time delay Sensor spatial coverage Time to detect
Dimensional Parameters	The properties or characteristics inherent in the physical entities whose values determine system behavior and the structure under question, even when not operating	Signal-to-noise ratio Bandwidth, frequency Operations per second Aperture dimensions Bit error rates Resolution Sample rates Antijamming margins Cost

From: Llinas, Ch 20 [6].

For IFS performance analysis, we typically use measures of performance (MOP) to determine the system quality where some MOEs can be viewed as information quality (e.g., timeliness, accuracy). To determine effectiveness, we have to test the measures for robustness as well as the metric usefulness in HLIF tasks.

16.4.2 High-Level Information Fusion MOEs

High level MOEs can be viewed from the discussion of the MOM criteria needs as shown in Table 16.5.

Table 16.5 Criteria Measures

Level 1 Criteria	*Level 2, 3 Criteria*
Accuracy	Correctness in reasoning
Repeatability/consistency	Quality of decisions/advice/ recommendations
Robustness	Intelligent behavior
Computational complexity	Adaptability in reasoning (robustness)

From: Llinas, Ch 20 [6].

From Table 16.5, both quality and robustness are addressed with correctness and intelligence to support decision making. Decision-making requires metrics for understanding and situational awareness goal-directed intelligent behavior as shown in Table 16.6.

Table 16.6 Situation Awareness

		Phase	
		Process	*Outcome*
Goal	Tactical (Short-term)	Situation assessment	Situation awareness
	Strategic (long-term)	Sense making	Understanding

From: [51].

Using the definition of HLIF, we can determine a summary set of metrics for situation, threat, and user assessment as shown in Table 16.7.

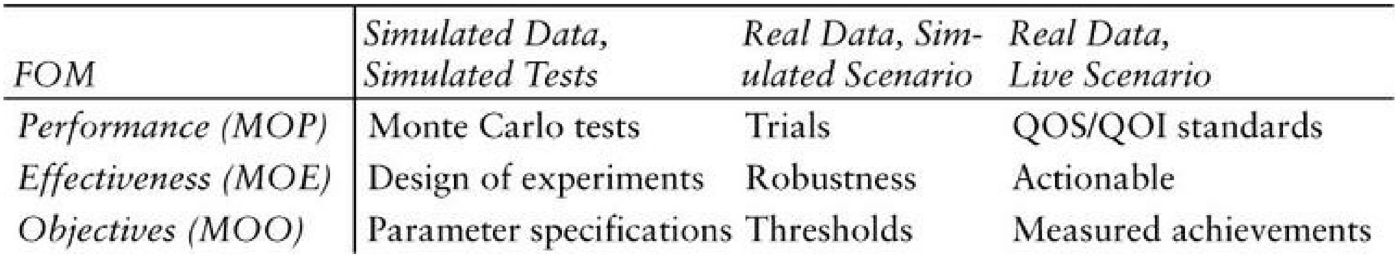
Table 16.7 High-Level MOEs

	Situation	*Threat*	*User*
MOFE	Comprehension	Survivability	Command
MOE	Awareness	Resilience	Timeliness
MOP	Detection	Risk assessment	Actionable information

16.4.3 Organizational Effectiveness

There are many ways to measure the effectiveness of an organization. Using an example of information systems functions, we can measure quality, productivity, and efficiency [52]. A system provides information exchange with the environment to the organization (teams) through information processing, profitability, flexibility, and adaptability [53, 54]. Based on the thresholds and objectives of the test, we can determine the system capability to perform in the field. Testing the system and the organization in a real scenario is required to meet measures of objectives (MOOs), detailed in Table 16.8.

Table 16.8 Testing Figure of Merit (FOM) Measures



FOM	Simulated Data, Simulated Tests	Real Data, Simulated Scenario	Real Data, Live Scenario
Performance (MOP)	Monte Carlo tests	Trials	QOS/QOI standards
Effectiveness (MOE)	Design of experiments	Robustness	Actionable
Objectives (MOO)	Parameter specifications	Thresholds	Measured achievements

Sproles [55] discussed effectiveness evaluation issues. Such issues as real-world events [56] and knowledge representation [57] determine the efficacy of the system. However, there is a need for modeling [58] and system evaluation [59, 60].

In Section 16.3, we discussed quality measures and in Section 16.4, we discussed various effectiveness metrics; however, these metrics need to be tested to constitute robustness. Given the many references of HLIF exemplar metrics, analysis, and discussion, one issue to address is whether or not there is an information fusion gain.

16.4.4 Information Gain

Information gain is the ability of the system to provide improvement – such as in a receiver operator curve [61] or entropy analysis [62]. Das [33] describes a measure of effectiveness for decision trees using information gain.

Within the above algorithm, a measure of effectiveness, called information gain, [33] of an attribute A is computed via the procedure *Gain(Training Set, A)*. The attribute that provides the maximum information gain is placed at the root of the decision tree. The information gain metric is an information theoretic measure of how much entropy is revealed by a specific attribute. Given a collection S of c class labels, the entropy is defined as:

$$Entropy(S) = -\sum_{i=1}^{c} p_i \log_2(p_i) \tag{16.2}$$

where p_i is the proportion of S belonging to class i. The formula for computing information gain for an attribute A with respect to a set of instances S is:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} Entropy(S_v) \tag{16.3}$$

where the sum is taken over all possible values of the attribute A, and S_v is the subset of A for which the attribute A has the value v. Das gives an example to account for movement over various weather conditions (normal, rain, foggy), visibility (clear, poor), and mobility (move, slow, stop).

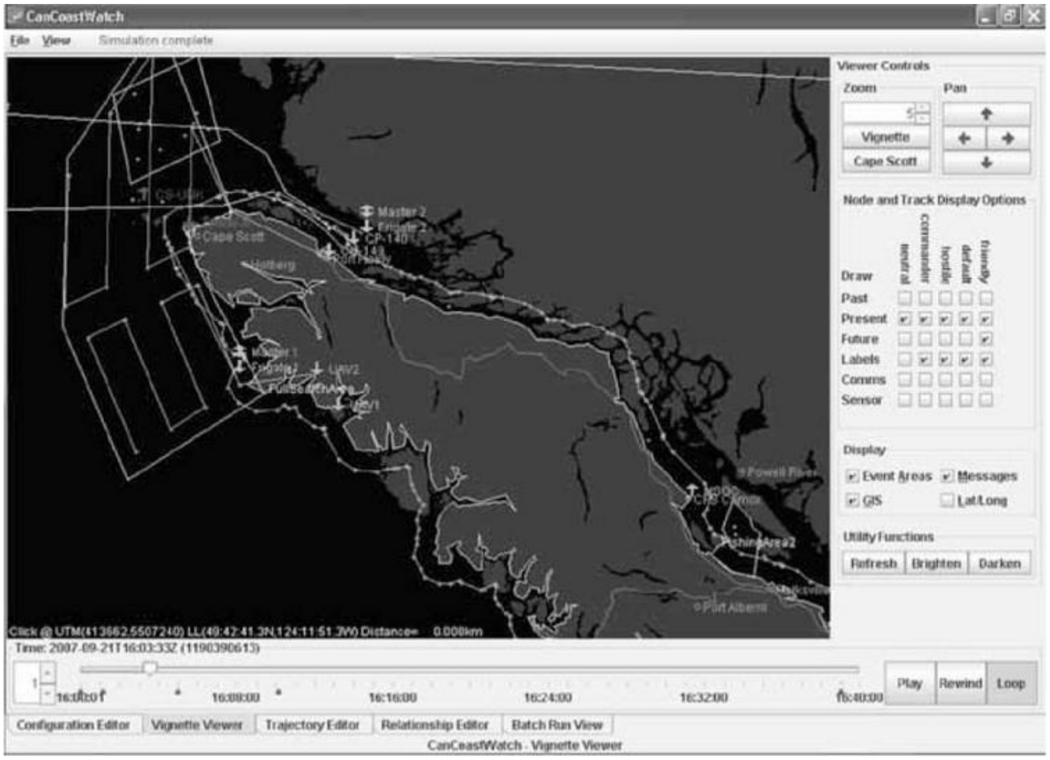
Situation Awareness Example

To determine HLIF MOEs, we need to consider information gain, quality, and robustness.

- InfoGain = value-added aggregation of elements of a situation (e.g. ability to link different regions of activity into a common temporal/spatial operational picture)
- Quality = timeliness for actionable information, uncertainty reduction, and information confidence
- Robustness = coping with real-world variation

$$\text{Fusion Effectiveness} = \text{Info Gain} * \text{Quality} * \text{Robustness} \tag{16.4}$$

Our example comes from a need to protect and address user needs to protect a coastal area [63–65]. Using the CanCoastWatch (CCW) testbed renamed the INFORM Lab described in Chapter 7, Li et al. [66] looked at high-level data in a goal-driven, net-enabled distributed data fusion system, as shown in Figure 16.3.



Draw	neutral	commander	hostile	default	friendly
Past	☐	☐	☐	☐	☐
Present	☑	☑	☑	☑	☑
Future	☐	☐	☐	☐	☑
Labels	☐	☑	☑	☑	☑
Comms	☐	☐	☐	☐	☐
Sensor	☐	☐	☐	☐	☐

☑ Event Areas ☑ Messages
☑ GIS ☐ Lat/Long

Click @ UTM(413662.5507240) LL(49:42:41.3N,124:11:51.3W) Distance= 0.000km
Time: 2007-09-21T16:03:33Z (1190390613)

Figure 16.3 Example of maritime domain awareness.

In the maritime domain awareness system, the goal is to provide search and coverage of possible activities. Parameters used in the search include: search area, time of last visit, detections from radar, sonar and detections from video, and communication. Low level SAR, EO/IR, and radar information provide object track and identity (ID) information. The situational evidence is gathered from the commanders needs through goals. Observations provide the ability of a user to observe, orient, decide, and act over the track and ID information. Both the IFS and the user reasons over the situation and determines a level of confidence in the analysis.

Wehn et al. [67] looked at the decision function as related to planning and resource management. Goals were to evaluate the system effectiveness for net-enabled operations such as synchronization, scheduling, and search. Using the goals of HLIF planning, effectiveness, and information content, we revisit the scenario with the general HLIF metrics.

In a cooperative scenario, detected objects can be confirmed; however for the non-cooperative case, the detected objects cannot be communicated with. Our MOE (Table 16.9) includes information gain (IG) (confidence of track and ID information), information quality (IQ) (data fusion coverage area), and robustness (whether the information is consistent or needs multiple verifications). Robustness can be repeated observations or confirmed through communication.

Table 16.9 Variables of HLF-MOE

IG	*IQ*	*Robustness*
Fusion of radar, sonar, and video	Uncertainty with area covered	Repeated measurements

Using the search area, we wish (assuming the same number of measurements) to optimally utilize the sensors. We look at three scenarios where different data comes from two heterogeneous sensors. We calculate the information gain (same sensor with different looks), the quality of the data (as related to the sensor resolution as a measure of confidence), and robustness as a measure of consistency. Assuming that the system is robust (which could be better modeled with real world data) we determine the effectiveness of the three approaches shown in Figure 16.4 in the high traffic areas (upper left of Figure 16.3).

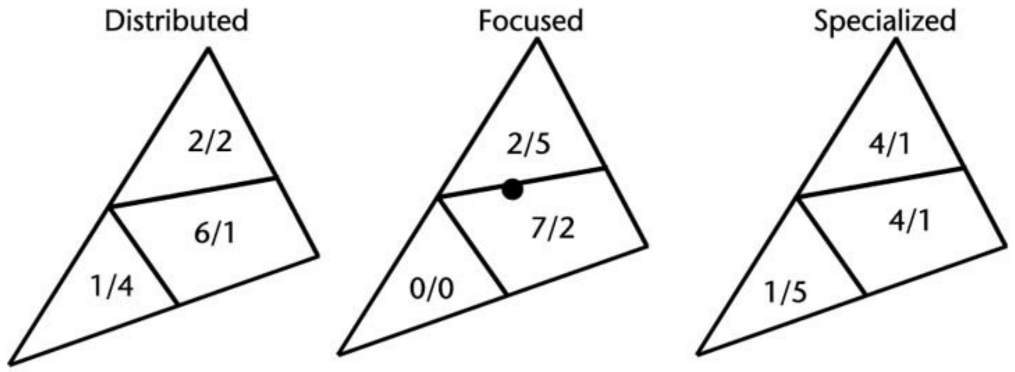


Figure 16.4 Information gain evaluation (a) distributed, (b) focused, and (c) specialized maritime domain awareness approach (EO looks/radar looks).

Case 1 (*distributed approach*) shows that the high-resolution EO sensor (9 looks) is focused at the areas with more traffic, while the low-resolution radar (7 looks) looks at the periphery. Case 2 (*focused approach*) only concentrates at the areas of traffic, discounting low probability information in the areas with less traffic. Finally, Case 3 (*specialized approach*) partitions the sensor capabilities with the high-resolution sensor looking at the high dense areas, while the low-resolution sensor is scanning the areas with less traffic. Results are shown in Table 16.10.

Table 16.10 Results of MOE for SA

	Distributed	*Focused*	*Specialized*
Info gain	0.23	0.18	0.29
Info quality	0.68	0.85	0.625
Effectiveness	0.157	0.154	0.18

From this simple example, we see that the focused and distributed approaches give about the same effectiveness with a tradeoff in IG and IQ. One challenge in the focused approach is that the low-probability areas (which could be the areas the operators need the most help) are discounted. The specialized approach validates its effectiveness in that the best sensors are applied to the situations that they can monitor. The information gain is larger, which increases effectiveness since the same distribution of looks over similar quadrants yield about the same IQ. Thus, it is better to have a wider and specialized sampling to be more effective, which should be intuitive.

Figure 16.5 presents an analysis of robustness, information gain, and information quality that lead to a repeated analysis of a measure of effectiveness. For the MOE update, a Bayes analysis (described in Chapter 13) was used in a repeated update of the MOE. Five trials were done for each time step (robustness) with a random interval of sensor sampling at each time step. The information gain was calculated as in (16.3). The sensor quality was determined from the EO/SAR sensors, as described in Chapter 13, used to calculate the variance of the sensor spatial coverage.

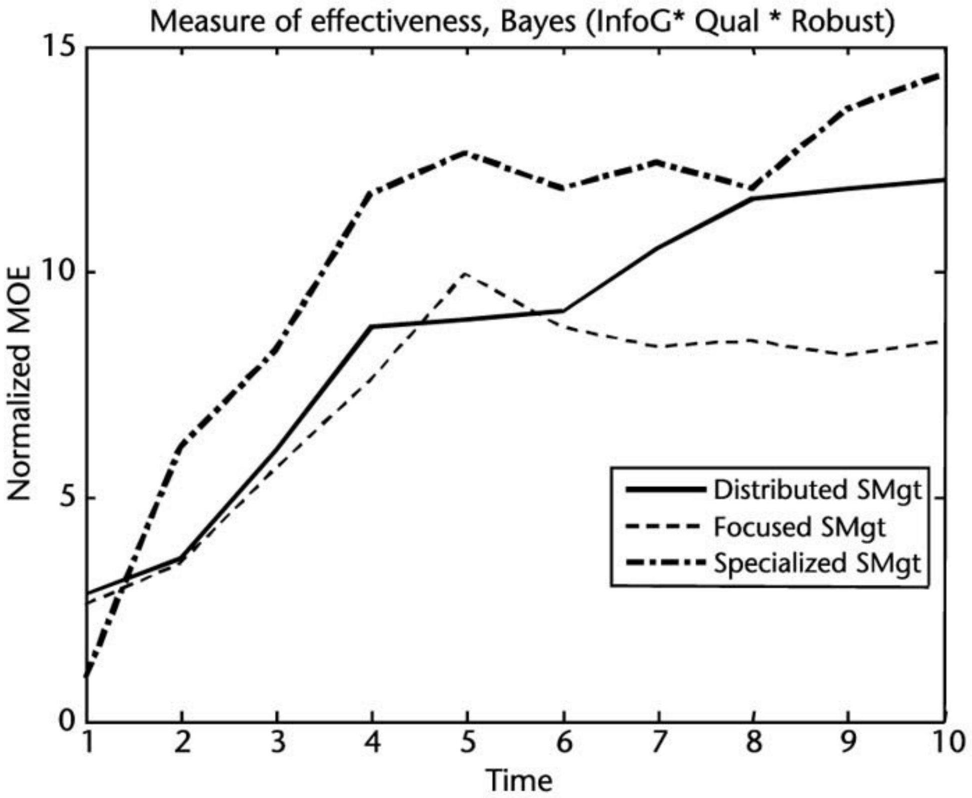


Figure 16.5 Measure of effectiveness evaluation for an INFORM Lab scenario.

The CCW scenario (described in Chapter 11) and the INFORM testbed (described in Chapter 7), and the coalition testbeds (described in Chapters 6, 8, and 12) emphasize the need for effectiveness metrics. The MOE analysis can be coordinated with future developments in visualization (described in Chapter 9), user-machine agreement coordination (described in Chapter 8), and information management (described in Chapter 5). The track/ID scenario relates to the operation condition modeling (described in Chapter 13) and the C-OODA model (described in Chapter 10) to support situation analysis and awareness, as outlined in Chapters 2–4.

Conclusions

In order to address issues for HLIF measures of effectiveness, this chapter brings together prior research to postulate a simplistic taxonomy to establish a discussion. Since standard LLIF tracking and identification performance metrics are available, a similar set of HLIF MOE standards is needed. For an IFS to be operational, there needs to be a verified information gain and validation of robustness over various situational operating conditions. In a simple scenario of the INFORM Lab testbed, we simulated the case for effectiveness as a compiled from information gain, quality, and robustness parameters.

rences

[1] Klein, L. A., *Sensor and Data Fusion: A Tool for Information Assessment and Decision Making*, SPIE Press, 2004.

[2] Hall, D. L., and S. A. McMullen, *Mathematical Techniques in Multisensor Data Fusion*, Norwood, MA: Artech House, 2004.

[3] Blasch, E., and P. Valin, "Track Purity and Current Assignment Ratio for Target Tracking and Identification Evaluation," *International Conference on Information Fusion*, 2011.

[4] Djiknavorian, P., D. Grenier, and P. Valin, "Analysis of Information Fusion Combining Rules Under the DSm Theory using ESM Inputs," *International Conference on Information Fusion*, 2007.

[5] Waltz, E., and J. Llinas, "System Modeling and Performance Evaluation," *Multisensor Data Fusion Systems*, Norwood, MA: Artech House, 1990.

[6] Llinas, J., "Assessing the Performance of Multisensor Fusion Processes," in *The Handbook of Multisensor Data Fusion: Theory and Practice*, D. Hall and J. Llinas (eds.), CRC Press, 2001.

[7] Theil, A., L. J. H. M. Kester, and E. Bosse, "On Measures of Performance to Assess Sensor Fusion Effectiveness," *International Conference on Information Fusion*, 2000.

[8] Van Laere, J., "Challenges for IF Performance Evaluation in Practice," *International Conference on Information Fusion*, 2009.

[9] Blasch, E., P. B. Deignan Jr., S. L. Dockstader, M. Pellechia, K. Palaniappan, and G. Seetharaman, "Contemporary Concerns in Geographical/Geospatial Information Systems (GIS) Processing," *Proceedings of IEEE National Aerospace Electronics Conference (NAECON)*, 2011.

10] Vicente, K. J., *Cognitive Work Analysis*, *Toward Safe, Productive and Healthy Computer-based Work*, Lawrence Erlbaum, 1999.

11] Jousselme, A-L., P. Maupin, and E. Bosse, "Formalization of Uncertainty in Situation Analysis," *DSTO Conference*, 2003.

12] Auger, A., and J. Roy, "Expression of Uncertainty in Linguistic Data," *International Conference on Information Fusion*, 2008.

13] http://en.wikipedia.org/wiki/Effectiveness.

14] Lambert, D. A., "Situations for Situation Awareness," *International Conference on Information Fusion*, 2001.

15] Salerno, J., "Information Fusion for Situational Awareness," *International Conference on Information Fusion*, 2003.

16] Pannetier, B., and J. Dezert, "Extended and Multiple Target Tracking: Evaluation of an Hybridized Solution," *International Conference on Information Fusion*, 2011.

17] Blasch, E., R. Breton, and P. Valin, "Using the C-OODA Model for CIMIC Analysis," *Proceedings of IEEE National Aerospace Electronics Conference (NAECON)*, 2011.

18] Wei, M., G. Chen, et al., "Game-Theoretic Modeling and Control of Military Operations with Partially Emotional Civilian Players," *Decision Support Systems*, Vol. 44, No. 3, 2008.

19] McMichael, D., and G. Jarrad, "Grammatical Methods for Situation and Threat Analysis," *International Conference on Information Fusion*, 2005.

20] Salerno, J., M. Hinman, and D. Boulware, "Building a Framework for Situational Awareness," *International Conference on Information Fusion*, 2004.

21] Kettani, D., and J. Roy, "A Qualitative Spatial Model For Information Fusion and Situation Analysis," *International Conference on Information Fusion*, 2000.

22] Svenson, P., T. Berg, P. Hörling, M. Malm, and C. Mårtenson, "Using the Impact Matrix for Predictive Situational Awareness," *International Conference on Information Fusion*, 2007.

23] Schubert, J., "Evidential Force Aggregation," *International Conference on Information Fusion*, 2003.

24] Stotz, A., and M. Sudit, "Information Fusion Engine for Real-time Decision-making (INFERD): A Perceptual System for Cyber Attack Tracking," *International Conference on Information Fusion*, 2007.

25] Svenson, P., "Equivalence Classes of Future Paths for Sensor Allocation and Threat Analysis," *International Conference on Information Fusion*, 2004.

26] Waltz, E., *Knowledge Management in the Intelligence Enterprise*, Norwood, MA: Artech House, 2003.

27] Valin, P. "Unified Framework for Information Fusion," *International Conference on Information Fusion*, 2001.

28] Shahbazian, E., D. E. Blodgett, and P. Labbé, "The Extended OODA Model for Data Fusion Systems," *International Conference on Information Fusion*, 2001.

29] Duquet, J-R., M. Gregoire, M. Lohrenz, K. Kesavadas, E. Shahbazian, T. Smestad, A. Vanderbilt, and M. Varga, "Fusion and Automation—Human Cognitive and Visualization Issues," *IST-043/RWS-006 Working Group 6 Report*, 2005.

30] Matheus, C., M. Kokar, K. Baclawski, J. A. Letkowski, C. Call, et al., "SAWA: an Assistant for Higher-level Fusion and Situation Awareness," *Proceedings of SPIE*, Vol. 5813, 2005.

31] Bosse, E., J. Roy, and S. Wark, *Concepts, Models, and Tools for Information Fusion*, Norwood, MA: Artech House, 2007.

32] Lambert, D. A., "Tradeoffs in the Design of Higher-level Fusion Systems," *International Conference on Information Fusion*, 2007.

33] Das, S., *High-Level Data Fusion*, Norwood, MA: Artech House, 2008.

34] Blasch, E., and Hong, L., "Data Association through Fusion of Target Track and Identification Sets," *International Conference on Information Fusion*, 2000.

35] Chen, H., G. Chen, E. Blasch, and T. Schuck, "Robust Track Association and Fusion with Extended Feature Matching," *Optimization & Cooperative Control Strategies*, M. J. Hirsch et al. (Eds.), LNCIS 381, Berlin, Heidelberg: Springer-Verlag, 2009.

36] Blasch, E., "Level 5 (User Refinement) Issues Supporting Information Fusion Management," *International Conference on Information Fusion*, 2006.

37] Nilsson, M., and T. Ziemke, "Information Fusion: A Decision Support Perspective," *International Conference on Information Fusion*, 2007.

38] Kessler, O., and F. E. White, "Data Fusion Perspectives and Its Role in Information Processing," in *Handbook of Multisensor Data Fusion: Theory and Practice*, *2nd Edition*, M. Liggins, D. Hall and J. Llinas (eds.), CRC Press, 2008.

39] http://en.wikipedia.org/wiki/Overall_equipment_effectiveness.

40] Johnson, M. E., and K. C. Chang, "Quality of Information for Data Fusion in Net Centric Publish and Subscribe Architectures," *International Conference on Information Fusion*, 2005.

41] Yu, B., and K. Sycara, "Learning the Quality of Sensor Data in Distributed Data Fusion," *International Conference on Information Fusion*, 2006.

42] Hossain, M. A., P. K. Atrey, and A. El Saddik, "Modeling Quality of Information in Multi- Sensor Surveillance Systems," *IEEE Int. Conf. on Data Engineering Workshop*, 2007.

43] Perry, W., D. Signori, and J. Boon, *Exploring the Information Superiority: A Methodology for Measuring* *the Quality of Information and Its Impact on Shared Awareness*, RAND Corporation, 2004.

44] http://en.wikipedia.org/wiki/Quality_of_service.

45] Phister, P. W., and I. Ploisch, "Data Fusion 'Cube': A Multi-Dimensional Perspective," *International Command and Control Research and Technology Symposium*, 2002.

46] Wang, R., and D. Strong, "Beyond Accuracy: What Data Quality Means to Data Consumers," *Journal of Management Information Systems*, Vol. 12, No. 4, p. 5–34, 1996.

47] http://en.wikipedia.org/wiki/Information_quality.

48] Green, J. M., and R. W. Johnson, "Towards a Theory of Measures of Effectiveness," *International Command and Control Research and Technology Symposium*, 2000.

49] Broman, V., and J. Pack, "Measures of Effectiveness for the Distributed Data Fusion Problem," *Tech. Report 1648*, Navy Research and Development Center, 1994.

50] VanDyke, J. P., J. L. Tomkins, and M. D. Furnish, "Measures of Effectiveness of BMD Mid-Course Tracking on MIMD Massively Parallel Computers," *Sandia Report*, SNAD95- 0798, 1995.

51] http://en.wikipedia.org/wiki/Situation_awareness.

52] Campbell, J. P., "On the Nature of Organizational Effectiveness," in *New Perspectives on Organizational Effectiveness*, P. S. Goodman and J. M. Pennings (eds.), Jossey-Bass, 1977.

53] Myers, B. L., L. A. Kappelman, and V. R. Prybutok, "A Comprehensive Model for Assessing the Quality and Productivity of the Information Systems Function: Toward a Contingency Theory for Information Systems Assessment," *Information Resources Management Journal*, Winter 1997.

54] Bietzel, S. M., E. C. Jensen, A. Chowdhury, et al., "Recent Results on Fusion of Effective Retrieval Strategies in the Same Information Retrieval System," *ACM Symposium On Applied Computing*, 2003.

55] Sproles, N., "Coming to Grips with Measures of Effectiveness," *Systems Engineering*, 2000.

56] Cordesman, A. H., "Afghanistan and Measure of Effectiveness," www.csis.org/Burke/reports, July, 2008.

57] Schuck, T., and E. P. Blasch, "Description of the Choquet Integral for Tactical Knowledge Representation," *International Conference on Information Fusion*, 2010.

58] Smith, N., and T. Clark, "A Framework to Model and Measure System Effectiveness," *International Command and Control in RTS*, 2004.

59] Lingard, D. M., and D. A. Lambert, "Evaluation of the Effectiveness of Machine-Based Situation Assessment," *Artificial Life—Borrowing from Biology*, Vol. 5865, Springer Berlin, 2009.

60] Lefebvre, E., M. Hadzagic, and E. Bosse, "Information Quality Assessment in Fusion Systems," *COGnitive Systems with Interactive Sensors Conference*, 2009.

61] Blasch, E. P., J. Hoffman, and J. Petty, "Defining a Fusion Gain – System Operation Characteristic Curve," *Proceedings of SPIE*, Vol. 4380, 2001.

62] Stephanou, H. E., and S-Y. Lu, "Measuring Consensus Effectiveness by a Generalized Entropy Criterion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 4, 1988, pp. 544–554.

63] Shahbazian, E., M. J. DeWeert, and G. Rogova, "Findings of the NATO Workshop on Data Fusion Technologies for Harbor Protection," *Proceedings of SPIE*, Vol. 6204, 2006.

64] Nilsson, M., J. van Laere, T. Ziemke, and J. Edlund, "Extracting Rules from Expert Operators to Support Situation Awareness in Maritime Surveillance," *International Conference on Information Fusion*, 2008.

65] Waller, D., M. R. MacLeod, and T. McCallum, "Measures of Effectiveness and Performance for the Northern Watch Station," *DRDC CORA TM 2008-053*, July 2009.

66] Li, Z., H. Leung, P. Valin, and H. When, "High Level Data Fusion System for CanCoast-Watch," *International Conference on Information Fusion*, 2007.

67] When, H., R. Yates, P. Valin, A. Guitouni, E. Bosse, A. Dlugan, and H. Zwick,“A Distributed Information Fusion Testbed for Coastal Surveillance,” *International Conference on Information Fusion*, 2007.

CHAPTER 17

Summary

Erik P. Blasch (United States), Éloi Bossé (Canada), and Dale A. Lambert (Australia)

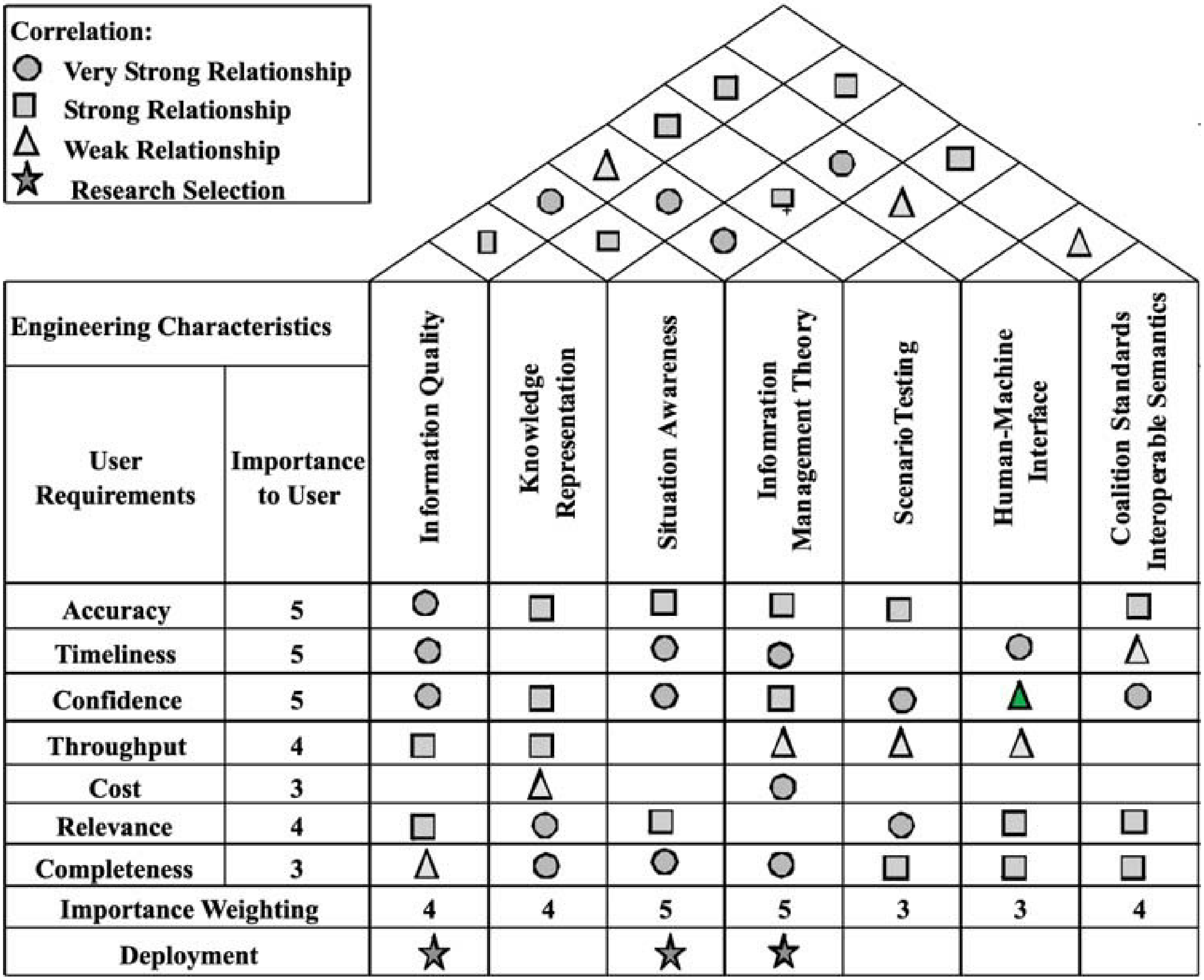
From the many topics published in the literature on Information Fusion, we highlighted the unique emerging developments in high-level information fusion (HLIF) with a focus on situation assessment (SA) and impact assessment (IA). SA and IA are enabled by supporting data and information fusion methods, and require elements of information management, user appreciation, and evaluation strategies for the transition of emerging products to operational systems. HLIF developments are based on the data, metrics, and scenarios of interest that highlight pragmatic HLIF system improvements for decision support needs.

In this text, we explored the theoretical and representational issues in HLIF. To enable HLIF, we focused on SA and IA, situation awareness (SAW), user support, measures of effectiveness, scenario-based design, and information management. From the various models presented in this chapter, six levels of information fusion were defined (see Chapter 2). Level 0/1 are traditionally thought of as low-level information fusion (LLIF) products, while Levels 2–6 are the HLIF techniques. As a summary, Table 17.1 maps the issues in the text to the levels of information fusion.

Table 17.1 Theoretical Developments Mapped to Levels of Information Fusion

	Situation Awareness	*Information Management*	*Scenario-based Design and Evaluation*	*Measures of Effectiveness*
L0–Registration	√	—	—	—
L1–Object assessment	√	—	√	—
L2–Situation assessment	√	√	√	√
L2–Impact assessment	√	√	—	√
L4–Resource management	—	√	√	√
L5–User refinement	—	√	√	√
L6–Mission management	—	√	√	√

From Table 17.1, we see that SA is a key enabler of future information fusion system designs and complements the many other recent texts in information fusion from Artech [1–6]. The various theoretical frameworks presented in the text include the information fusion-situation assessment (IFSA) process model, the state transition data fusion (STDF) functional model, the interpreted systems (IS) formal model, and the cognitive observe-orient-decide-act (C-OODA) cognitive model. All frameworks provide a translation and coordination between the various information fusion levels, with a focus on information representation. In developing the IFSA (Chapter 2), STDF (Chapter 3), IS (Chapter 4) and C-OODA (Chapter 10) frameworks, SA resides in all models and is essential for HLIF designs from across the international cooperation of United States, Australia, Canada, United Kingdom, and New Zealand. The Coalition Distributed Information Fusion Testbed (CDIFT) utilized a simulated maritime environment to (1) support the development of information fusion techniques, (2) coordinate human-machine needs, and (3) test information and resource management strategies. Figure 17.1 integrates the various themes of the text from information representation to theoretical system design implementations.



Correlation:
● Very Strong Relationship
□ Strong Relationship
△ Weak Relationship
☆ Research Selection

		Engineering Characteristics						
User Requirements	Importance to User	Information Quality	Knowledge Representation	Situation Awareness	Infomration Management Theory	Scenario Testing	Human-Machine Interface	Coalition Standards Interoperable Semantics
Accuracy	5	●	□	□	□	□		□
Timeliness	5	●		●	●		●	△
Confidence	5	●	□	●	□	●	▲	●
Throughput	4	□	□		△	△	△	
Cost	3		△		●			
Relevance	4	□	●	□		●	□	□
Completeness	3	△	●	●	●	□	□	□
Importance Weighting		4	4	5	5	3	3	4
Deployment		☆		☆	☆			

Figure 17.1 HLIF design choices based on a house of quality assessment.

From Figure 17.1, raw data products processed by LLIF for object location and identity, support SA. SA theoretical models for information representation and processing impact information management, human-machine interface, and scenariobased designs. Future SA developments require adequate data, information, and knowledge representation to allow standard information fusion techniques (i.e., Bayes rule) to be appropriately applied. For example, the use of operational conditions from a scenario-based design (Chapter 11) can be considered elements of information management that can be refined by users either through information requests that afforded by a user-defined operating picture (UDOP, Chapter 9), a legal agreement protocol (Chapter 8), or as agents in an information management system (Chapters 5–7).

Current Trends

We sought to present and detail some key elements of the multinational developments that are considered essential for the development, assessment, evaluation, and fielding of HLIF systems. The exemplar testbeds (Chapters 6 and 7), including the CDIFT and the Information Fusion and Resource Management Laboratory (INFORM Lab), showcase current developments. Future testbed developments and analysis will be reported in ongoing studies and analysis through comparisons of HLIF methods. However, there is still a need for the HLIF community to develop standard metrics of effectiveness (MOEs) that can be used to upgrade international standards, such as the NATO Standardization Agreement (STANAG) formats (e.g., STANAG 4162-Combination of Identity Declarations).

It is our hope that the readers gained insights into the theories, representations, strategies, and issues surrounding HLIF system design for multinational purposes. A key idea is that for systems to operate internationally, there is a need for common data structures, information management protocols, interoperability semantics, and verification and validation methods.

Common data sets, metrics, and testbeds serve a purpose in evaluation to better knowledge acquisition, standardize data representations, and compare theoretical methods. Figure 17.2 highlights the user requirements for Information Fusion quality of service (QOS) metrics (Chapter 16) as mapped to engineering requirements using the quality functional deployment methodology of a "house of quality," The house of quality serves as a systems design tool to capture the interactions between the customer requirements and systems capabilities and to highlight the correlated areas of development needs. From Figure 17.2, we see that information quality and management are strongly correlated with the need for systems-level situation awareness.

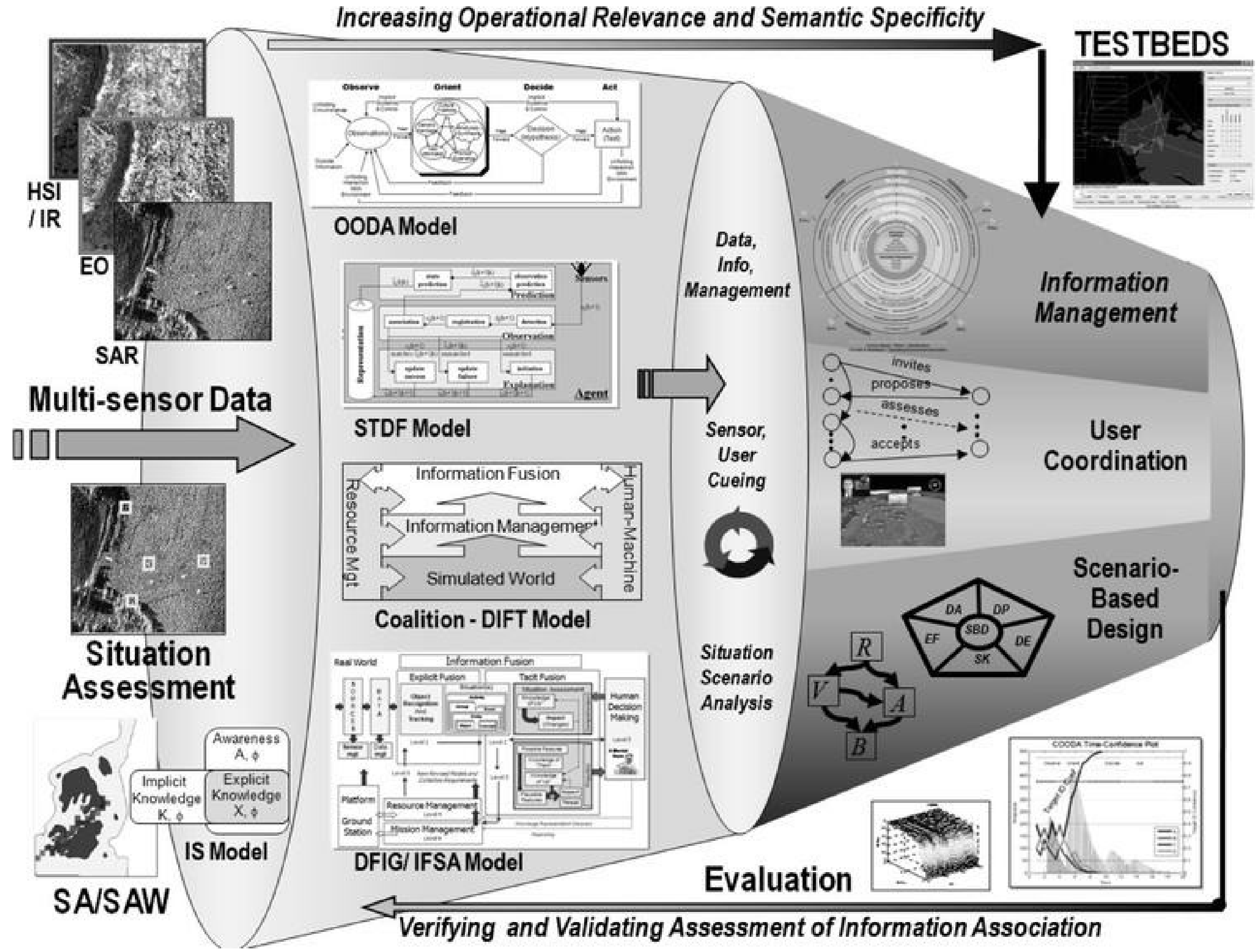


Figure 17.2 Synthesis of the book.

Future

Past Artech books [1–6] (in this series) are important to understanding the developments in information fusion and can serve as valuable complementary references to this text. This book serves as a handbook of current international developments, which focus on collaborative efforts between and among nations written by mostly government authors. The compiled material brings together contemporary issues for the emerging science of high-level information fusion. Future assessment and collaboration of information fusion developments will stem from the ideas in the text and with the growth of the information age, emerging topics such as data mining, sensor exploitation, interoperable resource management, diverse user integration, information dissemination, international standards, and business practices will be important. Future analysis, results, and documentation, will help continue to mature the science of high-level information fusion.

We invite all readers to bring their ideas and refinements to continue to coalesce the science of HLIF.

rences

[1] Waltz, E., and J. Llinas, *Multisensor and Data Fusion*, Norwood, MA: Artech House, 1990.

[2] Blackman, S. and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Norwood, MA: Artech House, 1999.

[3] Waltz, E., *Knowledge Management in the Intelligence Enterprise*, Norwood, MA: Artech House, 2003.

[4] Bossé, É., J. Roy, and S. Wark, *Concepts, Models, and Tools for Information Fusion*, Norwood, MA: Artech House, 2007.

[5] Das, S., *High-Level Data Fusion*, Norwood, MA: Artech House, 2008.

[6] Hall, D. L., and J. M. Jordan, *Human-Centered Information Fusion*, Norwood, MA: Artech House, 2010.

About the Editors

Erik P. Blasch received a B.S. in mechanical engineering from the Massachusetts Institute of Technology in 1992 and M.S. degrees in mechanical engineering ('94), health science ('95), and industrial engineering (human factors) ('95) from Georgia Tech and also attended the University of Wisconsin for an M.D./Ph.D. in mechanical engineering/neurosciences until being called to active duty in 1996 to the United States Air Force. He completed an M.B.A. ('98), M.S.E.E. ('98), M.S. econ ('99), M.S./Ph.D. psychology (ABD), and a Ph.D. in electrical engineering from Wright State University and is a graduate of Air War College. From 2000–2010, Dr. Blasch was the information fusion evaluation tech lead for the AFRL Sensors Directorate—COMprehensive Performance Assessment of Sensor Exploitation (COMPASE) Center, adjunct professor with Wright State University, and a reserve officer with Air Office of Scientific Research. From 2010–2012, Dr. Blasch was an exchange scientist to Defence R&D Canada at Valcartier, Quebec in the Future Command and Control (C2) Concepts group. He received the 2009 IEEE Russ Bioengineering Award and compiled over 30 top ten finishes as part of robotic teams in international contests including winning the '91 American Tour del Sol solar car, the '93 Aerial Unmanned Vehicle, and the '94 AAAI mobile robotics competitions. He is a past president of the International Society of Information Fusion (ISIF), a member of the IEEE AES Board of Governors, a SPIE Fellow, and active in AIAA and ION. His research interests include target tracking, information/sensor fusion, pattern recognition, and biologically inspired robotics.

Dale A. Lambert received a B.Sc. in computer science, a first class B.A (Hons) in philosophy, and a B.A. in mathematics from The Flinders University of South Australia. He subsequently undertook postgraduate master's degree work in mathematical logic, before completing a Ph.D. in artificial intelligence under a full Flinders University Research Scholarship. A Grad. Cert. in management from the University of South Australia followed, together with completion of the Executive Leadership Development Program. He has served as a postgraduate supervisor on behalf of several universities and is a frequent conference and journal reviewer. Dr. Lambert is the research leader of intelligence processing and analysis within Australia's Defence Science and Technology Organisation, having previously served as head of C2 Situation Awareness, head of the DSTO Fusion for Situation Awareness Initiative and head of Human Systems Integration. Prior to those appointments, Dr. Lambert participated in, and then led, a section working in the areas of sensor fusion, tracking, and intelligent control. These appointments collectively involved the development and delivery of an assortment of science and technology solutions to various clients, including foreign nations, and many public representations in the national print, radio, and television media. Dr. Lambert is the recent outgoing chair of The Technical Cooperation Panel (TTCP)’s Command, Control, Communications and Information Systems (C3I) Technical Panel on Information Fusion (TTCP C3I TP1), having held that position since late 2006, and is still its Australian National Lead. Dr. Lambert is married with two sons.

Éloi Bossé received a B.A.Sc., M.Sc. from Université Laval in 1981 in electrical engineering and Ph.D. degrees in 1990 from Université Laval in conjunction with Ottawa/Carleton Universities. In 1981 he joined the Communications Research Centre, Ottawa, Canada, where he worked on high-resolution spectral analysis and radar tracking in multipath. In 1988 he was transferred to the Defence Research Establishment Ottawa (DREO) and in (1992) to the Defence Research and Development Canada Valcartier (DRDC Valcartier) where he worked on information fusion and decision support. He has published over 160 papers in journals, conference proceedings, and technical reports. Dr. Bossé has been head of the C2 Decision Support Systems Section at DRDC Valcartier from 1998 to 2010 and was the chair for the TTCP C31 TP1 He has recently worked at defining an S&T program to address the challenge of an integrated command and control for National Defence. He currently holds an academic position at Université Laval as an industrial research chair in information fusion and decision support. Dr. Bossé was the executive chair of the 10th International Conference on Information Fusion (FUSION 2007), held in July 2007 in Québec City, Canada. He is coauthor *Concepts, Models, and Tools for Information Fusion*, Artech House, Norwood, 2007 (Non-Atomic Military Research and Development—NAMRAD Principals 2006 achievement AWARD).

List of Contributors

Editors

Erik P. Blasch, Ph.D.
AFRL Exchange Scientist to Defence Research and Development Canada–Valcartier
Decision Support Systems Section
Defence Resarch and Development
Canada–Valcartier
2459 Pie-XI Blvd North Québec, Québec G3J 1X5, Canada
erik.blasch@gmail.com

Éloi Bossé, Ph.D.
Electrical and Computer Engineering Department
Université Laval
2325, rue de l'Université
Québec, Québec G1V 0A6, Canada

Dale A. Lambert, Ph.D.
Research Leader, Intelligence Processing and Analysis
Room L205.2.H.25
Defence Science and Technology Organisation
PO Box 1500
Edinburgh, South Australia 5111
dale.lambert@dsto.defence.gov.au

Authors

Clinton Blackman
United Kingdom

Elizabeth K. Bowman, Ph.D.
U.S. Army Research Laboratory
Computational and Information Sciences
Directorate
Aberdeen Proving Ground, MD 21005
elizabeth.k.bowman.civ@mail.mil

Richard Breton, Ph.D.
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
richard.breton@drdc-rddc.gc.ca

Anne-Claire Boury-Brisset
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
Anne-Claire.Boury-Brisset@drdc-rddc.gc.ca

Greg Chase

DSTO (Defence Science and Technology Organisation), Australia
PO Box 1500
Edinburgh, South Australia 5111
gregchase@adam.com.au

Adel Guitouni, Ph.D.
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
adel.guitouni@drdc-rddc.gc.ca

Jens Happe
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada

Justin Henley
DSTL (Defense Science and Technology Laboratory), United Kingdom
Portsdown West
Fareham
PO176AD, United Kingdom
JTHENLEY@mail.dstl.gov.uk

Michael Hinman
Air Force Research Laboratory
Information Directorate
525Brooks Road
Rome, NY 13441-4505
michael.hinman@rl.af.mil

Peter Houghton
DSTL (Defense Science and Technology Laboratory), United Kingdom
Portsdown West
Fareham
PO176AD, United Kingdom
PDHOUGHTON@mail.dstl.gov.uk

Paul Hyden, Ph. D.
US Naval Research Lab
4555 Overlook Ave S.W
Washington, DC 20375-5337
paul.hyden@nrl.navy.mil

Anne-Laure Jousselme, Ph. D.
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
anne-laure.jousselme@drdc-rddc.gc.ca

James Kraft

Ashley G. Lambert

Snezana Mitrovic-Mimic
Department of Mathematics
Simon Fraser University
Surrey, British Columbia, Canada
snezanam@sfu.ca

Justin Nevitt
US Naval Research Lab
4555 Overlook Ave S.W.
Washington, DC 20375-5337
justin.nevitt@nrl.navy.mil

Mark Linderman, Ph. D.
Air Force Research Laboratory
Information Directorate
525Brooks Road
Rome, NY 13441-4505
mark.linderman@rl.af.mil

Luc Pigeon
Defence Research and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
Luc.Pigeon@drdc-rddc.gc.ca

Robert Read
DTA (Defence Technology Agency)
Private Bag 32901
Auckland 0744, New Zealand
r.read@dta.mil.nz

Stephen Russell, Ph. D.
US Naval Research Lab
4555 Overlook Ave S.W.
Washington, DC 20375-5337
stephen.russell@nrl.navy.mil

John J. Salerno, Ph. D.
Air Force Research Laboratory
Information Directorate
525 Brooks Road
Rome, NY 13441-4505
john.salerno@rl.af.mil

Elisa Shahbazian, Ph. D.
OODA Technologies
4891 Grosvenor Ave
Montréal, Quebec H3W 2M2
elisa.shahbazian@ooda.ca

George P. Tadda
Air Force Research Laboratory
Information Directorate
525Brooks Road
Rome, NY 13441-4505
george.tadda@rl.af.mil

Pierre Valin, Ph. D.
Defence Resarch and Development Canada–Valcartier
2459 Pie-XI Blvd North
Québec, Québec G3J 1X5, Canada
pierre.valin@drdc-rddc.gc.ca

Steven Wark
Defence Science and Technology Organisation
PO Box 1500
Edinburgh, South Australia 5111
Steven.Wark@dsto.defence.gov.au

Hans When
MacDonald, Dettwiler and Associates Ltd.,
3800Commerce Parkway
Richmond, BC V6V 2J3, Canada
hw@mdacorporation.com

Index

A

- Abductive reasoning, 334
- Accuracy/purity, 322
- Activities
  - actor-centric, 243, 244
  - decomposition, 240, 242
  - defined, 23
  - information management, 107
  - validation, 240
- Activities of interest (AOIs), 315, 317–18
  - baseline score, 324
  - computation, 323
  - score, 322–23
  - tracks, 323
- Activity process diagrams, 245, 247
- Agents
  - capacity, 180
  - characteristics, 132–33
  - collaboration and, 132
  - intelligent, 132
  - OODA, 156, 159–61
  - protocols for, 86
  - proxy, 147
  - reasoning schemes, 84
  - scope and functionality, 133
  - sensing, 97
  - time step, 76–77
  - as web services, 131
  - in workflows, 131
- Agreement, 180–81
- Air search ground-based Doppler radar, 48
- Air Tasking Order (ATO), 117
- Assessments
  - confidence, 261–62
  - contract, 187
  - damage, 318
  - impact, 74, 316
  - JDL, 46–47
  - scenario, 76
  - threat, 316
  - *See also* Situation assessment (SA)
- Atlantis countries, 253
- Attack tracks, 320
- ATITTUDE TOO cognitive model

- defined, 173
- software identities, 179

Australian contribution perspectives, 4–5

Automatic target classification (ATC)
- evaluation, 281–82
- object detection and identification by, 280
- OC modeling, 282–83
- systems, 279
- training OCs, 291

Automation, 176–78
- lower-level, 177
- mixed initiative strategy, 177

Awareness, 93, 94
- individual, improving, 201
- of information sources, 209–10

B

Basic Probability Assignment (BPA), 98–99

Batch run editor, 167

Bayes fusion, 286

Bayesian networks, 28

Bayes model, 286

Bayes net model, 293, 294

Behavior Simulation Toolbox, 311, 312

Belief, modeling, 98

Boltzmann’s constant, 49

Boolean, 28

Business Processing Execution Language (BPEL), 129, 130

C

Canadian contributions perspectives, 5–6

Capacity, 180

Certainty, 181–82

Checking algorithms, 95–96

Classifiers, 38–39

Coalition approach
- Air Attack of the Celtic Straits, 255
- Air Attack of the Convoy, 255
- Alliance Convoy, 254
- Blueland ground-based radars, 255, 257–58
- combat search and rescue (CSAR), 273–75
- commercial air corridors, 254, 257
- conclusion, 276
- convoy, 257
- events and order of battle, 258
- Fusion 2+, 258–60
- higher-level COP (HICOP), 268–72

  indicators of collector behavior (ICB), 260–63
  introduction to, 251–52
  Maritime Routes, 255
  Missile Deployment, 255
  platforms, sensor models, trackers, 256–58
  Redland Airborne Surveillance, 255
  Redland Warships, 254, 257
  scenario, 252–55
  State Transition Data Fusion (STDF) model, 263–68
  Submarine Attack, 255
  urban operations, 272–73
  vignette components, 254–55
  Whale Migration, 255
Coalition Distributed Information Fusion Testbed (CDIFT), 137–52
  architecture, 143–51, 256
  CoAX, 141–43
  collaboration models, 137–39
  conclusion, 151–52
  defined, 137, 255
  distributed architecture, 140
  dynamic resource management, 141
  example information flows, 151
  functional components, 143, 255–56
  heterogeneous systems integration, 140
  human-machine interface layer, 148–51
  information fusion layer, 146–47
  information management layer, 144–45
  joint development, 138–39
  joint ownership, 139
  loose coupling between components, 141
  real-time performance, 140
  requirements, 139–41
  resource management layer, 147–48
  simulated information feeds, 139–40
  simulation layer, 143–44
  technology demonstration, 138
  technology sharing, 138
  technology showcase, 137–38
Coalition Secure Management and Operations System (COSMOS), 114
CoAX, 141–43
  CDIFT and, 143
  defined, 141
  illustrated, 142
  technologies, 141
Cognitive OODA (C-OODA), 16, 215–28
  advantages of, 217
  C2 model, 227
  cognitive process in, 223
  comparisons, 215

  - control states, 224
  - defined, 216
  - as descriptive model, 226–27
  - development, 219
  - discussion and conclusions, 227–28
  - first order control model with time delay, 226
  - input/output delay models for, 227
  - introduction to, 215–17
  - loop, 219–23
  - modeling detail, 226
  - overall time response, 228
  - SHOR model for action, 220–21
  - simulation layer (CDIFT), 223–27
  - situation assessment models, 219–20
  - skills-rules-knowledge model, 221–22
  - specifications of time delays, 224
  - state space control module, 225
- Cognitive work analysis (CWA), 221
- Combat search and rescue (CSAR), 273–75
  - combat search techniques, 274
  - defined, 273
  - phases, 274
  - Sea King, 275
  - Tornado, 275
- Combined Air Operations Centre (CAOC), 273, 275
- Command, Control, Communications and Information Systems (C3I), 128
- Command and Control (C2) system, 269
- Command and Control Information Exchange Data Model (C2IEDM), 112
- Common operational picture (COP), 195, 196
- Common Recognized Operational Picture (CROP), 195
- Communities of Interest (COI), 120
- Computational Tree Logic (CTL) operators, 87
- Confidence
  - assessing, 261–62
  - defined, 262
  - measure, 321
- Configuration editor, 167
- Context
  - defined, 120
  - extraction, 120, 121
- Context Aware Access Control (CAAC), 120, 121
- Contracts
  - agreement, 180–81
  - assessments, 187
  - capacity, 180
  - certainty, 181–82
  - consideration, 182
  - formation, 179–82, 183–88
  - invitations, 179–80, 184–85

- offer and acceptance, 181, 188
- performance, 182, 188
- pre-agreed monetary compensation, 183
- preliminary proposal, 181
- proposals, 185–87
- remedies, 182–83, 188
- trust ratings, 183

Control of Agent Based Systems (CoABS), 132, 141
Cost utility, 322
Current trends, 350–52

D

Damage assessment (DA), 318
DAML (DARPA Agent Markup Language), 141
Data fusion
- defined, 34
- DFIG, 37
- JDL model, 35
- roots, 34–35
- situation awareness and, 36
- STDF, 38

Data Fusion Information Group (DFIG) model, 215
- components, 20–21
- defined, 18–19
- illustrated, 19, 36
- levels, 19–20
- OODA in relation to, 218
- situation awareness and data fusion, 37

Data information ratio (DIR), 320
Data standards, 112
Decentralization, 174–76
Decision-level fusion (DLF), 293
Decision-making models, 217
Decision support, 299–300
Deductive reasoning, 334
Defense Lines of Development (DLOD), 199
Defense Science and Technology Organization (DSTO), 146–47, 269
- multi-agent architecture, 146
- virtual advisor, 270

Dempster-Shafer methods, 26, 28
Derived operating conditions, 284
Design challenge, 2, 9
Design of experiments (DOE), 282
Discretization Toolbox, 304, 307–8
Distributed architecture, 140
Dual Node Network (DNN), 173

E

Effectiveness, 32–33
  evaluation issues, 341
  organizational, 340–41
  See also Measures of effectiveness (MOEs)
Efficacy, 332
Efficiency, 332
Electronic countermeasures and reconnaissance (ECR), 275
Emergency locator transmitter (ELT), 274
Endsley SAW model, 17, 219–20
Entities, 23
Entity relationship diagrams, 245, 246
Entropy, 28
Environmental operating conditions, 289–90
Epistemic challenge, 1, 8
Evaluation challenge, 2, 9–10
Events, 23
Exploration problem, 303–4
Extended operating conditions (EOCs), 281–82, 284–85, 295
EXtensible Access Control Markup Language (XACML), 122
  architecture, 145
  defined, 144
  policy management and enforcement, 145
EXtensible Markup Language (XML), 156, 167

F

Figure of merit (FOM) measures, 343
Force Protection Condition (FPCON), 120
Fragmentation, 321–22, 326
Fusion 2+ testbed, 146, 258–60
  into CDIFT, 260
  defined, 258
Fusion process, 43–47
  general form of, 45–46
  JDL assessments, 46–47
  prediction, observation, and explanation, 43–45
Future, 352–53
Fuzzy logic, 28

G

Game theory, 85
Generic Intelligence Processor (GIP), 260
Global states, 85
Graphs, visibility-based pursuit evasion, 301–2
Graph searching, 301
Groups, 23

H

Haiti UDOP, 208–9
Higher-level COP (HICOP), 268–72
  compatibility, 149
  elements, 150
  fully-automated system for, 272
  multimedia narrative, 270–72
  need for, 268
High-level information fusion (HLIF)
Australian contribution perspectives, 4–5
  Canadian contributions perspectives, 5–6
  challenges, 1–2
  coalition approach, 251–76
  as form of reasoning, 334
  information management support, 105–34
  LLIF versus, 332, 333–34
  measures of effectiveness (MOEs), 331–45
  MOEs, 339–40
  performance metrics, 332
  science of, 7–10
  United States contributions perspectives, 6–7
Human-machine interface layer (CDIFT), 148–51
Hypertext Transfer Protocol (HTTP), 52

I

Impact assessment (IA), 316
Indicators of collector behavior (ICB), 260–63
  candidate cluster identification, 261
  CDIFT application, 263
  confidence assessment, 261–62
  defined, 260–61
  inferring intent, 262–63
Information
  common syntax and semantics, 115
  contextual, 120–23
  dynamic policy control of access to, 144
  importance of, 122–23
  packaging for sharing, 115
  redundant, suppression of, 123
  relevance of, 123
  universal services, 116
  unstructured, 123
Information actors
  defined, 110
  elements of, 111
  illustrated, 110
Information Age, 173, 175

Information exploitation (IX) systems, 118–19
Information fusion
  decision quality, 337
  defined, 82
  IM support to, 118–31
  layer (CDIFT), 146–47
  lower levels, 83
  low-level versus high level, 333–34
  model advantages, 82
  models, 82
  MOEs, 339–42
  semantic synonyms, 21–22
  syntactic algorithms, 21–22
  theoretical developments mapped to, 350
Information fusion SA (IFSA) model
  activities of interest and, 317–18
  conclusions, 327
  data information ratio, 320
  defined, 315
  example, 323–27
  illustrated, 316
  integrated, 24–25
  introduction to, 315–17
  metrics, 321–23
  risks, 319–20
  situation assessment concept and, 317–20
Information fusion systems (IFSs)
  evaluation, 332, 335
  function of, 315
Information gain, 333, 341–42, 344
Information lifecycle, 118–19
Information management (IM), 105–34
  activities, 107
  actors, 110
  from agent perspective, 132–33
  best practices, 107, 114–17
  challenges in coalition environment, 112–14
  in coalition environment, 114–15
  complexity reduction, 115–17
  conclusions, 133–34
  contextual information, 120–23
  control and flexibility, 117
  dedicated infrastructure, 116
  defined, 105–6
  functions, 116
  goal, 106
  information sharing, 115
  information spaces, 110
  layer (CDIFT), 144–45

    managed information objects, 109
    model, 107–12
    model illustration, 108
    model utility, 110–12
    as service, 126–28
    service layers, 110, 112
    in supporting the enterprise, 107–8
    support to information fusion, 118–31
    syntactic and semantic interoperability, 119–20
    unstructured information, 123–26
    value-based, 108
    workflow, 128–31
Information quality (IQ), 337–39, 344
    defined, 337
    measures, 338
    purposes of, 338–39
    standards development, 337
Information sharing, 115
Information sources
    awareness of information sources, 209–10
    selecting, 210
Information spaces
    coalition, 112–13
    defined, 110
INFORM Lab, 155–71
    agent affiliations and relationships, 164–65
    architectural approaches, 158
    architecture, 156–67
    batch run editor, 167
    components, 157
    conclusion, 171
    configuration editor, 167
    default communicator, 162–63
    defined, 155
    extension mechanisms, 166–67
    fusion and resource management, 160
    goals, 163
    implementation, 167–68
    introduction to, 155–56
    measures of effectiveness (MOEs), 344–45
    nodes, 156
    nodes behavior, 157
    nodes overview, 158
    noncooperative search vignette, 169
    parameterization, 166
    as platform independent, 167
    platforms, 161–62
    pluggable objects, 166–67
    relationship editor, 167

  services, 165–66
  situation evidence, 163–64
  tests and validation, 168–71
  trajectory editor, 167
  vignettes, 168
  vignette viewer, 168
  XML interfaces, 167
Integrated information fusion situation assessment (IFSA)
  defined, 24–25
  illustrated, 25
Integration
  management, 178
  UC^2, 178–79
Interface challenge, 2, 8–9
Interpreted Algorithmic Belief Change System (IABCS), 90
Interpreted algorithmic systems, 88
Interpreted belief change systems, 89–90
Interpreted plausibility systems, 88–89
Interpreted systems, 85–87
  actions, 85–86
  algorithmic, 88
  belief change, 89–90
  defined, 86
  knowledge operators, 87
  plausibility, 88–89
  types of, 88–90
Interpreted systems semantics, 81–100
  agent reasoning schemes, 84
  background, 85–90
  conclusions, 99–100
  defined, 83
  introduction to, 81–85
  restricting, 100
  as situation analysis blueprint, 90–96
  surveillance scenario, 96–99
Invitations, 179–80, 184–85

J

JBI Phoenix, 127, 128
JDL model, 215
  assessments, 46–47
  defined, 35
  as functional model, 41
  illustrated, 35, 263
  Levels, 37
  states in the world, 41–43
Joint Battlespace Infosphere (JBI), 127, 128, 144

Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM), 112, 119
Joint development, 138–39
Joint ownership, 139

K

Kalman filter, 57–58
Knowledge of them (KOT), 318–19
Knowledge of us (KOU), 318–19
Knowledge representation (KR), 20

L

Laplace notation, 225
Legal agreement protocol, 173–89
    automation, 176–78
    computation, 183–88
    conceptualization, 173–79
    contract formation, 179–82
    contract performance, 182
    contract remedies, 182–83
    decentralization, 174–76
    formalization, 179–83
    integration, 178–79
    sample vignette, 188–89
    ubiquity, 176
Level 0 fusion
    explanations, 46
    process, 45
    signal, 47–52
    textual, 52–55
Level 1 fusion
    explanations, 46–47
    process, 45
    signal, 56–60
    textual, 60–63
Level 2 fusion, 63–73
    assessment system challenge, 65
    conditional probability of state of affairs, 71
    explanation process outcomes, 71–73
    explanations, 47
    formal theory of uncertainty, 67
    illustrated, 65
    multisource fusion, 64
    partial states of affairs, 69
    prediction process, 66–67
    process, 45
    sets of possible worlds, 67, 68
    situations, 63

- states of affairs probabilities, 70
- table, 73

Level 3 fusion, 47–55
- coarse of action process, 76
- explanation process outcomes, 77
- explanations, 47
- illustrated, 75
- impact assessment challenges, 74
- prediction process, 74
- process, 45
- projection process, 75–76
- scenario assessment, 76
- scenarios, 73–74
- summary, 77
- table, 77
- time step agent, 76–77

Linear Time Logic (LTL), 87

Local states, 86–87

Low-level information fusion (LLIF), 331–32
- defined, 331
- HLIF versus, 332, 333–34
- MOEs, 339

M

Managed information objects (MIO)
- defined, 109
- determination of, 149
- elements of, 109
- metadata, 149–50

Maritime domain awareness example, 342–45

Markov chains, 28

Measures of effectiveness (MOEs), 316, 331–45
- background, 333
- determination of, 342
- evaluation standard for, 333
- HLIF, 339–40
- information fusion, 339–42
- information gain, 341–42, 344
- introduction to, 331–33
- LLIF, 339
- organizational effectiveness, 340–41
- variables, 343

Measures of force effectiveness (MOFEs), 332

Measures of merit (MOM), 331, 339
- categories, 340
- defined, 339

Measures of performance (MOPs), 316

Mephisto semantic framework, 181

Metrics, 321–23
  accuracy/purity, 322
  AOI score, 322–23
  confidence, 321
  cost, 322
  HLIF performance, 332
  information fusion quality, 336–39
  information gain, 341–42
  standardization, 336
  throughput, 322
  timeliness, 322
  for various disciplines, 337
MightyMoRiver project, 209
Military Strike Vignette, 239
Minimum Spanning Tree (MST), 261
Modular OODA (M-OODA), 16, 216, 222–23
  component specifications, 223
  control modeling, 222
  defined, 222
Multihypothesis tracking (MHT), 91
Multilateral Interoperability Programme (MIP), 119
Multisensor tracking, 91
Multisource fusion, 64

N

Notions of situation awareness (SAN), 81

O

Objects
  features observation, 58
  indexed, 58
  Level 1 assessment, 42
  managed information (MIO), 109
  pluggable, 166–67
Obscuration fraction, 289
Observables, 42
Observe-Orient-Decide-Act (OODA) loop, 16, 82
  cycle, modeling, 218–19
  DFIG model relationship, 218
  extended, 216
  multiplayer, 218–19
  nodes overview, 158
  phases, 217–18
  use of, 216
  See also OODA agents
Ontology Web Language-Description Logics (OWL-DL), 141
OODA agents

  affiliations and relationships, 164–65
  communicator, 162
  components, 159–61
  coordination, 160
  deciding function, 170
  defined, 156
  description of current situation, 163
  distributed feedback loop, 170
  goals, 163
  parts illustration, 159
  platform modeling of, 161
  user-defined relationships, 164–65
  in vignettes, 168
Operating conditions (OCs)
  ATC training, 291
  categories, 287
  conditioning on, 292–93
  defined, 279
  derived, 284
  direct, 283–84, 289–90, 291
  environment, 280, 281, 289–90, 293
  example, 287–92
  extended, 284–85, 295
  indirect, 283–84, 289, 291
  sensor, 280, 281, 290–91
  sensor-based classifier, 280–81
  standard, 284–85, 295
  target, 280, 281, 287–89
Operating condition scenario modeling, 279–95
  Bayes fusion, 286
  Bayes model, 285
  design, 285–86
  example operating conditions, 287–92
  experiment design for scenarios, 282–83
  illustrated, 292
  introduction to, 279
  scenario parameters, 282
  terminology, 283–85
Organization, this book, 3
Organizational effectiveness, 340–41
Organization for the Advancement of Structured Information Standards (OASIS), 122

P

Paradigm challenge, 1, 8
Persistent change, 40
Picturing
  challenges, 195–97

  defining own pictures, 200
  impact, 198
  purposes of, 200–201
  universality, 197
Platforms
  default implementation, 162
  INFORM Lab, 161–62
Pluggable objects, 166–67
Policies
  combining, 122
  federated, 122
  mediation, 122
Precision, 321
Predict-Match-Extract-Search loop, 82
Projection, 24
Proposals, 185–87
Proxy agents, 147
Pursuit-evasion (PE) games, 301

Q

Quality, 333
Quality of information (QOI), 331, 335
Quality of service (QOS), 331, 335, 336–37
Queries, in situation analysis, 99

R

Range-Doppler-azimuth cells
  direct mapping between, 51
  illustrated, 50
Reasoning methods, 334
Receiver operating characteristic (ROC) curves, 325, 326, 327
Recognition primed decision (RPD) model, 17–18
Recognized air picture (RAP), 274
Recognized ground picture (RGP), 274
Reference Model for Visualization (RM-Vis), 206–7
Relationship editor, 167
Resource management, 147–48
Robustness, 333, 344
Rosson’s and Carroll’s framework, 237–38
RPD model, 220
Rules of the game, 85

S

Scenario-based design (SBD), 233–49
  activities decomposition, 240, 242
  activities validation, 240

    activity process diagram, 245, 247
    actor-centric activities, 243, 244
    benefits of, 235
    conclusion, 248–49
    design activity, 244
    diagram, 242
    entity relationship diagrams, 245, 246
    findings on methodology, 234–39
    framework, 235–36, 248–49
    framework overview, 236
    methodology, 234
    military C2 framework, 236–39
    Military Strike Vignette, 239
    procedure, 240
    process based on Atlantis problem scenario, 240–48
    proposed framework, 236–39
    prototype, 248
    top-down waterfall, 244
    warship activities, 243
Scenario-based evaluation, 281–82
Semantic challenge, 1, 8
Semantic registration, 266–67
Sensables, 161
Sensor-based classifier operating conditions, 280–81
Sensor operating conditions, 290–91
Sensors
    assignment, 300
    creation, 162
    placement, 300–301, 303
    sensables, 161
Service layers, 110, 112
Service Oriented Architecture (SOA), 126
    agility and flexibility, 126
    environments, 127
    governance, 126–27
    middleware approach, 128
    modularity and reuse, 126
SHOR model, 220–21
Signal fusion (Level 0), 47–52
    explanation outcomes, 52
    goal, 48
    illustrated, 48
    observation process, 49
    signal detection, 50
    signal registration process, 50
    table, 53
Signal fusion (Level 1)
    explanation process outcomes, 59–60
    illustrated, 57

  object assessment output, 57
  state vector estimate data structure, 58
  state vector prediction process, 59
  state vector time steps, 56
  summary, 60
  table, 60
  uncertainty, 56–57
Simple Mail Transfer Protocol (SMTP), 52
Simulation layer (CDIFT), 143–44
Situation analysis (SAN), 81, 84
  defined, 95–96
  formalization, 84
  process formalization, 90–96
  queries, 99
  situation, 91
  situation awareness (SAW), 91–94
  situation perception and comprehension, 94–95
Situation analysis toolbox (SAT), 299–312
  Behavior Simulation Toolbox, 311, 312
  components, 300
  conclusions, 312
  countersmuggling vignette, 305–6
  defined, 300
  Discretization Toolbox, 304, 307–8
  exploration, 303–4
  illustrated, 305
  introduction to, 299–300
  software, 304–5
  State Generation Toolbox, 304–5, 308–9
  State Searching Toolbox, 305, 309–10
Situation assessment (SA), 14
  challenges and limitations, 15
  concept, 317–20
  Data Fusion Information Group (DFIG) model, 18–20
  defined, 13, 15, 316
  framework, 22
  functions and methods, 26, 27
  integrated information fusion (IFSA), 24–25
  issues/challenges, 14, 27
  metrics, 27, 29
  model based on activities of interest, 20–24
  models, 18–20
  processing methods, 28
  user models, 20
Situation assessment reference model
  defined, 23
  illustrated, 24
Situation awareness (SAW), 33–34
  components of, 34

  C-OODA, 219–20
  data fusion and, 36
  defined, 13, 15–16, 33, 94
  DFIG, 37
  Endsley model, 17
  example, 342–45
  introduction to, 14
  issues/challenges, 14, 27
  metrics, 29
  models, 17–18
  OODA model, 16
  process model, 319
  recognition primed decision (RPD) model, 17–18
  reference model, 317
  situation analysis, 91–94
  STDF, 38
  strengths and weaknesses, 33–34
  surveillance scenario, 97–98
Situation evidence (SE), 163–64
Situations
  defined, 23
  defined in terms of state transitions, 92
  generated by motion and sensing strategies, 300–304
  illustrated, 43
  Level 2 fusion, 63
  notion of, 84
  perception and comprehension, 94–95
  shift in understanding, 64
  situation analysis, 91
  in state spaces, 84–85
  surveillance scenario, 96–97
Skills-rules-knowledge model, 221–22
Social contracting protocols, 178–79
Standardization Agreement (STANAG) formats, 350
Standard operating conditions (SOCs), 284–85, 295
State Generation Toolbox, 304–5, 308–9
States
  classifications, 39–40
  global, 85
  JDL model, 41–43
  local, 86–87
  representation, 264–65
State Searching Toolbox, 305, 309–10
States of affairs
  conditional probability of, 71
  partial, 69
  probabilities, 70
State transition data fusion (STDF) model, 23, 263–68
  abstract, 44

  framework, 44
  fusion process, 43–47
  higher-level process, 267
  illustrated, 264
  for Level 2 situation assessments, 264, 265
  motivation, 36
  observation, 266–67
  prediction and explanation, 267–68
  premises, 263–64
  semantic registration, 266–67
  situation awareness and data fusion, 38
  state representation, 264–65
  state transitions, 38–43
State transitions
  classification, 38–39
  conceptualizations, 41
  situations defined in terms of, 92
  states, 39–40
  transitions, 41
Structure, this book, 2–7
Subject matter experts (SMEs), 245
Suggested Upper Merged Ontology (SUMO), 120
Surveillance scenario
  belief, revision, and update, 98–99
  illustrations on, 96–99
  scheme, 96
  situation, 96–97
  situation analysis, 99
  situation awareness (SAW), 97–98
Synthesis, this book, 351
Synthetic Aperture Radar (SAR), 280
System challenge, 2, 9

T

Tactical Reports (TACREPs), 260
Target detection, 294
Target operating conditions, 287–89
  general list of, 288
  in OC model, 289
  target class, 287
Team OODA (T-OODA), 216
Technical Cooperation Program (TTCP), 1, 2, 3, 4, 233, 252
Technology
  demonstration, 138
  sharing, 138
  showcase, 137–38
Technology, emotion, culture, and knowledge (TECK-OODA), 16
Temporal operators, 87

Text-oriented tabular entry (TOTE) displays, 200
Textual fusion (Level 0), 52–55
  explanation process outcomes, 55
  goal, 53
  illustrated, 54
  observation process, 53
  table, 55
  token observation, 54–55
Textual fusion (Level 1)
  explanation process outcomes, 63
  goal, 60–61
  illustrated, 61
  parser, 60, 62
  prediction process, 61–62
  summary, 63
  table, 63
  token parse tree, 62
  token sequences, 63
  word state vectors, 60
Threat assessment (TA), 15, 316
Throughput, 322
Timeliness, 322
Trajectory editor, 167
Transitions
  change classifications, 39
  from state classifications, 41
Trust ratings, 183

U

Ubiquitous Command and Control (UC2)
  defined, 174
  framework, 174
  inconsistency management, 176
Unified Modeling Language (UML), 159, 239
United States contributions perspectives, 6–7
Universal Core (UCore), 119
Unmanned Air Vehicle (UAV), 165
UN Security Council (UNSC), 253–54
Unstructured information
  characterization, 123
  data integration, 125–26
  extraction and annotation techniques, 124–25
  information management functions and, 124
  management and exploitation of, 123–26
  mechanisms for contextual diversity, 125
  multimedia sources, 125
  storage and dynamic structure, 126
  textual data, 124

  transformation goal, 124
  transformation methods, 125
Urban operations, 272–73
User Defined Operating Picture (UDOP), 149, 151, 193–212
  adoption of, 208
  alternative pipeline model, 205
  architecture view, 204
  awareness of information sources, 209–10
  capability, 201–3
  characteristics of, 201–3
  conclusions, 211–12
  defined, 193
  end user expertise, 211
  feasibility, 207–8
  flexibility, 194
  functions, 194
  future capability realization, 204–9
  Haiti, 208–9
  impact assessment challenges, 197
  instantiation, 208
  introduction to, 193–94
  need-to-know constraints, 211
  picture definitions, 200
  picturing capability challenges, 195–97
  picturing purposes, 200–201
  pipeline, 205, 210
  remaining issues, 209–11
  selecting information sources, 210
  technical characteristics, 202
  universality, 197
  user configurable layered services supporting, 203
  as user-modifiable configuration, 194
  users/user needs, 198–200
  visualizations, exploitation of, 205–7
  way forward, 208–9

V

Vignettes
  INFORM Lab, 168–69
  noncooperative search, 169
  sample, 188–89
Vignette viewer, 168
Virtual advisors, 270
Virtual Battlespace, 271
Visibility-based pursuit evasion, 301–2
Visualizations, 201
  composition graph, 206
  configuration of, 205–6

  - exploitation, 205–7
  - provision of guidance on, 207
  - ROC curve, 327
- Visualization Toolbox, 305, 311

W

- Web services
  - agents and humans as, 131
  - interfaces, 148
- Workflow management systems, 130
- Workflow(s), 128–31
  - agents and humans in, 131
  - components, 129–30
  - concurrent interacting, 130
  - defined, 128–29
  - subworkflows, 129–30