

INTELLIGENT DECISION SUPPORT SYSTEM SOFTWARE FOR MODELLING UPGRADE DESIGN OF MARITIME HELICOPTERS FOR ANTI-SUBMARINE MISSIONS – STATUS REPORT II

Srinivas Jonnalagadda Arvind K. Sinha Peter Hoffmann
The Sir Lawrence Wackett Centre for Aerospace Design Technology
School of Aerospace, Mechanical and Manufacturing Engineering
RMIT University, GPO Box 2476V, Melbourne, Victoria, 3001, Australia.
(Tel: +61-3-9645 4536 Fax: +61-3-9645 4534)
(Correspondence: arvind.sinha@rmit.edu.au)

Daniel P. Schrage
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150, USA
(Tel: +1-404-894-6227 Fax: +1-404-894-2760)
(E-mail: daniel.schrage@ae.gatech.edu)

Abstract

Future global crisis and conflicts in littorals will require aerial platforms, including helicopters, with enhanced anti-submarine capabilities in order to counter the increased submarine threat. The design and manufacture of new aircraft with enhanced capabilities is highly expensive and time consuming. A mid-life upgrade of existing platforms with new on-board systems is generally the preferred option for enhancing mission capability. The pre-upgrade design analysis process is intricate and iterative. An intelligent, knowledge-based software tool that ensures speed and accuracy needs to be developed. This paper describes “Intelligent Decision Support System” a Software tool in development, to simulate the pre-upgrade design analysis process.

1. Introduction

In future global crisis and conflicts the prevalent submarine threat present in littorals will require aerial platforms, including helicopters, to possess enhanced anti-submarine capabilities [4]. The design and manufacture of new aircraft capable of defeating these future threats is highly expensive and time consuming. A mid-life upgrade of existing platforms with new on-board systems is generally the preferred option for enhancing mission capability and overcoming new threats [5]. The trade-off between established in-service technologies and advanced technologies is complex and requires detailed synthesis for the formalisation of a coherent whole [6].

To achieve optimum integration of mission systems in an upgrade design, it is essential to capture

functional and technical knowledge and to automate the design and manufacturing processes [1].

Sinha et al. [7, 8, 9, and 10] adopted a systems approach to address the mid-life upgrade process for mission capability enhancement. This was conceptualised as an ‘input-process-output’ configuration which considered the disparate design parameters - mission capability, flight performance, reliability, maintainability and cost.

Jonnalagadda et al. [2] presented a multi-agent based automation framework to automate the systems methodology of Sinha et al [7, 8, 9, and 10]. The automation framework description included the description of the baseline modules. Each module was self-contained using distributed system paradigm providing input to the knowledge base for design decision support.

In this paper, an overview of the automation framework is presented followed by the architecture and development tools of the “Integrated Decision Support System” software which is being developed. Description includes the usage scenario, user interface, and application architecture.

2. Mid-Life Upgrade Systems Methodology

The generic “mid-life upgrade system” developed by Sinha et al. [7, 8, 9, and 10] is presented in Figure.1.

The mid-life upgrade system identifies the integration of missions through decision support system to formulate a mission equipment package for upgrade. The airframe to be upgraded is then analysed on the basis of several design parameters: a) mission capability; b) flight

performance; c) maintainability; d) reliability; and e) cost. A multi parameter evaluation graph is then proposed for identification of an optimal upgrade design.

Automation framework of the aforementioned mid-life upgrade system identified the following modules: a) mission system identification; b) mission payload design; c) multi-parameter analysis; and d) decision support and e) robustness. The input command and output was through a man-machine interface. A database provided inputs for all computations.

The mid-life upgrade system and its proposed automation are revisited from a system perspective to develop an upgrade system structure to enhance anti-submarine capabilities. The system is viewed in an 'input-process-output' configuration. Three components of the system were identified to address the following functional characteristics (attributes): a) Man-Machine Interface – User inputs and machine codes; b) Mission System- Attack, counterattack, reconnaissance and surveillance; and c) In-Service Helicopter- Mission capability, flight performance and logistic support. The environment of the operation is to address the threat, time of operation, weather and the sea state (Table 1).

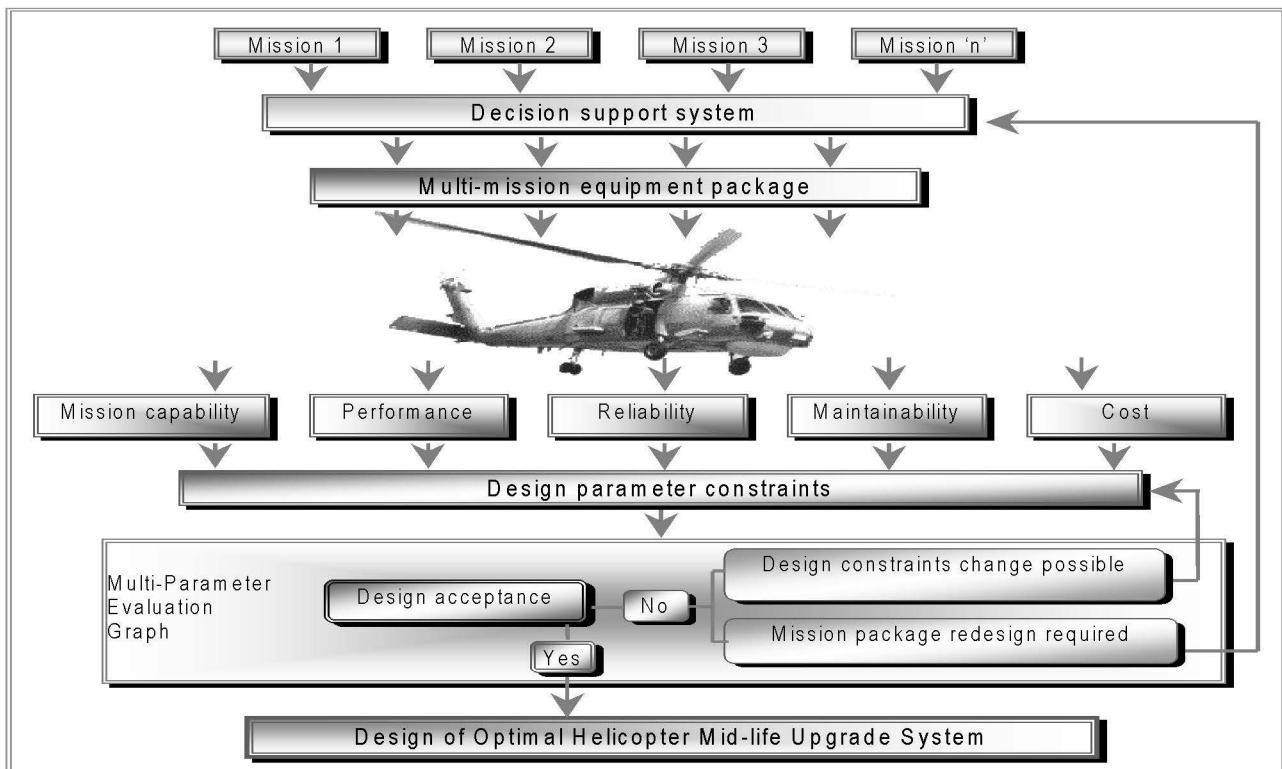


Fig 1 Mid-Life Upgrade System [Ref 12]

3. Automation of Mid-Life Upgrade Systems Methodology

The automation model of Jonnalagadda et al. [2] is presented in Figure 3. The mid-life upgrade of maritime helicopters for anti-submarine missions comprised of the five baseline modules, which can be either integrated or distributed to form an intelligent decision support system presented in Figure 4:

1. In-service Helicopter module (IHM): Provides design details of the selected in-service maritime helicopter to be upgraded.

2. Anti-submarine mission requirements module (ARM): Determines mission requirements based on operational needs and the operational environment.
3. Knowledge base module (KBM): Computing methodology and integration of the state-of-the-art mission systems by implementing reuse of relevant information to produce optimum upgrade design option.
4. Mission systems technology module (MST): Provides functional characteristics of system technologies offering improved anti-submarine mission capability.

5. Man-Machine Interface module (MMI): The man-machine Interface provides user system interaction.

4. Knowledge base module (KBM)

The Knowledge Base Module identifies the state-of-the-art mission systems from the defined operational and environmental needs provided by the anti-submarine mission requirements module. It is also tasked for the mission systems integration into an in-service helicopter and the evaluation of alternative upgraded helicopter designs. The sub-modules of KBM are as follows:

1. Knowledge management sub-module: Creates, collects and processes knowledge required for achieving different goals.
2. Database management sub-module: Stores and manages operational, environment and mission systems data received from the onboard database.
3. Design baseline sub-module: Maintains a baseline designs once the optimum upgrade design option is generated. The initial design constraints identified will form the first baseline. The further baselines are dependent upon choice of mission systems integrated by the user.
4. Mission systems Identifier sub-module: Compares the functional characteristic of the mission system from MST, the translated anti-submarine mission requirements from the ARM, IHM and identifies ideal mission systems for the mid-life upgrade. These are stored in the database. The output will be the ideal mission systems for mission payload design.
5. Mission payload design sub-module: Integrate ideal mission systems and design alternative mission payloads according to the priority based on their relative functional dependence and degree of contribution in mission success.
6. Design parametric analysis sub-module: Performs unit level sub-system test with the alternative mission payloads for acceptance of disparate design parameters- flight performance, mission capability, maintainability, reliability and cost.
7. Verification and validation sub-module: Performs both unit level and system verification and validation after integration for the system effectiveness of the upgraded design option.
8. Design robustness analysis sub-module: Tests the robustness of the design decision against temporal uncertainties.

9. Coordinator sub-module: Refines, interprets and present to the agents and other sub-modules the relevant information to perform the required task.

5. Intelligent Decision Support System

The Intelligent Decision Support System (IDSS) software to aid the automation of systems methodology for modelling upgrades of maritime helicopters; includes operational, technological analyses and logistic support analysis. To generate an optimum upgrade design the IDSS characteristics need to be the following:

- a) A distributed set of components connected by a network;
- b) Fusion of data from various operational and environment sources or systems;
- c) Inherit the capabilities to conduct complex computations and handle large volume of data;
- d) The software should have the knowledge of previous missions and configurations; that provides different scenarios to the user including visualisations;
- e) Software be modular, support reuse and scalable;
- f) Easy to access and user-friendly navigation; and
- g) Adhere to various software engineering standards.

The pre-upgrade design analysis process is intricate and iterative. To reduce this complexity a knowledge-based automation framework has been developed Figure 4. To develop the "Intelligent Decision Support System" software incorporating multiple-agents more than one tool is needed. There is a wide range of readily available Common Development Tools (CDT) ranging from Java development kit to CLIPS, to even simplistic tool such as Microsoft Excel [3]. Rapid Application Development (RAD) tools such as the Microsoft.NET Framework [11] a new computing platform that simplifies application development in the highly distributed environment of the Internet. It designed to fulfil the following objectives:

- a) To provide a consistent object-oriented programming environment where object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely;
- b) To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party; and
- c) To build all communication on industry standards to ensure that code based can integrate with any other code.

Inputs		Anti-Submarine Mission Requirements (Attributes)	Output	
Operational Needs	Attack & Counterattack	Armament	Attack & Counterattack Sub-Mission Capability	Anti-Submarine Mission Capability
		Observation		
		Navigation		
		Communication		
		Self Attack		
		Knowledgebase		
		Crew Activity		
		Fire Power		
	Reconnaissance & Surveillance	Situation Awareness	Reconnaissance & Surveillance Sub-Mission Capability	
		Communication		
		Navigation		
		Crew Activity		
		Armament		
		Survivability		
Environment Needs	Threat	Armament	All Threat	
		Observation		
		Survivability		
	Weather	Tropical	All Weather	
		Hot Weather & Desert		
		Cold Weather		
		Thunderstorms & Turbulence		
		Snow		
	Time of Operation	Day	Round-the-Clock	
		Day/ Night		
		Night		

Tab 1 Inputs, Mission Requirements and Outputs of Mid-Life Upgrade System for Anti-Submarine Missions

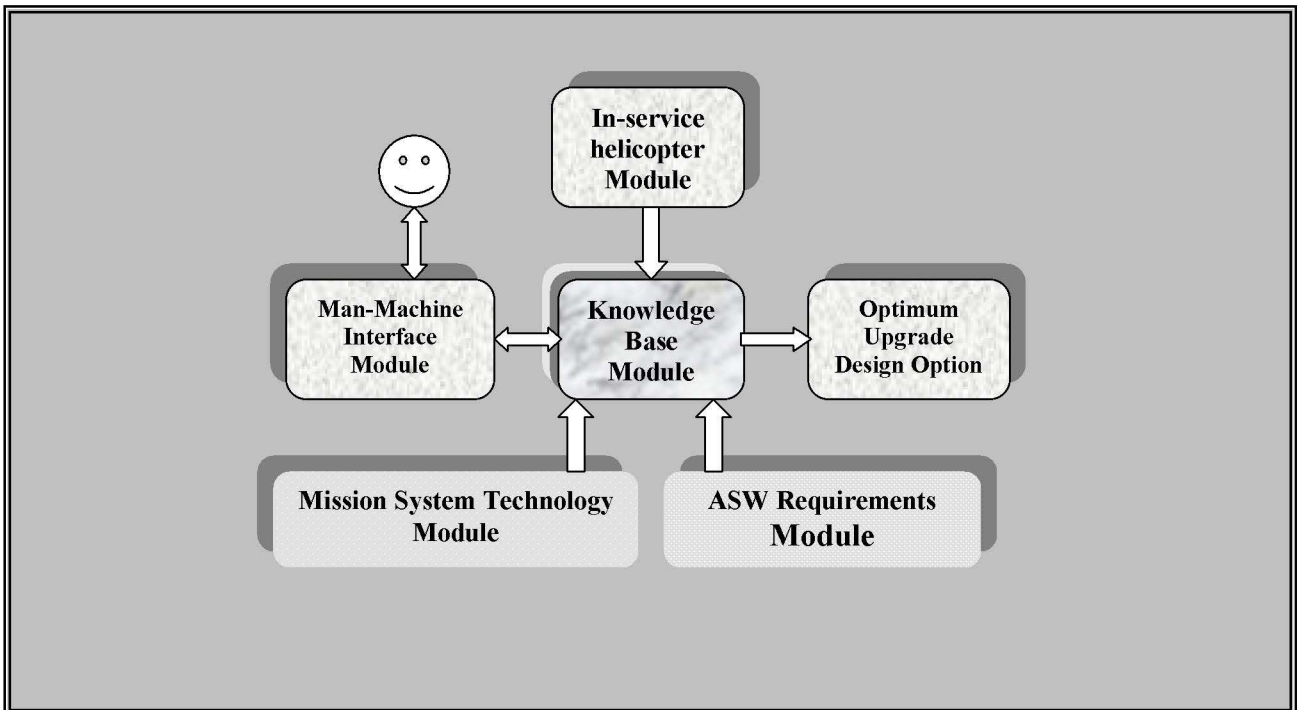


Fig 3 Automation Process to Enhance Anti-Submarine Capabilities through Mid-Life Upgrade [Ref 2]

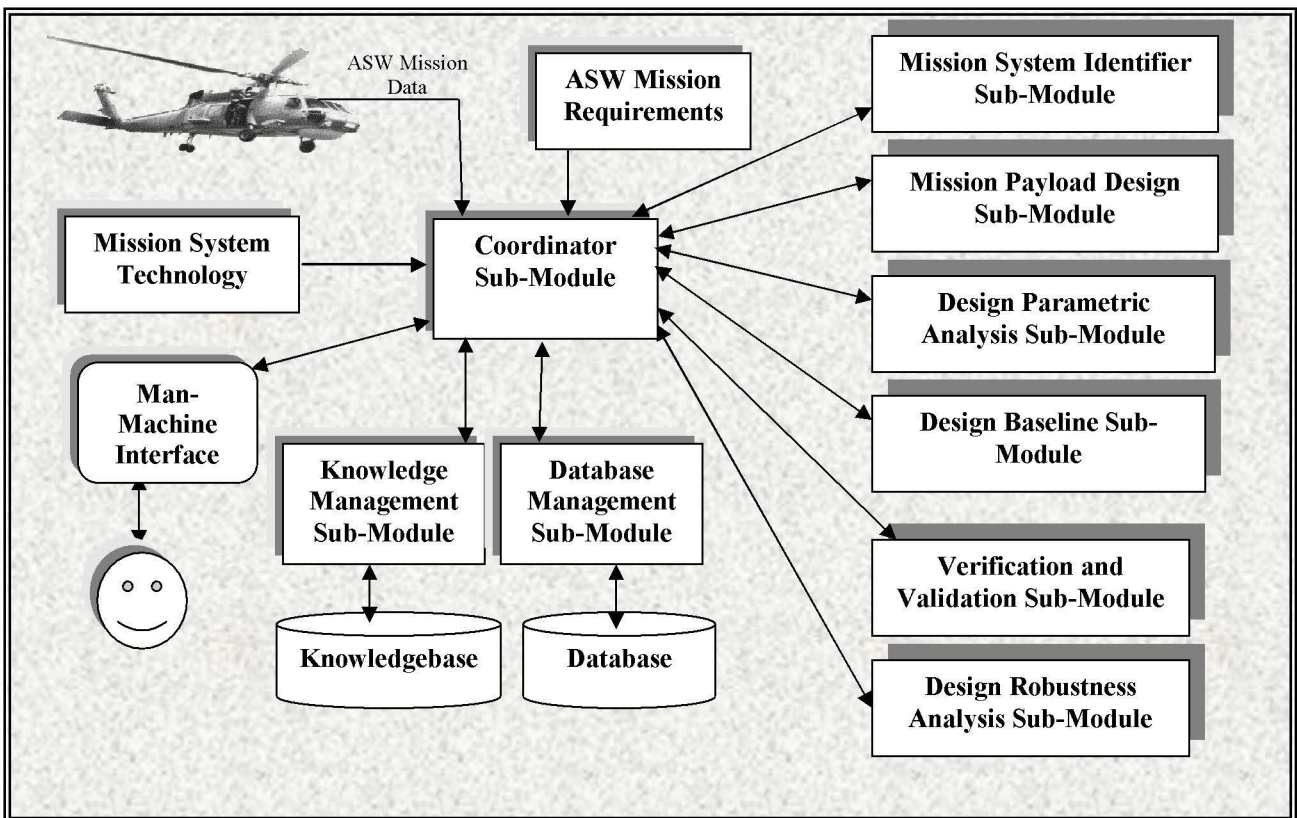


Fig 4. Framework for Intelligent Decision Support System to Enhance Anti-Submarine Mission Capabilities

5.1 Software Architecture

In this research project it is considered that the modules shown in Figure. 4 can either be integrated or distributed. The software agents communicate with a common platform representing a shared blackboard in client-server architecture. The iteration representing the pre-upgrade design analysis process is automatically handled by the Coordinator sub module. The architecture is 3-tier with the database and the knowledge base holding the operational and mission systems data at the lower tier. Knowledge base module is at the middle tier. The man-machine interface is at the top layer. The shows the “Intelligent Decision Support System” software architecture is presented in Figure 5.

5.1.1 Man-Machine Interface

The user interface for the IDSS software consist both the web-based and windows-based graphical user interface. The web-based graphical user interface is used is responsible for collecting the ASW mission data even from a remote site and also providing a uniform standardised mission data useful for creating mission profile and also mission systems identification for a particular environment conditions. The windows-based graphical user interface provides powerful tool for designing alternative mission payloads while viewing different configurations of payloads and providing operational and technological analyses in the form of 2D and 3D visualisations. User restrictions can be put on the user access to both forms of graphical user interface providing a level of abstraction. Microsoft.NET [11] provides the development experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.

5.1.2 Knowledgebase and agents

The Knowledgebase and Agents of the IDSS provide the functions to the automation process and to conduct the mid-life upgrade computations design decisions. Various complex functions and mega formulas can be programmed into user defines sub-routines providing the flexibility of using relevant mathematical and geometric knowledge with reason for a specific decision software tool which simplifies user-system interaction. Interface to other desired software such as Microsoft Excel can be easily provided during the tool development to handle large mass of data. Tools such as Microsoft Visual C# and Visual Basic can provide

rich set of function that can be packaged into a single tool IDSS.

5.1.3 Database

The data and information about the system is stored in the Database module which is in the lower layer. The data is “Relational Database” and hence uses Structured Query Language (SQL). The Data Agent and the Coordinator Agent communicate in passing the helicopter operational and mission systems data to the database. Most of the “Relational Database Management Systems” and Web-based systems can understand SQL. The knowledge base consists of information about the previous helicopter configuration, mission and environmental conditions. All the knowledge is represented in extensible Markup Language (XML). This helps in a standardised exchange of information across different agents distributed and the internet.

5.1.4 Interfaces

The IDSS provides communication between the agents and also to external applications. The HTTP (Hypertext Transfer Protocol) is used to communicate while using the web-based GUI. The data and knowledge exchange for web-based GUI uses Simple Object Access Protocol (SOAP). SOAP is a language, platform independent communication protocol supported by Microsoft and is based on XML [12].

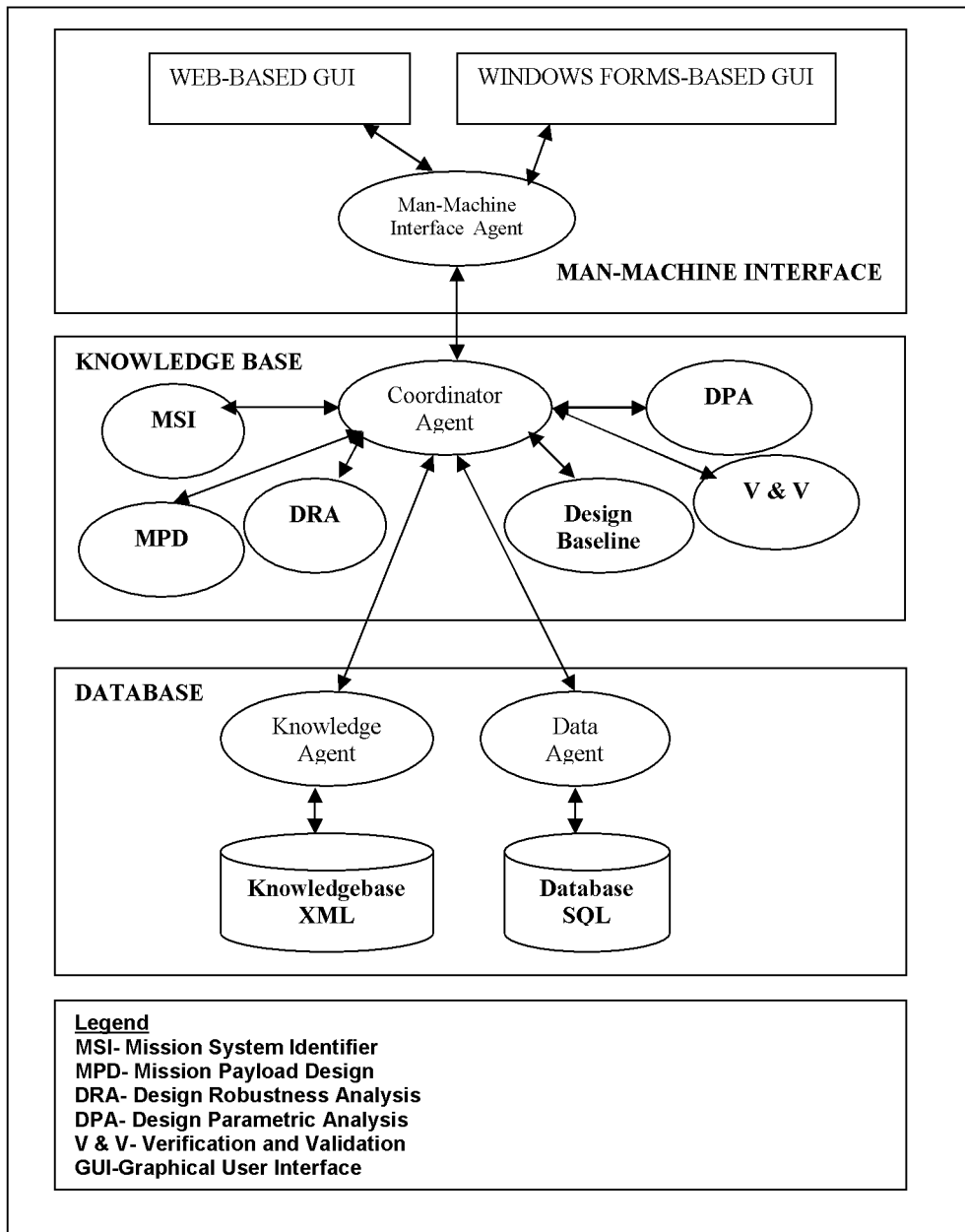


Fig 5 Intelligent Decision Support System Architecture

6 Results and Discussion

The distributed architecture of the IDSS provided the basis for the identification of various CDTs and RAD tools. A wide variety of commercial off-the-shelf ubiquitous software can be used for the development of IDSS to meet the following objectives:

- a) The system consists of distributed set of components connected by a network.
- b) A graphical user interface that is ubiquitous.
- c) Fusion of data from various operational and environment sources or systems.
- d) Inherit the capabilities to conduct complex computations and handle large volume of data.
- e) Knowledge of previous missions and configurations; providing different scenarios to the user including visualisations.
- f) Software be modular, support reuse and scalable.
- g) Adhere to various software engineering standards.

7 Conclusion

The software architecture developed adopting a systems approach considered various commercial-off-the-shelf software to automate the upgrade design process. The study provides a promising avenue for complex development of IDSS to inherit capabilities of handling large volume of data, automating knowledge and information re-use to assess different mission payloads.

8 References

- [1] Aiyappan R. *Knowledge Driven Automation*. <http://www.expresscomputeronline.com/20021223/tech1.shtml>, 2002, access date: 23/02/2004.
- [2] Jonnalagadda S, Sinha A K, Hoffmann P, and Schrage D P. Modelling Upgrade Design of Maritime Helicopters for Anti-submarine Missions- Project Framework. *Proceedings of the 60th Annual Forum and Technological Display of American Helicopter Society*, Baltimore, MD, 08-10 June, 2004.
- [3] Kusumo R and Sinha A K. Common Development Tools For An Integrated Decision support System for Mid-Life Upgrade Analysis. *22nd Congress of the International Council of the Aeronautical Sciences*, 613, 613.1-613.8, Toronto, Canada, 8-13 September, 2002.
- [4] Morgan J. Anti-Submarine Warfare-A Phoenix for the Future. <http://www.chinfo.navy.mil/navpalib/cno/n87/usw/autumn98/anti.htm>, 1998, access date: 16/04/2003.
- [5] Mulholland D. USAF puts upgrades before platforms, *Jane's Defence Weekly*, Vol. 35, i.23, pp.5, 2000.
- [6] Roth B A, and Marvis D N, *A Stochastic Approach to Designing Affordable, Environmentally Acceptable Systems*, www.asdl.gatech.edu, 2002.
- [7] Sinha A K, Kam B H, and Wood L A. A Comparative Analysis of Decision Support Systems for Helicopter Multi-Mission Modifications. *First International Seminar on the Development and Utilization of Helicopters in South Asia Pacific*, Bangalore, India, 1995.
- [8] Sinha A K, Kam B H, Wood L A, & Caterson J. Toward a Compromise Solution for Helicopter Multi-Mission Modification Decisions. *Proceedings of 52nd Annual Forum and Technology Display of American Helicopter Society*, Washington, D C, U.S.A, 4-6 June 1996.
- [9] Sinha A K, Bil C, and Scott M L, Design of Payloads for Mid-life Upgrade of Maritime Helicopters: Stage I, II, III and IV. *Proceedings of 3rd Australian Pacific Vertiflite Conference on Helicopter Technology*, Canberra, A.C.T, 12-14 July 2000a.
- [10] Sinha A K, Bil C, and Scott M L. Design of Payloads for Mid-life Upgrade of Maritime Helicopters: Stage V and VI. *International Conference on Systems Thinking in Management*, Melbourne, Victoria, 8-10 November 2000b.
- [11] Microsoft.Net Framework Documentation <http://www.msdn.microsoft.com>
- [12] Learn SOAP, www.w3schools.com, access date: 3/07/2004.