

OPERATOR INTERFACE AND SITUATION PERCEPTION IN HIERARCHICAL INTELLIGENT CONTROL: A CASE STUDY

Hui-Min Huang
 Mechanical Engineer
 National Institute of Standards and Technology
 Gaithersburg, Maryland 20899
 Phone: (301) 975-3427
 huang@cme.nist.gov

ABSTRACT

This paper covers some operator interface issues of a hierarchical intelligent control system. The logical structure of the operator interface function in reference to the NIST Real-time Control System (RCS) architecture has been investigated. An emphasis is that the operator interface should be integrated seamlessly with existing systems. A design issue is to allow operators to be involved in system operations in different degrees. Operators at different control levels have different perceptions of the world. Software service personnel and control operators may require totally different types of information.

1. INTRODUCTION

The researchers at the Intelligent Systems Division of the National Institute of Standards and Technology (NIST) supported an ARPA submarine automation project (ARPA Order No. 7829; the ARPA Maritime Systems Technology Office). This project used the SSN 637 class nuclear submarine data as the model. The study focused on parts of the maneuvering and engineering support functions and developed a series of software systems [1, 2] to demonstrate the results.

The core technology that the researchers apply to the submarine automation problem is the NIST Real-time Control System (RCS) reference model architecture. Although numerous papers have been published describing the RCS theory and applications [3, 4, 5], the issue of operator interface requires systematic studies. This paper focuses on the operator interface issues in RCS based control systems. The researchers view that, in the RCS architecture, the operator interface function forms a logically distinct structure parallel to the control

hierarchy. In RCS, a central concept is multiple levels with distinct, but successive, levels of abstraction. The researchers attempt to apply the same concept to the operator interface structure. The goal is to form a unified and comprehensive logical structure for the development, operation, and service of RCS applications.

1.1 The Control System

A brief summary of the submarine control system is given here to facilitate the understanding of this operator interface issue. The left side of *Figure 1* describes the submarine automation control hierarchy. A box represents a controller within the hierarchy. The command controller handles the highest level control, namely, the execution of the missions. Such control is achieved by assigning tasks to and coordinating the behavior of the two subordinates, the Maneuver and the Engineering Systems controllers. The tasks that these two subordinates execute are at a lower level of abstraction with higher resolution and contain a higher level of detail. Similarly, these two controllers complete their tasks by:

- * decomposing their tasks and assigning the resulting sub tasks to their subordinate controllers, propulsion, helm, and depth, and ventilation and diesel, respectively.
- * coordinating the execution of the subordinate controllers.

The same principle applies to still lower level controllers. Propulsion coordinates the DC motor, the clutch, and the

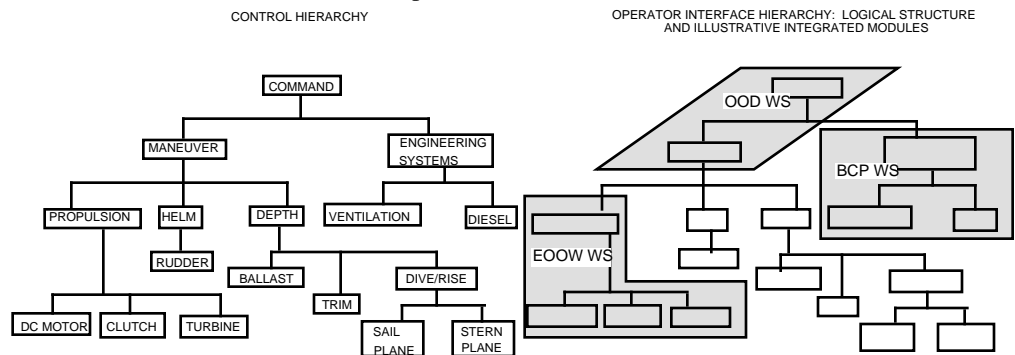


Figure 1: The illustrative submarine automation control hierarchy

turbine controllers. The lowest level typically contains actuator controllers.

Each controller has a sensory processing and a world modeling function. Their responsibility is to perceive the state of the world in real-time and to support the aforementioned decision making process. See [2] for a detailed description of the submarine control system.

The right hand side of *Figure 1* describes the operator interface hierarchy, which will be described later.

1.2 Basic Principles for Operator Interface

Our basic principles for operator interface design are:

- * to have a well-defined overall structure and well-defined roles for individual operators,
- * to integrate seamlessly with the system control,
- * to integrate seamlessly with the current operating environment,
- * to match user requirements and expertise,
- * to allow emergency procedures,
- * to optimize both real-time performance and operator workload,
- * to inform intuitively, in real-time and as much as operators require, but not to overload, and
- * to support system service and development, including testing and simulation.

The remainder of this paper describes and illustrates these principles.

2. THE ORGANIZATION AND FUNCTIONING OF THE CONTROL OPERATOR INTERFACE

The researchers propose that each of the controllers in an RCS hierarchy is logically associated with an operator interface (OI) node. This concept facilitates a well-defined overall structure for OI. In *Figure 1*, the right side shows a hierarchy for OI, which is an image of the control hierarchy, shown at the left side of the figure.

2.1 Levels of Commanding Authority

To support hierarchical system real-time control, in RCS, the operator interface functions are divided into six types of levels of commanding authority. Specific applications may not contain all the levels. From the highest to the lowest levels of authority, the level types are:

Level 6 -- application domain level, or mission level, operator. This is the highest level. The operator enters and monitors the overall command for the entire control systems. In our case study, the command controller operator belongs to this level.

Level 5 -- group level operators. These handle control activities involving either multiple coordinated control entities or multiple coordinated groups of control entities. Our particular case does not involve this function.

Level 4 -- equipment or task level operators. These operators deal, through their corresponding controllers, with the control activities of individual major physical entities. In the hierarchy shown in *Figure 1*, the maneuver and the engineering systems controller operators are of this type.

Level 3 -- elementary move (emove) level operators. In our case study, the propulsion, helm, and depth controller operators belong to this level. The operators may monitor or command such elementary move control activities as avoiding obstacles, including the Arctic region ice keels, and such control activities as avoiding kinematic limits, including the possible instability of the submarine caused by large bubble (pitch) angles.

Level 2 -- primitive (prim) level operators. These operators must ensure the dynamic achievability and smoothness of the kinematically sound control activities. The dive/rise, trim, and ventilation operators are of this type.

Level 1 -- actuator level operators. Through their corresponding controllers, these operators deal with direct environmental interaction, such as the electrical and mechanical signals that drive the actuators toward the goals.

This hierarchical organization also allows a problem with a high level of abstraction to be logically and smoothly transitioned to a set of sub problems with low levels of abstraction. In this way, the operator interface for a complex problem can be shared by a set of operators with well defined and limited responsibilities.

2.2 Watch Station Displays

For either well-established industry areas or any types of systems in which manual operations dominate, it is beneficial to introduce automation in an evolutionary, as opposed to revolutionary, manner. Automation and intelligent control should be incrementally integrated in. Therefore, the author proposes that the logical OI structure must be designed to be compatible with the current submarine operating environment.

The researchers have developed three watch station (WS, a submarine term, meaning an onboard location at which crew members are assigned to perform pre-specified duties) graphic panels to provide the input and output capability

for operators during real-time control. The three displays are shown in the three shaded areas in *Figure 1*. These correspond to the layout of actual submarine operational compartments [2] and are also mapped to the control hierarchy.

The operator interface (OI) must display the necessary information for all the controllers, in real-time, to enable the interaction between the control hierarchy and the submarine operators. Note that the objective of the OI is

not to mimic the current submarine operating environment. Instead, the number and types of WS displays are determined by considering the following three factors: the operator workload [4], understandability and acceptability by the current submarine operation community, and the efficiency of hierarchical system control.



Figure 2: The Officer of the Deck Watch Station Display

The three WS displays are:

- * The Officer of the Deck watch station (OOD WS), which serves as the OI of the command and maneuvering controllers of the control hierarchy.
- * The Engineering Officer of the Watch watch station (EOOW WS), which serves as the OI of the propulsion controller and all its subordinates.
- * The Ballast Control Panel watch station (BCP WS), which serves as the OI of the engineering systems, ventilation, and diesel controllers.

The OOD WS, illustrated in Figure 2, displays the crucial maneuvering data for the OOD, including (from left to right) the bubble angles, the heading and speed, and the depth. The WS also includes two text-based message areas: the left side showing the command that the command controller is outputting (for maneuver) and the right side showing the announcement that the controller is making. When a lubrication (lube) oil fire is reported through the commanding channels, the command controller immediately announces the message of “ENG ROOM FIRE, ALL HANDS ON EABs (emergency air breathers).” Meanwhile, the COMMAND text area starts displaying “PREP FOR EMER VENT,” meaning that the command controller is ordering the maneuver controller to execute the displayed command: to prepare for emergency ventilation.

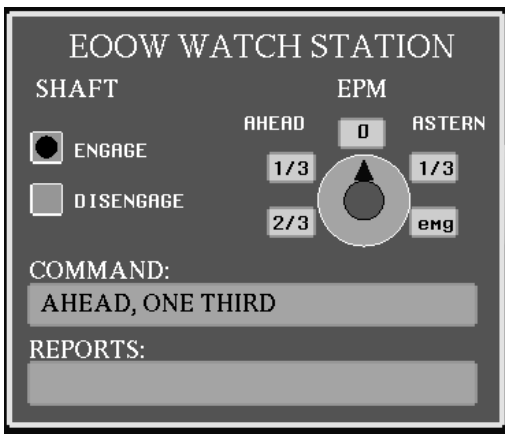


Figure 3: The Engineering Officer of the Watch Watch Station Display

The Engineering Officer of the Watch watch station, shown in Figure 3, employs two buttons for engaging or disengaging the main shaft clutch. This WS also employs a speed control knob for the Emergency Propulsion Motor (EPM). In addition, this WS has two text-based message windows. The command text window normally displays the command that the propulsion controller is executing. When the propulsion controller requests the operator to perform a command, this command appears in this window. Meanwhile, the window will shade in yellow.

The REPORT message window displays useful messages for the Engineering Officer of the Watch operator. In Figure 3, EOOW must execute a ONE_THIRD_SPEED command, which is a sub command of the “PREP FOR EMER VENT” command that Maneuver is executing.

The Ballast Control Panel watch station functions similarly to the other two WSs. See [2] for additional details.

2.3 Operator Interface Data Types

The following summarizes the data types developed for operator interactions:

* Input from the operators: Status, Command selections, Environmental variable settings, and Actuator override devices.

* Output to the operators: Continuous operational data, including ship depth displays and paths; Digital operational data, including ship speed displays; Discrete activities, such as commands, announcements, watch station reports; Operator input requests; Schematic diagram; Errors and recommendations; Controller performance, such as execution time; and Debug data, including command/status, world data.

3. LEVELS OF OPERATOR INVOLVEMENT DURING OPERATIONS

RCS allows operators to be involved in system operations at various degrees. In this application, the researchers started experimenting with five degrees of access control, giving the operators from the least to the most amount of authority to interact with individual controllers in the hierarchy:

- * Monitor, or to be informed.
- * Respond to system requests.
- * Alter system behavior by issuing new commands.
- * Manual override.
- * Modify system, or to reconfigure control hierarchy.

The access control should be applied to both the controllers and the tasks. The former means that each controller in the hierarchy has a logical operator interaction function with assigned degrees of operator access control. The latter means that each step in a task plan may be assigned certain degrees of operator access control. For example, operator intervention may not be allowed during the execution of a safety related task. Our experiments focused on applying the operator interaction to controllers. Further research is required to integrate these two aspects.

The author uses a two-dimensional matrix to visualize our access control concept, as shown in Figure 4. Monitoring is the default degree of authority, which is available to all the operators.

The next unit on the horizontal axis, respond to system requests, does not allow operators to initiate interruptions to system operations. The system design does not allow the propulsion operator to issue new commands. Rather,

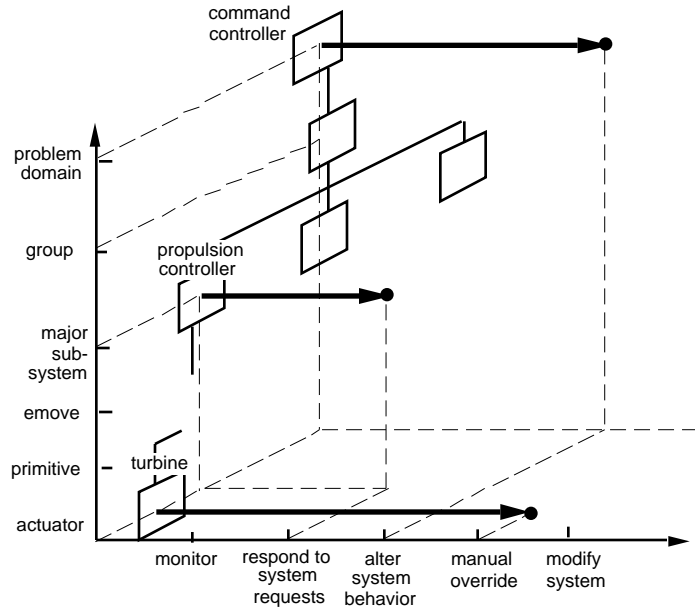


Figure 4: Different degrees of operator involvement

the operator is only authorized to perform requested actions, such as CHANGE_TO_EPM, as described in section 2.2. The operator, then, manually, disengages the main turbine shaft, engages the EPM shaft, and clicks the button on the display to report the completion. In this case, the operator serves as the EPM controller. This man-in-the-loop control operation also suggests that our method supports integrating the automation technology to legacy systems with manual operations.

Another example of responding to system requests is for operators to make decisions for the control system. The submarine may run into some salinity disturbances close to coast where the depth controller can not maintain the ship’s depth. The error messages and a set of options are displayed for Maneuver. The operator selects a best command for the controller to perform [1].

The next unit on the horizontal axis, alter system behavior, means that the operator initiates graceful changes to system operations. In other words, operator commands are integrated with the ongoing system automatic operation. The command controller operator can alter system behavior by issuing a new mission command and

having both the new and the existing mission commands scheduled and executed.

The next unit on the horizontal axis, manual override, means that the operator takes over the control and the automatic control commands are ignored, suspended, terminated, or aborted. The lowest level operators can block the automatic commands and directly manipulate the actuators [1]. When the submarine is about to hit ice keels, the helm operator must be allowed to enter an emergency turning command.

Reconfiguring control hierarchy means that the operator can re-align the command chain of the hierarchy or can expand or reduce the control capability of the controllers. In an early version of the submarine control system, a FORTH based programming language was used. This allowed the system operation to be halted, new task plans to be programmed in and incrementally loaded, and the system operation to be resumed. Such capability was abandoned later in favor of using C as the programming language.

4. User Expertise and Hierarchical Situation Perception

As stated in our basic principles, operator interface information must meet the user requirements and expertise [6]. All users are unique. They require different types of data. In one respect, data are processed and assimilated according to the levels of the operator commanding authority.

In the RCS hierarchical environment, different levels deal with events and commands at different resolutions and time scales. Temporal and spatial integration must be performed when lower level information, either discrete events or continuous data, is delivered to higher levels. This integration process creates a different perception. Operators only need to understand the situations at two levels of resolution: their own level and the level below.

In Figure 5, the author illustrates that the high-level control level, Maneuver, is integrating the information received from the three spatially distinct subordinates, propulsion, depth, and helm. At the first instance, propulsion reports that the submarine has achieved the required speed. At the second instance, helm reports that the areas of concern have been cleared of hostile objects. At the following instance, depth reports that the submarine has reached the required depth. Next, propulsion reports that the emergency motor has been engaged. Maneuver summarizes all the information and concludes that it is ready to perform emergency ventilation.

Note that the world modeling function within a control node may involve the following two types of knowledge transformation:

- * from raw data to concise data and
- * from the raw or concise data to conventional format that match the operator expertise.

5. Supporting Other System Functions

The focus of this paper is on the operators' roles in system control. There are, however, additional types of operators that support the control systems, including supporting system development and service, simulation, and training. A system servicer may prefer to review raw data, as opposed to view concise and processed data often preferred by a control operator. The author provided a detailed description of these issues in [1].

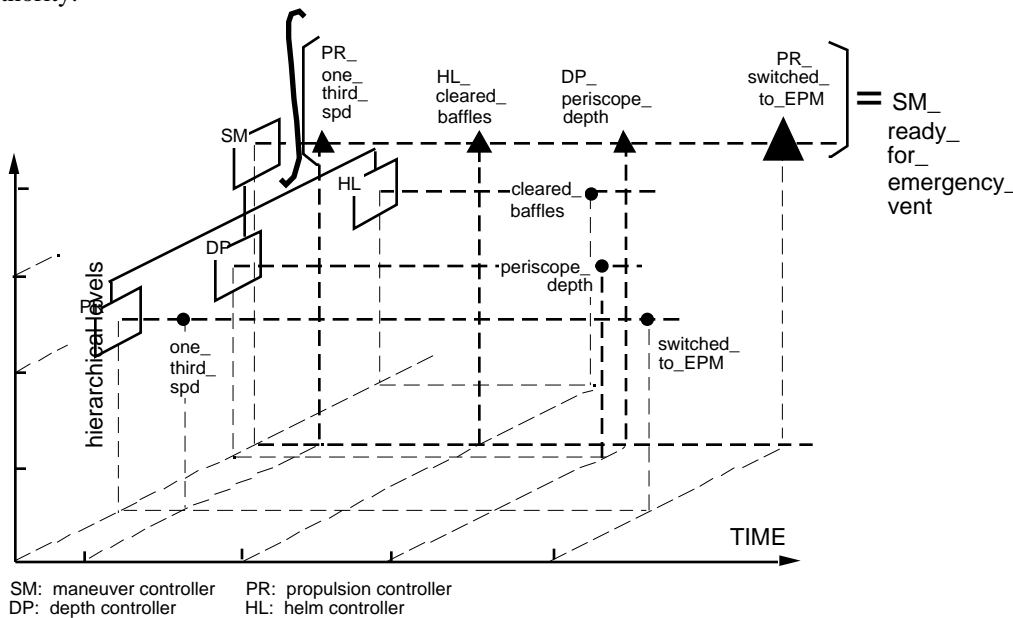


Figure 5: hierarchical situation evaluation

6. SUMMARY AND ISSUES

The results on studying the submarine automation operator interface issue were presented. A logical structure for operator interface to support hierarchical real-time control system control was proposed. The following are some specific results:

- * Well-defined operator interface logical structure.
- * User interface organized as watch stations to suit distributed operational environment.
- * Different levels of operator involvement.

To improve the operator interface system, the resulting illustrations must be generalized. Some specific topics include:

- * Allowing only authorized operators for the individual watch station displays.
- * Allowing operators to select commands or parameters from within the system task structure.

Additionally, there is an issue of whether an operator interface function is to support the operation of a RCS controller or operators themselves are to serve as controllers. As an initial effort, in this paper, the author references the latter situation as "man-in-the-loop" control. However, further study is needed to provide rigorous guidelines and methods to distinguish these two roles.

ACKNOWLEDGMENTS

Captain Robert Lowell and Dr. Ed Carapezza of ARPA have been the sponsors of this project at various stages. They have provided valuable technical insights to the problem. Dr. James Albus, Division Chief of the Intelligent Systems Division of NIST, and Dr. Anthony Barbera of the Advanced Technology and Research (ATR) Corporation have been leading the research of RCS since they originated RCS two decades ago at NIST. Richard Quintero of NIST managed and established the technical objectives for this project. M. L. Fitzgerald, Nat Frampton, Philip Feldman, and Clyde Findley of ATR and Ron Hira, Ross Tabachow, and Will Shackelford of NIST participated in various stages of this project.

REFERENCES

[1] Huang, H., Hira, R., and Quintero, R., "A Submarine Maneuvering System Demonstration Based On The NIST Real-Time Control System Reference Model," in *Proceedings of The 8th IEEE International Symposium on Intelligent Control*, Chicago, Illinois, 1993, pp. 376-381.

-
- [2] Huang, H., "An Architecture and a Methodology for Intelligent Control," *IEEE Expert*, Volume 11, Number 2, April 1996, pp. 46-55, IEEE Computer Society, Washington, D. C.
- [3] Albus, J.S., "Outline for a Theory of Intelligence," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21, No. 3, May/June 1991, pp. 473-509.
- [4] Huang, H., Quintero, R., and Albus, J.S., "A Reference Model, Design Approach, and Development Illustration toward Hierarchical Real-Time System Control for Coal Mining Operations," Book Chapter in *Control and Dynamic Systems, Advances in Theory and Applications*, Volume 46, Academic Press, 1991, pp.173-254.
- [5] Szabo, S., Scott, H.A., Murphy, K.N., Legowik, S.A., Bostelman, R.V., "High-Level Mobility Controller for a Remotely Operated Unmanned Land Vehicle," *Journal of Intelligent and Robotic Systems*, 5, 199, pp. 63-77.
- [6] Mayhew, D. J., *Principles and Guidelines in Software User Interface Design*, Prentice Hall, Englewood Cliffs, NJ, 1992.