1990

A Software Prototype of the Message Processor in Navy C31 Station

Luqi; Davis, Tony

Naval Postgraduate School

Luqi, T. Davis. "A Software Prototype of the Message Processor in Navy C31 Station", Technical Report NPS 52-90-010, Computer Science Department, Naval Postgraduate School, 1990.
https://hdl.handle.net/10945/65236

NPS52-90-010

NAVAL POSTGRADUATE SCHOOL

Monterey, California

A SOFTWARE PROTOTYPE OF THE MESSAGE PROCESSOR IN NAVY C^3I STATION

LUQI and DAVIS, TONY

August 1989

Approved for public release; distribution is unlimited.

Prepared For:

National Science Foundation
Washington, DC 20550

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral R. W. West, Jr.
Superintendent

Harrison Shull
Provost

This report was prepared in conjunction with research funded by the National Science Foundation.

[redacted]

LUQI
Assistant Professor
of Computer Science

Reviewed by:

[redacted]

for ROBERT B. MCGHEE
Chairman
Department of Computer Science

Released by:

[redacted]

KNEALE T. MARSHALL
Dean of Information
and Policy Science

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
NPS52-90-010

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Computer Science Dept.
Naval Postgraduate School

6b. OFFICE SYMBOL (if applicable)
52Lq

7a. NAME OF MONITORING ORGANIZATION
National Science Foundation & ONR

6c. ADDRESS (City, State, and ZIP Code)
Monterey, CA 93943

7b. ADDRESS (City, State, and ZIP Code)
Washington, DC 20550

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
National Science Foundation

8b. OFFICE SYMBOL (if applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
NSF CCR-8710737

8c. ADDRESS (City, State, and ZIP Code)
Washington, DC 20550

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
A SOFTWARE PROTOTYPE OF THE MESSAGE PROCESSOR IN NAVY C^3I STATION (U)

12. PERSONAL AUTHOR(S)
Luqi and Davis, Tony

13a. TYPE OF REPORT
Progress

13b. TIME COVERED
FROM Mar TO Dec 89

14. DATE OF REPORT (Year, Month, Day)
August 1989

15. PAGE COUNT
40

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Command, Control, Communication and Intelligence (C^3I) systems are hard real-time systems that assist a force commander in understanding a tactical situation within his area of responsibility. The system provides the commander with the necessary information to enable him to evaluate options and direct appropriate responses. This paper focuses on a sub-module of the entire system, the Message Processor (MP), which is responsible for message passing of data between ships and aircraft. The Message Processor is a time constrained system, that is, it must meet real time processing deadlines in order for the C^3I system to provide the force commander with timely information. The complexities of design of large real time systems, such as the Message Processor, has often resulted in low programming productivity and reliability. These problems may be alleviated through the modeling of the system by rapid design prototypes. The Prototyping System Description Language (PSDL) is well suited to the construction of these rapid prototypes. The requirements of real time systems may be firmed up in an iterative approach through PSDL executable prototypes. This iterative approach clarifies requirements and eliminates a large amount of wasted effort currently spent on software development

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
[X] UNCLASSIFIED/UNLIMITED [ ] SAME AS RPT. [ ] DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
Luqi

22b. TELEPHONE (Include Area Code)
(408) 646-2735

22c. OFFICE SYMBOL
52Lq

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

traditional software life cycle models. PSDL is designed to operate within a software support environment which contains a software-base management system, a syntax directed editor, a design database and an execution support subsystem. This paper presents a PSDL prototype of the Message Processor subsystem suitable for execution within this software support environment.

A Software Prototype of the Message Processor in a Navy C3I Station

- Modeling and Specification of Hard Real-Time Systems in PSDL

Luqi
Tony Davis

Computer Science Department
Naval Postgraduate School
Monterey, CA 93943

TECHNICAL REPORT NPS 52-89-15

ABSTRACT

Command, Control, Communication and Intelligence (C^3I) systems are hard real-time systems that assist a force commander in understanding a tactical situation within his area of responsibility. The system provides the commander with the necessary information to enable him to evaluate options and direct appropriate responses. This paper focuses on a sub-module of the entire system, the Message Processor (MP), which is responsible for message passing of data between ships and aircraft.

The Message Processor is a time constrained system, that is, it must meet real time processing deadlines in order for the C^3I system to provide the force commander with timely information. The complexities of design of large real time systems, such as the Message Processor, has often resulted in low programming productivity and reliability. These problems may be alleviated through the modelling of the system by rapid design prototypes. The Prototyping System Description Language (PSDL) is well suited to the construction of these rapid prototypes. The requirements of real time systems may be firmed up in an iterative approach through PSDL executable prototypes. This iterative approach clarifies requirements and eliminates a large amount of wasted effort currently spent on software development using traditional software life cycle models.

PSDL is designed to operate within a software support environment which contains a software-base management system, a syntax directed editor, a design database and an execution support subsystem. This paper presents a PSDL prototype of the Message Processor subsystem suitable for execution within this software support environment.

I. INTRODUCTION

A. PROTOTYPE SYSTEM DESCRIPTION LANGUAGE

In February 1989 the Naval Research Advisory Committee conducted a review of the Next Generation Computer Resources [Ref. 1:p. 1-55]. Among the committee's findings was the conclusion that embedded real time microcomputer systems are proliferating and that the Navy should encourage the use of ruggedized equipment in protected environments. The committee also concluded that the Navy should redirect its system prototyping effort away from its current practice of prototyping standard Navy-wide computer resources. This effort should, in the committee's opinion, be directed towards prototyping computer resource that use ruggedized commercial equipment and incorporation of commercial standards.

There has been an ongoing effort at Naval Postgraduate School (NPS) Computer Science Department to develop a low cost Naval Tactical Data System (NTDS) display using commercially available computer resources, such as the ruggedized SUN workstation. In conjunction with this program, the Computer Science Department has been developing software tools to aid the software generation process. The tools consist of an automated software support environment which is essential for the rapid construction of system prototypes. The Prototyping System Description Language (PSDL) has been developed for use within this environment. The environment contains a software-base-management system, a syntax-directed editor with graphics capabilities, a design database and an execution support subsystem [Ref. 2:p. 29].

These tools are necessary to ensure high programming productivity and reliability of military software. One of the more promising methods to reach this goal, and the goals of the Naval Research Advisory Committee, is by the use of rapid prototypes. PSDL is especially suited to the production of prototypes for large real-time systems, as well as being useful for prototyping typical Ada applications. Rapid prototyping has been found to be an effective technique for clarifying requirements and eliminating the large amount of wasted effort currently spent on developing software to meet incorrect or inappropriate requirements in traditional software life cycles

[Ref. 3:p. 1409]. The requirements of these large software systems may be firmed up iteratively in a rapid prototyping approach through the use of executable prototypes that are designed in PSDL.

This paper presents a PSDL prototype of a C^3I system Message Processor (MP) module. The language is well suited for requirements analysis, feasibility studies and the design of large embedded systems. In addition, PSDL has constructs designed to enforce and record timing constraints, which are an essential part of real-time system specification.

B. C^3I MESSAGE PROCESSOR DESCRIPTION

A Command, Control, Communication and Intelligence (C^3I) System assists a commander in understanding a tactical situation within his geographical area of responsibility. The system provides the commander with the necessary information to enable him to evaluate options and direct appropriate responses. There are many sub-modules within a complete C^3I system, but this paper focuses on the module that is responsible for message processing. Such a module would find application within a Naval Tactical Data System (NTDS) and is analogous to the current link software module employed on NTDS Link-11 and Link-4 capable ships.

The Message Processor (MP) component of a C^3I system is responsible for routing:

- track reports,
- participating unit directives,
- status reports, and
- combat directives,

between a Direction System (DS) with onboard sensors (OS), analogous to the NTDS, and the Ship and Aircraft radio transceivers. The MP provides Ship-Link track identification numbers and reporting responsibility conflict resolution for 256 current tracks for broadcast over a Ship Link (SL) and/or Aircraft Link (AL). This provides a common datum for the Participating Units (PU) in the link.

The MP is also responsible for maintaining status information and participation-mode information for PUs, in addition to prompting DS system operators (OT) for responses to in-coming and out-going directives. A basic system architecture is illustrated at Figure 1.1.

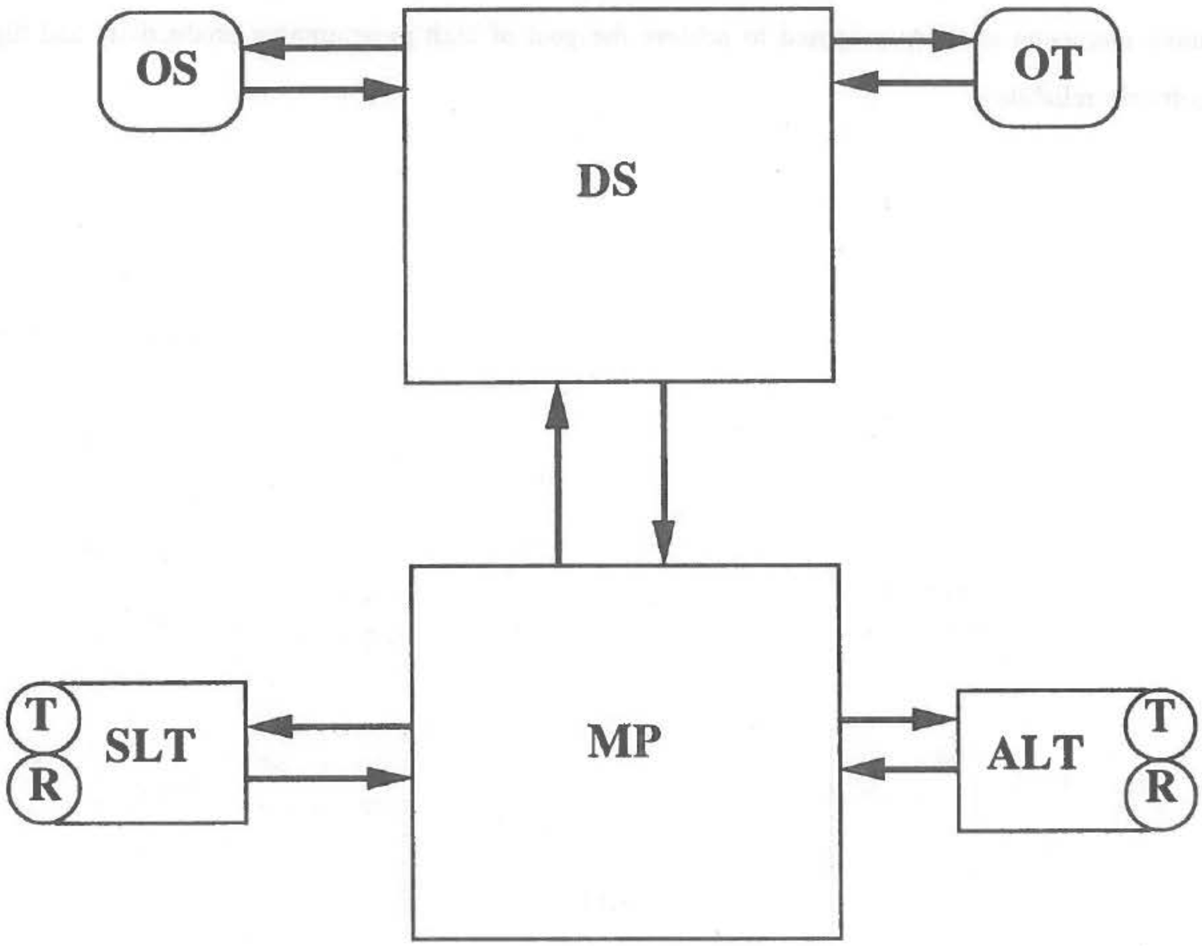


Figure 1.1 - Basic C3I System Archictecture

Such systems are characterized by the requirement to perform real time processing. The ability, or inability, of such systems to meet these real-time processing deadlines may affect tactical outcomes. The complexities of large real-time systems, such as this C^3I system, has often resulted in low programming productivity and reliability. However, rapid prototyping in PSDL is one of the more promising methods proposed to achieve the goal of high programming productivity and high software reliability.

2. MESSAGE PROCESSOR REQUIREMENTS

A. GENERAL REQUIREMENTS

The Message Processor (MP) is responsible for offloading the following data:

- Message routing information,
- Ship link track management,
- Unit status,
- Directive management, and
- Communication link functions.

The message processor software must satisfy the following general functionality and performance requirements [Ref. 4:p. 15]:

- MP routes incoming and outgoing (via Ship and Aircraft Link Transceivers) radar track reports between the DS, SL and AL. The MP maintains a SL-track id assignment for the most critical of up to 256 current tracks being broadcast over the SL. The MP resolves conflicting SL-track id allocations of incoming SL-track reports to provide a common track-reference datum for the ships and aircraft within a geographical link node.
- MP manages directives to and from the ship in which it resides.
- MP maintains status and participation-mode information from the Unit-Status (US) and Participating-Unit (PU) messages that are broadcast over the link via the AL and SL transceivers.

These functional specifications may be used to construct a module decomposition diagram. This diagram is realized by analyzing the detailed requirements of the MP system and assigning one or more modules to meet those requirements. The detailed requirements are presented in Section 2.B, while the module decompositions are presented in Section 2.C.

B. DETAILED REQUIREMENTS

This section discusses the detailed requirements necessary for the MP to satisfy the general requirements described in the previous section.

1. Requirement # 1

SL shall control the SLT to implement the Master Unit (MU) or Participating Unit (PU) role in the Round-Robin protocol. There shall be only one MU per node at any time. The MU is analogous to the term Net Control Ship (NCS) used in NTDS capable ships and node is analogous to an entire NTDS link-11 net [Ref. 4:p. 15].

2. Requirement # 2

A DT report from DS shall if possible be assigned a free or lowest-priority track-id (if current id is null) and broadcast over SL within two SL cycles of its arrival at the MP [Ref. 4:p. 19].

3. Requirement # 3

Each ST that is broadcast over SL shall be routed to DS if it is consistent with the receiving ship's current SL tracks; otherwise one of the two inconsistent track reports shall be selected (to become the current track with that id) and routed to DS. The ST is analogous to a remote track in NTDS link-11 capable ships [Ref. 4:p. 19].

* Amendment - Each ST that is broadcast over SL shall be routed to DS if it is consistent with the receiving ship's current SL tracks; otherwise a correlation conflict warning shall be sent to the DS. Both tracks shall be selected and routed to the DS for resolution by the system operator. The system operator must have control over track conflicts, MP conflict resolution recommendations may be forwarded to the DS to assist the operator.

4. Requirement # 4

In the absence of communication failures, a concurrent allocation of the same SL track id to two inconsistent tracks by different ships shall be resolved [Ref. 4:p. 20].

5. Requirement # 5

In the absence of communication failures, a concurrent assumption of reporting responsibility for the same SL track shall be resolved [Ref. 4:p. 20].

6. Requirement # 6

A base unit MP shall route to AL those SL tracks with ids that are designated by DS (with TA messages) for reporting over AL (irrespective of which ship has reporting responsibility). This routing shall persist until a TA message retracts it, or the track is dropped [Ref. 4:p. 20].

7. Requirement # 7

MP shall forward outgoing directives to their destination units over the appropriate links [Ref. 4:p. 21].

8. Requirement # 8

If a reply to an outgoing directive is overdue then the MP shall re-issue that directive if the destination unit is still a link participant. Re-issuance shall continue every six link cycles thereafter until a reply is received, the directive is cancelled, or the destination is no longer a link participant [Ref. 4:p. 21].

9. Requirement # 9

On receipt of RD(unit) from the DS, MP shall send to DS each unresolved directive from Ownship to the unit specified [Ref. 4:p. 21].

10. Requirement # 10

An incoming directive from SL to Ownship (OS) shall be routed to DS; other incoming directives shall be deleted [Ref. 4:p. 21].

*Amendment - All incoming directives from SL to Ownship, or other PUs shall be routed to DS. The option to receive directives sent to other ships and aircraft should be retained by the DS and its operators, since information sent to other units in the link helps the commander to compile an overall battle picture.

11. Requirement # 11

The MP shall prompt the DS (the station operator) for response to an incoming directive before its deadline has passed [Ref. 4:p. 21].

12. Requirement # 12

On receipt of a RD(Ownship) from the DS the MP shall send to DS each unresolved directive to Ownship [Ref. 4:p. 21].

13. Requirement # 13

Each PU message from DS shall be routed to the appropriate ship or aircraft link [Ref. 4:p. 21].

14. Requirement # 14

Each incoming PU message from SL shall be routed to DS [Ref. 4:p. 21].

15. Requirement # 15

Each incoming US message shall be routed to DS [Ref. 4:p. 22].

16. Requirement # 16

On receipt of a request status(unit) (RS(u)) message from DS, the most recent PU and US messages from the unit shall be sent to DS [Ref. 4:p. 22].

17. Requirement # 17

MP shall route to DS the link-status (SLS and, for base units an ALS) reports on communication errors and timeouts from participating SL or AL units [Ref. 4:p. 22].

18. Requirement # 18

Each track report in a packet received from a ship or aircraft shall be processed or discarded within one-link cycle of its arrival at the MP. That is, before another updated or new track report can arrive from the same unit [Ref. 4:p. 22].

19. Requirement # 19

SL and AL polling-cycle durations shall be at least 0.5 second [Ref. 4:p. 22].

20. Requirement # 20

Each DT message from DS to MP shall be transmitted over SL within two link cycles of its arrival at MP unless there are more than 255 words of message data to be transmitted in the next outgoing packet; in that event queued messages shall be deleted or replaced in order of priority until the next transmission opportunity.

The message handling priorities, in descending priority are [Ref. 4:p. 22]:

- ISL, IAL, SLS and ALS.
- Incoming and outgoing directives.
- Outgoing messages (from base units) to remote units (aircraft).
- Incoming SL and AL track reports.
- Outgoing SL track reports
- All other messages (including responses to directives).

*Amendment - The message priorities should be amended as follows:

- ISL, IAL, SLS and ALS.
- Hostile Quick Response Tracks (such as pop-up missiles). These targets shall be identified by a special field in the track descriptor.
- Incoming and outgoing directives, including responses to directives.
- Outgoing messages (from base units) to remote units (aircraft).
- Incoming and outgoing hostile and unknown DS, SL and AL track reports.
- Incoming and outgoing friendly SL and AL track reports.
- All other messages.

21. Requirement # 21

Each incoming SL or AL track report shall be processed and routed to DS within five (SL or AL) cycles of its observation time; otherwise it shall be discarded [Ref. 4:p. 23].

22. Requirement #22

No more than 50 DT messages/second will arrive at the MP from the DS. This assumption places an upper bound on the minimum calling period of the module responsible for forwarding the messages over the link [Ref. 4:p. 22].

C. MODULE DECOMPOSITION

Based on the system requirements presented, it is possible to construct a module decomposition diagram, consisting of modules that are responsible for implementing these requirements. Each lower level module is responsible for implementing the higher level module requirements. This decomposition is illustrated at Figure 2.1.

The PSDL language has been used to prototype this system and the detailed description of each module, with its corresponding PSDL prototype, shall be discussed.

1. C³I System

This module is responsible for the complete implementation of the system described. This is the highest level module, and includes the MP, DS, AL and SL modules. This high level module has been included only for completeness, the main focus of the prototype shall be on the MP module. The data streams of interest are those emanating from, and terminating on the MP module. These data streams are either input from, or output to, the DS, SL and AL modules. There are no real-time constraints, since the module has to perform the high level goal of providing situation assessment and advice to a C³I system operator. The prototype is represented at Figure 2.2.

Since this is the highest level module, it is responsible for implementation of all the requirements discussed.

2. Message Processor

This module, identified as the OPERATOR in the PSDL description at Figure 2.3, is responsible for meeting the general requirements of managing:

- Radar Tracks,
- Directives, and
- Participating Units.

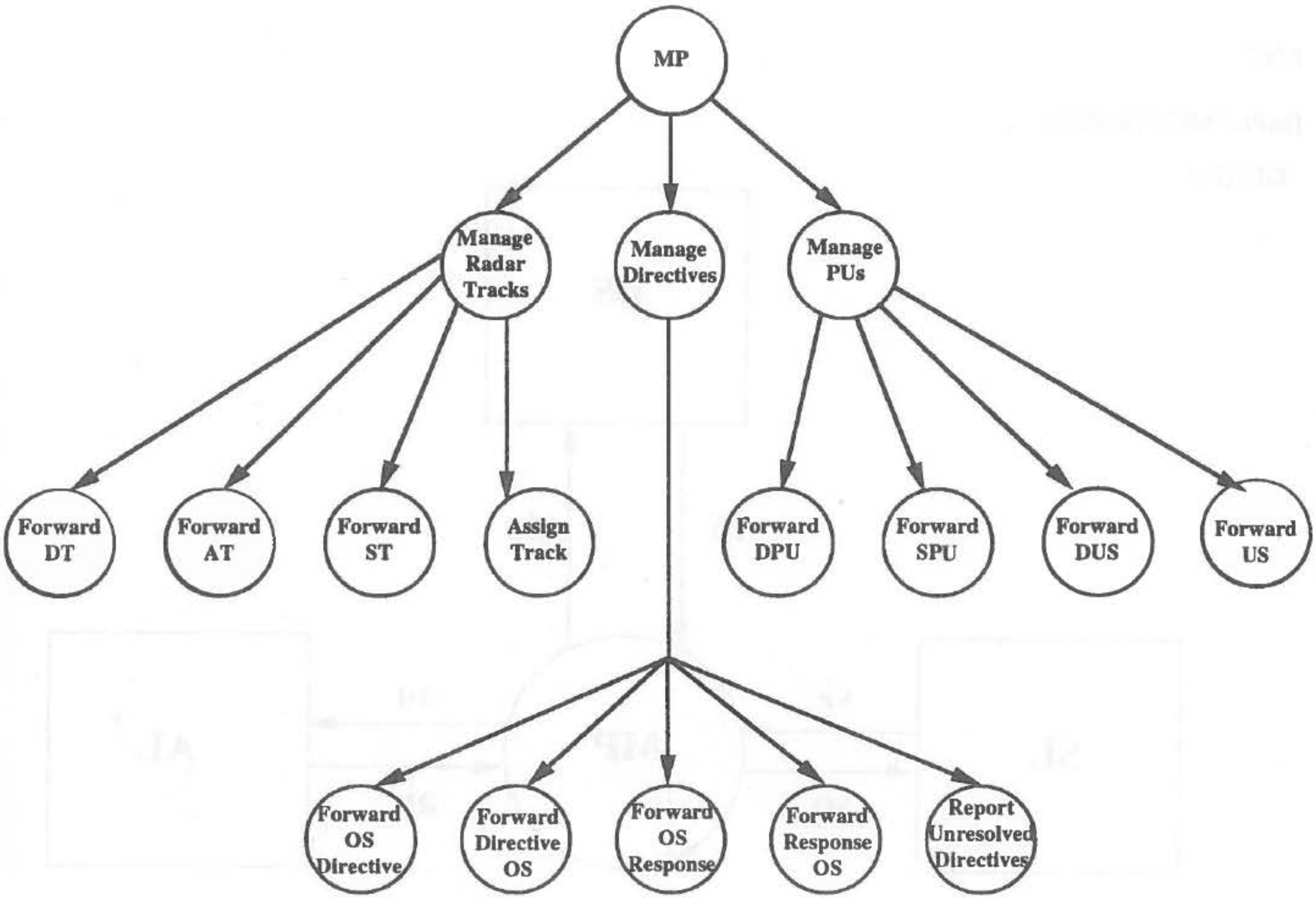


Figure 2.1 - C3I Module Decomposition

OPERATOR c3i_station

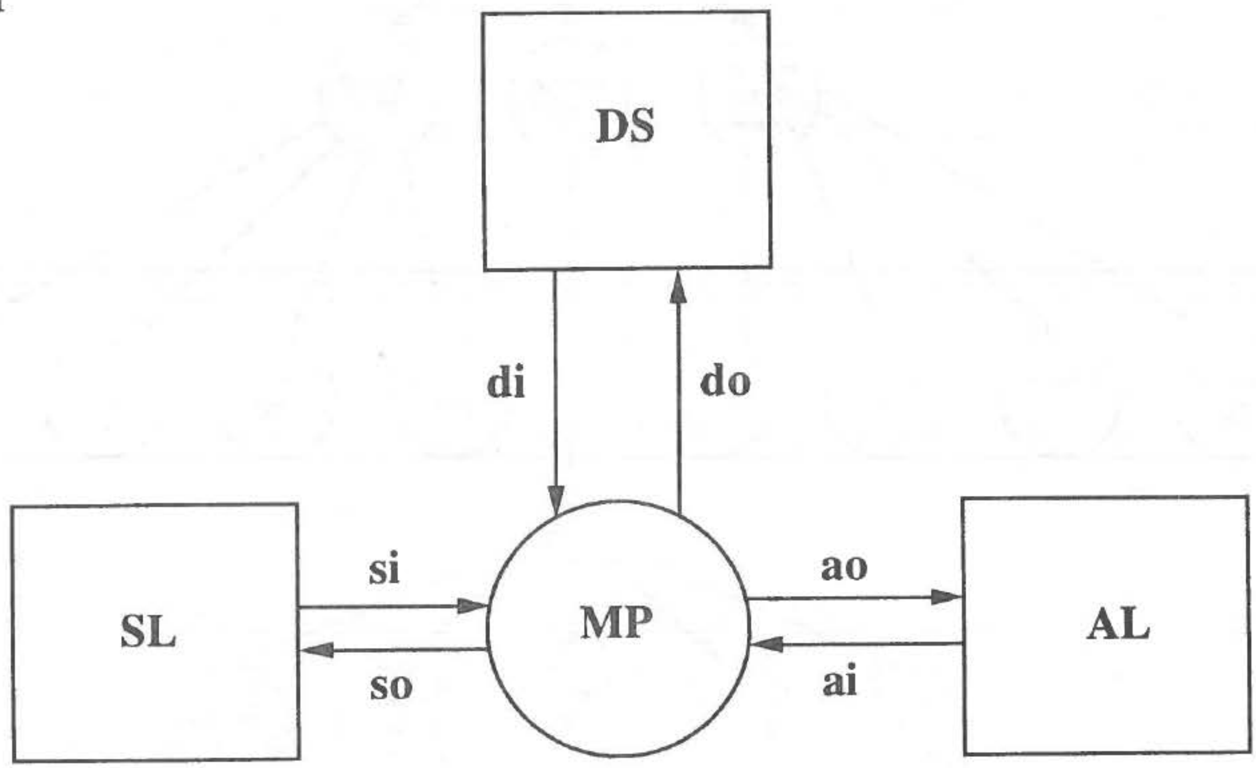
SPECIFICATION

DESCRIPTION {Provides situation assessment and advice to a c3i operator}

END

IMPLEMENTATION

GRAPH



DATA STREAM di, do, si, so, ai, ao: streams

DESCRIPTION {DS: direction system,
MP: message processor,
SL: ship link,
AL: aircraft link}

END

Figure 2.2 - PSDL Description of C3I Station

Consequently there are three sub-modules, each is responsible for meeting these general requirements. The input/output to/from the MP is in the form of data streams. There is one input/output data stream for each of the modules identified in the C³I system module. These data streams may be decomposed into specific data streams for each of the sub-modules within the MP.

The information contained in each of the streams is module specific. The sub-module MANAGE_RADAR_TRACKS is concerned with the assignment of track numbers, the management of the track database and resolution of conflicting track ids for example. Consequently it will only receive data streams that are concerned with tracks:

- dt - a track report from DS.
- at - a track report from AL.
- st - a track report from SL.
- dta - a track assignment from DS.

and will output reports for tracks:

- mdt - a track report from MANAGE_RADAR_TRACKS to the DS module.
- mat - a track report from MANAGE_RADAR_TRACKS to the AL module.
- mst - a track report from MANAGE_RADAR_TRACKS to the SL module.

Each of the data streams output from the three sub-modules is "bundled" to a composite data stream from the MP module. A similar "bundling" is applied to the input data streams.

The MP module has no timing constraints, other than the general requirement for timely execution of the program. The specific timing constraints are hidden within each of the three sub-modules. The MP module is responsible for implementation of all the requirements at section 2.B .

3. Radar Track Manager

This module is responsible for processing all SL and AL tracks. This module maintains the database for up to 256 tracks of interest and is responsible for resolution of reporting responsibility and inconsistent track ids. Specifically this module is responsible for implementation of the

OPERATOR mp

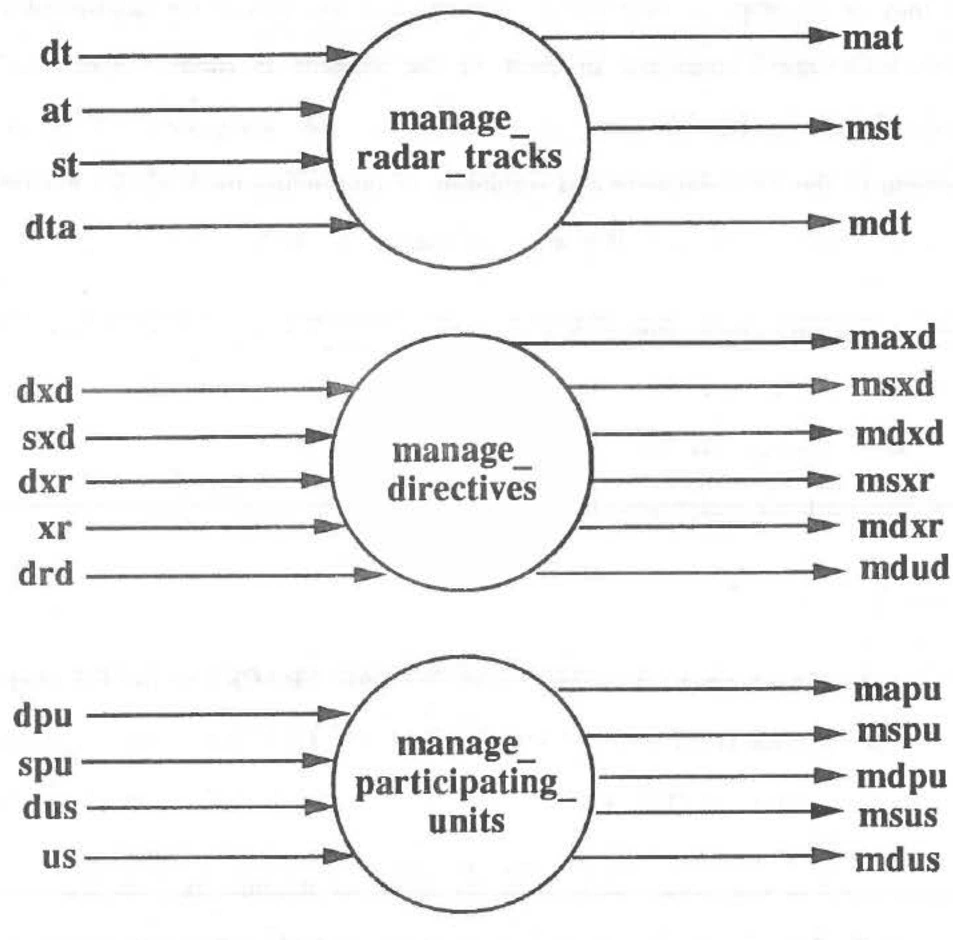
SPECIFICATION

INPUT di: streams, si: streams, ai: streams
OUTPUT do: streams, so: streams, ao: streams

END

IMPLEMENTATION

GRAPH



DATA STREAMS
di: streams, dt: track_report, dta: track_assignment, dxd: directive,
dxr: response, drd: unit_id, dpu: participating_unit, dus: unit_status,

si: streams, st: track_report, sxd: directive, xr: response,
spu: participating_unit, us: unit_status,

ai: streams, at: track_report, xr: response, us: unit_status

Figure 2.3 - PSDL Description of Message Processor

do: streams, mdt: track_report, mdxd: directive, mdxr: response,
mdud: set[directive], mdpu: participating_unit;

so: streams, mst: track_report, msxd: directive, msxr: response,
mspu: particpating_unit, msus: unit_status,

ao: streams, mat: track_report, maxd: directive,
mapu: participating unit

END

Figure 2.3 - PSDL Description of Message Processor Continued

following requirements:

- Requirement # 2,
- Requirement # 3,
- Requirement # 4,
- Requirement # 5,
- Requirement # 6,
- Requirement # 18, and
- Requirement # 20.

The decomposition of this module is shown in Figure 2.4. This PSDL description shows that this module is a state machine. The initial value of the state machine is the empty set (corresponding to an empty database), and the state is changed with the addition, deletion and update of tracks within the track database.

The changes to this state machine are effected through the input data streams dt, st, at and dta in conjunction with the value of the state variable sl_tracks. Thus the previous value of the machine may determine the next value when the data stream is input. The value of sl_tracks is also output and fedback to the input. This module outputs the data streams mdt, mat and mst signifying radar track reports to the DS, AL and SL modules respectively.

This module is also subject to timing constraints, imposed by the initial requirements. These requirements result in minimum calling periods and maximum response times for the sub-modules concerned. For the OPERATOR forward_dt, these times are calculated by considering the following facts:

- Requirement # 19 states that the minimum link cycle duration shall be at least 0.5 seconds.
- Requirement # 20 states that each DT message from DS shall be transmitted over the appropriate link within 2 link cycles of its arrival at the MP.
- Requirement # 22 states that no more than 50 DT messages/second will arrive at the MP from the DS.

Thus there is a total of 1000 ms to process a maximum of 50 DT messages. Therefore the minimum calling period of the module is (1000/50) ms, ie 20 ms. The maximum response time will be the same as the minimum link cycle duration of 500 ms. The module must be capable of output at least every 500 ms, if it is to satisfy the possible situation of only two link participants with few messages to transmit.

For the OPERATORS forward_at and forward_st the times were calculated by considering the following facts:

- In the worst case the modules must be capable of forwarding up to 256 tracks/3 modules (assuming uniform distribution over the DS, AL an SL modules) per link cycle. This is a maximum of 85.3 tracks per OPERATOR per link cycle. Some may argue the worst case is if all 256 tracks are to be forwarded by one of the DS, AL or SL modules. This is a highly unlikely scenario, seldom does an NTDS battle picture contain only one type of target, hence a uniform distribution is a more valid assumption.
- The minimum link cycle duration per requirement # 19 is 500 ms.
- Each track report in a packet received from a ship or aircraft shall be processed or discarded within one link cycle of its arrival at the MP, as per requirement #18.

Thus the OPERATORS forward_at and forward_st must be capable of outputing the results every (500/85.3 ms) or 5.8 ms. Thus the modules may not be called more frequently than every 5.8 ms. The maximum response time for these two modules is 500 ms (the minimum link cycle time), as described for the OPERATOR forward_dt.

The assign_track sub-module is responsible for the allocation of track numbers to all tracks passed to the MP by the DS. This module maintains the database for up to 256 tracks of interest and is responsible for all track conflict resolution. There are no time constraints upon this module other than a general efficiency constraint.

a. Latency

The module Manage_Radar_Tracks must also account for latencies due to DMA access times and Radio Frequency (RF) propagation delays. The RF latency between transmission sites and receivers may be calculated by considering:

- Maximum distance between two ships is 1024 nautical miles (nmi),
- Maximum distance between two aircraft is 850 nmi, and

OPERATOR manage_radar_tracks

SPECIFICATION

INPUT dt, st, at: track_report, dta:track_assignment

OUTPUT mat, mst, mdt: track_report

STATES sl_tracks:set[track_description],
-- state varaiable representing the radar track database
INITIALLY { }

DESCRIPTION {manages the radar track information in the message processor

END

IMPLEMENTATION

GRAPH

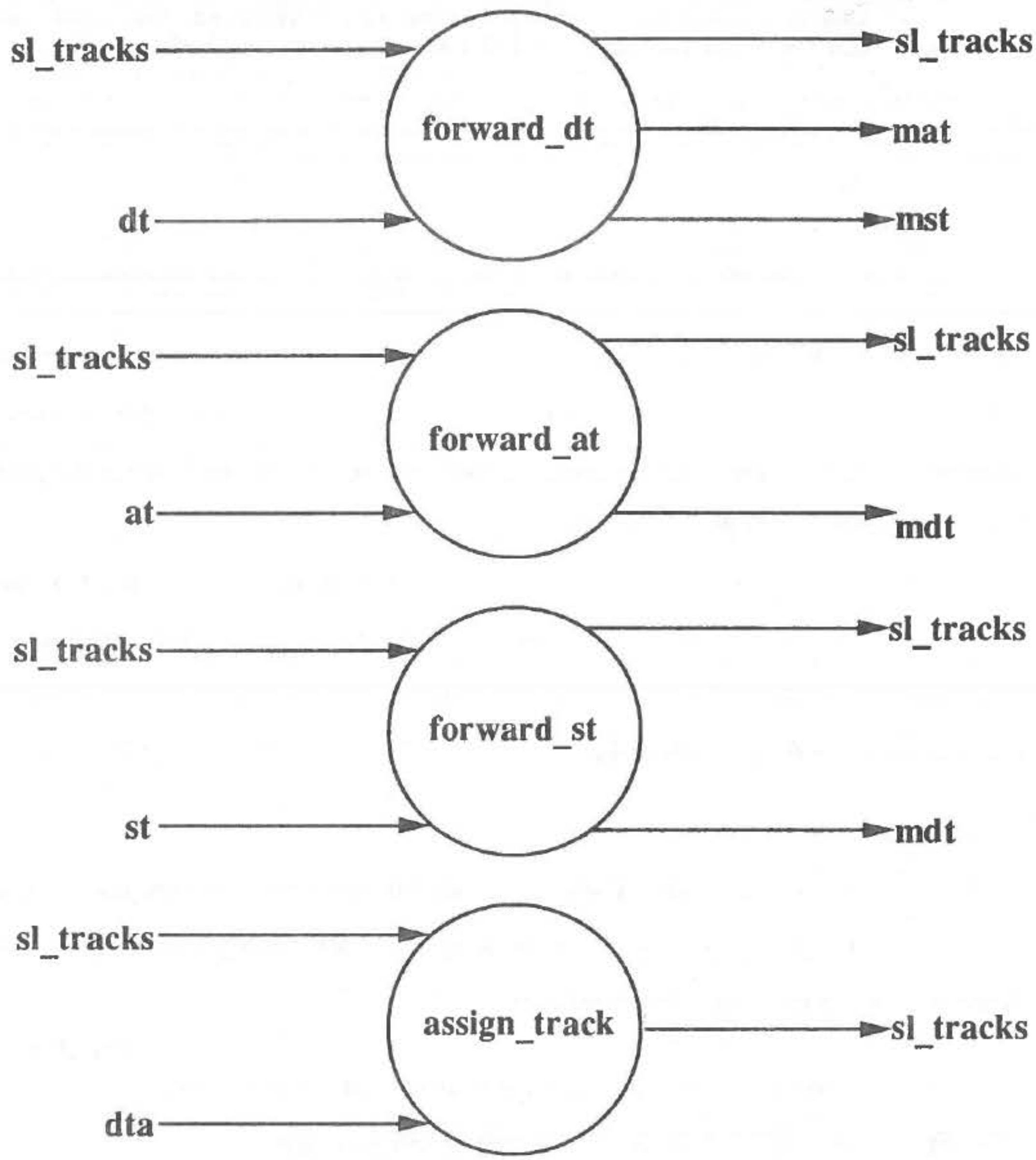


Figure 2.4 - PSDL Description of Radar Track Manager

CONTROL CONSTRAINTS

```
OPERATOR forward_dt  MINIMUM CALLING PERIOD 20 ms
                     MAXIMUM RESPONSE TIME  500 ms
    -- worst case: each DT message from DS must be
    -- forwarded over the link within 2 ship link cycles
    -- worst case: no more than 50 DT messages/second shall
    -- arrive from DS to the MP
    -- 2 ship link cycles = 1000 ms available to forward a
    -- maximum of 50 DT messages, gives minimum
    -- calling period of  (1000/50) seconds = 20 ms
    -- Minimum link cycle duration is 500ms, thus each module
    -- must be capable of completing computation within this
    -- time to handle worst case of a two ship link with few
    -- messages to forward, thus maximum response time is
    -- 500 ms
    -- allocate 500 ms to mp and 500 ms to sl
    -- Max latency considering both RF propogation delay
    -- and DMA latency is 7.25 ms for ship link and 6.2 ms
    -- for aircraft link

OPERATOR forward_at  MINIMUM CALLING PERIOD  5.8 ms
                     MAXIMUM RESPONSE TIME  500 ms

    -- system maximum track load is 256 tracks
    -- radar track manager contains 3 modules responsible
    -- for forwarding messages over link
    -- worst case assumption is that this module will have
    -- 256/3 track messages for forwarding within one
    -- link cycle of 500ms, therefore this module must
    -- be capable of forwarding one message every (500/256)x3
    -- seconds = 5.8 ms, which gives the minimum calling
    -- period of the module
    -- Maximum response time as for forward_dt
    -- Max latency for aircraft link is 6.2 ms

OPERATOR forward_st  MINIMUM CALLING PERIOD  5.8 ms
                     MAXIMUM RESPONSE TIME   500 ms

    -- Minimum response time as for forward_at
    -- Maximum response time as for forward_at
    -- Max latency for ship link is 7.25 ms

END
```

Figure 2.4 - PSDL Description of Radar Track Manager Continued

- RF propagation through air is 163,732 nmi per second.

These figures give the latency due to RF delays for each of the two available links; the ship link and the aircraft link.

The latency for each is as follows:

- For the ship link 6.25 ms (1024 nmi/163,732 nmi/sec), and
- For the aircraft link 5.2 ms (850 nmi/163,732 nmi/sec).

DMA latency is dependent on the implementation architecture, however for purposes of illustration assume that the architecture implements message passing between the MP and the SLT and ALT using 10 Mbit/second DMA channels. If the packets to be transmitted consist of 256 words, with each word consisting of 32 bits, then the latency due to DMA will be (256x32/10000000) seconds, or approximately 1 ms.

The maximum latency will therefore be the sum of the worst case RF propagation delay, that is the ship link case and the DMA latency, which results in a worst case latency of 7.25 ms. For purposes of prototyping these delays are negligible when compared to the overall ship link cycle time of 500 ms and have been ignored. As more sophisticated iterations of the prototype are produced these latencies may assume a more important role.

4. Directives Manager

The PSDL prototype for the module is shown at Figure 2.5. This module contains sub-modules responsible for the management of the different types of directives that are likely to be encountered by the MP:

- Directives from Ownship to other PUs are managed by module forward_ownship_directive,
- Directives forwarded to Ownship by other PUs are managed by module forward_directive_to_ownship,
- Responses to directives from other PUs are managed by module forward_ownship_response,
- Response to Ownship directives from other PUs are forwarded to the DS by module forward_response_to_ownship, and
- Unresolved directives, either from or to Ownship, are actioned by the module

OPERATOR manage_directives

SPECIFICATION

INPUT dxd, sxd: directive, dxr, xr: response, drd: unit_id

OUTPUT maxd, msxd: directive, msxr, mdxr: response,
mdud:set[directive]

STATES directives:set[directives]
-- state varaiable representing unresolved directives
INITIALLY { }

DESCRIPTION {manages the directives to/from the message processor}

END

IMPLEMENTATION

GRAPH

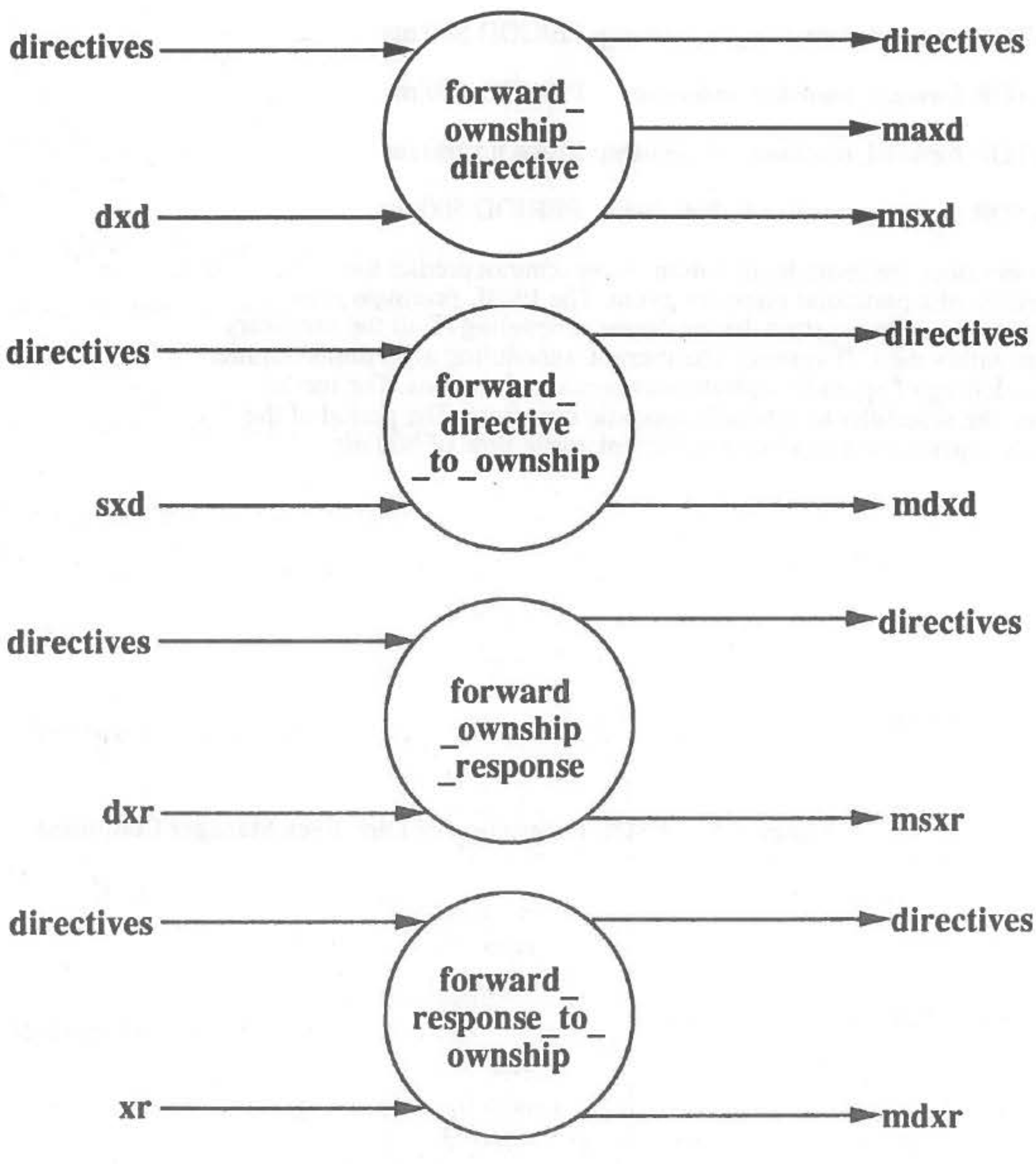


Figure 2.5 - PSDL Description of Directives Manager

CONTROL CONSTRAINTS

OPERATOR forward_ownship_directive PERIOD 500 ms

OPERATOR forward_directive_to_ownship PERIOD 500 ms

OPERATOR forward_ownship_response PERIOD 500 ms

OPERATOR forward_response_to_ownship PERIOD 500 ms

OPERATOR report_unresolved_directives PERIOD 500 ms

```
-- both operators are sporadic in nature, ie we cannot predict the
-- occurence of a particular directive event. The PSDL protoype is to
-- be a part of a larger system that performs scheduling of all the necessary
-- events within the C3I system. The current scheduling algorithms require
-- the modelling of sporadic operators as periodic functions. The model
-- allows the scheduler to schedule sporadic operators. The period of the
-- sporadic operators is modelled as the link cycle time of 500 ms.
```

END

Figure 2.5 - PSDL Description of Directives Manager Continued

report_unresolved_directives.

This module is responsible for implementing the requirements that are related to the management of directives, specifically:

- Requirement # 7,
- Requirement # 8,
- Requirement # 9,
- Requirement # 10,
- Requirement # 11, and
- Requirement # 12.

The OPERATOR manage_directives is also a state machine, similar to the Radar Track Manager. In this case the state variable is altered by the status of unresolved directives, this in turn will cause each of the sub-modules to take appropriate action to ensure re-issuance of the directive or prompt the DS operator for a response to a directive.

The OPERATORs forward_ownship_directives, and forward_directive_to_ownship are subject to the timing constraint set by the minimum link cycle time of 0.5 s (500 ms). The number of directives per link cycle is assumed to be one for the purposes of this prototype.

The OPERATORS are sporadic in nature, that is we cannot predict the occurrence of these events. Within the current support environment these sporadic operators must be modelled as periodic events, each with a defined period of the link cycle time of 500 ms. This is a necessary requirement for the scheduler algorithms within the support environment to function correctly. This may change when a larger database of scheduler algorithms is available; with algorithms that can take sporadic operators as input.

5. Participating Units Manager

For each unit in the link its current role in the link (either MU or PU) must be maintained. The PU manager is responsible for the maintenance of this data and is modelled in PSDL at figure 2.6. This module maintains the link information, namely status and participation-mode information that are obtained from Unit-Status (US) and Participating-Unit (PU) messages. These messages are transmitted by all link participants periodically.

The PU manager is responsible for the remaining requirements:

- Requirement # 1
- Requirement # 13
- Requirement # 14
- Requirement # 15
- Requirement # 16, and
- Requirement # 17.

Requirement number 19 is a link protocol requirement and is not applicable to the MP. Responsibility for this requirement rests with the modules of the C^3I station that are responsible for the Aircraft and Ship Links and attached radio transceivers.

The sub-modules within the OPERATOR manage_participating_units are:

- OPERATOR forward_dpu, the sub-module responsible for forwarding DS participating unit messages to the AL and SL links,
- OPERATOR forward_spu ensures that PU messages received over the SL are forwarded to the DS,
- OPERATOR forward_dus forwards unit status messages from the DS to SL, and
- OPERATOR forward_us is the sub-module responsible for forwarding units status messages from other PUs to the DS.

OPERATOR manage_participating_units

SPECIFICATION

INPUT dpu, spu: participating_units, dus, us: unit_status

OUTPUT mapu, mspu, mdpu: participating_unit, msus, mdus: unit_status

DESCRIPTION {manages the link participants within a node}

END

IMPLEMENTATION

GRAPH

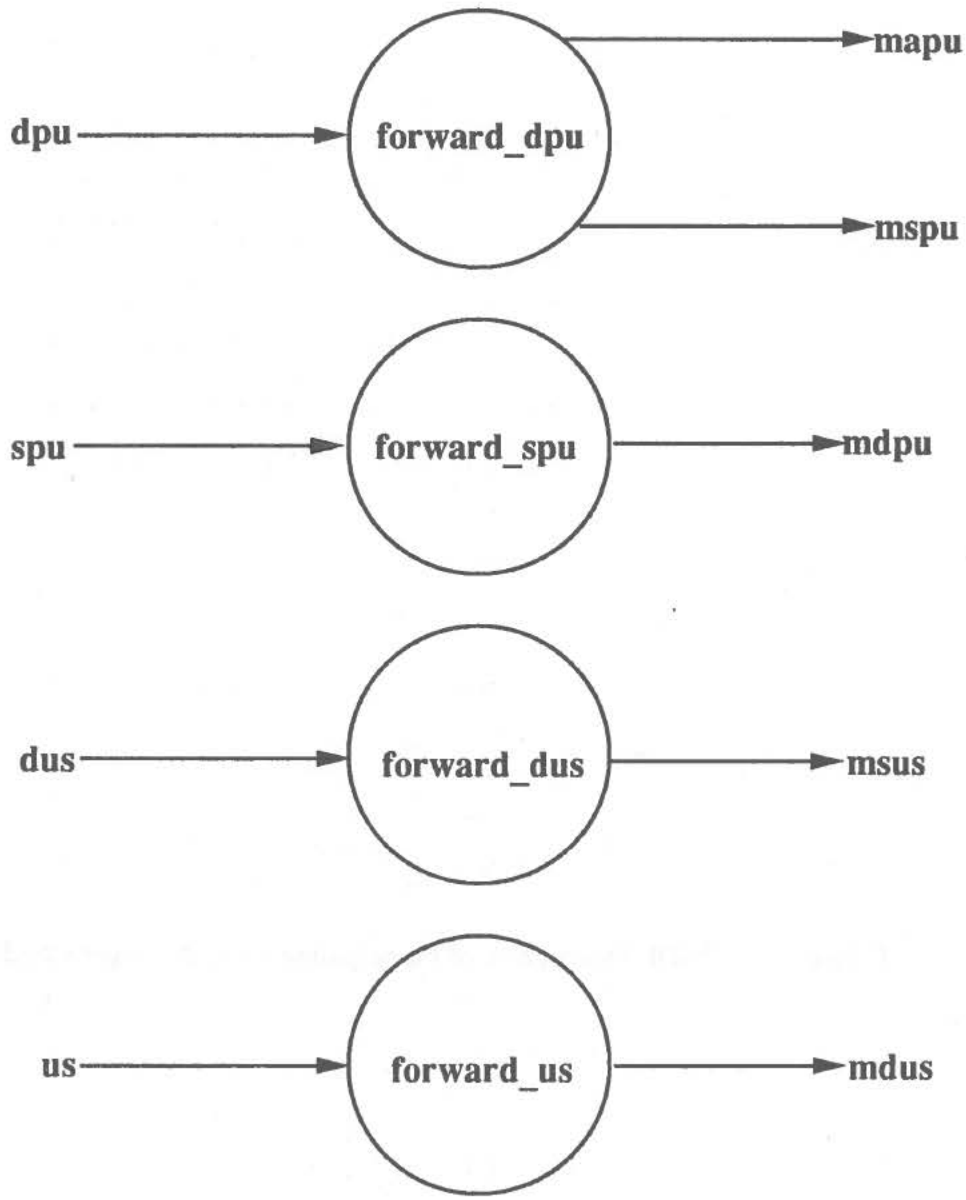


Figure 2.6 - PSDL Description of Particpating Units Manager

CONTROL CONSTRAINTS

```
OPERATOR forward_dpu PERIOD 500 ms

OPERATOR forward_spu PERIOD 500 ms

OPERATOR forward_dus PERIOD 500 ms

OPERATOR forward_us   PERIOD 500 ms

-- PU  & US information is forwarded once each
-- link cycle to  maintain communication protocols
-- the link cycle time minimum is 500ms
-- hence these modules must produce output at
-- 500 ms periods
```

END

Figure 2.6 - PSDL Description of Participating Units Manager Continued

DEFINITIONS

A. GENERAL GLOSSARY

AL	Aircraft Link
ALS	AL Status
ALT	Aircraft Link Transceiver
AT	AL Track Report
BASE UNIT	A PU with aircraft control responsibilities. Each base unit may control a maximum of 10 aircraft.
BT	Begin Transmission
C^3I	Command,Control,Communication and Intelligence
DMA	Direct Memory Access
DS	Direction System
DSP	Direction System Processor
DT	Direction System Track
ET	End Transmission
GATEWAY UNIT	Designated ship responsible for linking a node to other nodes and land based communication centers via a global satellite network.
IAL	Initialize AL
ID	Identification
ISL	Initialize SL
LS	Link Status
MH	Message Handler
MP	Message Processor
MU	Master Unit
NMI	Nautical Miles
NODE	A geographical cluster that consists of ships and aircraft with unique identifiers. Ships are within 50 nmi of one another; aircraft within 800 nmi of carriers within their nodes.
NTDS	Naval Tactical Data System
NS	Navigation System
OS	Onboard Sensor Device
OT	Operator Terminal
OWNSHIP	Unit identifier of the ship that contains the subject MP instance.
PSDL	Prototype System Description Language
PU	Participating Unit

RD	Request Directive
RF	Radio Frequencies
RS	Request Status
SL	Ship Link
SLS	SL Status
SLT	Ship Link Transceiver
ST	Ship Link Track Report
TA	Directs ST with specified ST identifier to AL (base units only)
US	Unit Status
X	X(Combat,Velocity,Destination)
XD	X Directive from Ownship
XR	X Response from Ownship

B. C³I SYSTEM DEFINITIONS

Node	A geographical cluster which consists of ships and aircraft. Ships are assumed to be within 1024 nmi of each other, while aircraft are assumed to be within 800 nmi of their aircraft carrier or base unit.
Node Limits	There shall be no more than 40 participating ships per node. There shall be no more than 10 base units per node.
Master Unit	There is at any one time a Master Unit. The current MU is the last ship that transmitted an ISL message (initializing the link), unless that ship has subsequently: • remained silent for >= 0.5 s, or •broadcast a PU message with protocol mode = MU, followed by a subsequent LS message. In the first event there is no MU until some ship broadcasts an ISL.
Base Unit	Ships that are capable of controlling aircraft. Base units are assumed to be capable of controlling up to 10 aircraft. Each Base Unit issues PU messages that insert, delete, or change the protocol level of remote units (aircraft).

Gateway Unit	The designated ship that is responsible for linking the node to other nodes and land based command centers.
AL-cycle	An interval during which each AL participant has an opportunity to send its messages.
SL-cycle	An interval during which each participating ship has an opportunity to send outgoing messages over the SL.
SL Track Ids	SL track ids are integers in the range 0..255; 0 is the null id. An SL track is free provided that it is non-null and either: • no track report with that id has arrived since link initialization, or • its track has not been updated for two or more SL link cycles. otherwise it is allocated and the last received SL track report with that id is current.
Track Consistency	Two DT or ST messages are consistent provided that either: • one or both track ids are null or they have distinct track ids, • they have the same SL track id and sender, or • they have the same SL track id, different senders, and are compatible. Otherwise they are inconsistent.
Compatible Tracks	Two ST or DT messages are compatible if the probability that they are tracks of the same object is >= 0.95; otherwise they are incompatible.
Overdue Directives	A directive is overdue when its sender has not received a reply, has not issued a cancellation and six or more link cycles (of the link over which it was sent) have elapsed since it was issued.

Unresolved Directives — A directive is unresolved if its sender has not received a comply or refusal and the directed unit is still a link participant.

C. TYPES

COMBAT_DIRECTIVE	is	subject	: track_id,
		action	: {identify,engage,fire disengage}
DIRECTIVE	is	receiver	: unit_id,
		action	: union[c: combat_directive, v: velocity_directive, p: position_directive],
		polarity	: boolean
PARTICIPATING_UNIT	is	id	: unit_id,
		role	: {full_response,limited_response,no_response, new_unit,deleted_unit,master_unit}
POSITION_DIRECTIVE	is	position	: vector[3,real]
RESPONSE	is	sender	: unit_id,
		receiver	: ship_id,
		action	: union[c: combat_directive, v: velocity_directive, p: position_directive],
		comply	: {Acknowledge,Affirmative,Negative}
TRACK_ASSIGNMENT	is	id	: track_id,

		polarity	: boolean,
TRACK_REPORT	is	id	: track_id,
		source	: unit_id,
		position	: vector[4,real],
		description	: track_description
UNIT_STATUS	is	id	: unit_id,
		resource	: resource_id,
		quantity	: nat
VELOCITY_DIRECTIVE	is	velocity	: vector[3,real]

D. DATA STREAMS

ai	AL, Input Data Streams
ao	AL, Output Data Streams
at	AL, Track Report
di	DS, Input Data Streams
do	DS, Output Data Streams
dpu	DS, Participating_Unit
drd	DS, Unit_Id
dt	DS, Track_Report
dta	DS, Track_Assignment
dus	DS, Unit_Status
dxd	DS, Directive
dxr	DS, Response
mapu	Manage Participating Units, Participating_Unit
mat	Manage Radar Tracks, Track_Report

maxd	Manage Directives, Directive
mdpu	Manage Participating Units, Participating_Unit
mdt	Manage Radar Tracks, Track_Report
mdud	Manage Directives, Set[Directive]
mdxd	Manage Directives, Directive
mdxr	Manage Directives, Response
mspu	Manage Participating Units, Participating_Unit
mst	Manage Radar Tracks, Track_Reports
msus	Manage Participating Units, Unit_Status
msxd	Manage Directives, Directive
msxr	Manage Directives, Response
si	SL, Input Data Stream
so	SL, Output Data Stream
spu	SL, Participating_Unit
st	SL, Track_Report
sxd	SL, Directive
us	Unit_Status
xr	Response

LIST OF REFERENCES

[Ref. 1] Naval Research Advisory Committee, *Next Generation Computer Resources*, February 1989.

[Ref. 2] Luqi, Berzins, V., *Rapidly Prototyping Real_Time Systems*, IEEE Software, September 1988.

[Ref. 3] Luqi, Berzins, V., Yeh, R.T., *A Prototyping Language for Real_Time Software*, IEEE Transactions on Software Engineering, Vol. 14, No. 10, October 1988.

[Ref. 4] Brown, T., *Design of a Naval C^3I Message Processor from Requirements*, Kestrel Institute, October 1988.

DISTRIBUTION LIST

(1)	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
(2)	Dudley Knox Library Code 0142 Naval Postgraduate School Monterey, CA 93943	2
(3)	Center for Naval Analysis 4401 Ford Avenue Alexandria, VA 22302-0268	1
(4)	Director of Research Administration Attn: Prof. Howard Code 012 Naval Postgraduate School Monterey, CA 93943	1
(5)	Chairman, Code 52 Computer Science Department Naval Postgraduate School Monterey, CA 93943-5100	1
(6)	Chief of Naval Research 800 N. Quincy Street Arlington, Virginia 22217	1
(7)	National Science Foundation Division of Computer and Computation Research Attn. Tom Keenan Washington, D.C. 20550	1
(8)	Naval Postgraduate School Code 52Lq Computer Science Department Monterey, CA 93943	100