THE JOINT ARCHITECTURE FOR UNMANNED SYSTEMS

Domain Model (DM)

Volume I

Version 3.2
10 March 2005

CHANGE SHEET

This change sheet includes all Inspection Forms incorporated into current and previous Domain Model versions.

Inspection Forms incorporated in version 3.2:

Date Approved	Tracking Number	Short Title
18 May 2004	DM-2004-05	Adding New Section 3.3.4 Functional Capability and Information Capability Abstract Levels.
18 May 2004	DM-2004-06	Revision of Section 4.1.5.3 of Weapon Payload for Three Levels of Control.
10 Aug 2004	DM-2004-07	Removal of Non-Technical Language Phrase from Section Preface.
10 Aug 2004	DM-2004-08	Update for Clarity.
10 Aug 2004	DM-2004-09	Update for Clarity.
10 Aug 2004	DM-2004-10	Update for Clarity.
10 Aug 2004	DM-2004-11	Adding New Section 7.0 Acronym List.
20 Aug 2004	DM-2004-14	Update for Clarity.

Inspection Forms incorporated in version 3.1:

Date Approved	Tracking Number	Short Title
19 Feb 2004	DM-2004-01A	Inclusion of version control
19 Feb 2004	DM-2004-02	Correcting TOC
19 Feb 2004	DM-2004-03A	Definition of Capability and addition of Glossary section
19 Feb 2004	DM-2004-04A	Automatic Configuration Functional Capability

Inspection Forms incorporated in version 3.0:

Date Approved	Tracking Number	Short Title
01 Dec 2003	DM-2003-01	Rewrite the Domain Model

Domain:
A set of relevant information, potentially including both entities and processes, over which an analysis is defined.

This document defines the scope of the Joint Architecture for Unmanned Systems (JAUS), the rationale for the scope, and the definition of the requirements. The Domain Model is intended for use by product developers, customers/users, and researchers. Product developers work with customers/users to define the capabilities that JAUS must support. Product developers work with researchers to focus research efforts on capabilities with no known solutions.

Product Developer:
Organization that decides the material solution to meet the customer/user requirements.

0.1 Definition of Domain Model

Within JAUS, the domain model is a representation of the unmanned systems' functions and information, built for the purpose of defining the JAUS requirements.

Customer/Users:
Organization that defines system capabilities and who will operate the final product.

0.2 Document Overview

This documented is comprised of the following sections:

Section 1:	Introduction	States the problem and defines the JAUS Domain, including JAUS objectives and constraints.
Section 2:	Domain Analysis	Assesses the problem domain, evaluates existing architectures and defines a solution.
Section 3:	JAUS Scope	Defines the scope and technical approach of JAUS.
Section 4:	Capability Model	Defines current and future customer/user requirements.
Section 5:	Reference	Applicable references.
Section 6:	Glossary	Definition of terms used throughout this document.
Section 7	Acronym List	Addresses the document acronyms for a quick terminology reference.

Researcher:
Organization that seeks a technical approach to a problem with no known or adequate solution.

1.1 Background

Unmanned systems reduce exposure of personnel to harmful environments, perform tasks not possible for humans, and provide cost effective solutions to repetitive tasks. As a result, a large number of unmanned system products are being introduced to the market. Furthermore, many of these systems are characterized as task dependent and non-interoperable. A standard open architecture is required to support the rapid and cost-effective development of unmanned systems.

1.2 Premise

This document develops the scope and technical approach for unmanned systems' architecture.

1.3 Architecture Objectives

Architecture:
The structure or structures of unmanned systems, which comprise of software elements, externally visible properties of those elements, and the relationships among them.

An analysis needs to be performed to select or develop an architecture to support the following objectives:

1. Support all classes of unmanned systems
2. Rapid technology insertion
3. Interoperable operator control unit
4. Interchangeable/interoperable payloads
5. Interoperable unmanned systems

Interoperability:
The ability of two or more subsystems to exchange information and to use the information that has been exchanged.

Interoperability is the ability of two or more subsystems to exchange information and to use the information that has been exchanged [IEEE 90]. The architecture along with sound systems engineering will specify how interoperability is accomplished.

1.4 Architecture Constraints

The architecture shall be constrained by:

1. Defense Acquisition System
2. Operational procedures
3. Intellectual property and data rights
4. Systems engineering
5. Research and development
6. Product acquisition

1.5 Available Standards and Architectures

The following are some of the standards and architectures available:

1. Joint Technical Architecture (JTA)
2. 4D/Real-time Control System (4D/RCS)
3. Rotorcraft Open Systems Avionics (ROSA)
4. Air Vehicle Standard Interface (AVSI)

1.6 Architecture Benefits

The net effect is more efficient development, reduced ownership cost, and an expanded range of vendors. Therefore, JAUS is an open architecture over the domain of unmanned systems that meets the objectives within the stated constraints.

An analysis of the JAUS architecture objectives is needed to understand what JAUS must support. Furthermore, JAUS must adhere to specific architecture constraints. By analyzing available standards and architectures, a determination can be made if a current standard or architecture exists that satisfies the architecture objectives. If one does not exist, then the analysis will identify why current standards and architectures are insufficient and propose a new standard.

2.1 Analysis of Architecture Objectives

2.1.1 Support all Classes of Unmanned Systems

Classes of unmanned systems include air, ground, surface (e.g., water), and subsurface systems and are not limited in size. Their size could range from small, throwable robots to ship-size unmanned vehicles. Therefore, to support all classes of unmanned systems, JAUS should ensure platform independence.

2.1.2 Rapid Technology Insertion

Funding constraints may prohibit unmanned systems from having certain customer/user-defined capabilities or a viable technical solution may not even exist. Systems should have the ability to be fielded in increments of operational capability. The operational capability is improved over time as technology improves and the customers/users gain experience with the systems. Therefore, JAUS should not impose a specific technical approach.

2.1.3 Interoperable Operator Control Units

Operator Control Unit: A human interface device allowing a human operator to control an unmanned vehicle.

Operator control units (OCUs) range in size from small, portable handheld devices to ones needing a climate controlled room, each with different levels of functionality. The smaller handheld OCU may have several of the same functions of the larger OCU, but not all. Therefore, OCUs can have a degree of inoperability based on their similar functions. JAUS should allow for the interoperability of operator control units.

Payload:
A component or set of mission oriented components carried by an unmanned system not necessary for vehicle operation.

2.1.4 Interchangeable/Interoperable Payloads

As innovative concepts and improved technologies rapidly change payload capabilities, customers/users and developers need to have the ability to update payloads without redesigning the unmanned system thus greatly reducing total life-cycle cost. Therefore, JAUS should allow technical advancements while not imposing specific hardware or software implementations.

2.1.5 Interoperable Unmanned Systems

Unmanned systems are mission enhancers. However, certain mission goals may not be achievable by a single unmanned vehicle, or more useful results are obtained using multiple unmanned systems. Therefore, JAUS should allow for communication between unmanned systems independent of platform type.

2.2 Analysis of Constraints

2.2.1 Defense Acquisition System

The Defense Acquisition System directs programs to use a modular open systems approach to ensure access to the latest technologies and products, and facilitate affordable and supportable modernization of fielded assets. An open systems approach is an integrated business and technical strategy that employs a modular design and, where appropriate, defines key interfaces using widely supported, consensus-based standards that are published and maintained by a recognized industry standards organization.

In order to maintain a technological edge, focus must be diverted to systems that can evolve more readily as technological advances keep pushing ahead. An open systems approach leads the way for transformation to the new paradigm by offering a more cost-effective way of keeping a technological edge. Therefore, JAUS should focus on open standards and architectures.

2.2.2 Operational Procedures

Although the marketplace has been flooded with unmanned systems, "how" they can be used has not been fully developed. Therefore, JAUS should facilitate diverse operational procedures.

2.2.3 Intellectual Property and Data Rights

JAUS should not infringe upon intellectual property and data rights.

2.2.4 Systems Engineering

The elements of systems engineering are to ensure the compatibility, interoperability, and integration of all functional and physical interfaces, and to ensure that the system definition and design reflect the requirements for all system elements. Systems engineering allocates functions to and defines information transfer between subsystems. Therefore, JAUS should not allocate functions to any particular element of the system and should not define or measure system capability performance.

2.2.5 Research and Development

Research and development is the process by which a technical solution is sought for a problem with no current resolution or to which an answer can be improved. Systems are often designed around a particular technology, and research and development breakthroughs can take a significant amount of time to transition into new or improved products. Therefore, JAUS should be independent of technology.

2.2.6 Product Acquisition

Product Acquisition is the process used to acquire quality products. The primary objectives of this process are to:

- Satisfy the needs of customer/users
- Acquire quality products in a timely manner at a fair and reasonable cost
- Focus on specific capabilities; “what” is needed, not “how” to do it.

Unmanned systems have generic functions applicable to all classes. These functions should be separated from mission specific functions to allow the greatest flexibility in vehicle usage. The JAUS architecture should be flexible enough to support a wide range of possible missions and be independent of operator use.

2.3 Analysis of Available Standards and Architectures

2.3.1 Joint Technical Architecture (JTA)

The Joint Technical Architecture is Department of Defense information technology standard that establishes a minimum set of rules governing information technology across all DoD systems. These standards include Information Processing Standards, Information Transfer Standards, Information Modeling, Metadata, and Information Exchange Standards, Human-computer Interface Standards, and Information Security Standards. Information Processing Standard describes Government and commercial information processing standards DoD uses to develop integrated, interoperable systems that meet the customer/users' information processing requirements. Information Transfer Standards describe the information transfer standards and profiles that are essential for information interoperability and seamless communications. It mandates the use of open systems standards used for Internet and the Defense Information System Network (DISN). Information Modeling, Metadata, and Information Exchange Standards describe the use of integrated information modeling and mandates applicable standards. This section mandates information standards, including message formats. Human-Computer Interface Standards provides a common framework for Human-Computer Interface design and implementation in DoD systems. The objective is the standardization of customer/user interface implementation options, enabling DoD applications to appear and behave in a reasonably consistent manner. Information Security Standards prescribe the standards and protocols to be used to satisfy security requirements. It provides the mandated and emerging security standards that apply to the JTA.

These standards are based on published Society of Automotive Engineers (SAE), Institute of Electrical and Electronics Engineers (IEEE), Government, and commercial interface, protocol, and architecture standards utilized to support interoperability across all DoD systems. To further support the interoperability of DoD systems, the JTA divides the systems into four defined domains and their associated subdomains. The domains are Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR), Combat Support, Modeling and Simulation, and Weapon Systems. These domains represent a group of systems that share common functional, behavioral, and operational requirements. The subdomains represent a subset of related systems, exploiting additional commonalities and addressing variances within the domain (4). Unmanned systems have been identified within several of these domains. The JTA addresses specific hardware, software, and informational standards that are applicable to unmanned system development, but does not address the format and content of that information.

2.3.2 4D/Real-time Control System (4D/RCS)

4D/RCS is a methodology for conceptualizing, designing, engineering, integrating, and testing intelligent systems software for vehicle systems with any degree of autonomy, from manually operated to fully autonomous. 4D/RCS demonstrates how intelligent unmanned vehicle systems can be integrated into any military command and control structure. It integrates the functional elements, knowledge representations, and flow of information so that intelligent systems can analyze the past, perceive the present, and plan for the future. It enables systems to assess the cost, risk, and benefit of past events and future plans, and make intelligent choices among alternative courses of action.

4D/RCS addresses engineering issues that must be considered when building intelligent unmanned systems that approach human levels of performance in responsiveness, dexterity, adaptability, reliability and cognitive understanding. It provides an engineering methodology by which military systems meet customer/user operational requirements (1). 4D/RCS plays an important role in the development of unmanned systems by supporting all unmanned system classes and rapid technology insertion. However, 4D/RCS only defines interfaces to the conceptual and semantic level but not to the syntactic, message, and transport levels.

2.3.3 Rotorcraft Open Systems Avionics (ROSA)

ROSA project uses commercial-off-the-shelf (COTS) electronics components and open systems specifications and standards developed for non-aviation applications. It will determine the degree to which those specifications and standards can be used to meet combat helicopter mission avionics for future rotorcraft. The primary objective is to reduce the cost of avionics for future rotorcraft (2). Therefore, it does not address all classes of unmanned systems and the interoperability of payloads.

2.3.4 Air Vehicle Standard Interface (AVSI)

AVSI is an effort to design modular payloads for unmanned air vehicles. Current problems in unmanned air vehicles include limited payload growth potential, restricted configuration, and reduced interoperability and compatibility. AVSI is an attempt to offer true “plug-n-play” capability that will maximize flexibility and reduce total ownership cost (5).

AVSI is working toward standardizing only unmanned aerial vehicle payloads. It is feasible for a payload to be interchangeable on any unmanned vehicle platform. Furthermore, it does not support all classes of unmanned systems and interoperable unmanned systems. Therefore, AVSI does not meet the JAUS architecture objectives to support all classes of unmanned systems.

2.4 Conclusion

JAUS:
A common language consisting of well-defined messages, enabling internal and external communication between unmanned systems.

Although current standards, guidelines, and architectures exist in developing unmanned systems and will continue to be incorporated in unmanned system development, they are still inadequate to facilitate the development of the entire domain of unmanned systems. The above mentioned architectures incorporate various SAE, IEEE, Government and commercial standards into standards that apply to some, but not all, issues related to the JAUS architecture. These standards and constraints describe elements associated with unmanned systems, such as: hardware and software requirements, interface protocols, frequencies, buses, etc., but do not fully satisfy all JAUS architecture objectives.

This analysis concludes that a common language consisting of well-defined messages, enabling internal and external communication between unmanned systems, as well as those standards and architectures mentioned above, are needed to fully satisfy the architecture objectives.

Furthermore, in order to be continually relevant, flexible and useful to the entire unmanned systems domain, JAUS is composed of five requirements, as shown graphically in Figure 2-1. They are the **philosophical underpinnings** of JAUS, and it is important that they be well understood. All JAUS messages are required to be:

1. Vehicle Platform Independent
2. Mission Independent
3. Computer Resource Independent
4. Technology Independent
5. Operator Use Independent

2.4.1 Vehicle Platform Independent

The domain of unmanned systems will include a wide spectrum of platform types. In order for JAUS compliant components to be interoperable, no assumptions about the underlying vehicle are made.

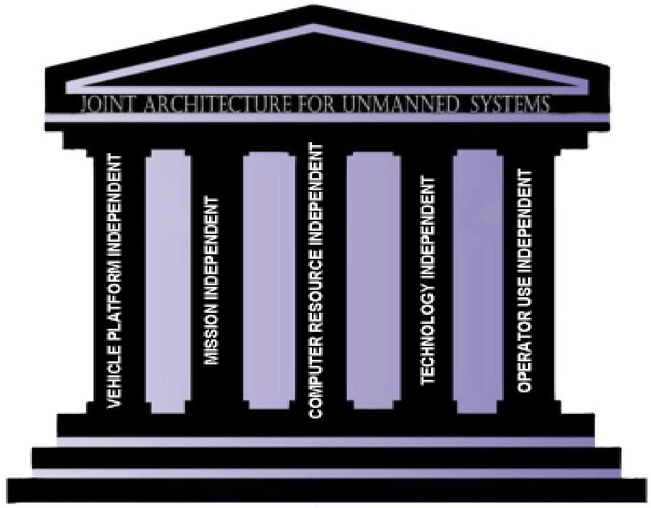

Figure 2.1 JAUS Independent Requirements

2.4.2 Mission Independent

The number and types of uses and missions for unmanned systems has the potential to be large, so JAUS must isolate mission specific functions from the generic functions. This allows the architecture to be independent of any specific mission or set of missions.

2.4.3 Computer Resource Independent

The growth in the computer industry has been enormous over the past several years and Moore's Law still holds true. Future unmanned systems must be able to capitalize on commercial advancements in computing and sensor technology. The issue for unmanned systems is two-fold. First, a single unmanned system must be able to evolve over its life cycle to accommodate new missions and greater degrees of autonomy. An architecture that imposes a specific hardware implementation reduces the opportunity to take advantage of future advancements. Second, each unmanned system prime contractor should have the flexibility to decide what computer hardware meets that particular system's requirements. Computer hardware that is appropriate for one unmanned system may not be appropriate for another. JAUS must maintain computer hardware independence in order to be applicable to all unmanned systems.

2.4.4 Technology Independent

JAUS does not specify any particular technology because there may be multiple solutions to a problem. An architecture that requires or is built around a particular technology solution may eliminate a superior solution. JAUS specifies the "what" not the "how." This requirement is essential for supporting evolutionary acquisition.

2.4.5 Operator Use Independent

The last requirement is that JAUS remain independent from any uses the operator will have for the unmanned system. This may "lock in" undesirable ways to use the system and "lock out" more beneficial ways. JAUS will not restrain the customer/user in determining the best approach to accomplish a mission.

3.1 JAUS Definition

Joint Architecture for Unmanned Systems (JAUS) is a common language enabling internal and external communication between unmanned systems. It incorporates a component based, message-passing architecture specifying data formats that promotes the stability of capabilities by projecting anticipated requirements as well as those currently needed. Furthermore, JAUS is open, scalable, and responsive to the unmanned systems communities' needs.

3.2 JAUS Documents

Joint Architecture for Unmanned Systems is comprised of six documents. They are as follows:

1. Domain Model (DM)—defines Unmanned Systems capabilities as requirements for the set of messages.

2. Reference Architecture (RA)—specifies the messages, formats, and data structures.

3. Standard Operating Procedures (SOP)—sets the procedures for the way the JAUS Working Group conducts its business.

4. Document Control Plan (DCP)—establishes the procedures for controlling the configuration of the JAUS baseline.

5. Strategic Plan (SP)—lays down the vision and mission statement; plans the education, expansion and acceptance of the architecture.

6. Compliance Specification (CS)—establishes policies, procedures, test, and reports for determining compliance to the JAUS standard.

3.3 Modeling Approach

The JAUS Domain Model is a tool with which customer/users can define both near and far term operational requirements for unmanned systems based on mission needs. By defining far-term capabilities, which may have never been demonstrated, the JAUS Domain Model then becomes a "road map" used by developers to focus research and development efforts to support future requirements.

By modeling the customer/user, the JAUS Domain Model represents three distinct elements: what the customer/user does, what the customer/user knows, and the devices the customer/user interfaces with. These elements are defined as the following: Functional Capabilities (FC), Informational Capabilities (IC), and Device Groups (DG), respectively. The Functional Capabilities and the Informational Capabilities may interface with the Device Group, but the JAUS Domain Model will not define these interfaces. These interfaces are open standards and may be defined by either the Joint Technical Architecture or by the Original Equipment Manufacturer's (OEM) specifications.

3.3.1 Functional Capability (FC)

Functional Capabilities are a set of capabilities that have similar functional purposes. The Domain Model divides Functional Capabilities into eleven categories that define what the unmanned system does. However, these Functional Capabilities are not limited to the below eleven categories. They are:

1. Command Capabilities make plans, assimilate data, make decisions, assign tasks, and handle unexpected events.

2. Maneuver Capabilities interface with all external maneuvering control devices and may include a set of sensors. The Maneuver Capability has a layered set of intelligent components that accept different levels of maneuvering instructions.

3. Navigational Capabilities are responsible for planning the path for an unmanned system between two or more geographical locations.

4. Communication Capabilities are responsible to manage communications between distinct unmanned systems and operator control units.

5. Payload Capabilities maintain control and provide status of payloads (e.g., reconnaissance/surveillance, target acquisition, weapons, Nuclear Biological Chemical (NBC) detector, etc.)

6. Safety Capabilities are responsible for the unmanned systems' safety.

7. Security Capabilities prevent unauthorized access and use of the unmanned system.

8. Resource Management Capabilities manage the vital resources of the unmanned system (e.g., power, fuel, bandwidth, signature, etc.)

9. Maintenance Capabilities are responsible for system maintenance, such as Built-in-Test Equipment, and have the capability to interface with external test equipment.

10. Training Capabilities have the responsibility of providing training support to the operator either internally (e.g., mission rehearsal or tutorials) or with external devices.

11. Automatic Configuration capabilities minimize operator configuration change.

3.3.2 Informational Capability (IC)

Informational Capabilities are groupings of similar types of information. They are global repositories of system data and are named according to their data type and define what the unmanned system knows or its informational capabilities. The ICs provide controlled access to the information for the entire unmanned system.

1. Vehicle Status Capabilities are responsible for providing the status of the vehicle to other Capabilities. Vehicle parameters, position, visual and audible data, and calendar.

2. World Model Capabilities maintain a repository of objects of interest including friendly and threat units, obstacles, staging areas, maps, routes, waypoints, and actions.

3. Library Capabilities maintain reference material, procedures, and performance data.

4. Log Capabilities have the responsibility to maintain a log of operational, maintenance, and training data.

5. Time/Date Capabilities provide time and date services to the unmanned system.

3.3.3 Device Groups (DG)

Device Groups are groupings of sensors and/or effectors that are used for similar functions. Specific devices are not defined with the Device Group.

3.3.4 FC and IC Abstraction Levels

The Domain Model will define different abstraction levels within functional and informational capabilities. Abstraction levels support evolutionary development and technology insertion. These abstraction levels should be independent of each other.

3.4 Compliance

Compliance is the process of validation and verification of unmanned system message traffic that adheres to predetermined message definitions, formats and rules. Currently, there are three levels of compliance. The first level is compliance between two unmanned vehicles or an unmanned vehicle and OCU. The second level is compliance between payloads, on-board controls, and sensing elements. The third level applies to software source code reuse. Furthermore, compliance at one level does not infer compliance at another, and the developer may, based on the developing system and program needs, extend compliance beyond that required of the customer/user.

This JAUS Domain Model is intended for use by product developers, customers/users, and researchers. Product developers work with customers/users to define the required unmanned system capabilities that JAUS must support. Product developers work with researchers to focus research efforts on future unmanned system capabilities with no known solutions.

4.1 Functional Capability

This section documents the collection and summary of user requirements. These requirements are presented in a context easily understood by both the user and the developer. The modeling approach presented in Para 3.3 is used.

4.1.1 Command Functional Capability

Mission planning:
Mission goals, pertinent supporting information, limiting factors, and vehicle parameters gathered from components or subsystems needed by unmanned systems to carry out the mission.

The Command FC describes the mission planning and decision-making roles for the unmanned system. It uses information gathered from onboard components or subsystems to make mission critical decisions. The Command FC has three categories: vehicle commander, team commander, and team-of-teams commander, each of these categories has varying levels of intelligence.

4.1.1.1 Vehicle Commander

Vehicle commander plans and makes decisions for a single unmanned vehicle. The vehicle commander issues directives to and receives reports from other components within the unmanned system. Additionally, the vehicle commander receives directives and sends reports to a team commander.

4.1.1.2 Team Commander

Team commander plans and makes decisions for a team or group of unmanned vehicles. It receives directives from the team-of-team commander and issues those directives to the vehicle commanders. Furthermore, the team commander issues reports to the team-of-teams commander.

4.1.1.3 Team-of-teams Commander

Team-of-teams commander plans and makes decisions for a team-of-teams of unmanned vehicles. It receives reports and issues directives to the team commanders.

4.1.2 Maneuver Functional Capability

Maneuver FC has the primary role of moving an unmanned system from one geographical location to the next. It has the ability to interface with all external vehicle actuation devices, which may include a set of sensors. This Functional Capability has three principal capabilities. The principal capabilities are Primitive Maneuver, Vector Maneuver, and Waypoint Maneuver, as shown graphically in Figure 4-1.

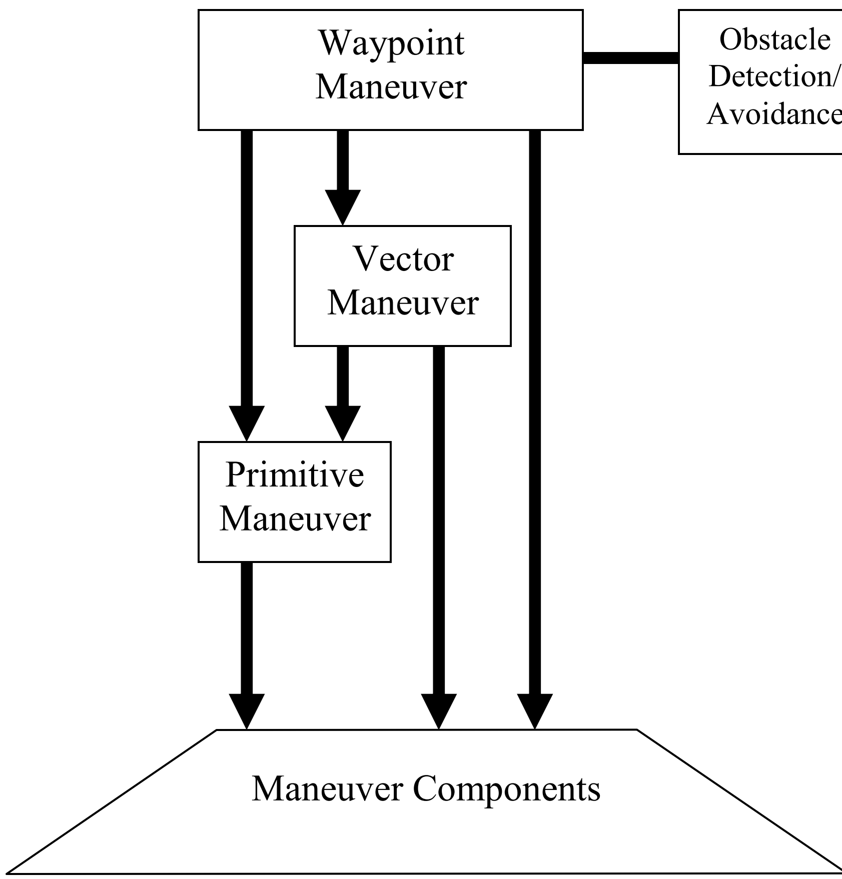

Figure 4-1 Maneuver Functional Capability Model

Primitive Maneuver capability controls the basic kinematics movement of unmanned system through a human interface device. The customer/user controls all aspects of the vehicle actuation devices such as steering, speed, braking, etc., as if the vehicle was manned. Vector Maneuver capability uses speed and direction to maneuver to the desired

geographic location, where as, Waypoint Maneuver capability uses a set of designated geographic points to maneuver to another location within given time constraints. Unmanned systems can be designed to incorporate one or any combination of the above principal capabilities. Vector and waypoint maneuvering can be translated into a primitive maneuvering. This means that customer/users can upgrade or downgrade to any mode of maneuverability regardless of the initial design.

Certain capabilities have been identified to enhance vehicle maneuverability. These capabilities include, but are not limited to object detection and avoidance, feature-following, and leader-follower. In obstacle detection and avoidance, unmanned systems accept input from a set of sensors to identify obstacles and override higher-level controls to prevent collision. The feature-following capability identifies terrain features to successfully maneuver from point to point. In leader-follower capability, unmanned systems follow the lead vehicle with a given set of parameters (e.g. time, speed, distance, etc). These abilities are not required, however, they provide a higher degree of capability to an unmanned system.

4.1.3 Navigation Functional Capability

The Navigation FC plans the path for an unmanned system between two or more geographical locations. Intelligent navigation refers to an unmanned system's ability to react with its environment while traversing from one point to another. In primitive navigation, an unmanned system moves from point to point without reacting to its environment. In semi-autonomous navigation, an unmanned system moves from point to point, reacts to its environment, but waits for corrective input from the customer/user. In autonomous navigation, an unmanned system travels from point to point making necessary corrections and/or changes to its original path without customer/user interaction.

4.1.4 Communication Functional Capability

The Communication FC manages the telecommunications of the unmanned system. Telecommunications is defined as communication from a distance; therefore, the Communication FC handles communication from subsystem to subsystem (e.g., operator control unit to unmanned system or unmanned system to unmanned system.) Furthermore, the Communication FC can handle all off board communication requirements, initiate interaction with other subsystems, and manage physical and logistical properties of the link.

4.1.5 Payload/Mission Module Functional Capability

The primary role for Payload/Mission Module FC is to handle and control all mission payloads/modules. Payloads can have different intelligence levels depending on customer/user requirements. Below are identified payloads used by unmanned systems.

4.1.5.1 Articulation

Articulation capability is the movement of mission models or tools in reference to the unmanned system. There are currently three articulation movements: primitive, repeating-path, and position and attitude. Primitive articulation is the positioning of the payload via human interface device (e.g., joystick) by a customer/user. Repeating-path articulation is movement over the same path, continuously. Spot-welding in auto manufacturing plants is an example of repeating-path articulation. Position and attitude articulation is orienting the payload, not the unmanned vehicle, to complete the mission. Positioning tools in order to reach difficult places is an example of position and attitude articulation.

4.1.5.2 Reconnaissance/Surveillance/Target Acquisition & Identification (RSTA-I)

RSTA-I capability detects and identifies potential threats or targets using various sensors and communication devices. RSTA-I can have varying levels of intelligence. In primitive RSTA-I, the customer/user sees the enemy target either by sight or by sensor feedback and must properly identify the target. In semi-autonomous RSTA-I, the unmanned system acquires and identifies the target and alerts the customer/user for action. In autonomous RSTA-I, the unmanned system acquires and identifies the target and reacts according to predefined mission parameters.

4.1.5.3 Weapon Payload

A weapon payload's primary mission is to engage targets.

4.1.5.3.1 Weapon Payload Primitive Capability

The operator controls the safety conditions of the weapon, ammunition capability, fire control, and fire rate. Additionally, the weapon payload should indicate primitive weapon's status.

4.1.5.3.2 Area Engagement Capability

The weapon payload can engage a defined zone.

4.1.5.3.3 Target Engagement Capability

The weapon payload can engage an identified target.

4.1.5.4 Environmental Sensing

Environmental Sensing is a payload that allows an unmanned system to acquire environmental information. Environmental information includes: weather, presence of nuclear, biological, and chemical Capabilities, and/or mapping of terrain features. This information can be sent to another subsystem or stored for future use by the unmanned system.

4.1.5.5 Manipulation

This payload allows an unmanned system to grasp, push, and/or pull objects. It encompasses combinations of articulation commands based on mission requirements and situational awareness (e.g., primitive, repetitive-path, and/or position and attitude). Whether it's opening a door or diffusing a bomb, the manipulation payload gives the unmanned system a degree of dexterity in handling objects.

4.1.6 Safety Functional Capability

The Safety FC ensures the overall safety of the unmanned vehicle and has primitive and/or intelligent capabilities. Primitive safety capability keeps the unmanned system safe from harmful situation through the interaction of the customer/user. An emergency stop button is an example of a primitive safety capability. With intelligent capability, the unmanned system based on its environmental sensing and navigational capabilities prevent the unmanned system from running into unsafe situations. Furthermore, it can override any commands from other subsystems that might cause an unsafe situation.

4.1.7 Security Functional Capability

The Security FC prevents unauthorized use of the unmanned system as well as the unmanned systems Informational Capabilities.

4.1.8 Resource Management Functional Capability

The Resource Management FC manages the vital resources of the unmanned system. Vital resources include: power, fuel, bandwidth, and heat signature.

4.1.9 Maintenance Functional Capability

The Maintenance FC is responsible for proper operation of the unmanned system and supports maintenance and repair. It functions as a Built-in Test (BIT) for onboard diagnostics and can interface with external test equipment for more advanced maintenance troubleshooting.

4.1.10 Training Functional Capability

The Training FC ensures that customer/users are properly trained in using the unmanned system. Possible training capabilities include: mission rehearsals, graded training tutorials, Hardware-in-the-Loop simulations, and other training devices.

4.1.11 Automatic Configuration Capability

The Automatic Configuration FC provides a mechanism that allows OCUs, payloads, and unmanned systems to be registered within the system and/or system of systems with limited operator input.

4.2 Informational Capability (IC)

Just as Functional Capabilities attempt to map what a customer/user does, Informational Capability maps what the customer/user knows. Informational Capability serves as global repositories of system data that are utilized by the unmanned system to accomplish its mission or task. Informational Capabilities are categorized as: Vehicle Status, World Model, Logs, Library, and Date/Time.

4.2.1 Vehicle Status Informational Capability

The Vehicle Status IC stores, records, and provides upon request information related to vehicle platform. Information related to vehicle platform includes:

1. Platform position and attitude
2. Engine Data

4.2.2 World Model Informational Capability

The World Model IC stores, records, and provides upon request information related to the environment influencing the unmanned system whilst executing its mission or task. This information includes:

1. Topographic layouts
2. Building layouts and blueprints
3. Terrain feature information
4. Objects of interest (last known and predicted location, last know and predicted path)

4.2.3 Log Informational Capability

Log IC records desired information during unmanned system operation. The Log IC is not limited to recording information from onboard sensors or payloads, but could also record information from other unmanned systems. Information that can be logged includes:

1. System failures (vehicle and payload)
2. Operator commands
3. Events during mission execution
4. Collaborative information

4.2.4 Library Informational Capability

The Library IC is a repository of operational information. Information stored in the Library IS can be mission specific or platform specific.

Mission Specific:
1. Procedures and tactics
2. Rules of engagement
3. Mission planning (assembly areas, mobility areas, avoidance areas, potential target locations, etc.)

Platform Specific:

1. Maintenance manuals
2. Online help
3. Training tutorials
4. Vehicle capability lists

4.2.5 Time/Date Informational Capability

Time/Date IC provides mission oriented time and date. It is used in conjunction with event logs and mission logs that require a time and date. Furthermore, it can be used to calculate elapsed time during an event or mission.

REFERENCES Section 5.0

Albus, James, et al. 4D/RCS: A Reference Model Architecture for Unmanned Vehicle Systems. Version 2.0, August 2002. <http://www.isd.mel.nist.gov/documents/albus/4DRCS_ver2.pdf>

Aviation Applied Technology Directorate, Systems Integration Division, Project Fact Sheet. Rotorcraft Open Systems Avionics. Ft. Eustis, VA. <http://www.aatd.eustis.army.mil/Business/FactSheets/ROSA/ROSA_FactSheet.asp>

[IEEE 90] Institute of Electrical and Electronics Engineers. *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries.* New York, NY: 1990

United States. Department of Defense. Joint Technical Architecture. Version 5.0, April 2003. <http://www-jta.itsi.disa.mil/jta/JTA50_20030404.pdf>

United States. Department of Defense. Modular Mission Payloads Architecture and Concept and Demonstration. NAVAIR. January 2002. http://uav.navair.navy.mil/payload/PnP_AUVSIWinter2002-Rev3b.pdf

Architecture	The structure or structures of unmanned systems, which comprise of software elements, externally visible properties of those elements, and the relationships among them.
Capability	Customer/user defined physical and functional requirements for unmanned systems needed to accomplish a desired mission or task.
Customer/Users	Organization that defines system capabilities and who will operate the final product.
Domain	A set of relevant information, potentially including both entities and processes, over which an analysis is defined.
Interoperability	The ability of two or more subsystems to exchange information and to use the information that has been exchanged.
JAUS	A common language consisting of well-defined messages, enabling internal and external communication between unmanned systems.
Mission Planning	Mission goals, pertinent supporting information, limiting factors, and vehicle parameters gathered from components or subsystems needed by unmanned systems to carry out the mission.
Operator Control Unit	A human interface device allowing a human operator to control an unmanned vehicle.
Payload	A component or set of mission oriented components carried by an unmanned system not necessary for vehicle operation.
Product Developer	Organization that decides the material solution to meet the customer/user requirements.
Researcher	Organization that seeks a technical approach to a problem with no known or adequate solution.

ACRONYM LIST Section 7.0

4D/RCS	4D/Real-time Control System
AVSI	Air Vehicle Standard Interface
BIT	Built-in Test
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance
CS	Compliance Specification
DCP	Document Control Plan
DG	Device Group
DISN	Defense Information System Network
DM	Domain Model
DoD	Department of Defense
FC	Functional Capability
IC	Informational Capability
IEEE	Institute of Electrical and Electronics Engineers
JAUS	Joint Architecture for Unmanned Systems
JTA	Joint Technical Architecture
NBC	Nuclear, Biological, Chemical
OCU	Operator Control Unit
OEM	Original Equipment Manufacturer
RA	Reference Architecture
ROSA	Rotorcraft Open Systems Avionics
RSTA-I	Reconnaissance/Surveillance/Target Acquisition & Identification
SAE	Society of Automotive Engineers
SOP	Standard Operating Procedures
SP	Strategic Plan