AD-A014 583

STRUCTURED ANALYSIS MODEL FOR NAVAL TELECOMMUNICATIONS PROCEDURES USERS MANUAL NTP 3

Clare Feldmann, et al

SofTech, Incorporated

Prepared for:

Naval Research Laboratory

July 1975

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

NRL Memorandum Report 3086

Structured Analysis Model for Naval Telecommunications Procedures Users Manual NTP 3

CLARE FELDMAN, KEN SCHOMAN,
AND DICK SNYDER

SofTech, Inc.
Waltham, Mass.

July 1975

NAVAL RESEARCH LABORATORY
Washington, D.C.

120

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Memorandum Report 3086	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) STRUCTURED ANALYSIS MODEL FOR NAVAL TELECOMMUNICATIONS PROCEDURES USERS MANUAL NTP 3		5. TYPE OF REPORT & PERIOD COVERED An interim report on a continuing NRL problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Clare Feldmann, Ken Schoman, and Dick Snyder		8. CONTRACT OR GRANT NUMBER(s) N00173-75-C-0503
9. PERFORMING ORGANIZATION NAME AND ADDRESS SofTech, Inc. 460 Totten Pond Road Waltham, Mass. 02154		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem B02-18 Project XF21-241-021-K211
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Naval Electronics Systems Command Washington, D.C. 20360		12. REPORT DATE July 1975 13. NUMBER OF PAGES 119
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Research Laboratory Washington, D.C. 20375		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES This project was carried out under the direction of Dr. Jorge Rodriguez. The contract manager was Dick Snyder; the contract team included Ken Schoman and Clare Feldmann.		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Structured analysis Activity model Data model Modularity Telecommunications		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Structured Analysis is a technique devised by SofTech to model processes. This report contains a Structured Analysis model depicting the activities and data involved in the production of naval messages using the procedures defined in the document, Naval Telecommunications Procedures Users Manual NTP 3.		

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

PREFACE

This report is an example of the use and applications of Structured Analysis, a technique devised by SofTech to model processes from operating systems for large-scale computers, to simple applications programs to message processing procedures in the Navy. To evaluate this technique, we requested SofTech to construct a Structured Analysis model of the process described in the document, Naval Telecommunications Procedures Users Manual NTP 3. This report contains the model and a complete description of how one reads and understands the Structured Analysis diagrams which comprise the model.

We feel that Structured Analysis may be useful in many areas. In particular, its use in system design may be valuable in achieving reliable software.

We would appreciate receiving comments, suggestions and possible uses for this technique.

Please send your responses to:

Helen Carter
Naval Research Laboratory
Code 5403D
Washington, D. C. 20375
(202) 767-3249

TABLE OF CONTENTS

Section 1

INTRODUCTION

The material contained within this document is a complete Structured Analysis model of the information in the document entitled Naval Telecommunications Procedures Users Manual NTP 3. This model has been produced for the Naval Research Laboratory to provide an in-depth example of the use of Structured Analysis. The model depicts the activities and data involved in the production of naval messages using the procedures defined in the aforementioned manual.

Structured Analysis is a new and practical way of coping with complexity. It is a way of getting problem analyses down on paper in a complete and comprehensive way, and in a form readily understood by people of all technical backgrounds. It is an orderly process of "top-down" decomposition of a subject into its constituent parts. Any chosen aspect of the subject can be exposed to whatever level of detail is necessary without losing comprehendability.

Structured analysis is a hierarchic discipline. The hierarchies represent levels of abstraction. A top-down view will see the forest before it sees the trees. Such a level of abstraction is useful for certain things, such as the exercise of management. At lower levels of abstraction the trees become important; here is where the work of the world generally takes place.

Both views, forest and trees, are necessary for completeness and rigor. Structured analysis provides that completeness and rigor in a single well- integrated discipline.

The model in this document was made from the viewpoint of the drafter of a naval message. This viewpoint shows best the inter-relationships of the various pieces of data in a naval message. At the same time the act of actually typing a naval message is also shown in the model. Hence the reader gets a graphical picture of the creation of the various pieces of information found in a naval message as well as a picture of the placement of this information on a naval message form.

The full Structured Analysis model consists of a model of the activities involved in the creation of naval messages and a model of the data which makes up this creation process. The final step in the modeling process is the tying together of these two models to form the single complete Structured Analysis model. The purpose of the tie is twofold. The primary purpose is to verify the correctness of the model. The tie serves to point out inconsistencies between the activity and data models such as data being used by an activity box on an activity diagram without that activity being shown as accessing that piece of data on the data diagrams. The secondary purpose is to allow one to easily see all activities which reference a piece of data without scanning the activity diagrams for all references to that data. The tie is done by methodically classifying each arrow in one model into the boxes of the other model, and then writing the node numbers of the selected boxes onto the arrows. When the annotation of all arrows has been completed on all diagrams in both models, the diagrams are said to be "tied together".

Section 2 contains a description of how one reads Structured Analysis diagrams. Section 3 contains the completed activity model, Section 4 contains the data model, and Section 5 contains the two models showing the annotations of the tie process.

Section 2

READER'S GUIDE TO STRUCTURED ANALYSIS

This section is composed of excerpts of other documents on Structured Analysis. Its intent is to provide instruction in reading and understanding Structured Analysis diagrams.

Section 1
READER GUIDE PURPOSE AND ORGANIZATION

What is a "Structured Analysis Model"? What is its form and what is it trying to accomplish? What is the best way to go about reading and understanding an SA* model? This document attempts to answer these questions.

The discussion begins with some basic concepts that are essential to understanding SA models. Turning to the mechanical details, various aspects of the diagram graphics conventions and form are discussed, followed by reading rules that have been found helpful in quickly gaining an understanding of the meaning being conveyed. Therefore, after studying this document, a student should be able to read any SA model in his area of expertise.

Since Structured Analysis is so widely applicable to many varied fields, an example in any of the fields would not be of universal interest and, in fact, would tend to generate interest in the technical content of the example, and thus distract the reader from the point being illustrated. Therefore, throughout the document, simple examples based upon a non-technical subject (which all readers should be familiar with) are used to illustrate each point. Since it is non-technical in nature, the reader must envision how the point being illustrated relates to his particular field.

---

*The abbreviation "SA" is used for the name "Structured Analysis" throughout this document

Section 2

BASIC CONCEPTS

There are certain basic concepts that underlie the Structured Analysis methodology which are essential to understanding SA models. The most basic is the concept of modularity. Using a blueprint-like organization, module diagrams are created by SA analysts (called "authors"), which show a "top-down" decomposition of the system being analyzed, from both an activity and a data-oriented viewpoint. These basic concepts are further expanded and described below.

2.1 Modularity

Everyone these days is familiar with plug-in "modules" used in electronic circuits. Each such module serves a well-defined function, and may be plugged into a circuit to replace an equivalent module, so long as its interfaces or connections match in shape and electronic characteristics. A new module that replaces an old module may not do its job in exactly the same way as its predecessor, but as long as its effect is the same, the total circuit will still perform satisfactorily. In fact, it may represent an improvement over the old module, performing its job more efficiently (requiring less power), or introducing less noise into the circuit.

Structured Analysis uses the concept of modularity to analyze complex system structures in a "divide and conquer" approach. By successively breaking down the subject matter into more and more, smaller and smaller, well-defined modules, the analyst finally arrives at small enough pieces so that the function of each individual module can be easily understood and its interfaces to other modules clearly seen. Thus, the complex system that was not capable of being understood in its total view can be well understood by seeing each of its modules and how they fit together. In fact, once this modular breakdown is created, replacement modules can be designed and "plugged in" to improve the performance of the total subject being modeled.

2.2 Blueprint-Like Form of Model

It is not sufficient merely to break down the complexity of a large system into small enough modules so that each can be understood. The problem is that it is very difficult for the human mind to comprehend the complex interfaces in such an elaborate picture and to see the total effect introduced by changes. Anyone who has tried to understand a large document describing each module in text form, or the bewildering maze presented by a wall-size chart showing the total operation of a complex system, will attest to this problem.

Structured Analysis tackles this interface structure problem much like the manufacturing industry did with the technique called blueprints. In addition to well-defined standards and conventions for employing drawings, text, and numbers, blueprints present many <u>different prints</u> or <u>views</u> of the subject. Some give high-level overviews. Some give alternate cross-sectional views. A series of "details" successively remove layer after layer of abstraction until the fine structure is exposed. But this fine structure is understood only because the reader has been exposed to each of the successive layers of detail. He is able to comprehend the entire subject by understanding each piece and how these pieces fit together, to whatever level he has examined the blueprints.

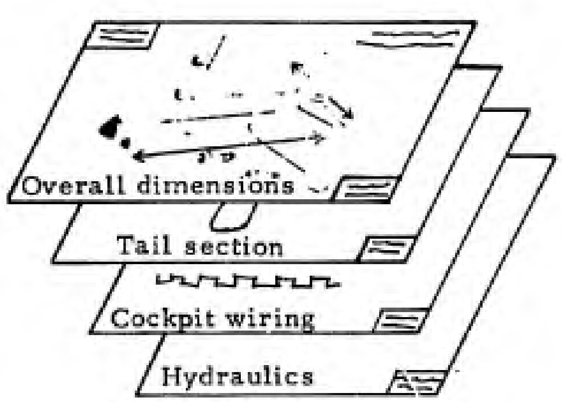


Figure 2.1 Typical Blueprint Breakdown

2.3 Top-Down Views

The principle of "top-down" system design gives an overall organization to the set of SA blueprints. It is based upon viewing a system from the highest-level, most general viewpoint, and then breaking down this view into successively finer and finer levels of detail. Structured Analysis adopts this top-down approach, and further refines the rules by the principle that any specific blueprint view is composed of not less than three and not more than six pieces. The top-down "decomposition" procedure thus presents first a very general view that is understandable by a wide class of readers--by top management down to the individual workers. As successively finer levels are shown, more and more of the detail is exposed, and the detail needed for complete understanding is attained. Note that various levels of the management personnel need not become familiar with these lower levels, since they may only be concerned with overview or briefing-levels of detail. It should be noted, however, that since the lower levels are derived directly as expansions of detail from the higher levels, all views are rigorously tied together into a single model and thus even top-level overviews represent the true picture of what is contained at lower levels, with nothing omitted or added. Meanwhile, the 3-6 rule ensures that neither too little additional detail to be meaningful nor too much to be understandable is introduced by each lower level presented.

The Structured Analysis top-down decomposition model is represented by a series of diagrams as shown in Figure 2.2. In Figure 2.2, the 3-6 details at each level are represented as numbered boxes, and we see that each individual detail box is decomposed into another diagram at the next lower level. Each diagram is called the "parent" of its 3-6 "children" diagrams, and each diagram is numbered in a Dewey-decimal manner, based upon the box number assigned in its parent view (e.g., diagrams 211, 212, and 213 are children of diagram 21). Each diagram in this structure is called a "node", and a complete layout of all nodes of a model in the form of a "table of contents" diagram is called a "node diagram" for the model (see Section 4.1).

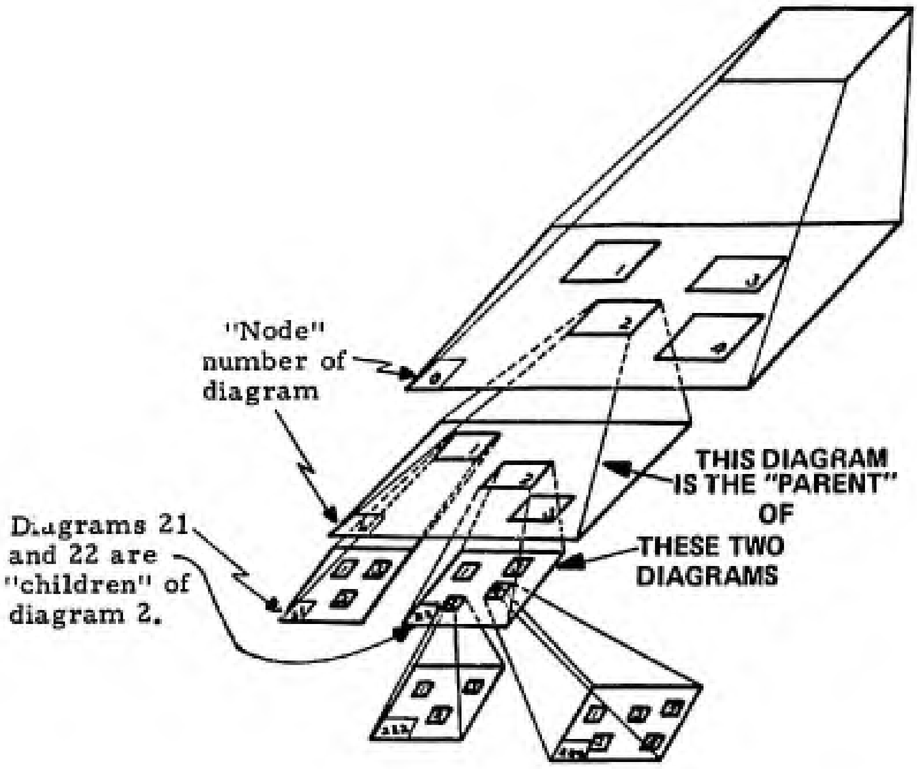


Figure 2.2   Structured Analysis Model

A complete analysis may require additional diagrams which do not fit into this top-down decomposition. These additional diagrams are used to highlight some particular point which may be obscure in a complex diagram. For example, a diagram may be used to show the major feedback paths in a sub-system. These extra SA diagrams are called "FEOs", where this acronym stands for "For Exposition Only", since they expose some aspect of a system but are not part of the top-down diagram structure that makes up the basic model.

2.4 Activity and Data Aspects

To describe any complex system completely, there are two basic aspects which must be covered: the functions performed by the system and the things upon which the functions act. In Structured Analysis, we call the functions "activities" and the things "data". (Note that the term "data" as used here covers more than the term "data" as used in computer applications or in mathematics--it includes anything which is not an activity.)

In Structured Analysis, we decompose a system once based upon its activities, and again, in a separate model, based upon its data. Thus, the same subject matter is examined twice: once from each aspect. This results in two separate top-down models. As a final step, the two models are tied together to show how details of one aspect correspond to details in the other aspect.

Actually, as will be seen in Section 3.3, each aspect presents a complete analysis of the subject. In the activity model, activities are emphasized and data items appear as the interfaces between them, whereas in the data model, data is emphasized and activities are included as interfaces. Therefore, each model contains both data and activities. However, the emphasis is quite important in determining the specific decomposition into levels of detail, and the reader will soon see how by building these two separate models with these two complementary emphases, we achieve a much richer understanding of the subject than is afforded by a single model.

Section 3
DIAGRAM SYNTAX

Structured Analysis diagrams are composed of boxes and arrows. Many box-and-arrow "languages" exist (flow charts, pert charts, block diagrams, etc.), but the conventions chosen for Structured Analysis are very carefully selected to provide very simple yet expressive component parts for the purpose of describing complex systems. In other words, the box and arrow conventions provide a very natural language for expressing system activity and data modules without becoming so complex as to make them difficult to read.

For connecting between top-down levels and for same-level module interfacing, the boxes and arrows of Structured Analysis correspond closely to the plug-in circuit modules discussed in the overview (Section 2.1). The activity or data described by the box can be removed and replaced by an equivalent module, much like electronic circuit module replacement (see Figure 3.1).

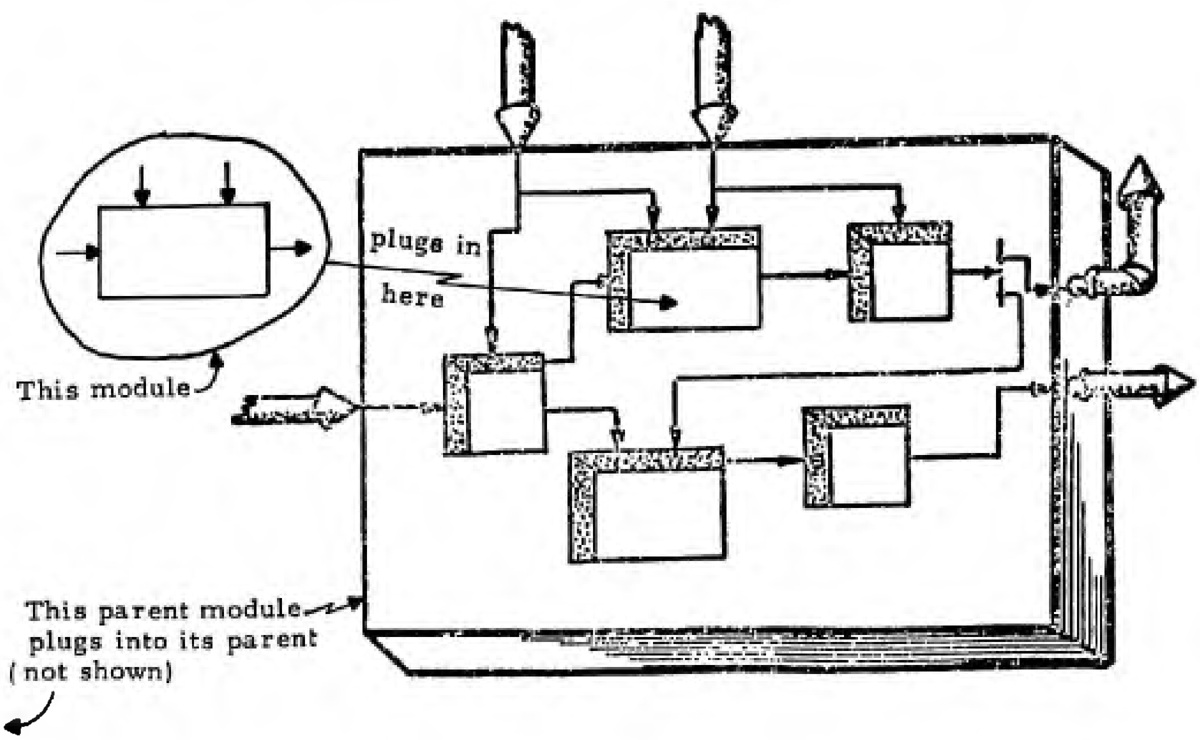


Figure 3.1 Plug-In SA Module

The following sections describe the SA box and arrow convention details, first discussing individual boxes, then box connection on a single diagram, and lastly, boxes connected across levels of decomposition.

3.1 Individual Box Conventions

3.1.1 Box Shape and Meaning

The four-sided box is the only shape of box used in Structured Analysis. Inside the box, the name of the activity (activity model) or data item (da*a model) is written. In the activity case, the name contains a verb (to express action) and in the data case it is a noun or noun phrase (to express a data item).

Figure 3.2 illustrates the two forms.

Figure 3.2 Activity and Data Box Titles

3.1.2 Arrow Interfaces

Arrows are used to connect boxes on a diagram, and represent <u>interfaces</u> between boxes. The four sides of a box are each assigned a specific meaning which defines the kind of arrow which may enter or leave that side of the box. Figure 3.3 defines this box/interface arrow convention, where more than one arrow may enter or leave a single side.

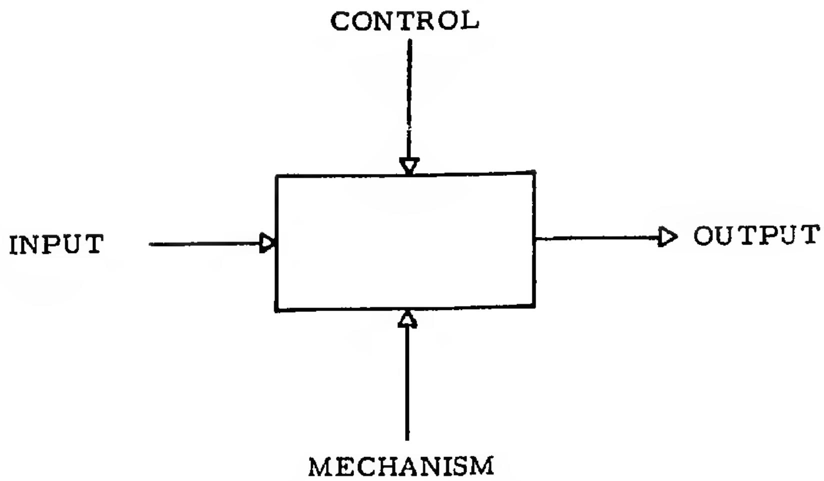


Figure 3.3 Box/Interface Arrow Definition

These rules are strictly obeyed in all SA diagrams. That is, an arrow is not permitted to leave the left side of a box, and an arrow entering the left side of a box always represents an input, etc.

3.1.3 Arrow Labels

As with box titles, titles are written along interface arrows to define the interfaces. The arrow title may simply be written nearby (above or below), or may be attached to an arrow by a "squiggle" if it would otherwise be ambiguous as to which of several arrows is intended, due to their close proximity on the diagram form. Figure 3.4 illustrates arrow label conventions.

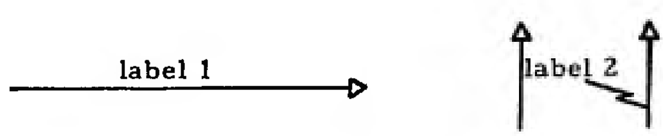


Figure 3.4 Arrow Labeling Conventions

In Figure 3.4, "label 1" labels the horizontal arrow and "label 2" labels the right-hand vertical arrow. In the latter case, a squiggle associates the label with the right-hand arrow rather than the nearby left-hand arrow.

3.1.4 Kinds of Arrows

Each of the four categories of interface arrow has a particular kind of interface it represents (the kind of thing represented by the arrow). In the activity diagram case, the interfaces between activity boxes are data items, since data is commonly passed back and forth between activities. In the data diagram case, the interfaces between data boxes are activities, since activities create and make use of data items, and therefore serve the interface role. The "mechanism" arrow is an exception to this rule, since it represents the mechanism necessary to "realize the box" (see complete definition below). In the activity case, the mechanism is the person or thing which serves as the "processor" which carries out the activity (or some portion of it). In the data case, the mechanism is the "storage device" used to hold the data item (or some portion of it).

3.1.5 Box/Arrow Convention Summary

The box interface conventions are summarized in Figure 3.5.

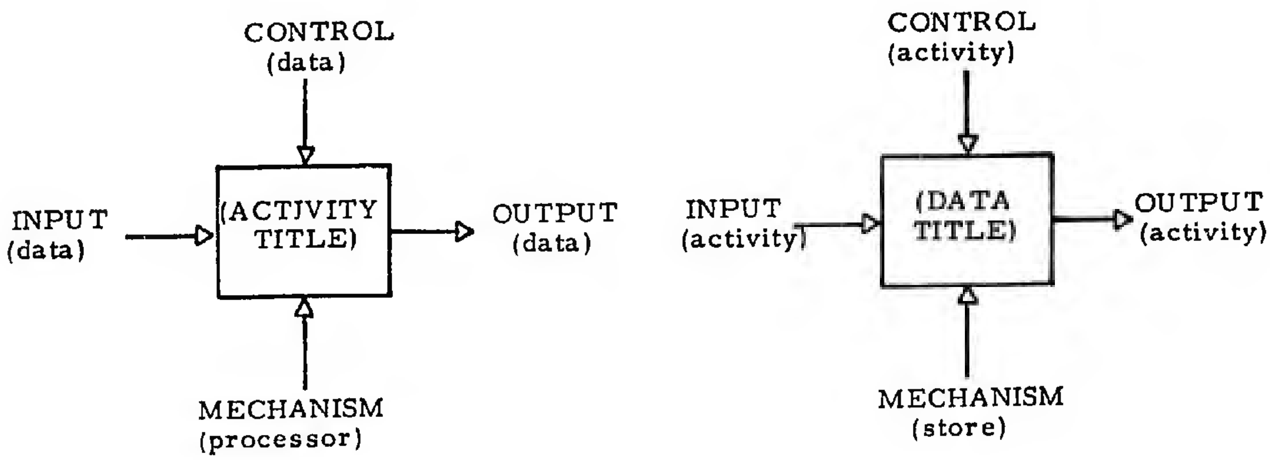


Figure 3.5 Box/Interfaces Arrow Conventions

Examples of interfaces are illustrated in Figure 3.6.

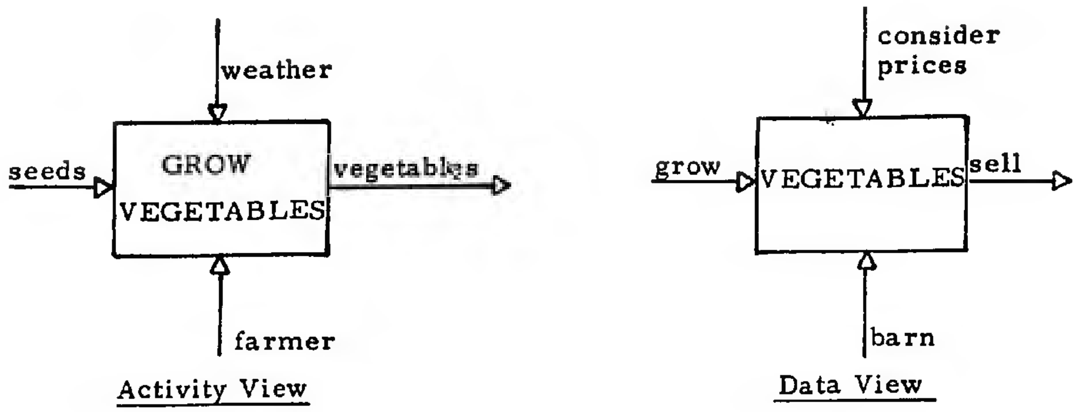


Figure 3.6 Examples of Interfaces

If you were to write equivalent sentences for the two boxes of Figure 3.6, they would say "the farmer grows vegetables from seeds subject to the control of the weather" and "vegetables are grown and kept in the barn, from which they are sold, considering current prices".

The example illustrates the usefulness of separating the concepts of input and control. There is a basic difference between an input, which (in the activity view) is converted by the activity into the output, and a control, which is never modified by the activity, but rather constrains how the activity functions in converting the input into the output. The control thus serves a completely different role than the input, and this distinction is important to the basic understanding of the system's operation.

In actual practice, a single interface arrow may include aspects of both input and control. In this case, the rule is that

> "an arrow is always considered as a control unless its obviously serves only as an input"

The reader should therefore not be too hasty to judge that an interface arrow is in error if he feels the input/control convention is wrong. He should attempt to consider the arrow as intended by the author before making too hasty a judgment.

3.1.6 Arrow Category Definitions

To summarize the arrow conventions, the four categories are defined as:

Activity Diagrams:

*Input: Data transformed by the activity into the output.

*Output: Data created by the activity.

*Control: Data used to control the process of converting the input into the output.

Mechanism: The processor which performs the activity (person, computer program, etc.)

Data Diagrams:

*Input: Activity which creates the data.

*Output: Activity which uses the data.

*Control: Activity which controls the creation or use of the data.

Mechanism: The storage device used to hold the data (buffer, computer memory, etc.)

* For convenience of reference, inputs and controls are sometimes called "entries", while outputs are sometimes called "exits".

3.1.7 Multiple Arrow Interfaces

In Section 3.1.2, single arrows were shown entering or leaving the four sides of a box, for simplicity. Actually, boxes frequently have multiple arrow interfaces. These are shown as separate arrows, as illustrated by Figure 3.7.

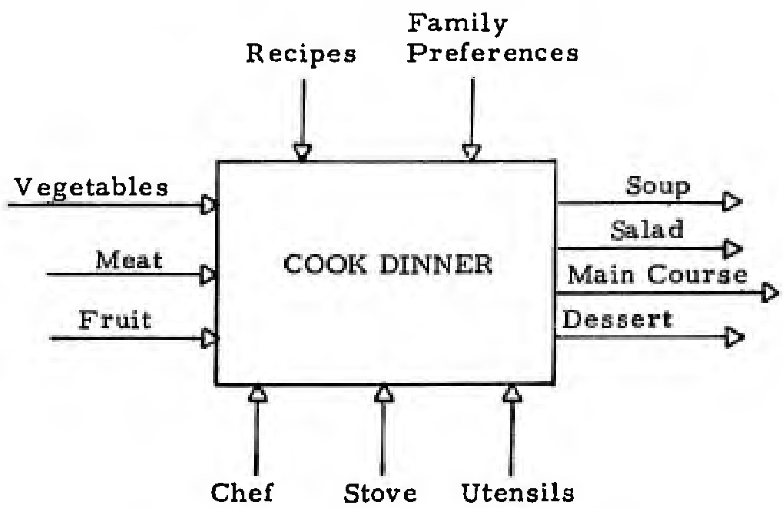


Figure 3.7 Example of Multiple Interfaces

Clearly, too many such arrows tend to clutter an SA diagram, and are discouraged. Instead, interface arrows are commonly "clustered" into a single arrow sometimes called a "pipeline" (but drawn the same way), and labeled with an appropriate, more general title which classifies the multiple contents of the pipeline. To see the fine interface detail shown by the breakout of the individual arrows, a lower-level diagram in the top-down structure may be examined. For example, the four output arrows shown in Figure 3.7 might be clustered into two pipelines labeled "main course" and "side dishes", where the "side dishes" pipeline contains the three arrows "soup", "salad", and "dessert", thus simplifying the interface arrows picture at this level of diagram.

This "clustering" technique plays an important role in creating understandable SA diagrams. It may be noted here that the level of generality of box and arrow titles should more or less match at any one diagram level, and that both box and arrow titles should grow successively more detailed as lower levels of diagrams are examined.

3.2 Connections Between Boxes

So far in the discussion, we have considered only individual boxes and their interface arrows. To form a complete Structured Analysis diagram, 3-6 boxes are drawn on an SA diagram form. The box titles represent a carefully chosen decomposition of a single box on the parent diagram, and cover the exact same subject matter, only in a lower level of detail. The input, output, and control arrows are then connected to show interfaces between the boxes. Figure 3.8 illustrates four connected boxes as their interface structure might appear on an SA diagram.

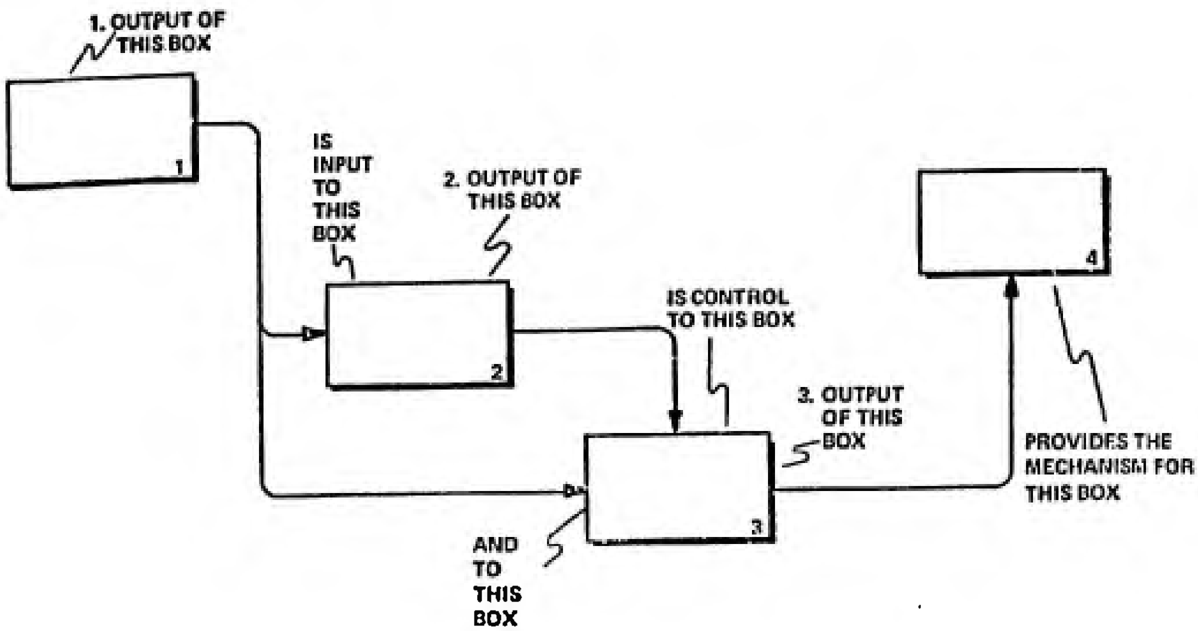


Figure 3.8 Multiple Box Interfaces

The basic connection rule is that <u>any</u> output arrow may provide <u>some</u> or <u>all</u> of the input, control, or mechanism to any other box. In fact, as illustrated by the output of box 1 in Figure 3.8 an output arrow may split, and provide entries or mechanisms to several boxes.

3.2.1 Constraint Relationships

In Structured Analysis, the connection of the output of one box to one or more other boxes shows a constraint relationship between those two boxes. That is, in an activity diagram, the box receiving the data is constrained, since it cannot begin its activity until the box providing the data sends it along. On a data diagram, the data cannot be used until the indicated activities supply it. This convention provides a subtle but important difference between common flow charts and Structured Analysis diagrams, in that SA diagrams can very naturally depict simultaneous activities.

As an illustration, consider Figure 3.8 as an activity diagram. Since box 1 provides input to both boxes 2 and 3, neither activity 2 nor 3 can begin until this data is provided. Once it is provided by box 1, box 2 may begin its action, but box 3 cannot start, since it must wait for box 2 to provide the needed control data, as shown. If the control a ow from box 2 to box 3 did not exist, both boxes 2 and 3 could have started and operated in parallel, using the output of box 1 (see Figure 3.9).

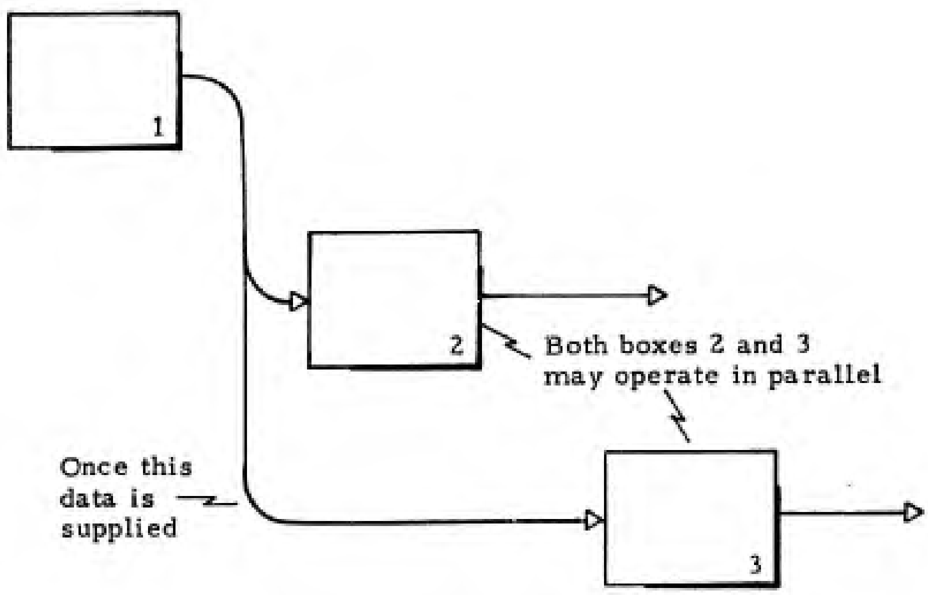


Figure 3.9 Simultaneous Action

Thus, sequence relationship can be shown, but the author is not constrained to simple sequences. This sort of flexibility is essential in properly modeling real-life system operation, and plays an important role in modeling various forms of feedback actions and in highlighting system initial conditions, both of which will become clear in the discussion below.

3.2.2 Specific Interface Formats

To describe the various forms of interface connections in more detail, the following sections begin with simple interface relationships, and build up to more and more elaborate relationships and common patterns found in Structured Analysis models. Examples are taken from the activity view to illustrate each point, but equivalent forms of relationships on data diagrams are equally straightforward.

3.2.2.1 Basic Input/Output Relationship

The basic input/output relationship is illustrated by a box which produces output which is, in turn, used as input by another box. Figure 3.10 illustrates this relationship.

Figure 3.10 Input/Output Example

Here, seeds grow into vegetables, which in turn are cooked to produce soup. The input/output relationship is clear, in that the data (seeds, vegetables, and soup are modified by the activities (grow and cook) and therefore cannot reasonably serve as controls.

3.2.2.2 Basic Output/Control Relationships

Instead of producing input to a second box, an activity may produce a control, as illustrated in Figure 3.11.

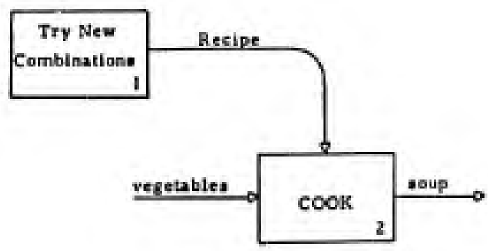


Figure 3.11 Output/Control Example

In Box 1, new combinations are tried, and combinations that work out well are formalized into recipes, which, in turn, control the cooking process. The recipe is clearly a control on the process, and does not make sense as an input, since it is not modified by the activity.

3.2.2.3 Basic Output/Mechanism Relationship

Finally, Figure 3.12 illustrates the output of a box used as a mechanism of a second box.

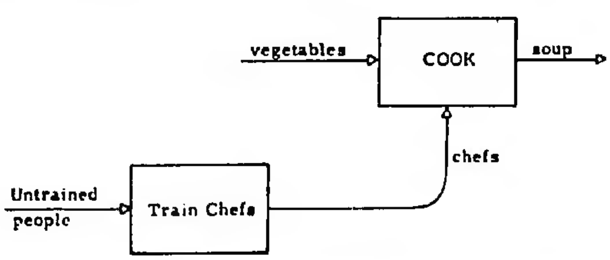


Figure 3.12 Output/Mechanism Example

This diagram shows the "chef training program" supplying chefs to man the cooking effort. This type of "activity-producing mechanism" combination is quite rare in SA diagrams, and is usually handled by simply naming the mechanism on the arrow, and providing the activity and data models of the mechanism separately.

3.2.3 Arrow Splits and Joins

Figure 3.8 above, shows one case of an arrow split in which the output of box 1 provides inputs to boxes 2 and 3. This is a common arrow pattern in Structured Analysis, and all manner of splits and joins are commonly used to show box relationships. To clarify whether all or part of the contents of an arrow follows which of the branches at splits and joins, SA uses the convention that all data flows along all branches unless otherwise indicated by a special label on an arrow branch. These conventions are summarized in Figure 3.13.

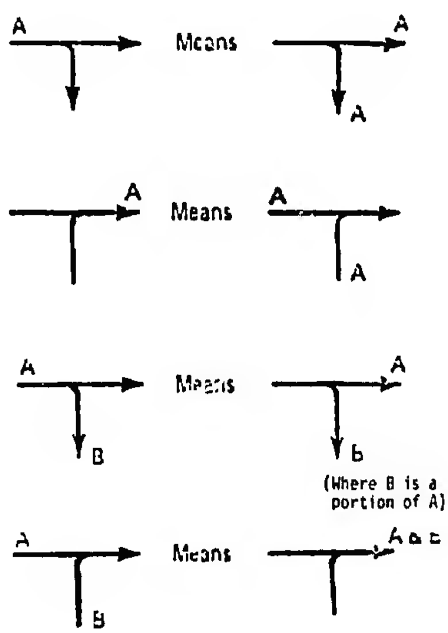


Figure 3.13 Arrow Branches and Joins

Figure 3.14 shows a complete SA diagram, illustrating various combinations of arrow splits and joins.

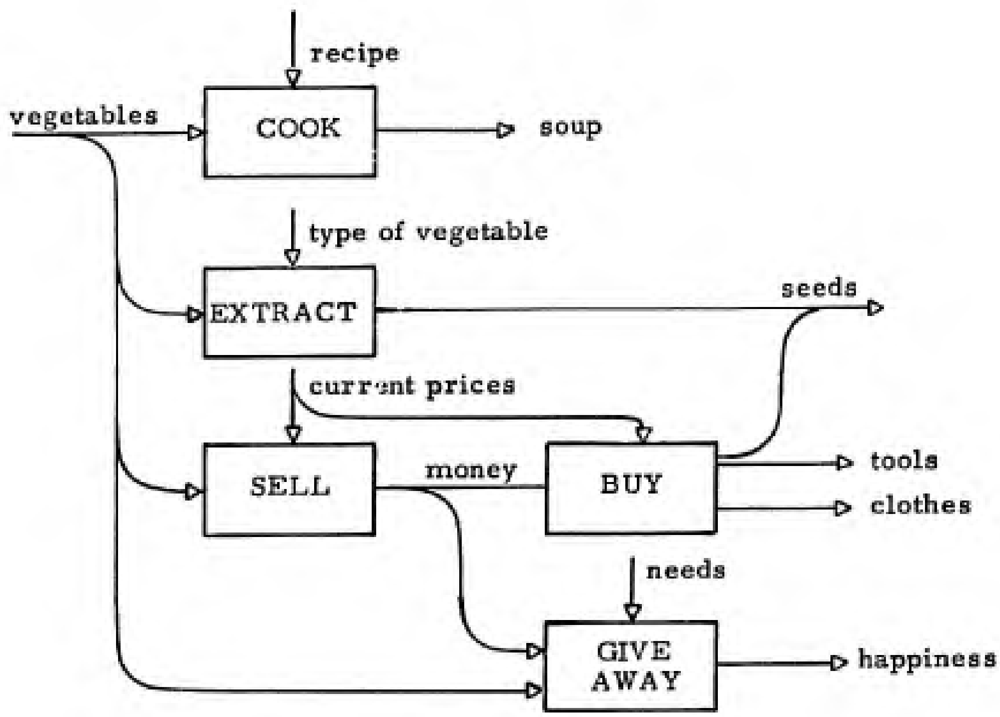


Figure 3.14 Example of Arrow Splits and Joins

Figure 3.14 says that vegetables can be cooked, sold, or given away, or that seeds can be extracted from them. The money obtained in the case of a sale can be used to buy seeds, tools, or clothes, or can be given away to produce happiness.

Arrow splitting is illustrated by the input arrow labeled "vegetables" which splits into four branches which provide the input to the four uses of vegetables shown. Since no labels are shown on the branches, this says that all vegetables go to all four boxes. If the label "tomatoes" had been written on the arrow branch entering the EXTRACT box, this would have indicated that only tomato seeds are to be extracted. In the output case, "money" is shown splitting and being used by either BUY or GIVE AWAY. In the GIVE AWAY case, both vegetables and money are given away, as indicated by the two input arrows entering the box. Arrow joining is illustrated by the arrow labelled "seeds", showing that seeds can either be extracted directly from vegetables, or bought. Again, since no labels appear on the branches of the arrow, it must be assumed that all kinds of seeds are on each branch. If the branch coming from BUY had been labeled "seed corn", this would indicate that only seed corn is bought.

3.2.4 Common Arrow Patterns

Although all arrows follow the conventions described above, and all necessary graphics are available in the Structured Analysis conventions discussed to this point, certain patterns occur so frequently that a convenient, special arrow notation has been adopted to cover these special cases.

3.2.4.1 OR Branches

A common situation is that a single box produces more than one output. For example, Figure 3.15 shows the BUY activity portion of Figure 3.14 as well as a possible child view of BUY, superimposed to show the source of the multiple outputs.

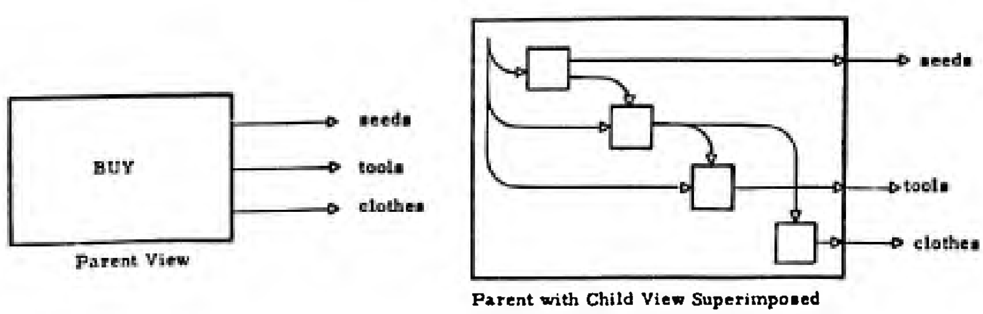


Figure 3.15 Multiple Outputs

Without any further parent diagram annotation and without looking at the child diagram, the parent diagram says that any or all of the three outputs may occur at any time as a result of the action of BUY. To see specifically which outputs are produced in any given circumstance, the reader must examine the lower-level "child" decomposition of the BUY activity.

One of the common situations is that one or another but not both outputs can occur simultaneously. To show this added constraint, the arrow OR structure is drawn, in the form

two-way branch three-way branch

Figure 3.16. OR Branch Structure

This says that at any given instant, A or B is produced, but not both. In the three-way case, A or B or C is produced, etc.

A corresponding (but less common) join structure is

two-way join three-way join

Figure 3.17. OR Join Structure

This says that A or B join, but not both, etc. Examples of OR branches are shown in Figure 3.18.

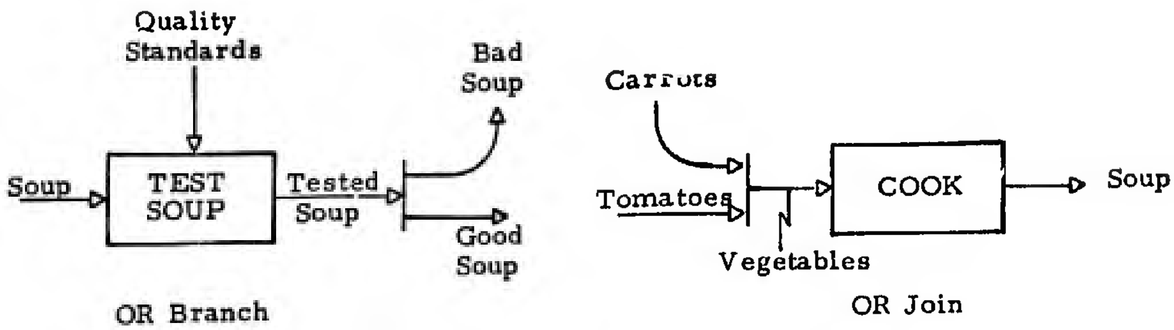


Figure 3.18 Examples of OR Branches and Joins

The OR branch portion of Figure 3.18 says that the soup is tested and judged to be either good or bad, and passed along or returned. The OR join portion of Figure 3.18 says that carrots and tomatoes are never combined into the same soup.

3.2.4.2 Iterations

To show two activities that iterate related data back and forth, the author, using basic SA notation, would draw

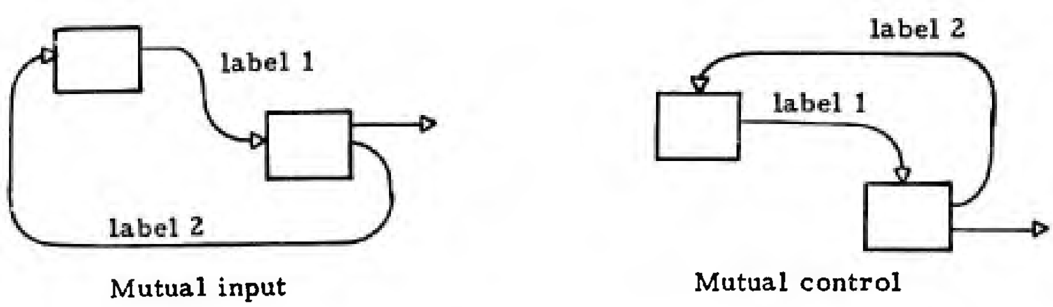


Figure 3.19 Iteration Patterns

Examples of these two situations are shown in Figures 3.20 and 3.21, respectively.

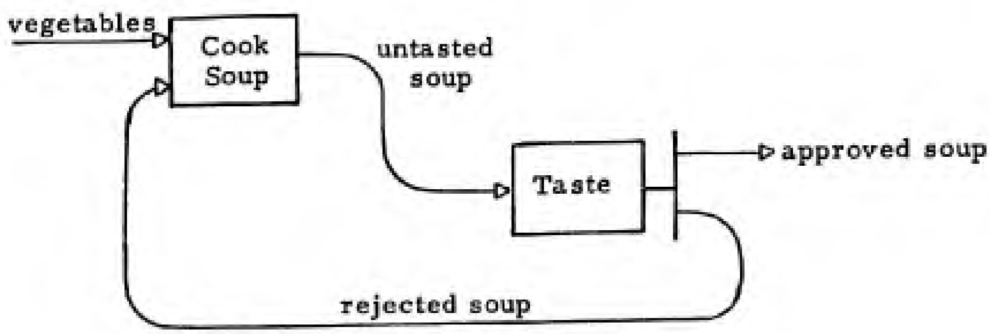


Figure 3.20 Mutual Input Example

Here, the soup is cooked and iterated through a quality control process until it is finally approved and released. The rejected soup is continually re-processed until it passes the taste test.

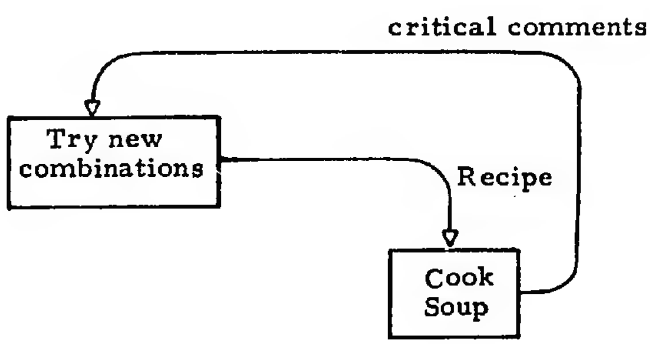


Figure 3.21 Mutual Control Example

In Figure 3.21, a feedback loop provides criticism of the recipes coming out of the "Try New Combinations" activity, thus causing possible future modifications.

To simplify the diagrams and to explicitly show the feedback situation, a "two-way arrow" form is used, as shown in Figure 3.22.

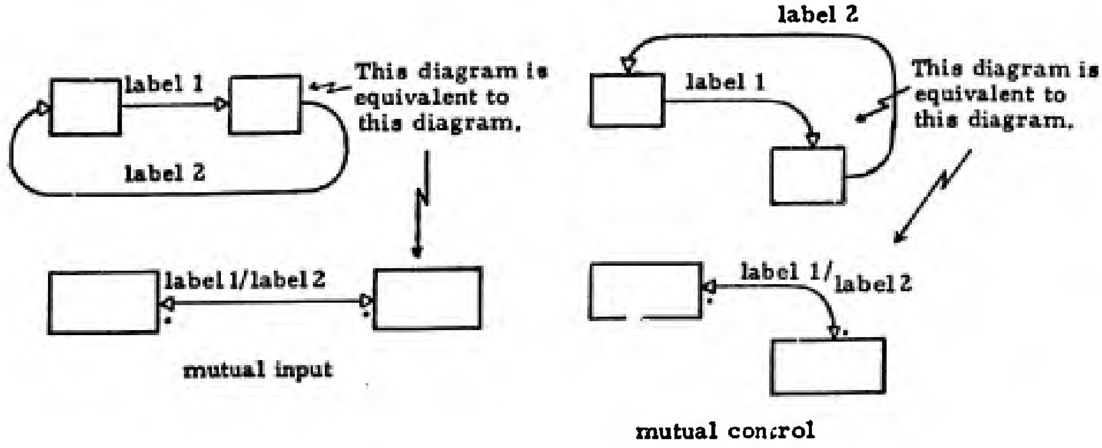


Figure 3.22 Two-Way Arrow Convention

The examples shown in Figures 3.20 and 3.21 are redrawn using the two-way arrow convention in Figure 3.22.

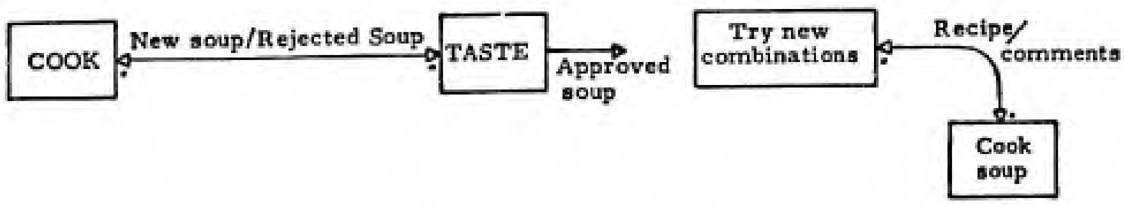


Figure 3.23 Examples of Two-Way Arrow Use

This two-way arrow format is in every way equivalent to the previous basic method which uses two separate arrows. It is used only where similar arrow contents are passed back and forth between two modules. It is not used in instances in which the arrow contents in the two directions are dissimilar. Notice several features of the two-way arrows:

- A single arrow is used, with arrowheads on both ends to show data flow in both directions.
- A small dot is shown below or to the right of the arrowhead to remind the reader that the arrow is two-way. (The two ends may be physically located far apart on the diagram, and the two-way nature may be unclear without the dots.)
- The two labels are separated by a slash ("/") to indicate forward/backward arrow contents.
- The "dominant" box with respect to the arrow contents is placed above and/or to the left of the other box on the diagram layout.

At first, the two-way arrow may seem strange because data is shown entering the output side of the box, but once this convention becomes familiar, the reduced diagram clutter and added information conveyed more than outweigh its initial strange appearance.

3.2.5 Arrow Connection Across Levels

To this point in the discussion, only simple-diagram arrow interfaces have been examined. When modules are connected to parent modules to form the top-down tree of diagrams comprising the model, arrows entering and leaving the diagrams (not connected at one end to a box on the diagram) hook up across the diagram-interface boundaries to form a complete, connected structure of interconnected diagrams. Figure 3.24 illustrates this connection across diagram levels, showing the children for boxes 1, 3 and 5 as though we were looking down upon the top-down decomposition and seeing the connected arrow structure across the diagram boundaries.

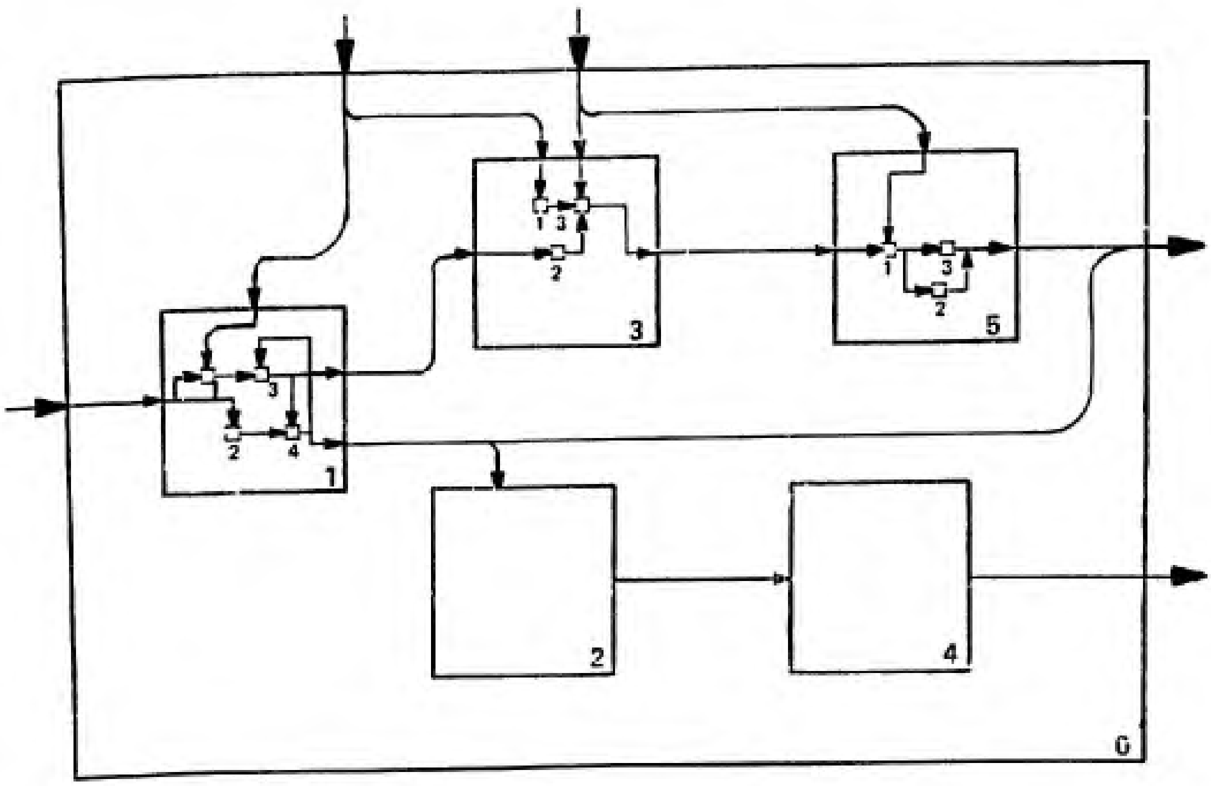


Figure 3.24 Arrow Connections Across Levels

[Note: This figure is for illustration only. Actual SA diagrams are never drawn this way.]

3.2.5.1 Parent Connection Notation

To connect arrows across the parent/child boundaries, a special arrow-matching "ICOM Code" is used. The name ICOM is derived from the four sides of an SA box (input, control, output, and mechanism), and the arrow code is constructed from the letter I, C, O, or M, followed by a number, assigned in a top-to-bottom and left-to-right order under each of the four categories. Figure 3.25 summarizes the ICOM arrow-numbering convention.

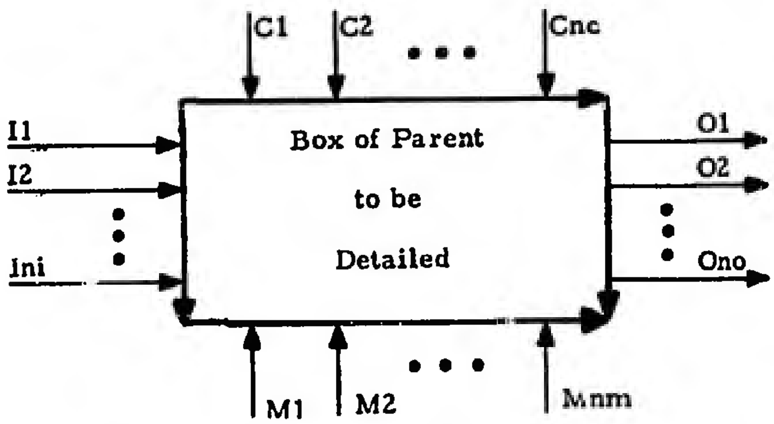


Figure 3.25 ICOM Code Numbering Convention

In the detail level (child) diagram, each boundary-crossing arrow (any arrow that does not have both ends connected to a box) is labeled with its parent-context ICOM code, in addition to its label.

Since every boundary-crossing arrow is one end of an arrow whose other end appears in the parent module, the descriptive phrases used to label the two arrows should match. However, the arrow label used on the lower level diagram may occasionally be phrased differently to give more detail or to match the terminology of the child diagram more closely than the corresponding higher level arrow. This is quite natural and acceptable, for the parent-level labels are more general and appropriate for that high-level view, whereas the lower levels need to consider more detail as the story unfolds.

It should be noted that occasionally the one-for-one arrow matchup between levels is not followed; specifically a higher-level arrow may be omitted when the arrow is only meaningful and useful where it is first introduced. Specific rules determine when it is permitted to omit a context arrow. Each such non-matching lower level arrow has its descriptive phrase enclosed in parentheses to show that the arrow is of the special, non-matching type.

Mechanism arrows are often omitted from the diagram since they are usually added after the model is complete, to show how the model is implemented. Also, if all boxes in a diagram use the same mechanism, a statement to this effect is more properly included at one place on the parent diagram or its accompanying text, rather than cluttering up the child diagram with several identically labelled arrows. Also, since many devices are generally employed at a higher level, these details are more properly uncovered by the lower-level breakdown. At the high level, the meaning is better conveyed by hiding such details inside the lower-level diagrams. Thus, a module diagram explicitly shows only as much information as is required for understanding its meaning.

As in Figure 3.24 Figure 3.26 shows an instance of a parent view with a possible child diagram layout inside (looking down from above) to show how the correspondence is made between the two levels.

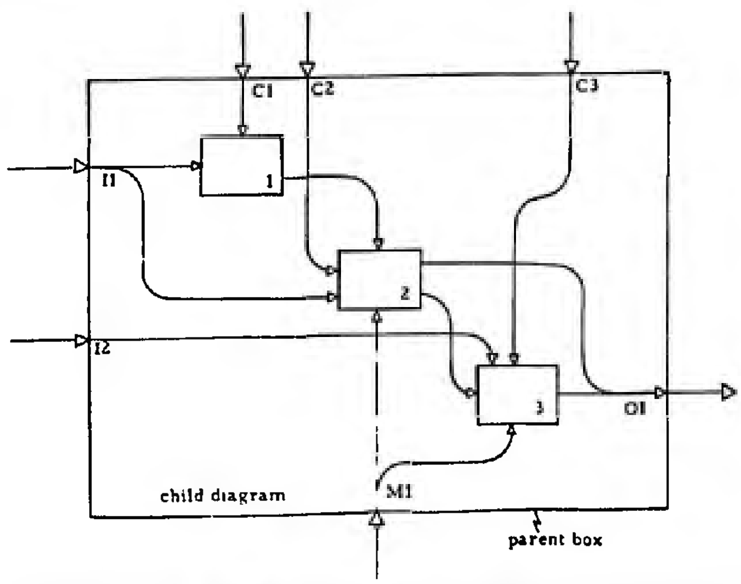


Figure 3.26 ICOM Code Connnection to Child Diagram

Several ICOM conventions are illustrated in Figure 3.26:

- ICOM codes are not written on the parent view. Codes are implicit from the arrow positions.
- Usage in the child view is not necessarily related to usage in the parent view. For example, I2 is used as a control to box 3 and C2 is used as an input to box 2, in the child diagram.
- Arrows in the child view commonly split or join to form the required interface, as shown by I1, M1, and O1.

In practice, the association of parent interface arrows with child details cannot, of course, use a "top-down", reduced-size view with all boundary-crossing arrows neatly lined up, as shown in Figure 3.26. In addition, arrow tails should not be drawn to the edge of a diagram, since this tends to clutter and obscure the diagram (for example, I2 would most likely be drawn as a short arrow entering the top of box 3, to avoid the multiple arrow crossings shown in Figure 3.26). However, the ICOM code is always written near the end of the arrow on the child diagram, and it is still a simple matter to envision the parent connections.

Two-way arrows may also be used as input, control, or output interfaces between levels. These multi-directional interfaces are slightly more complicated to follow, since there is only a single ICOM code associated with both directions. Two-way control and output arrow connections between levels are shown in Figure 3.27.

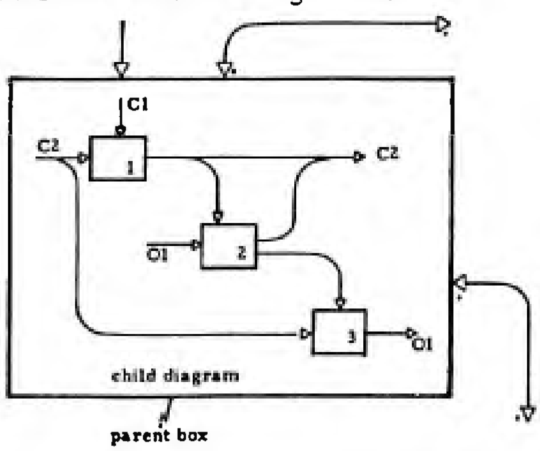


Figure 3.27 Two-Way Arrows Used as Interfaces Across Levels

The reader must be careful to connect both directions in these instances (for example, O1 is both an input to box 2 and an output of box 3).

3.3 The Activity/Data Model Relationships

As discussed in the initial overview, the complete Structured Analysis process results in two models: an activity model and a data model. Although dual notations are employed and although both models encompass the same subject matter, each view presents a very different organization of the material, and both are required for clarity and completeness.

To compare and contrast the two models, consider the purpose of each. The activity model decomposes the functions of the systems into their sub-functions, with diagram levels of decomposition showing a finer and finer breakdown of how the system operates, and with the data items serving as interfaces between these functions. Any individual activity diagram thus describes a well-bounded view of a module in this top-down system structure in a simple enough layout to permit a thorough understanding of its function and operation and how it relates to other modules which comprise the system. The reader of an activity model may probe as deeply as is required into lower decompositions or back off into the more abstract, higher levels of the model to orient his understanding of system operation.

The data model covers the same subject matter as the activity model, but emphasizes system data instead of activities, and decomposes the "things" of the system into sub-groups of things based upon the purpose which each serves. The data model thus complements the activity model by presenting a second view of the same subject, with a different emphasis. After carefully studying both models, the reader is left with a much richer understanding of the system decomposition than if only one model is studied.

For example, the data-based view shows in one place all activities creating and using an individual data item. All activities which modify a particular data item could be derived from the activity model by laboriously searching out all data arrows referencing the data item of interest and then examining which activities modify the data and which only read or pass along the data. In the data-based model, this information appears in a single place: on the input arrow to the data box. Similarly, high level data arrows on an activity diagram often contain primitive data elements that come from widely dispersed points in the data model.

Up to this point, only activity diagrams have been used for illustrations. As an example of a data model diagram, Figure 3.28 shows the fresh, data view of the same subject matter used for the activity diagram examples.

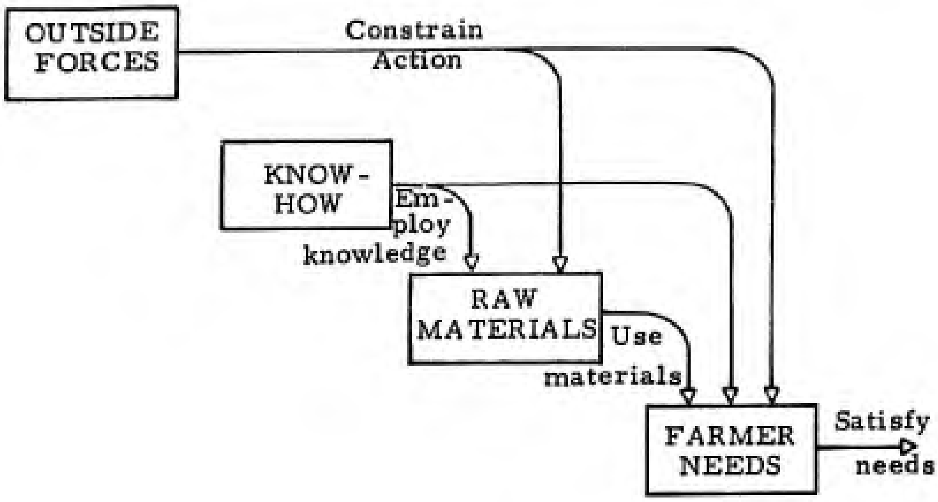


Figure 3.28 Sample Data Diagram

Recalling the activity view of the same subject, we see that all of the data items on the arrows can be considered to fall under one of the boxes shown in Figure 3.28:

Outside Forces:	Weather Market Prices Vendor Prices Lists of Needy People
Know-How:	Recipes Types of Vegetables
Raw Materials:	Seeds Vegetables Money
Farmer Needs:	Food (including "soup") Clothes Tools Happiness

It can be argued that money is a need rather than a raw material, or that vegetables are a sub-category of food rather than raw materials, but the activity diagrams show money as primarily a means of satisfying various needs, and that vegetables are used only to create soup, seeds, or money, but never eaten as an end product. Therefore, even though each view is basically independent, the activity view has influenced the data view somewhat, and the total model thus makes a consistent story.

As a wrap-up of the topic of Diagram Syntax, Figure 3.29 shows the "child" diagram for the "Raw Materials" box of Figure 3.28. The example also illustrates inter-level connection of parent/child diagrams via ICOM codes.

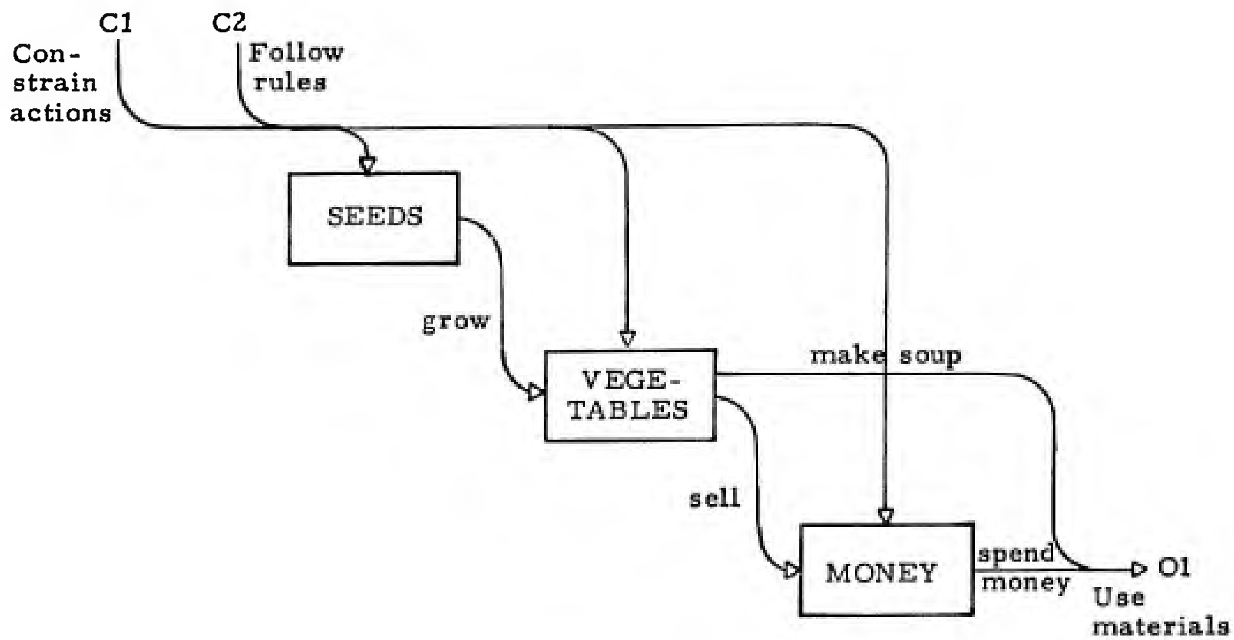


Figure 3.29 Child Diagram for "Raw Materials"

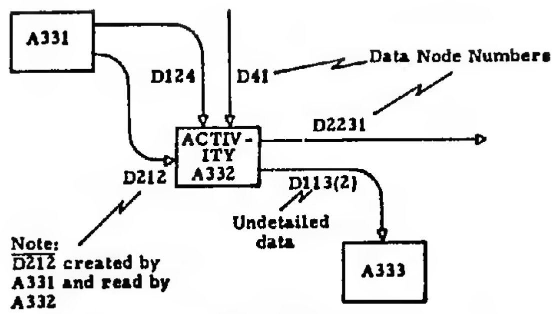
3.4 Activity/Data Model Ties

The activity and the data models are interlocked into the complete Structured Analysis model via a uniform cross-referencing scheme in which the explicit box numbers from one model are written on the arrows of the other model, and vice versa. To easily distinguish activities from data, each number is preceded by an A or a D, respectively.

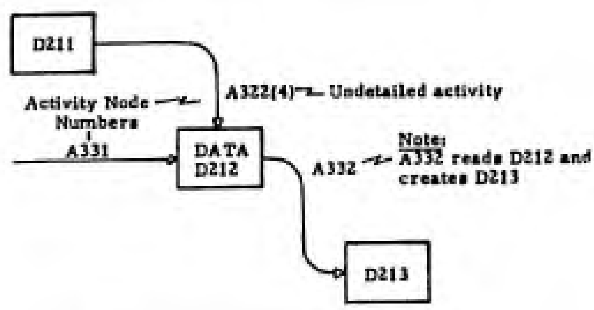
This information is not usually shown on the basic set of module diagrams, since these cross-reference numbers tend to obscure the diagrams. Cross-reference information is retained on a separate set of diagrams, and is used for specific purposes where complete detail is required.

When data node numbers are written on activity diagrams, the resulting model is called the "A/D Tie" model. When activity node numbers are written on data diagrams, the resulting model is called the "D/A Tie" model.

Figure 3.30 shows an example of a typical activity diagram and data diagram taken from A/D Tie and D/A Tie models. The data and activity node numbers are written along the arrows around Activity Box A332 and Data Box 212, respectively. Using the basic SA interface meanings, this arrow annotation concisely displays which pieces of data are used by which activities, and in what way. Parenthesized numbers following node numbers indicate that only a portion of the data item or activity is present on the arrow. This indicates that the referenced model was not carried to sufficient depth of detail to permit precise classification and, rather than mislead the reader, the classification was done to as great a depth as possible, and the sub-part of the indicated data item or activity is identified and listed at the end of the model, under the node and parenthesized sub-number shown.



Activity Diagram Tied to Data (A/D Tie)



Data Diagram Tied to Activity (D/A Tie)

Figure 3.30 Examples of A/D and D/A Ties

3.5 The Diagram Form

The rules, forms, and mechanical aids used in the practice of Structured Analysis are intended to aid the analyst, without becoming unduly restrictive or time-consuming. Perspicuity of insight and clarity of expression are not the only goals, for the equal importance in the requirement that everything must be on paper so that no vital information slips through the cracks or is conveyed by oral discussion and arm waving unavailable to the isolated reader.

A well known feature of forms and procedures design is the fact that people will either abuse or not use forms and procedures which are too complicated or which do not directly support their work. For this reason, the entire modularity discipline depends only on a single form -- the Diagram Form -- and that form has minimum structure and constraints. The form has been designed to support creativity under the assumption that the author recognizes his responsibility to communicate. Combined with the fact that the diagram form is a single standard size sheet for ease of filing and copying, the constraints on the author are minimal indeed and do support his prime responsibilities.

3.5.1 Publication Format

For publication, and the association of text with the diagrams, each module diagram is displayed on two facing pages. The module under current consideration is on the lower page. Related text material is on the upper page, with a reduced copy of the parent module to provide the complete context. The nested factor which is expanded on the lower page is highlighted or otherwise marked in the parent module. Each module has a title and a reference number. The pages are bound in sequence so that the node index forms a table of contents for the complete model. This format provides the definitive, primary vehicle for a Structured Analysis model, tying together all diagrams in the model along with their textual explanations and highlights. Figure 3.31 shows the publication format for SA module diagrams.

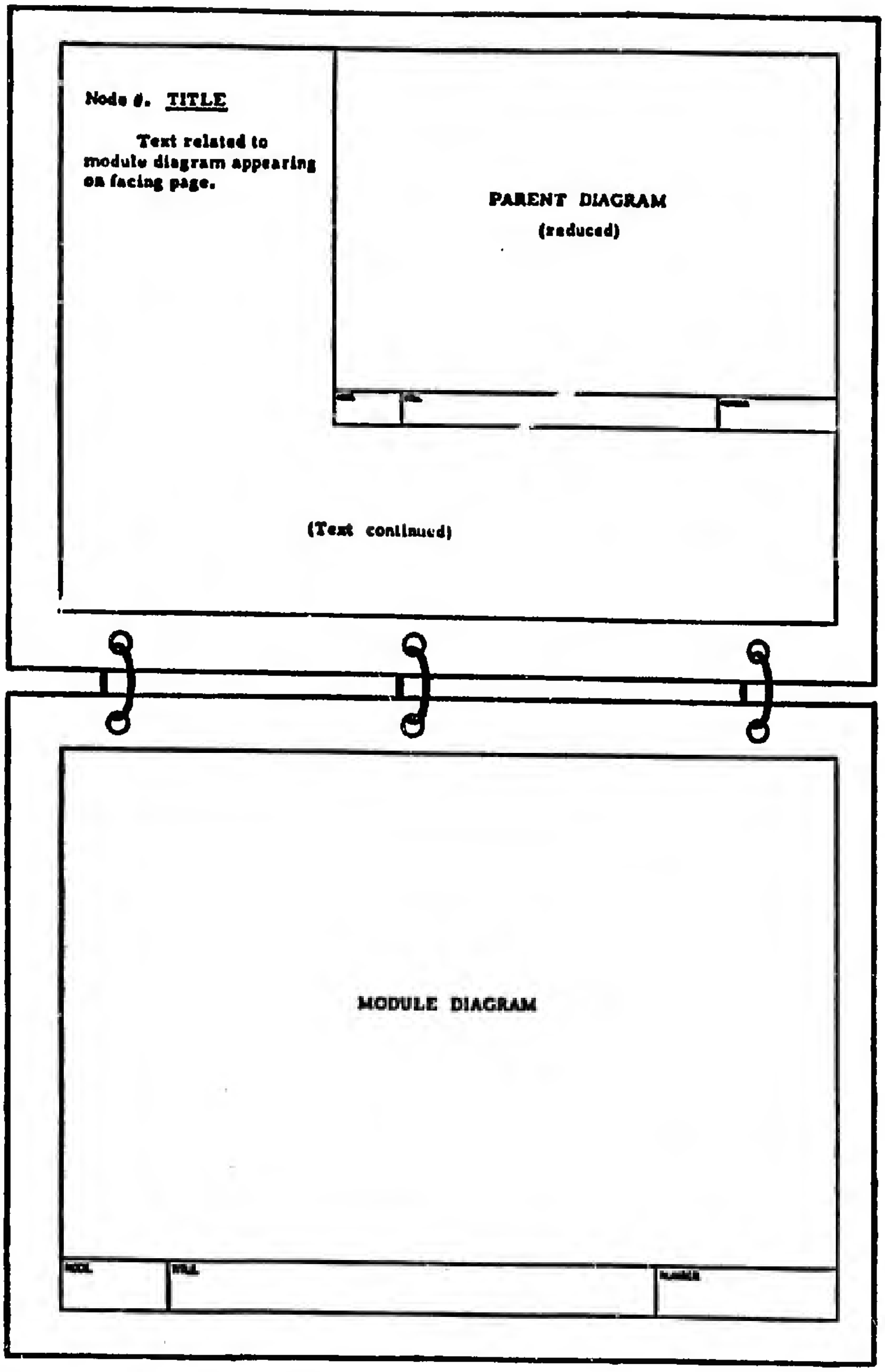


Figure 3.31 Publication Format for SA Module Diagrams.

3.5.2 Working Diagram Layouts

For convenience of handling and for general module-diagram reading, the "publication format" -- in which a standard notebook size document contains all diagrams in the model, and the reader has the current diagram, parent context, and text open to view at one time -- presents the diagram material in a reasonable form. However, in some instances it is helpful to be able to con template a larger segment of the model at one time than the publication format conveniently affords. For example, after having reviewed each individual diagram in detail, it may be desirable to view the entire model at once, to see if the whole thing makes sense and depicts a reasonable view of reality.

In these instances, it is helpful to attach the diagrams to a wall, or scatter them on a large table, laid out like the node diagram, so that it is easy to jump from one sheet to another. Technical Managers often find the wall-layout helpful so that they can view the current status or design layout whenever appropriate during daily operations. Perhaps a central meeting room wall might be profitably used by an entire team to maintain a model while it is in heavy use. Working groups may select whichever form best meets their specific needs.

Section 4
HOW TO READ MODULE DIAGRAMS

4.1 The Node Diagram Table of Contents

The module diagrams forming a model, branching downward from a single starting point, may be represented by a "node diagram", also called a "node tree" because of its branched structure, or by an equivalent "node index".

The node index appears on the first page of a model to form a table of contents. Scanning these box titles gives an overview of the decomposition structure. Figure 4.1 shows a typical node tree of diagrams forming a model, and its corresponding Node Index. The node diagram is usually not included in a draft version of a model, since the index form contains the same information and is more easily constructed. The node tree form is generally included in final publication format models only, and shows the model structure more explicitly then does the index form.

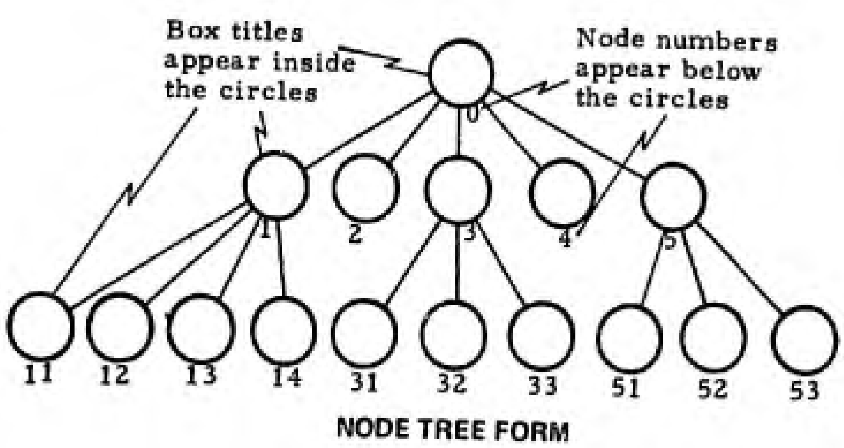


Figure 4.1 Typical Node Diagram and Index

Figure 4.1 shows the two general forms of node layouts. In practice, the activity model node numbers are preceded by the letter "A" and the data node numbers by the letter "D".

4.2 Basic Reading Sequence

The discipline of creating modules gradually unfolds complexity in steps that the reader's mind can put back together. This means that the reader of a model should also use a disciplined approach, so that his mind can follow the road to understanding so carefully laid out by the author. The following reading sequence is recommended, taking the "current modules" in top-down order:

1. Scan only the boxes of the current module to gain a first impression of the module decomposition.
2. Using the parent diagram, rethink the message of the parent module, observing the arrows feeding to and from the current module.
3. Referring back to the current module, see how and where each arrow from the parent context attaches to the factors in the current module, using the ICOM codes.
4. Then consider the internal arrows of the current module to see how it works in detail. Consider the boxes from top to bottom and from left to right. Examine the arrows by going clockwise around each box.
5. Finally, read the text for the current module to confirm or alter the interpretation gained from consideration of the diagrams themselves.

This 5-step reading sequence has been found to be the most effective in practice. In particular, it provides an orderly procedure which ensures that the major, important features of each diagram receive attention. Of course, if the reader mechanically follows the procedure without carefully thinking through the message at each step, little real meaning will be attained. In the end, the key to successful SA diagram reading is the careful study of each diagram by a reader knowledgeable in the field of the subject matter, pausing at step 4 to explore the diagram's decomposition and carefully laid-out arrow pathways to uncover the full meaning intended.

4.2.1 Reading the Diagram's Text

The purpose of the text accompanying each diagram is to highlight and point out specific noteworthy features of the diagram. In particular, the text does not present all of the same information contained in the diagram itself, but serves rather to highlight and emphasize important diagram features; the complete message is only conveyed by the diagram. For this reason, the reading sequence recommends reading the text last (step 5), as a wrap-up stage in which the reader should be well familiar with the diagram details and should be able to easily follow the detailed points presented in the text.

As an aid in connecting the points in the text to specific diagram elements, a shorthand box and ICOM code notation is employed in the body of the text, enclosed in parentheses. This reference code convention uses the box number (1-6) which appears in the lower-right corner of each box in the diagram and the ICOM code, to pinpoint arrows. If a box is referenced, the box number appears by itself (e.g., "(3)" or "(5)"). If an internal arrow is referenced, the box number is written first, followed by the ICOM code (e.g., "(1C2)" or "(2O3)"). If a diagram boundary arrow (one that connects to the parent diagram) is referenced, the ICOM code appears alone (e.g., "(C1)" or "(I2)").

Section 3

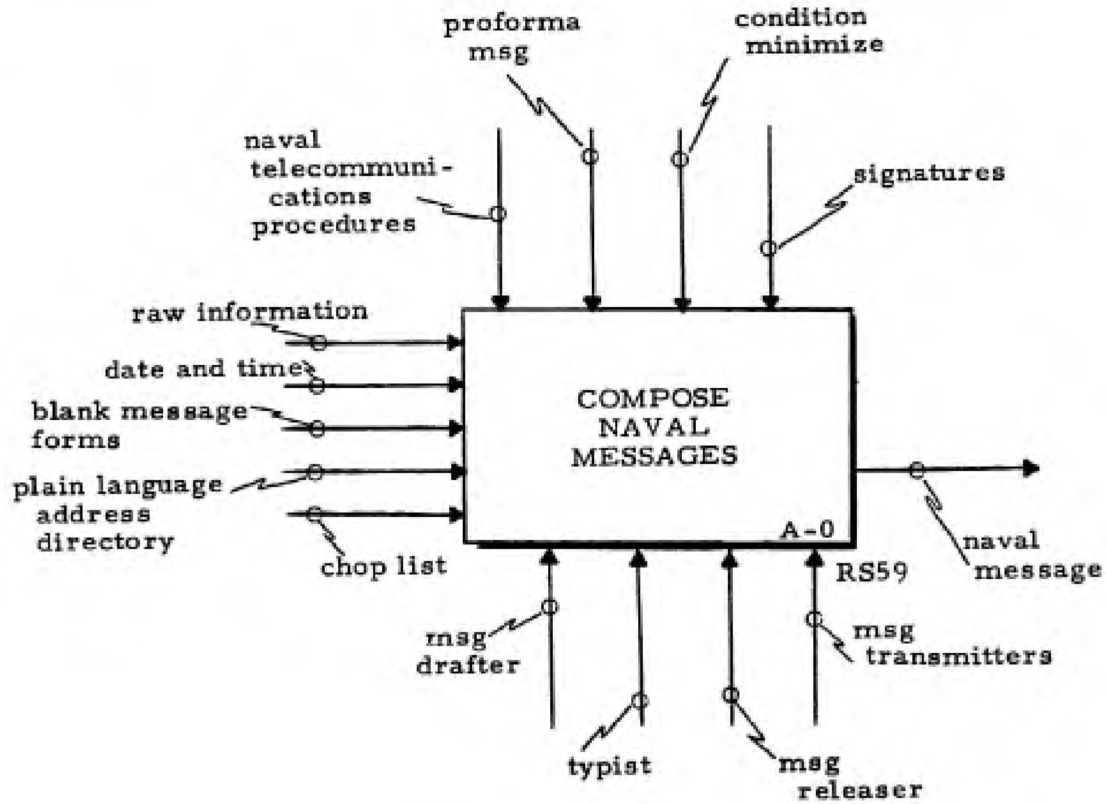
COMPOSE NAVAL MESSAGES ACTIVITY MODEL

Node Diagram for Compose Naval Messages

- (Top) COMPOSE NAVAL MESSAGES - PARENT CONTEXT
- A0 COMPOSE NAVAL MESSAGES
  - A1 DRAFT MESSAGE
    - A11 SELECT ADDRESSEES
      - A111 DETERMINE TYPE OF ADDRESSEE
      - A112 PREPARE GENERAL MESSAGE ID
      - A113 PRODUCE COMMERCIAL FIRM ADDRESS
      - A114 DETERMINE NEED FOR BOOK MESSAGE
      - A115 PROCESS COLLECTIVE ADDRESSES
      - A116 CLEAR CLASSIFIED MESSAGES FOR TREATY ORGANIZATIONS
    - A12 SELECT PRECEDENCE AND METHOD OF TRANSMISSION
      - A121 SUPPLY ROUTINE PRECEDENCE
      - A122 SUPPLY PRIORITY PRECEDENCE
      - A123 SUPPLY IMMEDIATE PRECEDENCE
      - A124 SUPPLY FLASH PRECEDENCE
      - A125 SUPPLY EMERGENCY PRECEDENCE
      - A126 SELECT TRANSMISSION SYSTEM
    - A13 PROVIDE DATE/TIME GROUP
    - A14 COMPOSE TEXT
      - A141 SUPPLY SSIC
      - A142 ADD PASSING INSTRUCTIONS
      - A143 ADD SUBJECT
      - A144 ADD REFERENCES
      - A145 WRITE TEXT
    - A15 CLASSIFY MESSAGE
      - A151 DECIDE IF MESSAGE SHOULD BE CLASSIFIED
      - A152 SELECT CLASSIFICATION
      - A153 DECIDE ON SPECIAL HANDLING MARKINGS
        - A1531 DECIDE IF SPECIAL CATEGORY
        - A1532 DECIDE IF LIMITED DISTRIBUTION REQUIRED
        - A1533 DECIDE IF MESSAGE SHOULD BE MARKED 'PERSONAL FOR'
        - A1534 DECIDE IF FOREIGN DISSEMINATION ALLOWED
        - A1535 DECIDE IF FOR NAVY EYES ONLY
      - A154 DECIDE IF DATA IS RESTRICTED
      - A155 DECIDE ON DOWNGRADING AND DECLASSIFICATION
    - A16 CHECK COMPLETED MESSAGE
  - A2 FILL IN MESSAGE FORM
    - A21 DO FORM SETUP
    - A22 CONSTRUCT HEADER
      - A221 FILL IN ALL PAGE BLOCKS
      - A222 FILL IN DRAFTER OR RELEASER TIME BLOCK
      - A223 FILL IN PRECEDENCE BLOCKS
      - A224 FILL IN CLASS BLOCK
      - A225 FILL IN MESSAGE HANDLING INSTRUCTIONS BLOCK
    - A23 CONSTRUCT ADDRESS COMPONENT
      - A231 FILL IN FROM LINE
      - A232 FILL IN TO LINE
      - A233 FILL IN INFO LINE
      - A234 FILL IN EXEMPTED ADDRESSES
      - A235 SPECIFY OTHER THAN ELECTRICAL TRANSMISSION METHOD
    - A24 CONSTRUCT TEXT
      - A241 FILL IN CLASSIFICATION LINE
      - A242 FILL IN SPECIAL HANDLING
      - A243 FILL IN PASSING INSTRUCTIONS
      - A244 FILL IN SUBJECT AND SSIC
      - A245 FILL IN REFERENCES AND TEXT
      - A246 FILL IN DOWNGRADING AND DECLASSIFICATION
    - A25 CONSTRUCT ADDITIONAL PAGE 1 INFO
      - A251 FILL IN SPECIAL INSTRUCTIONS BLOCK
      - A252 FILL IN DISTRIBUTION BLOCK
      - A253 FILL IN DRAFTER BLOCK
      - A254 FILL IN RELEASER BLOCK
      - A255 FILL IN SECURITY CLASS BLOCK
      - A256 SEE IF MORE TO TYPE
    - A26 GET OK FROM DRAFTER
  - A3 RELEASE MESSAGE
  - A4 SEND MESSAGE
  - A5 RESOLVE PROBLEMS

A-0 COMPOSE NAVAL MESSAGES - PARENT CONTEXT

This is the top level activity diagram for the composition of naval messages. Naval messages are the principal means by which commanders communicate. The composition of naval messages is standardized (C1) in terms of drafting procedures and typing procedures. This standardization is required due to the increasing level of automation of message processing functions (M4), and the concurrent requirement for messages to be constructed in an undeviating manner for automatic OCR recognition (I3). Further standardization for the purpose of automatic message routing is achieved by the use of standard addresses (I4). Messages are transmitted (O1) to the message addressees either by courier, electrically, or by one of the two major message processing systems which employ OCR equipment, the Local Digital Message Exchange (LDMX) and the Naval Communications Processing and Routing System (NAVCOMPARS).



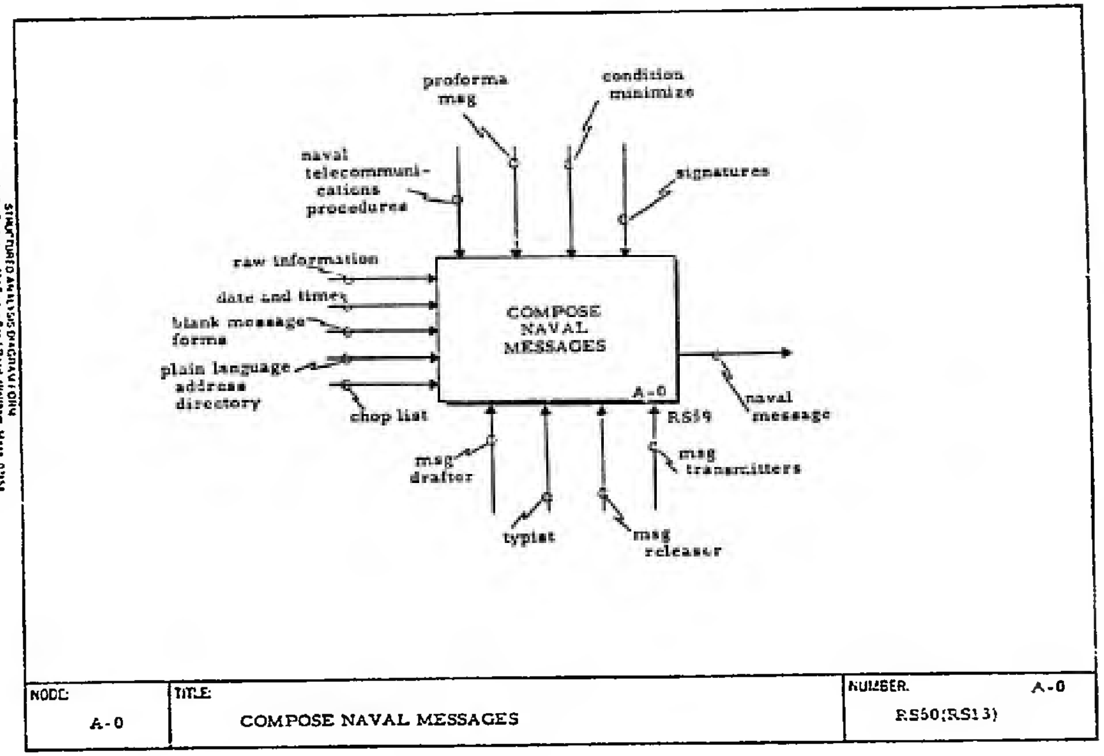
NODE:	TITLE:	NUMBER:
A-0	COMPOSE NAVAL MESSAGES	A-0 RS60(RS13)

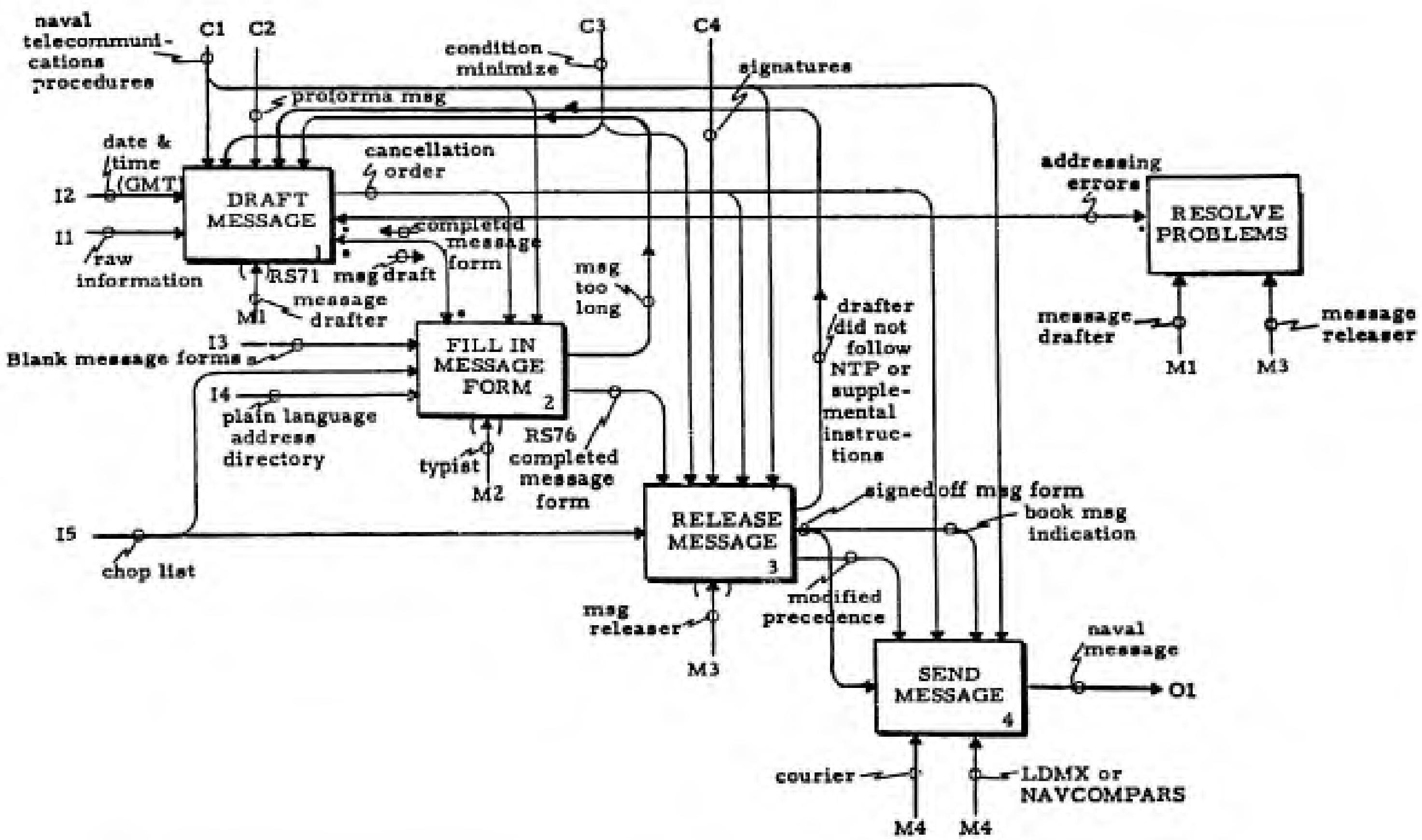
A0 COMPOSE NAVAL MESSAGES

The key steps in the composition of a naval message are the drafting of the message (1), the typing of the message on either an OCR form or other message form (2), the verification of the typed message (2C1 to 1O3) by the drafter (1M1), the obtaining of the required signatures (3C3) by the message releaser (3M1), and finally the transmission of the message (4O1) to the addressees by either courier (4M1) or one of the message processing systems (4M2).

The message drafter (1M1) may choose to cancel a message (1O1) at any time during the composition process. It is also the drafter's responsibility to review all messages during periods of minimize (1C2) to ensure that electrical transmission is essential.

The message releaser (3M1) has the responsibility for obtaining all message signatures (3C3) required (3I1) for a message. In addition, the releaser must inhibit the release of messages during periods of minimize (3C2), verify that the drafter has complied (3O1) with standardization or supplemental instructions (3C5), and verify or change the precedence associated with the message (3O2).





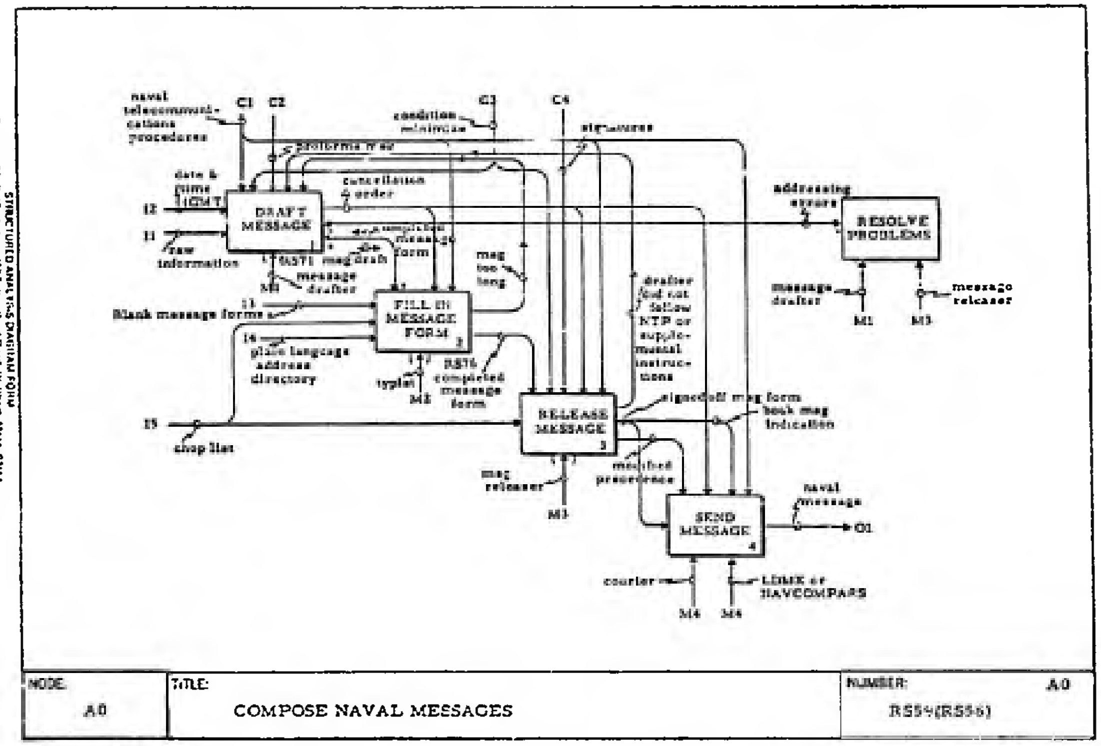
NODE: A0	TITLE: COMPOSE NAVAL MESSAGES	NUMBER: RS59(RS58) A0

A1 DRAFT MESSAGE

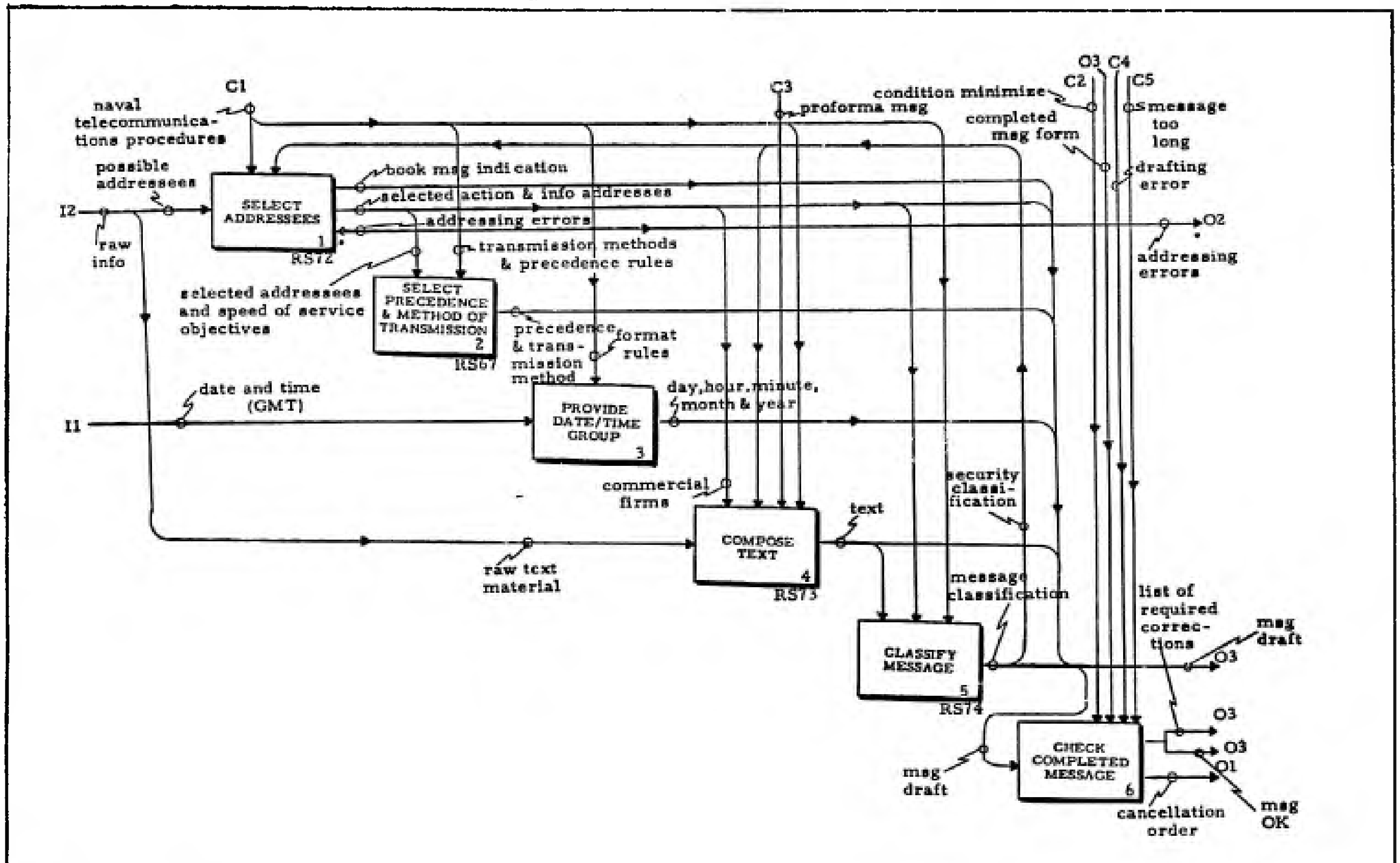
Message drafting involves the pulling together of four primary types of information. Most importantly of course there is the text (4O1) of the message itself. Depending upon the contents of the text material (5C1) and the addressees (5C2), the message receives a standard classification (5O1). The message addressees (1O2) are defined as being required to either take action on the message or to just use the message for their own information. In some situations (e. g., commercial contract bid request, a message may be destined for two or more addressees but no addressee need or should be informed of any other addressee. Such messages are called book messages (1O1). The final major component of the message is its precedence and its method of transmission (2). The precedence (2O1) indicates a desired writer to reader delivery time. The selected transmission method (2O1) may specify electrical transmission or courier delivery.

Once a message has been drafted (A1.O3) it is sent to be typed. When the typed draft is returned (6C2) it is checked for correctness. If errors are discovered (6), a list of corrections is sent back to the typist (6O1). Otherwise the message is given an OK by the drafter (6O1).

Note that the drafter must resolve certain types of problems. If the typist discovers that a message exceeds the page limit for NAVCOMPARS or LDMX (6C4), the message must be broken up into several messages. Similarly, the message may have to be redrafted if the message releaser discovers that the drafter did not follow the standard drafting procedures or the supplemental instructions (6C3).



STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech Inc., 460 Totten Pond Road, Waltham, Mass. 02154
Available in pads of 50 sheets



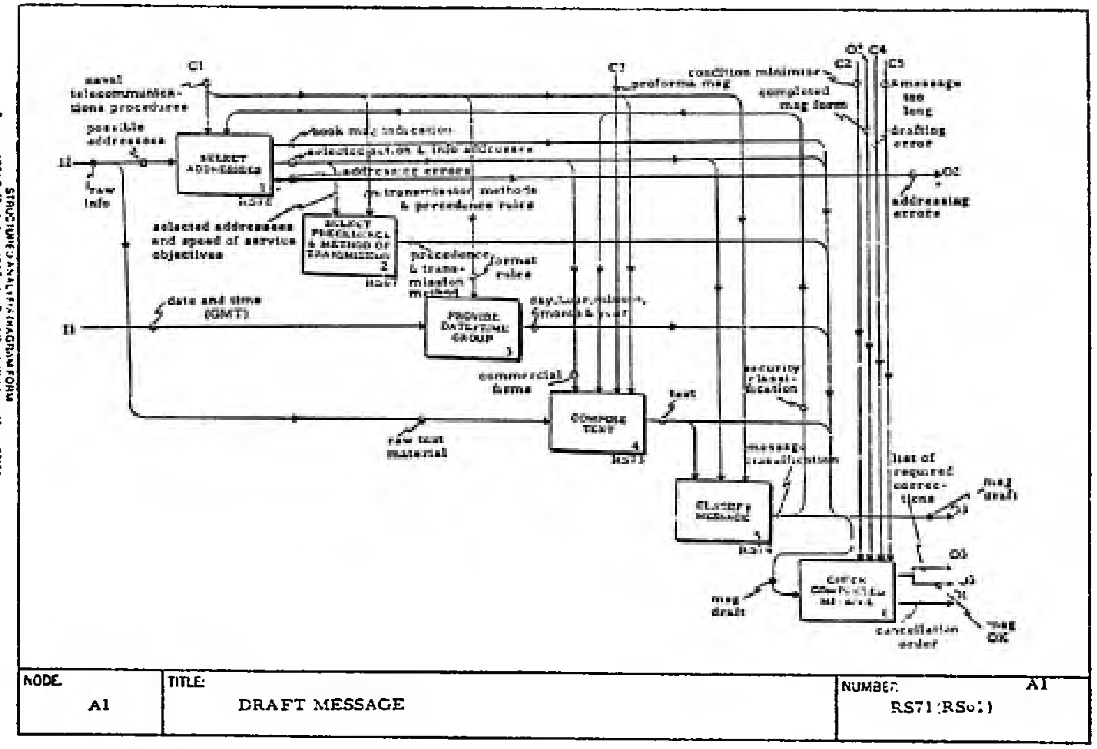
NODE:	TITLE:	NUMBER: A1
A1	DRAFT MESSAGE	RS71 (RS61)

A11 SELECT ADDRESSEES

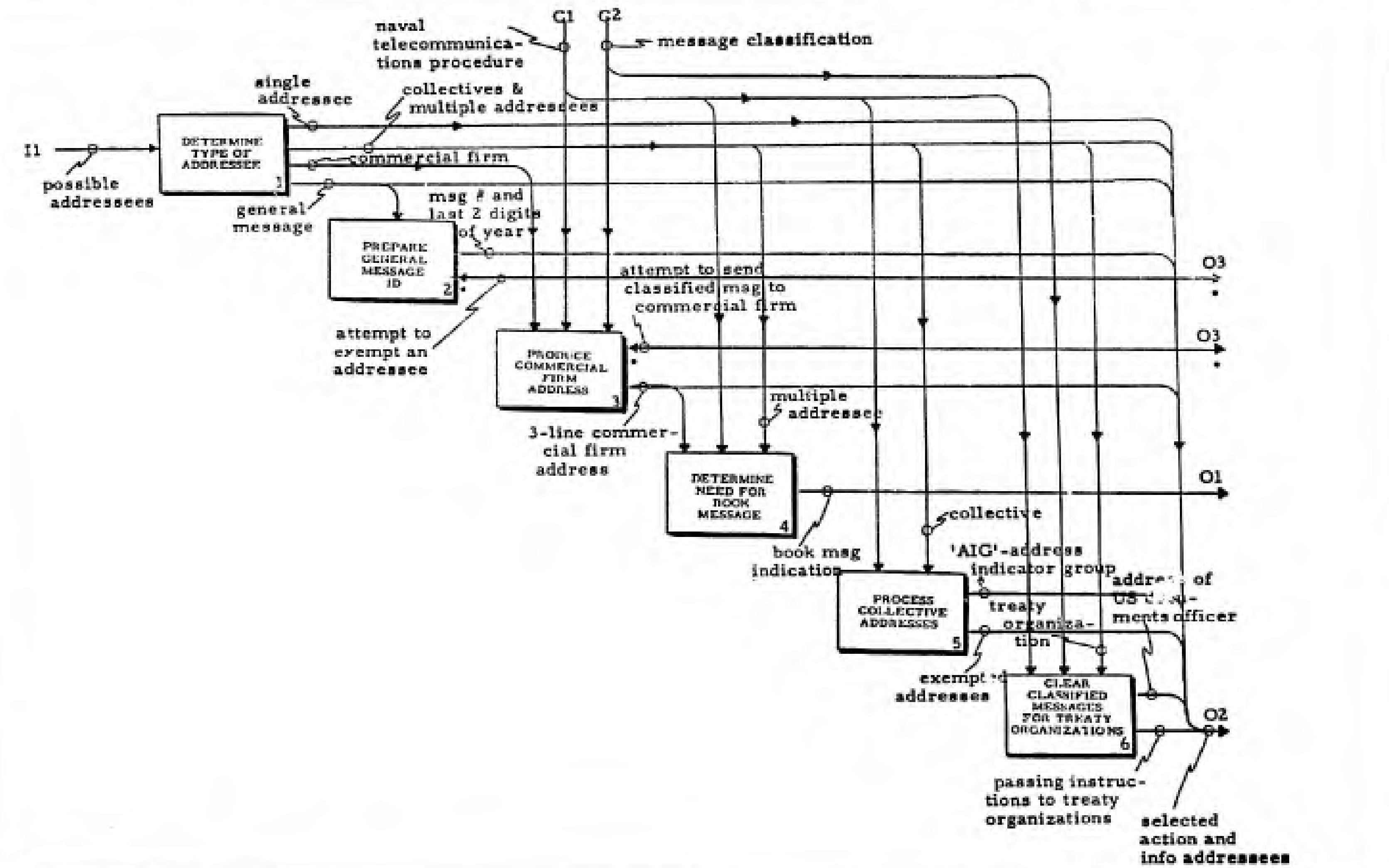
Messages may be addressed to single, separate activities (1O1), or a message may be addressed to a predetermined group of activities called a collective (1O2). A distinction is drawn between addressing messages to separate activities and collectives because addressing procedures are different for each case (1 and 5). When addressing messages to collectives, the originator may desire to exempt activities from receiving the message (5O2). Collectives are indicated by an Address Indicator Group (5O1).

In some cases, a message is to be sent to a commercial firm (1O3). In this case the firm's three line address must be used (3O2). Classified messages however (3C3) must not be sent to commercial firms (3O1). In addition, classified messages (6C2) which are to be released into Treaty Organizations (6C3) (e.g., NATO, SEATO, and CENTO) must be cleared (6) for disclosure authorization prior to transmission. Messages will be addressed to the appropriate U.S. documents officer (6O1) and will include internal passing instructions (6O2) indicating the ultimate addressee.

Finally, some messages are designed to meet recurring requirements for the dissemination of information to a wide, predetermined standard distribution. Such messages are called general messages (2). A consecutive serial number will be assigned to each general message (2O1) according to title, and this number consists of the message number, a slant sign, and the last two digits of the year. Note that exemption of addressees of general message is not permitted (2O2).



STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc. 460 Totten Pond Road, Waltham, Mass 02154
Available in pads of 50 sheets



NODE: A11	TITLE: SELECT ADDRESSEES	NUMBER: RS72(RS64) A11

A12 SELECT PRECEDENCE AND METHOD OF TRANSMISSION

The precedence system enables message drafters to indicate a desired writer to reader delivery time. There are five precedence categories: routine, priority, immediate, flash and emergency command precedence. The assignment of precedence indicates to the drafter the desired speed of delivery of addressees, to the telecommunications center the relative order of processing and delivery, and to the addressees, the relative order in which they should note the message (however, precedence has no direct effect on the time that a reply must be sent or on the precedence to be assigned to that reply).

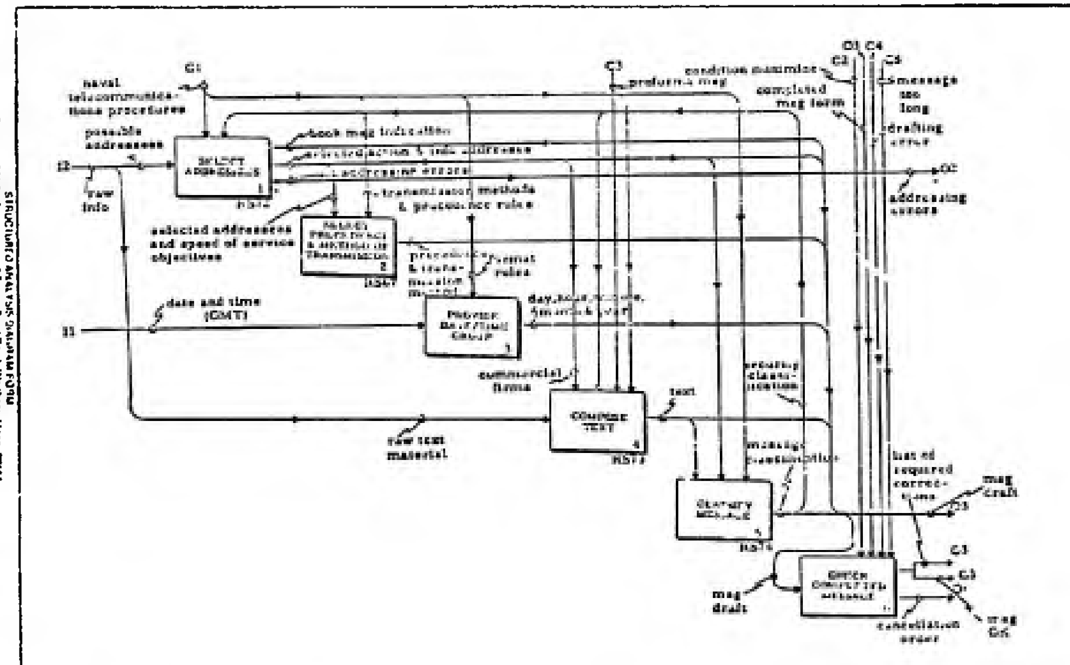
The assignment of precedence is the drafter's responsibility, although the releaser confirms (or may change) the assignment. In order for the system to work, assignment must be based on urgency rather than importance of the subject matter.

Routine (1) is the precedence assigned to all types of traffic which justify electrical transmission but which are not of sufficient urgency to require a higher precedence. Expected delivery time is six hours (1C2).

Priority (2) is reserved for messages which furnish essential information for the conduct of operations in progress. This is the highest precedence normally authorized for administrative messages. Expected delivery time is three hours (2C2).

Immediate (3) is the precedence reserved for messages relating to situations which gravely affect the national forces or populace and which require immediate delivery to addressees. Expected delivery time is 30 minutes (3C2).

Flash precedence (4) is reserved for initial enemy contact reports or operational combat messages of

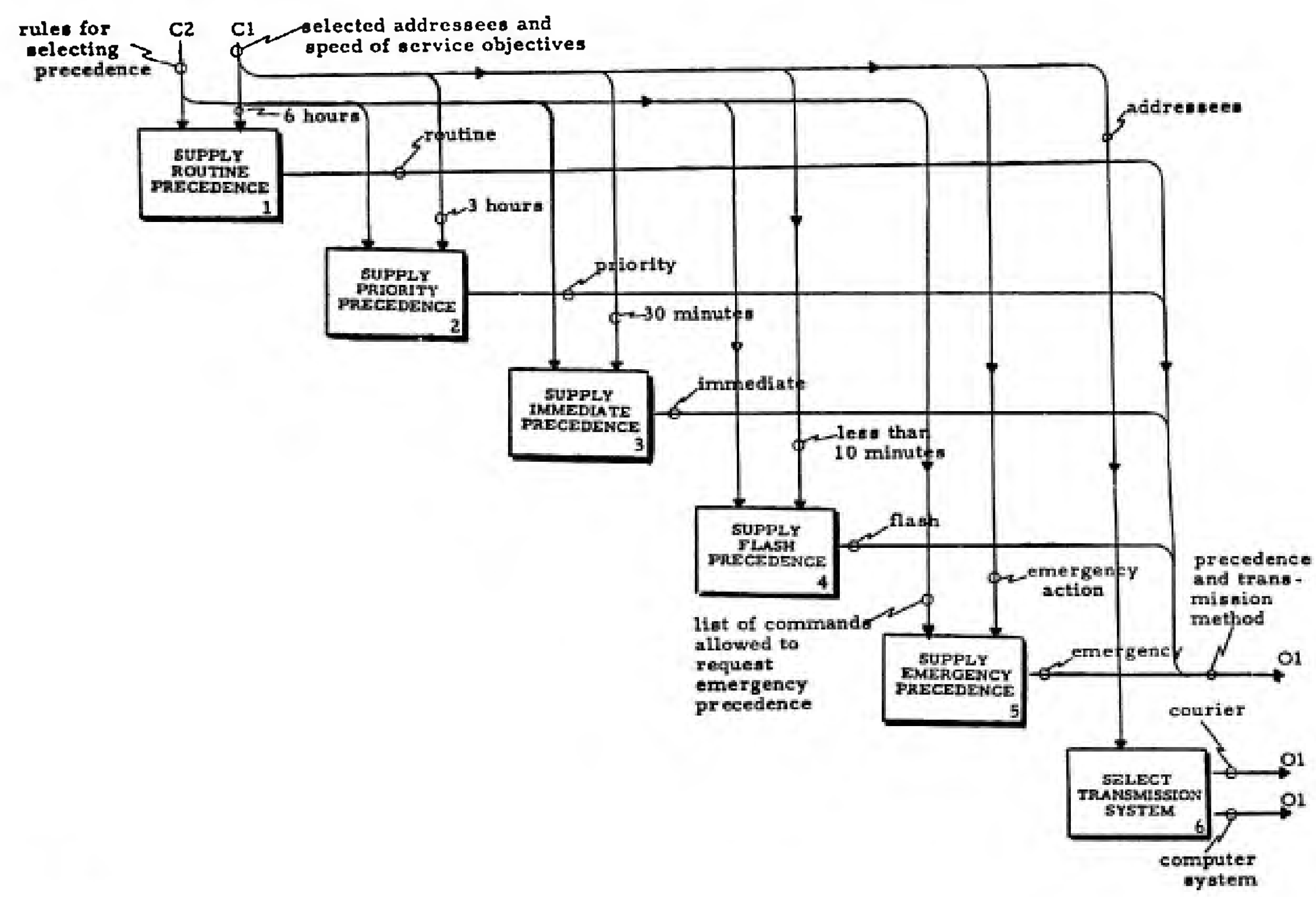


NODE: A1	TITLE: DRAFT MESSAGE	NUMBER: RS71 (RS..1) A1

STRUCTURED ANALYSIS DIAGRAM FORM
Form - 1974 SofTech, Inc. 460 Totten Pond Road, Waltham, Mass 02154
Available in pads of 50 sheets

extreme urgency. Message brevity is mandatory. Expected delivery time is less than ten minutes (4C2).

In addition to the four precedences listed above, a flash preempt capability designated Emergency Command Precedence (5) exists within the AUTODIN system. The use of the precedence is limited to the National Command Authority and certain designated commanders of unified and specified commands (5C1) and then only for specifically designated emergency action command and control messages (5C2).



NODE:	TITLE:	NUMBER:
A12	SELECT PRECEDENCE AND METHOD OF TRANSMISSION	A12 RS67(RS66)

A14 COMPOSE TEXT

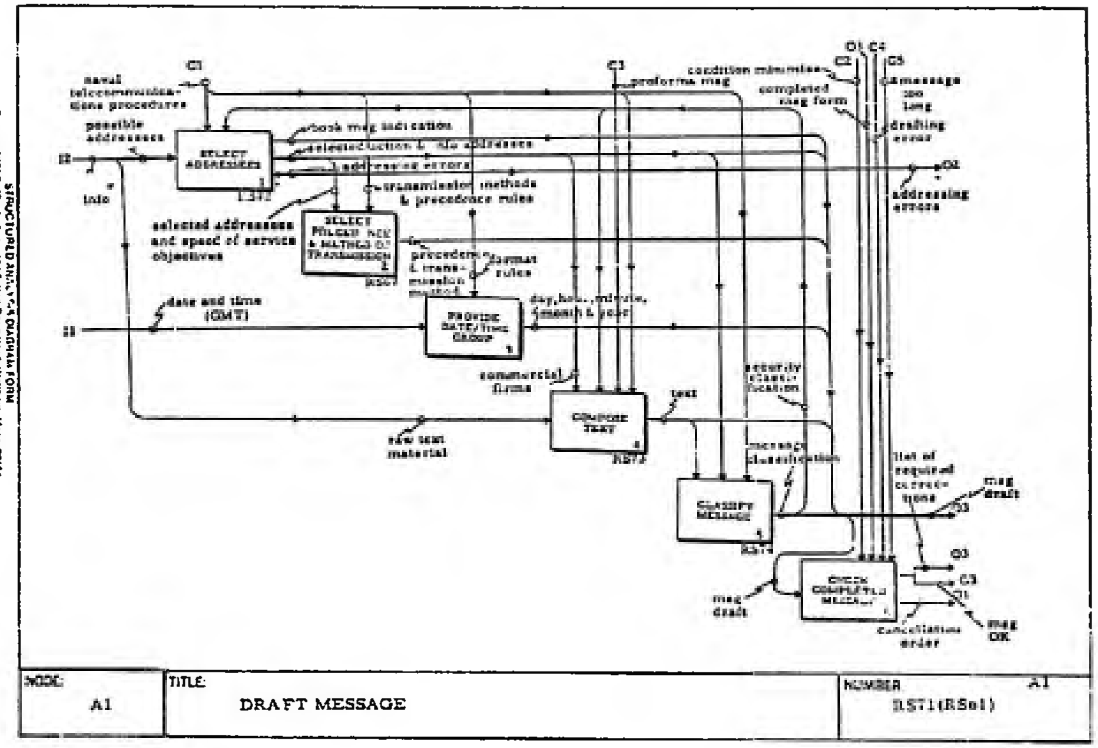
The text (5) is that part of the message which contains the thought or idea the drafter desires to communicate. Frequently, messages must address several subjects or several aspects of one subject. For this reason, textual material is divided into paragraphs and subparagraphs which are numbered and lettered consecutively, respectively (5O1). On classified messages, paragraphs must be appropriately classified (5C1).

The text may be preceeded by a subject line (3O1) when required for clarity. The subject line of a classified message must be marked with the appropriate classification symbol (3C1) if the subject line contains classified messages. A Standard Subject Identification Code (SSIC) (1O1) must be supplied for all Navy originated messages except when the message is addressed only to a commercial firm (1C2).

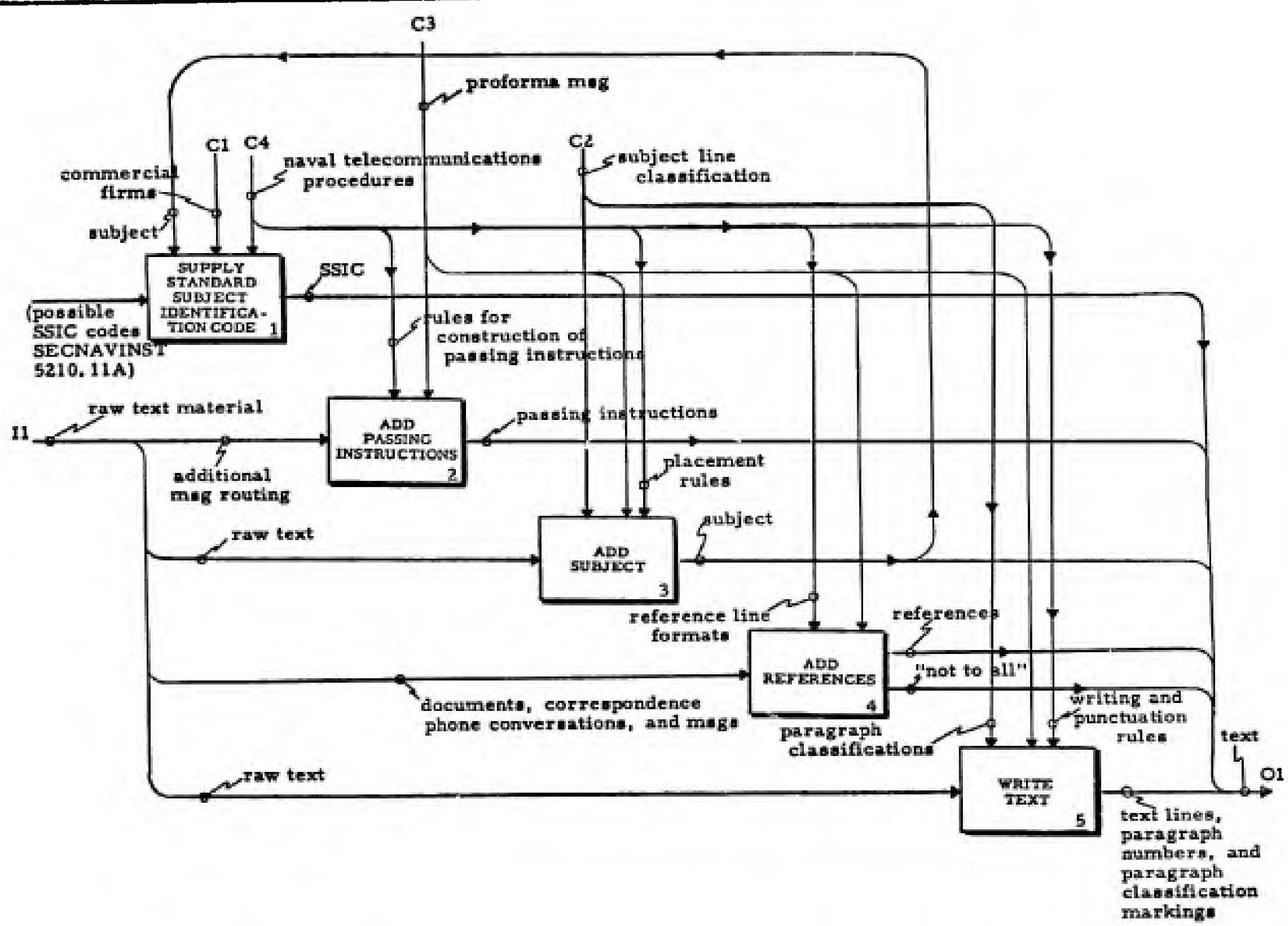
Reference lines (4O1) may be employed as an alternative to repeating lengthy references within the text of a message. References may be made to documents, correspondence, messages, or telephone conversations (4I1).

It is frequently necessary, particularly in the case of multiple address messages, to reference a document or message which all of the addressees do not hold. In such a case, the acronym NOTAL which means "Not to All" is appended in parenthesis to the reference (4O2).

Instructions within the text of a message naming a particular office or individual to whom the originator wants the message routed (2I1) are called passing instructions (2O1). Automated message processing systems do not route traffic according to passing instructions. While a manual oriented message processing center will route a message with passing instructions to the named office or individual, their inclusion will not limit distribution or cause special handling.



STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc. 460 Totten Pond Road, Waltham, Mass. 02154
Available in pads of 50 sheets



NODE: A14	TITLE: COMPOSE TEXT	NUMBER: A14 RS73(RS68)

A15 CLASSIFY MESSAGE

The classification of a message is the first line of the text in addition to any special handling markings. The classification must be one of the three classification designators (2O1) or the word UNCLAS for unclassified (1O1).

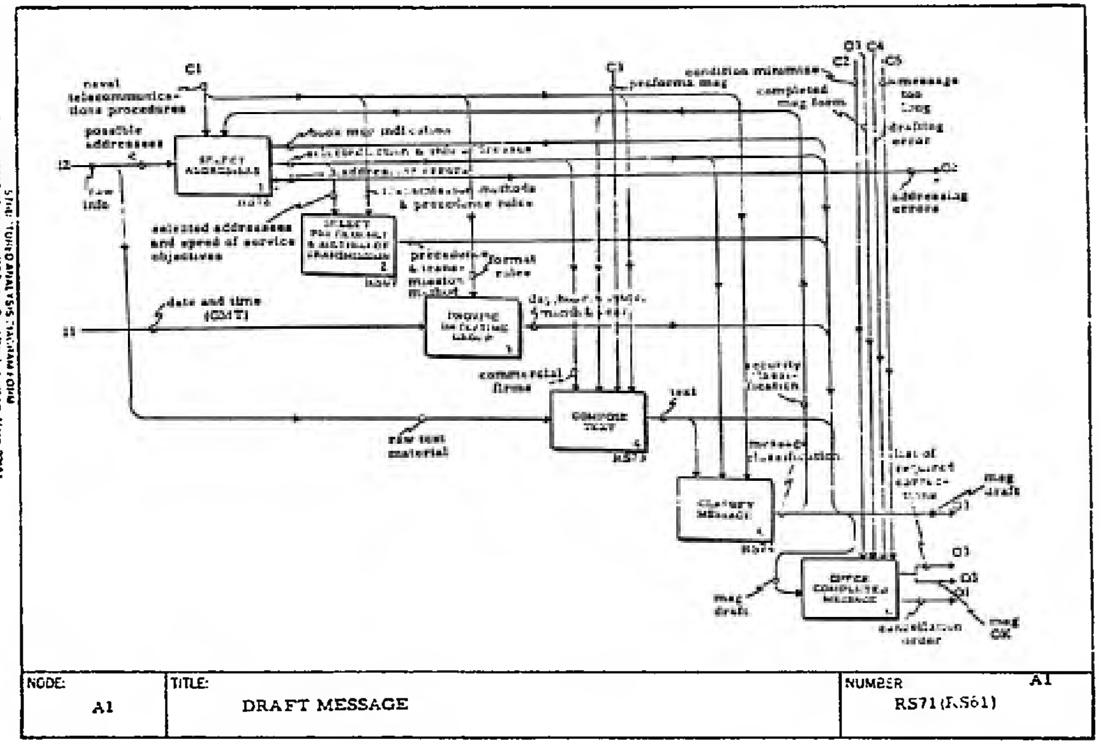
Unclassified messages, which meet the criteria of SECNAVINST 5720.42A, will be designated For Official Use Only (FOUO) (1O1), and handled in accordance with SECNAVINST 5570.2. The classification line of such a message would read "UNCLAS FOUO."

The caveat EFTO is a special handling designation which means Encrypt For Transmission Only. Except those message transmitted on foreign circuitry, and those addressed to foreign nations, NATO, CENTO, SEATO, and U.S. non-DOD agencies, all FOUO messages transmitted outside CONUS will be handled EFTO. The classification line of such a message would read "UNCLAS E F T O FOUO." (1O1)

Certain types of messages require special handling (3) in addition to that provided by the security classification. Markings which indicate special handling requirements are placed in the classification line immediately following the classification.

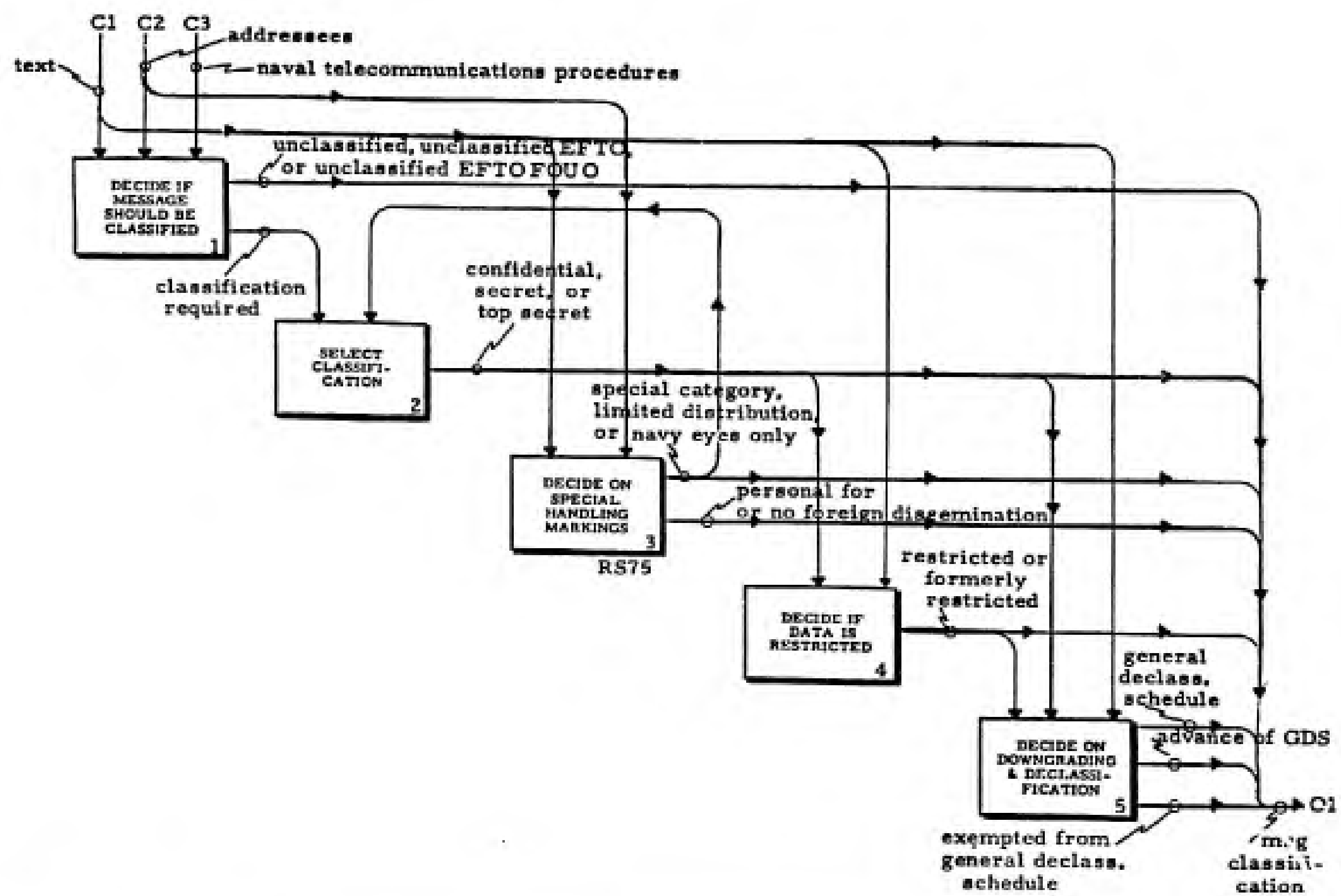
Certain naval nuclear propulsion information, in addition to being classified, is defined as Restricted Data or Formerly Restricted Data according to the Atomic Energy Act of 1954. When such information appears in the text of a message, it is necessary to add the markings (RD) or (FRD) (4O1) in addition to classification markings.

Downgrading and declassification markings must be applied to all classified messages. The markings



STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc. 460 Totten Pond Road Waltham, Mass 02154
Available in pads of 50 sheets

will show either "ADS" (5O2) to indicate declassification in advance of General Declassification Schedule; "GDS" (5O1) for General Declassification Schedule cases; or "XGDS (number of exemption category)" (5O3) to indicate exemption from the General Declassification Schedule. Information marked Restricted Data and Formerly Restricted Data is considered exempt from the General Declassification Schedule (5C1), and declassification markings need not be applied.



NODE:	TITLE:	NUMBER:
A15	CLASSIFY MESSAGE	A15 RS74(RS69)

A153 DECIDE ON SPECIAL HANDLING MARKINGS

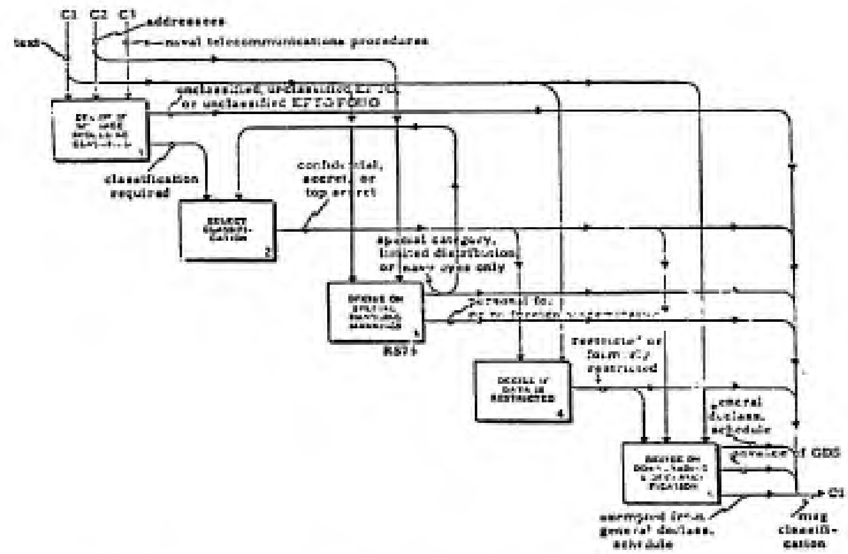
The special handling designator SPECAT (1O1) is the marking applied to Special Category messages. There are two types of SPECAT messages, those associated with code words or protects (1C1) (e.g., SPECAT SIOP-ESI), and those which are delivered initially only to a named individual (1C2).

The designator LIMDIS is the marking applied to messages which due to subject matter (2C1) require limited distribution within the addressed activity. The classification line of a secret message requiring limited distribution would read, "S E C R E T LIMDIS" (2O1).

The designator PERSONAL FOR (3O1) is the marking applied to those messages whose distribution is limited to the named recipient (3C1) (who may upon receipt direct further distribution).

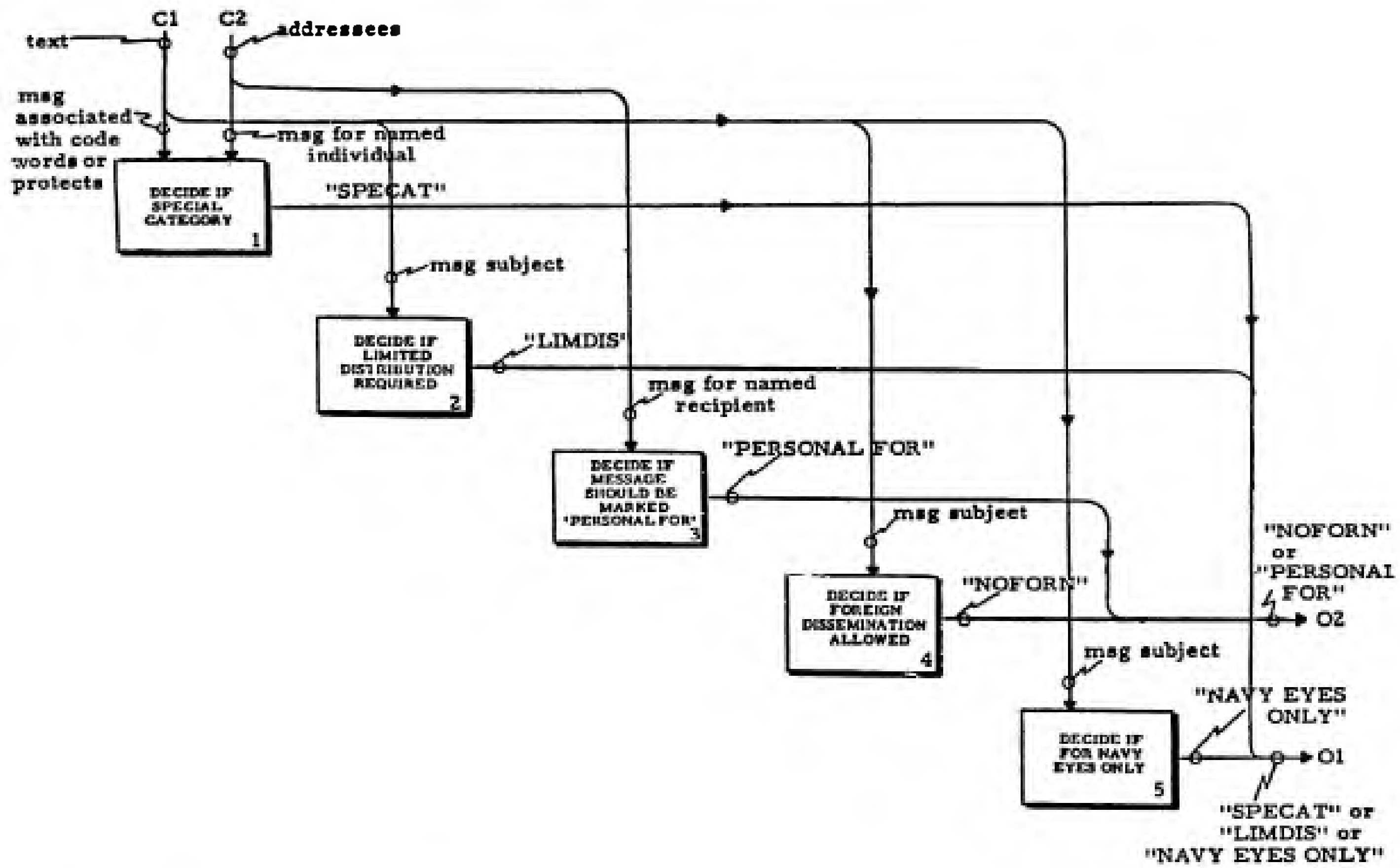
The designator NOFORN (4O1) is the shortened version of "No Foreign Dissemination." This marking may be applied only to intelligence information, which is normally excluded from general purpose Navy circuits, and Navy nuclear propulsion information (4C1).

The caveat NAVY EYES ONLY (5O1) may be applied to classified messages which warrant handling by Navy personnel only in order to preclude access by representatives of the other military services (5C1).



NODE: A15	TITLE: CLASSIFY MESSAGE	NUMBER: RS74(RS69) A15

STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc. 460 Totten Pond Road, Waltham, Mass. 02154
Available in pads of 50 sheets



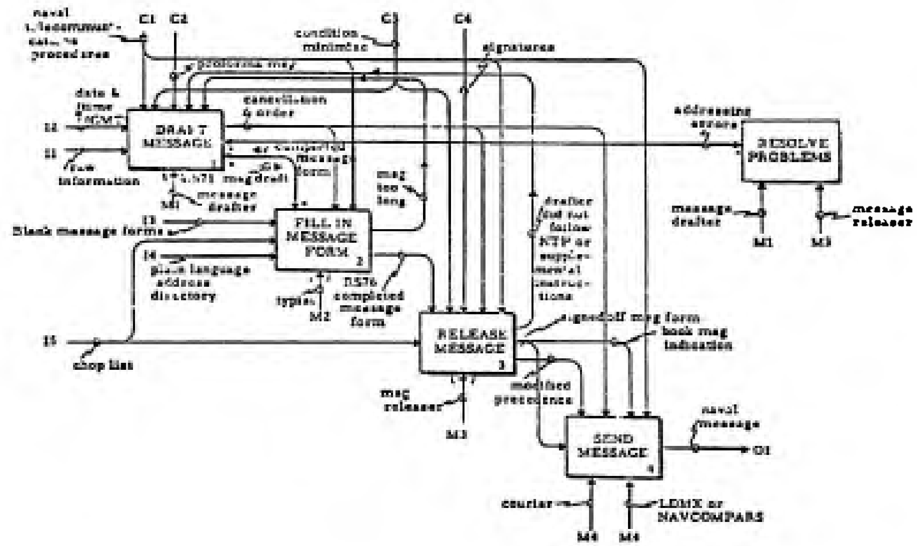
NODE:	TITLE:	NUMBER:
A153	DECIDE ON SPECIAL HANDLING MARKINGS	A153 RS75(RS70)

A2 FILL IN MESSAGE FORM

Once the message drafter has produced a message draft (A2. C1). the message must be typed either on OCR form DD-173 or other non-OCR forms (I11) depending on whether the message is to be sent by LDMX or NAVCOMPARS or some other non-computer based transmission system. If messages are to be typed on OCR forms, very specific formatting rules (A2. C3) must be followed by the typist. Since LDMX and NAVCOMPARS have specific page limits (40 and 99 pages, respectively) (1C1), the typist must inform the drafter (1O1) if an OCR message exceeds the page limits.

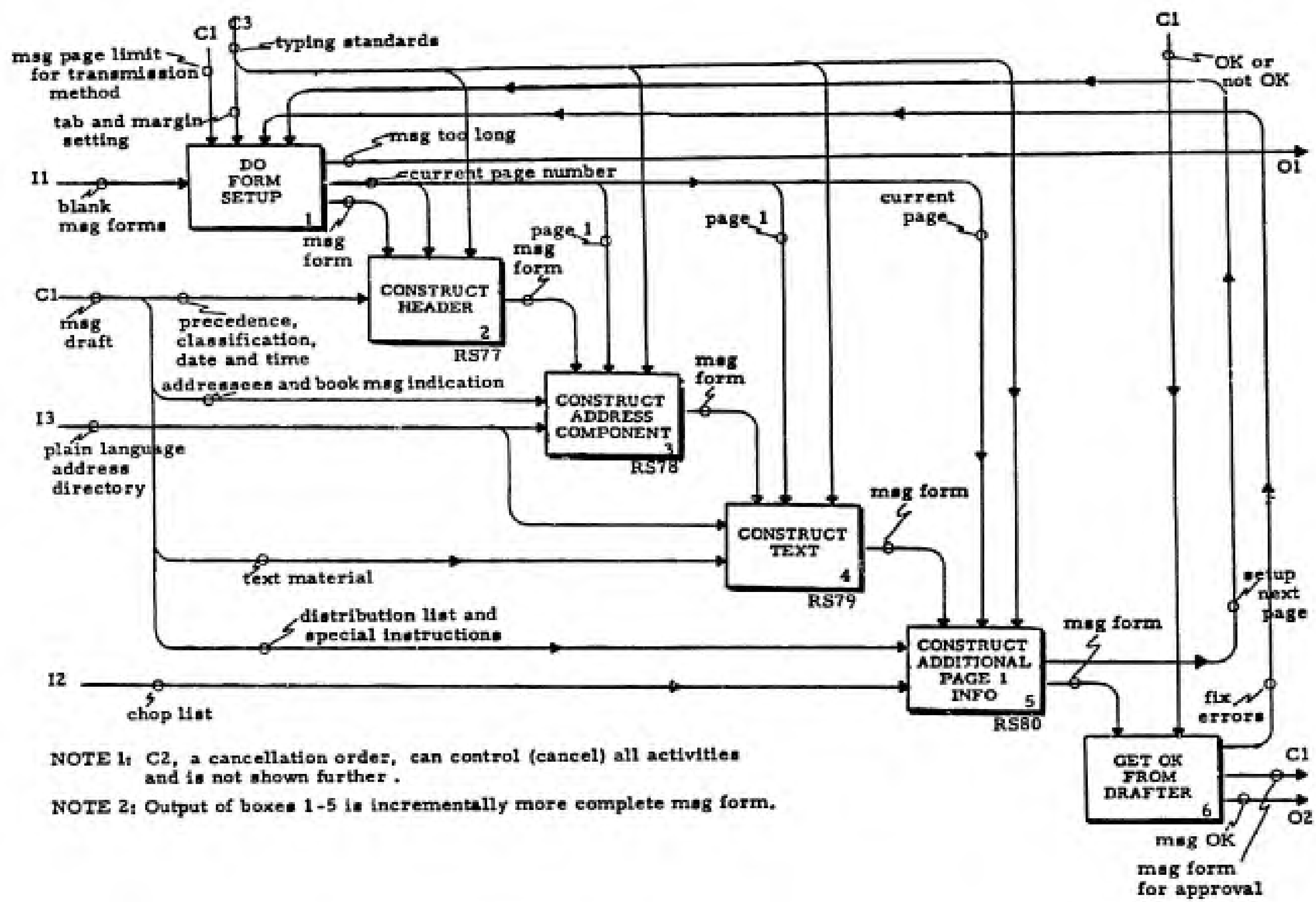
Certain information is typed on only the first page of a multipage message. The message header (2) contains the page number, drafter or releaser time block, precedence, classification, and message handling instructions (2I1 and 2C2). However, only the page number, drafter or releaser time, and classification must be repeated on all pages. Similarly, the address component (3) goes only on the first page (3C2) unless multiple pages are required to list all the addressees. Finally, the chops (5I2), special instructions (5I1), distribution block, drafter block, and releaser block are filled in on page one only.

Once the message has been typed, it must be submitted to the message drafter (6O2) for verification. If the message contains errors (6C2), they must be corrected (6O1). Otherwise, the message may be sent to the message releaser (6O3).



NODE:	TITLE:	NUMBER:
A0	COMPOSE NAVAL MESSAGES	RS5?(RS55) A0

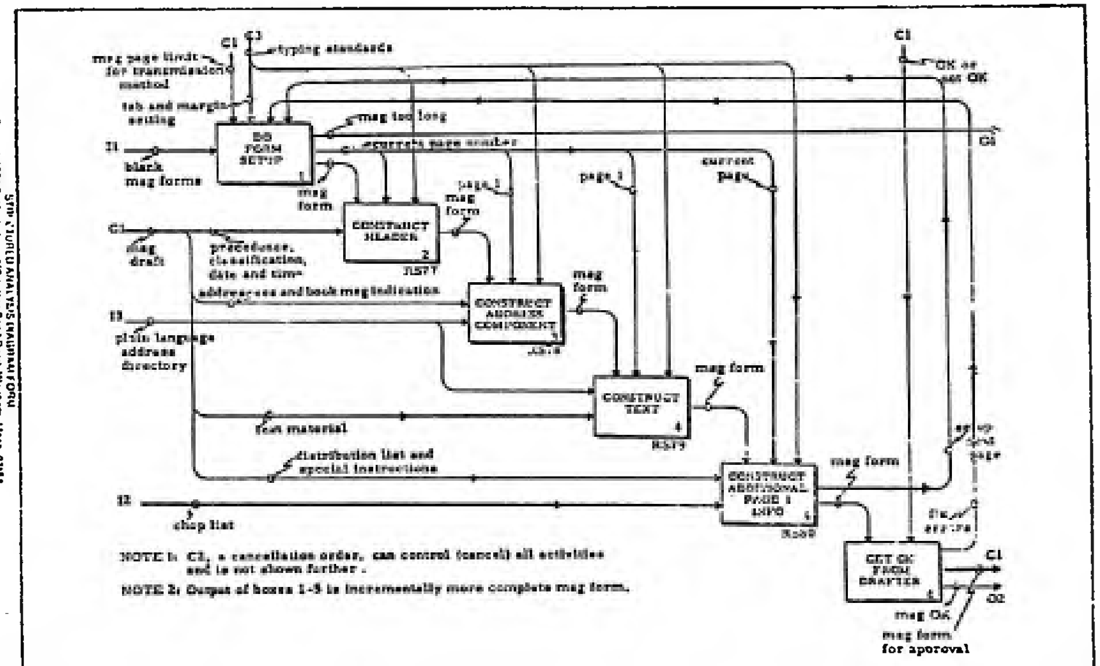
STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc., 460 Totten Pond Road Waltham, Mass 02154
Available in pads of 50 sheets



NOTE 1: C2, a cancellation order, can control (cancel) all activities and is not shown further.

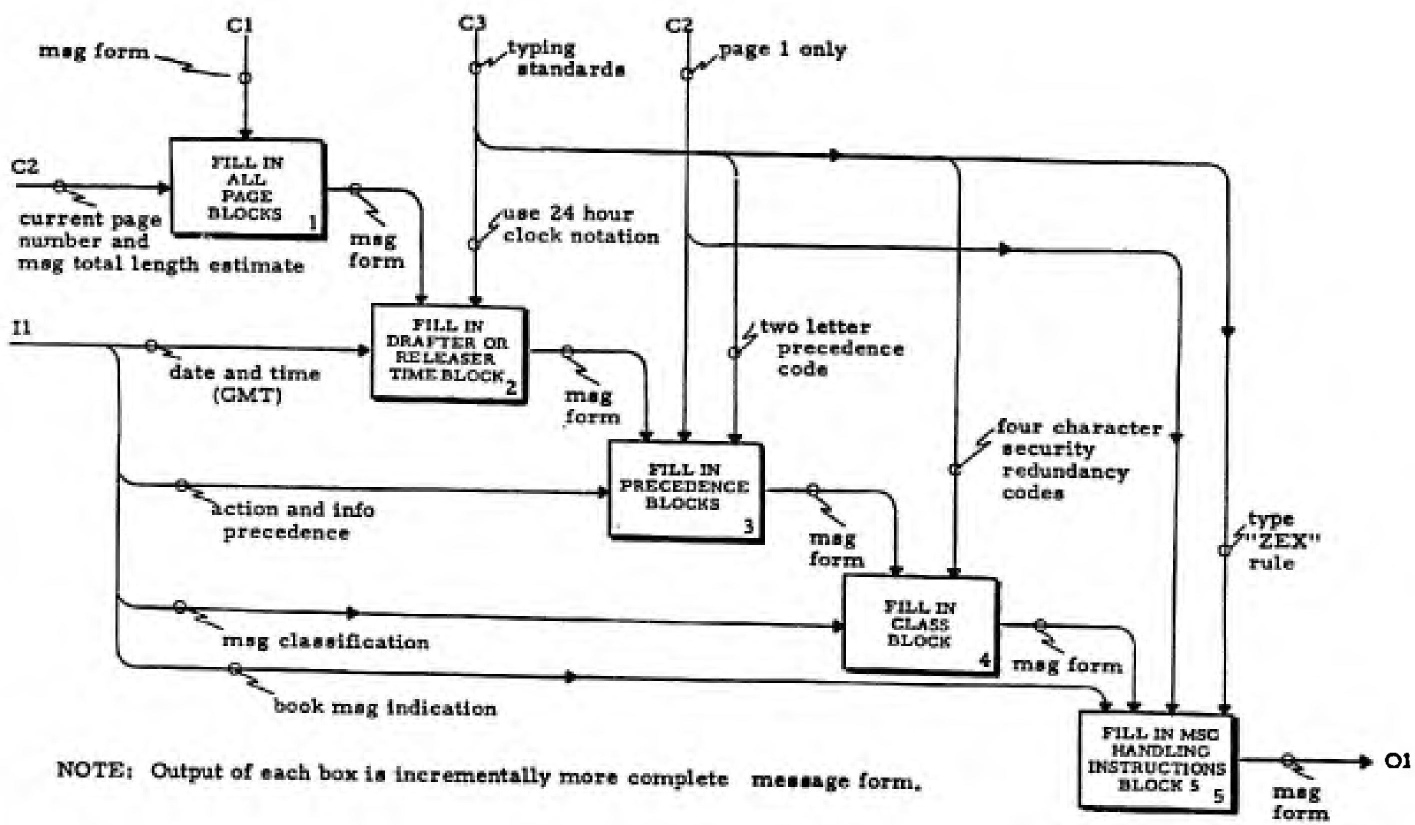
NOTE 2: Output of boxes 1-5 is incrementally more complete msg form.

NODE:	TITLE:	NUMBER:
A2	FILL IN MESSAGE FORM	RS76(RS14) A2



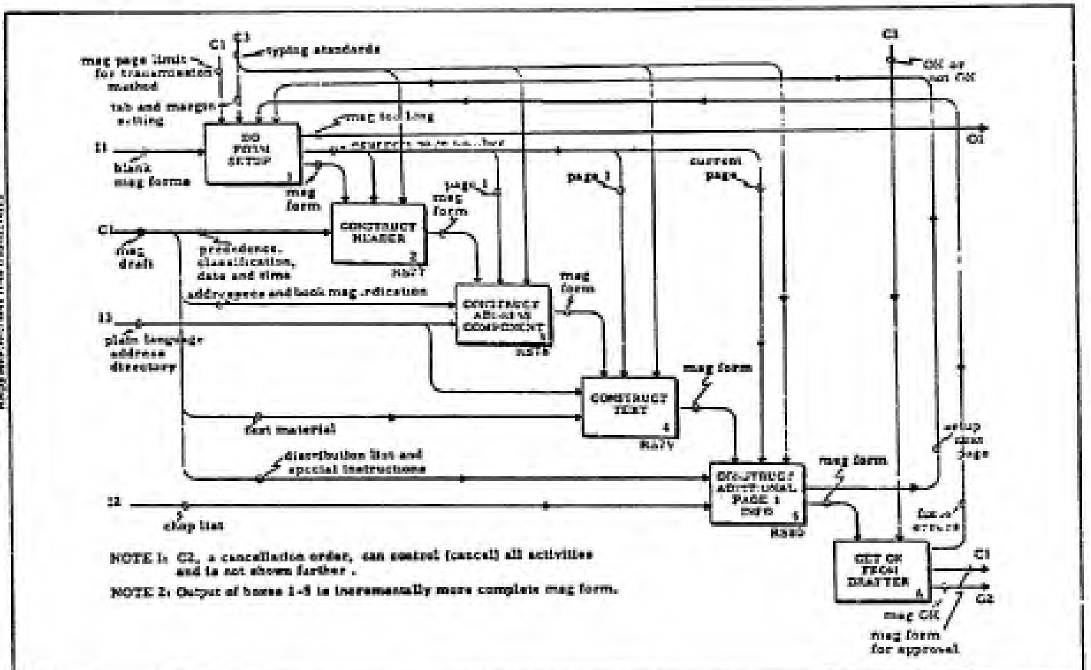
NODE:	TITLE:	NUMBER:
A2	FILL IN MESSAGE FORM	RS76(RS14) A2

STRUCTURED ANALYSIS DIAGRAM FORM
Form © 1974 SofTech, Inc. 460 Totten Pond Road, Waltham, Mass 02154
Available in pads of 50 sheets

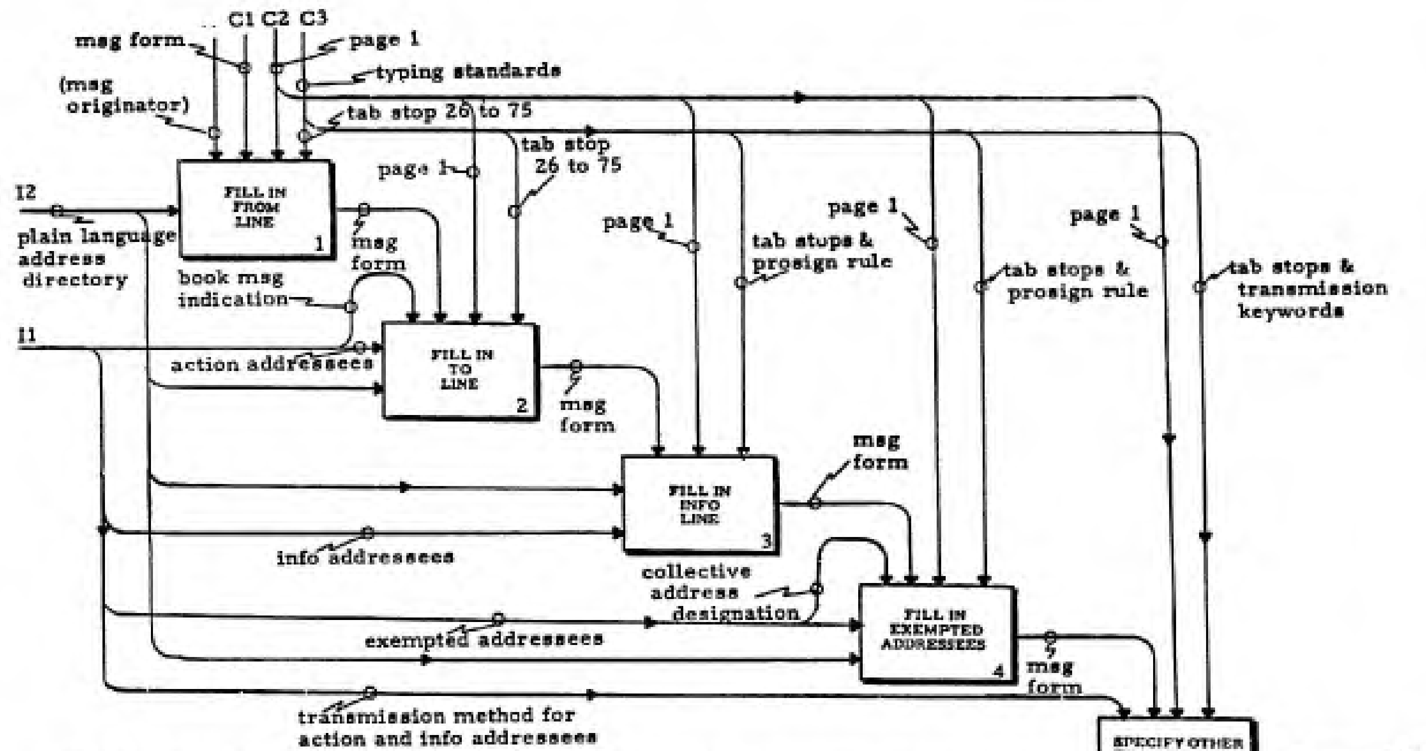


NOTE: Output of each box is incrementally more complete message form.

NODE: A22	TITLE: CONSTRUCT HEADER	NUMBER: RS77(RS17) A22



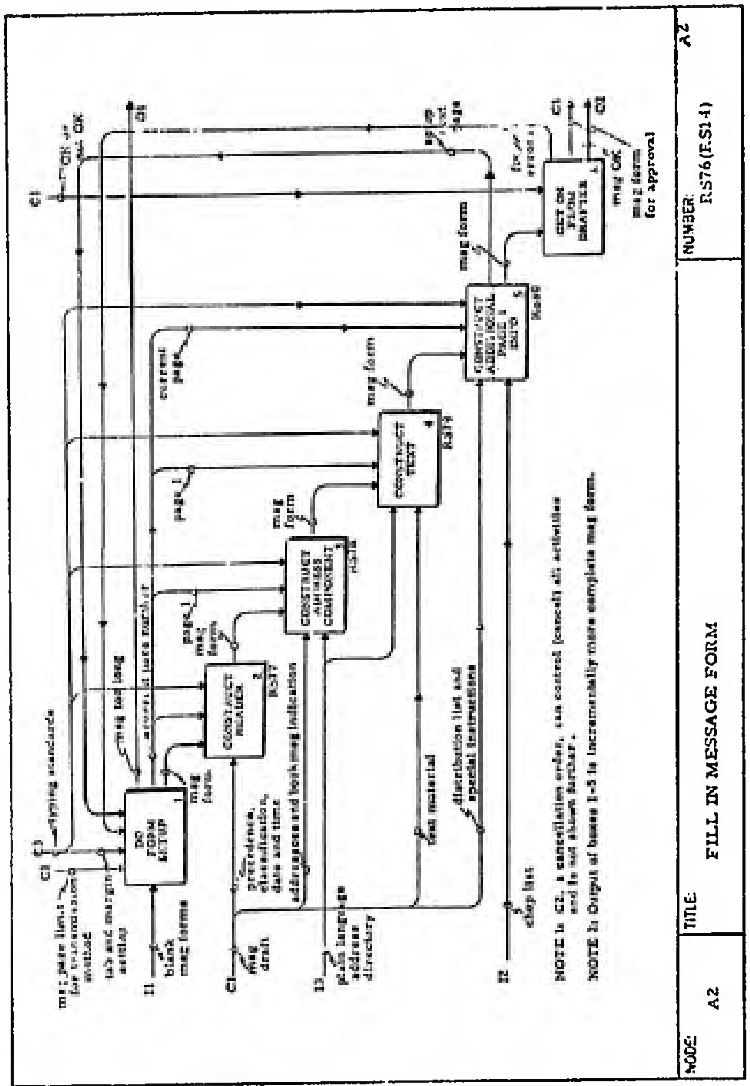
NODE:	TITLE:	NUMBER: A2
A2	FILL IN MESSAGE FORM	RS76(RS14)



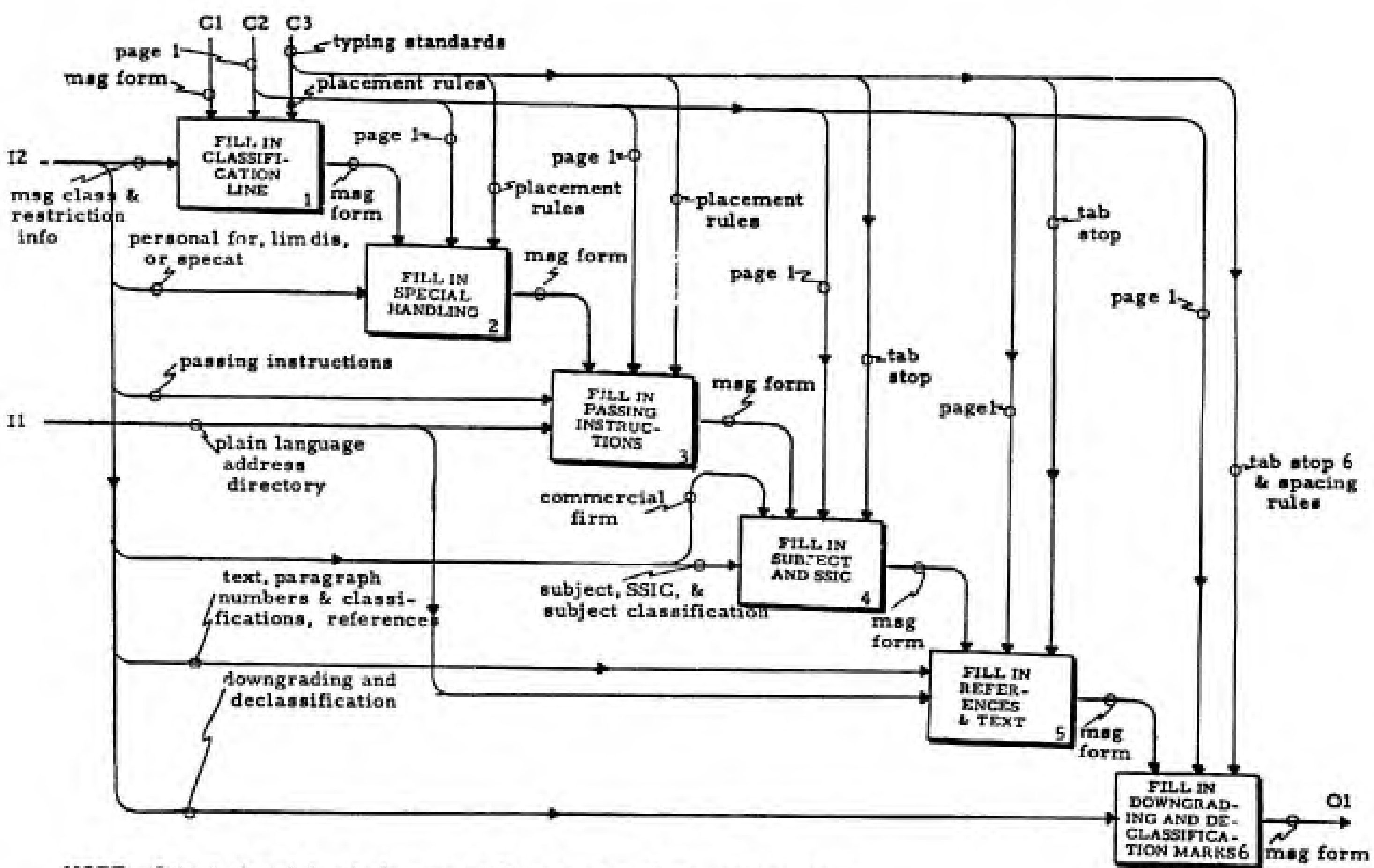
NOTE 1: Output of each box is incrementally more complete message form.

NOTE 2: No address component is constructed on any but the first page (except when additional pages are needed to continue the address component - not shown here).

NODE: A23	TITLE: CONSTRUCT ADDRESS COMPONENT	NUMBER: RS78(RS16) A23

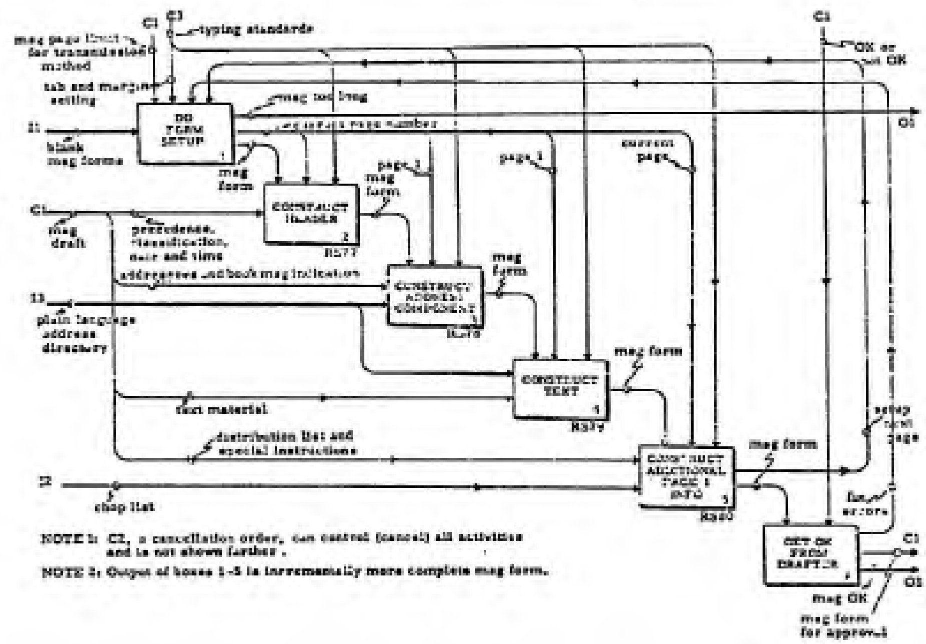


STRUCTURED ANALYSIS DIAGRAM FORM

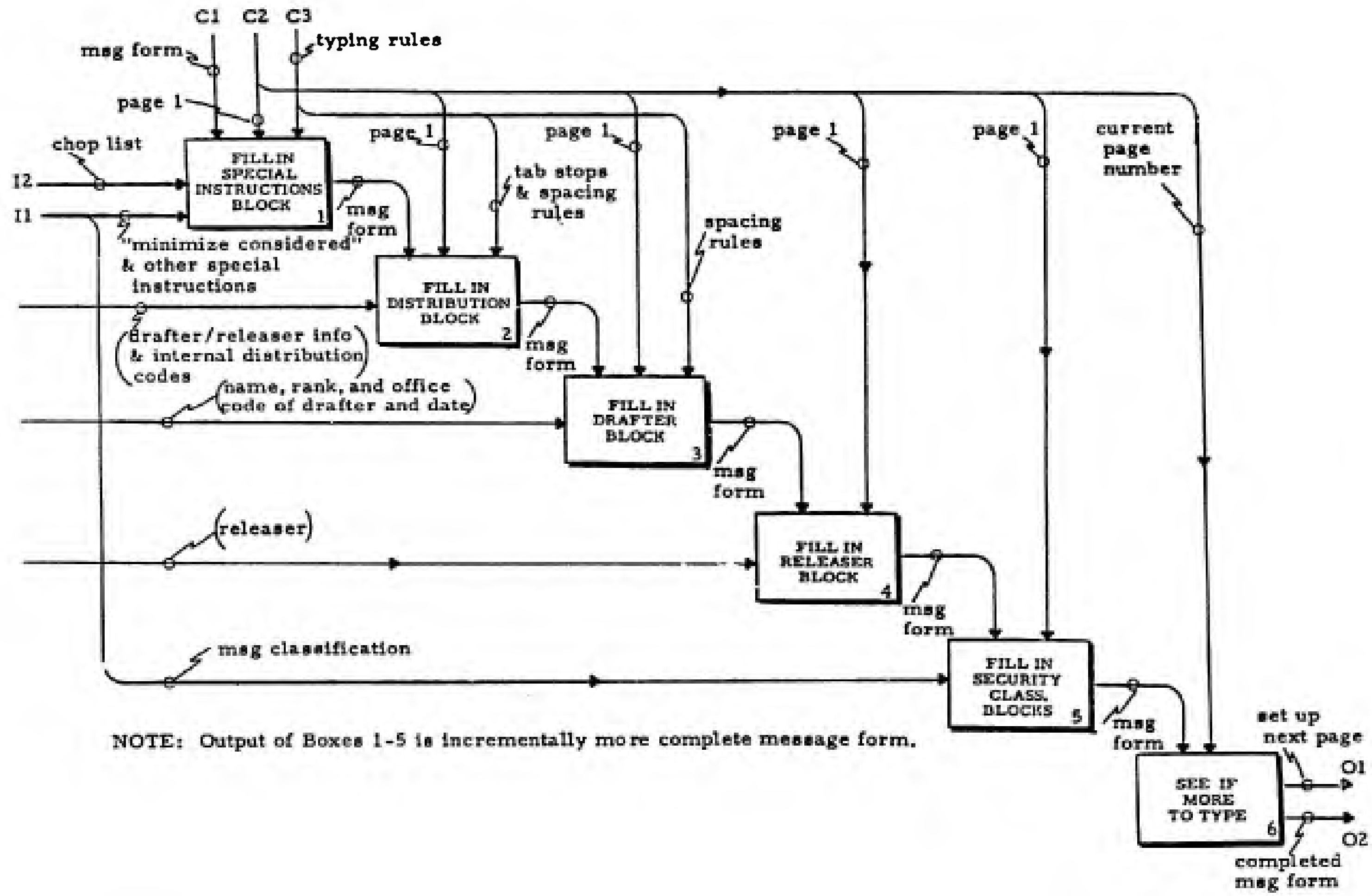


NOTE: Output of each box is incrementally more complete message form.

NODE:	TITLE:	NUMBER:
A24	CONSTRUCT TEXT	RS79(RS18) A24



NODE: A2	TITLE: FILL IN MESSAGE FORM	NUMBER: RS70(RS14) A2



NOTE: Output of Boxes 1-5 is incrementally more complete message form.

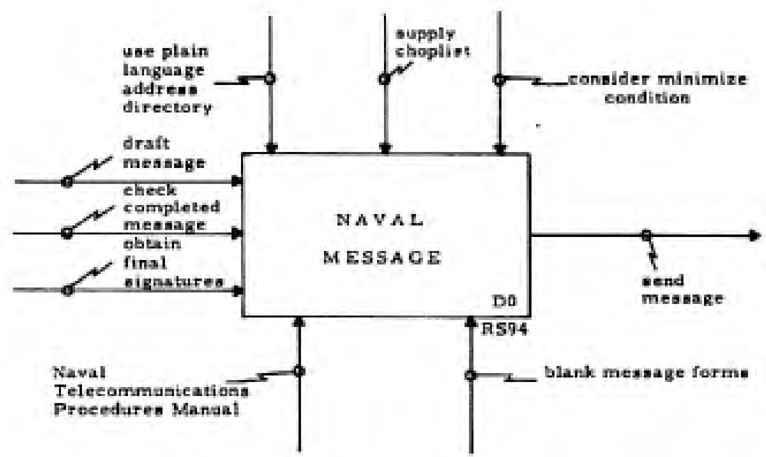
NODE:	TITLE:	NUMBER: A25
A25	CONSTRUCT ADDITIONAL PAGE 1 INFO	RS80(RS19)

Section 4

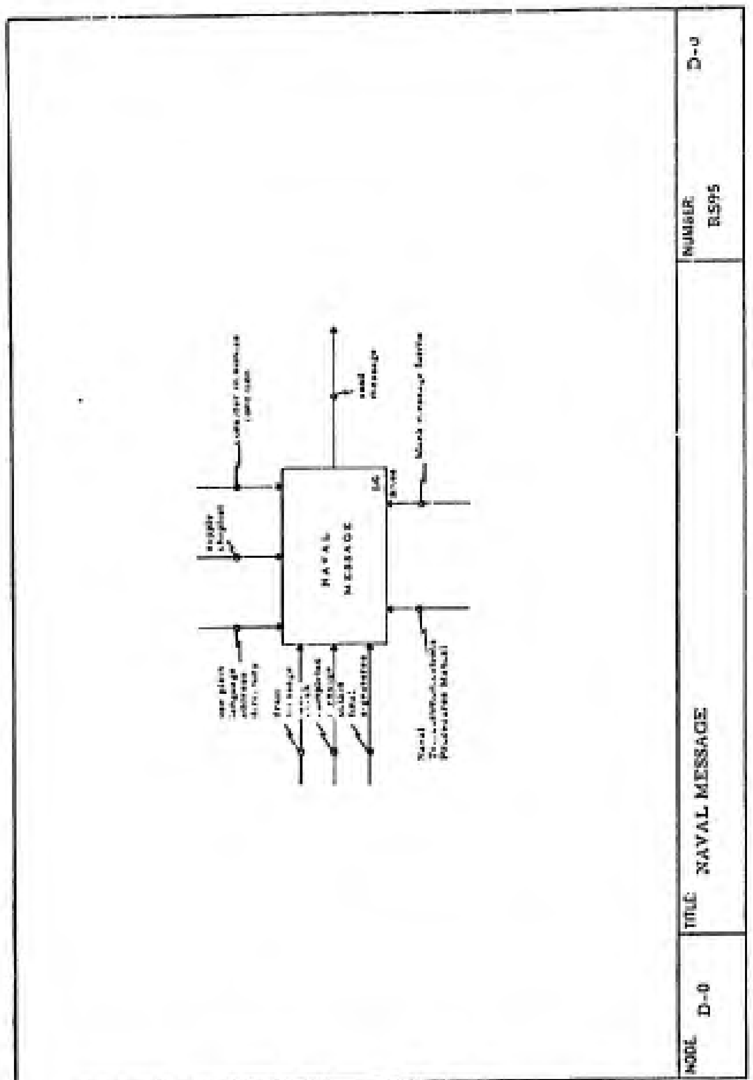
NAVAL MESSAGE DATA MODEL

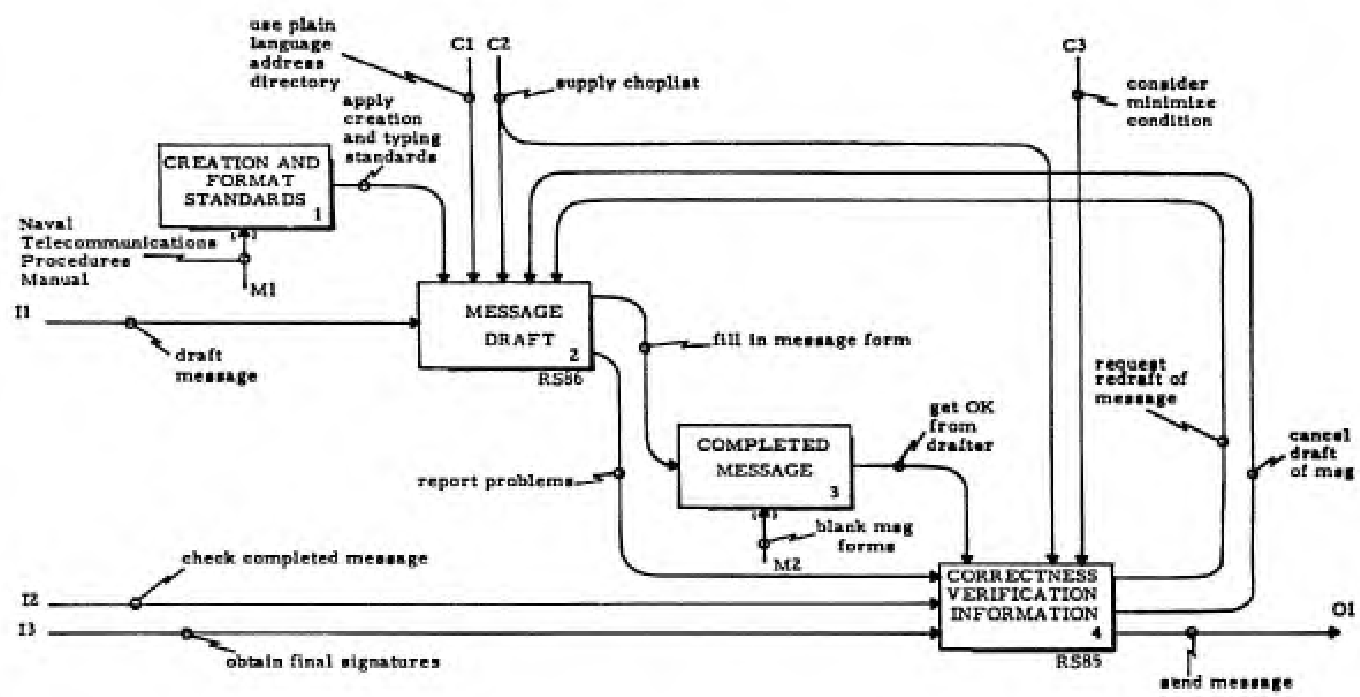
Node Diagram for Naval Message

(Top) NAVAL MESSAGE - PARENT CONTEXT
D0 NAVAL MESSAGE
- D1 CREATION AND FORMAT STANDARDS
- D2 MESSAGE DRAFT
  - D21 MESSAGE CLASSIFICATION
    - D211 SECURITY CLASSIFICATION
      - D2111 UNCLASSIFIED
      - D2112 FOR OFFICIAL USE ONLY
      - D2113 ENCRYPT FOR TRANSMISSION ONLY
      - D2114 CONFIDENTIAL
      - D2115 SECRET
      - D2116 TOP SECRET
    - D212 SPECIAL HANDLING
      - D2121 SPECIAL CATEGORY
      - D2122 LIMITED DISTRIBUTION
      - D2123 PERSONAL FOR
      - D2124 NO FOREIGN DISSEMINATION
      - D2125 NAVY EYES ONLY
    - D213 RESTRICTED
    - D214 DOWNGRADING AND DECLASSIFICATION
  - D22 ADDRESSING INFORMATION
    - D221 PASSING INSTRUCTIONS
    - D222 SINGLE ADDRESSEE
    - D223 COLLECTIVES AND MULTIPLE ADDRESSEES
    - D224 COMMERCIAL FIRM ADDRESS
    - D225 GENERAL MESSAGE
    - D226 BOOK MESSAGE INDICATION
  - D23 TRANSMISSION CONTROL
    - D231 ROUTINE PRECEDENCE
    - D232 PRIORITY PRECEDENCE
    - D233 IMMEDIATE PRECEDENCE
    - D234 FLASH PRECEDENCE
    - D235 EMERGENCY PRECEDENCE
    - D236 COMPUTER OR COURIER TRANSMISSION
  - D24 TEXT
    - D241 TEXT LINES
    - D242 REFERENCES
    - D243 SUBJECT
    - D244 SSIC
    - D245 PARAGRAPH NUMBERS
  - D25 MESSAGE OVERHEAD DATA
- D3 COMPLETED MESSAGE FORM
- D4 CORRECTNESS VERIFICATION INFORMATION
  - D41 DRAFTER OK OR NOT OK
  - D42 PROBLEMS
  - D43 CANCELLATION ORDER
  - D44 CHOP LIST
  - D45 SIGNATURES

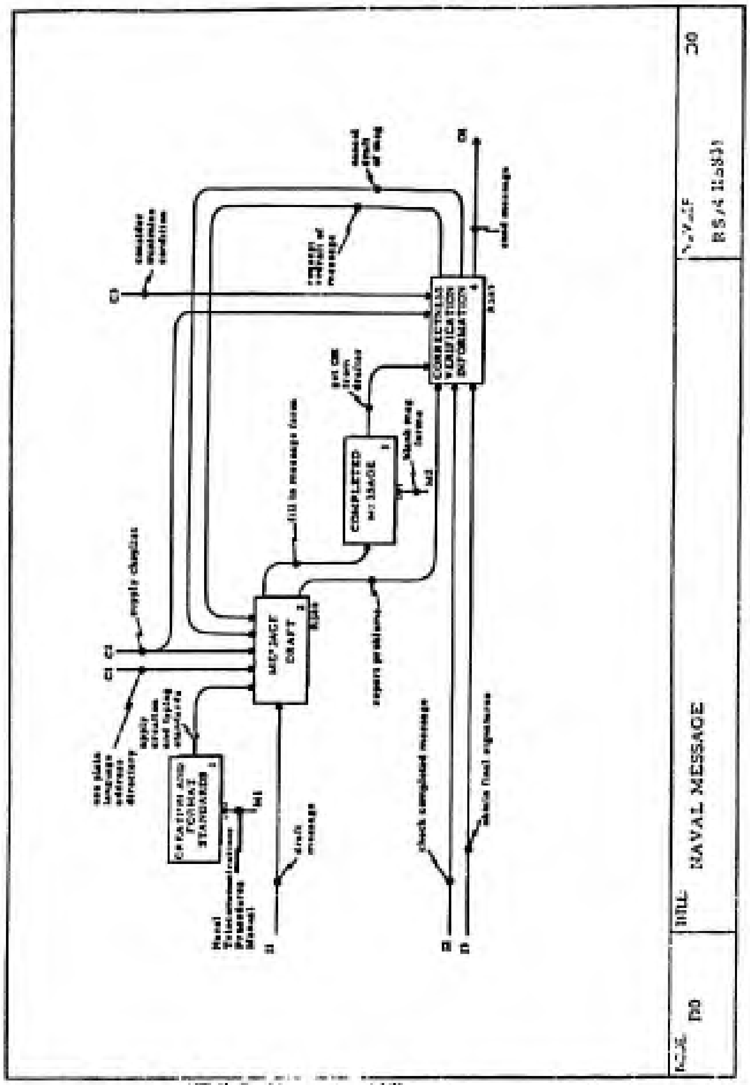


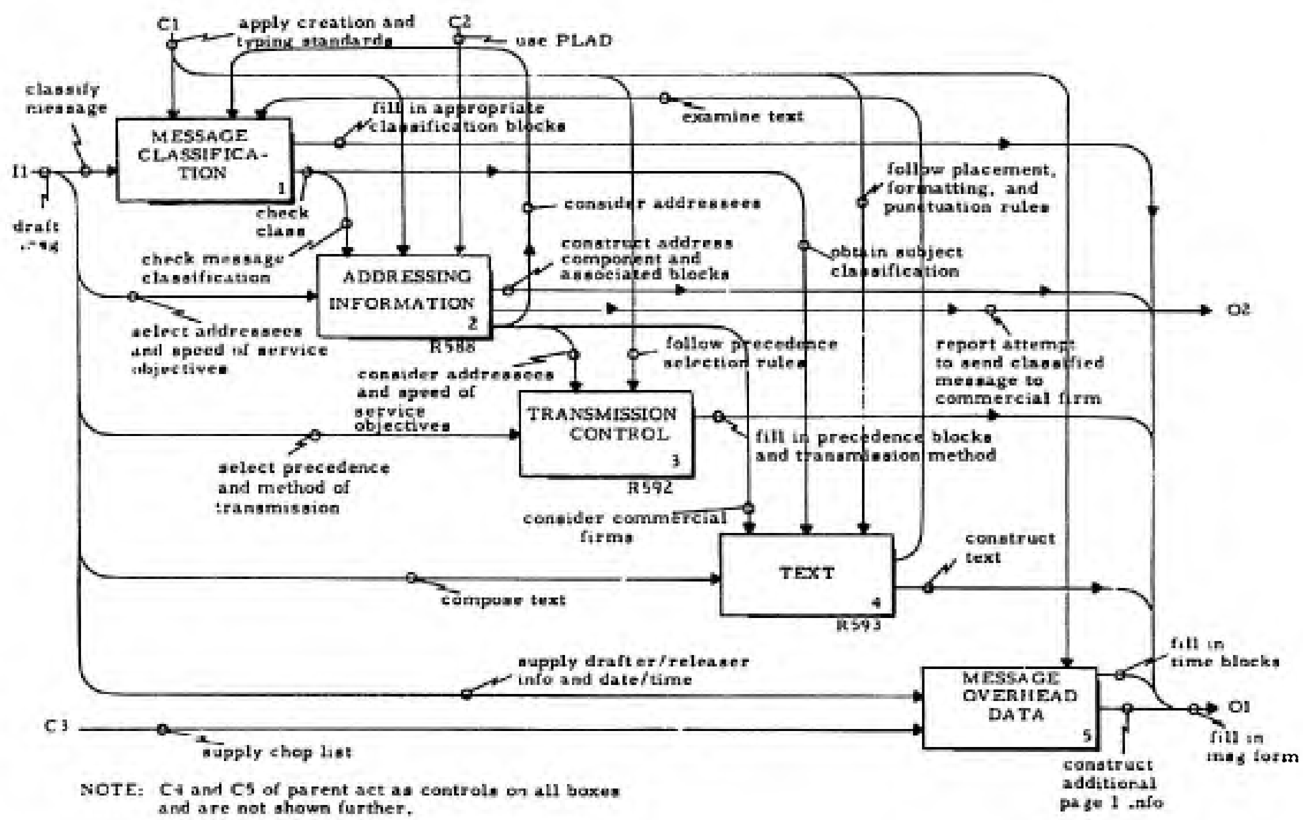
NODE: D-0	TITLE: NAVAL MESSAGE	NUMBER: RS95 D-0


NAVAL MESSAGE
NODE: D-0
TITLE: NAVAL MESSAGE
NUMBER: B595
D-0

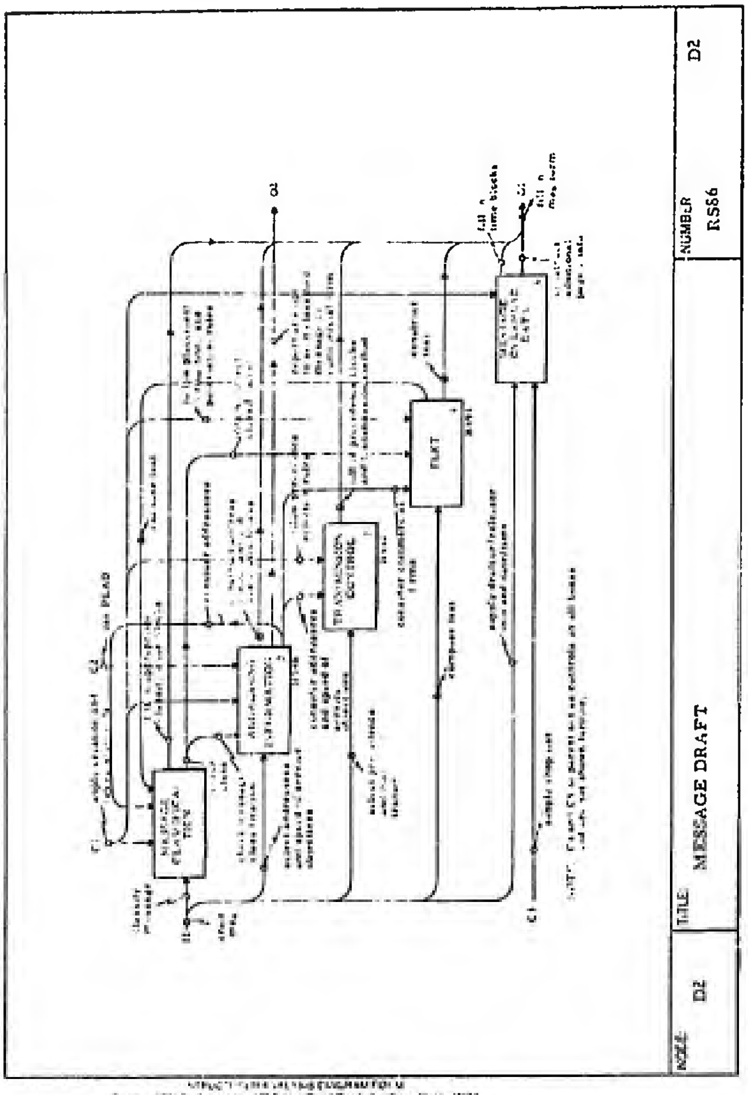


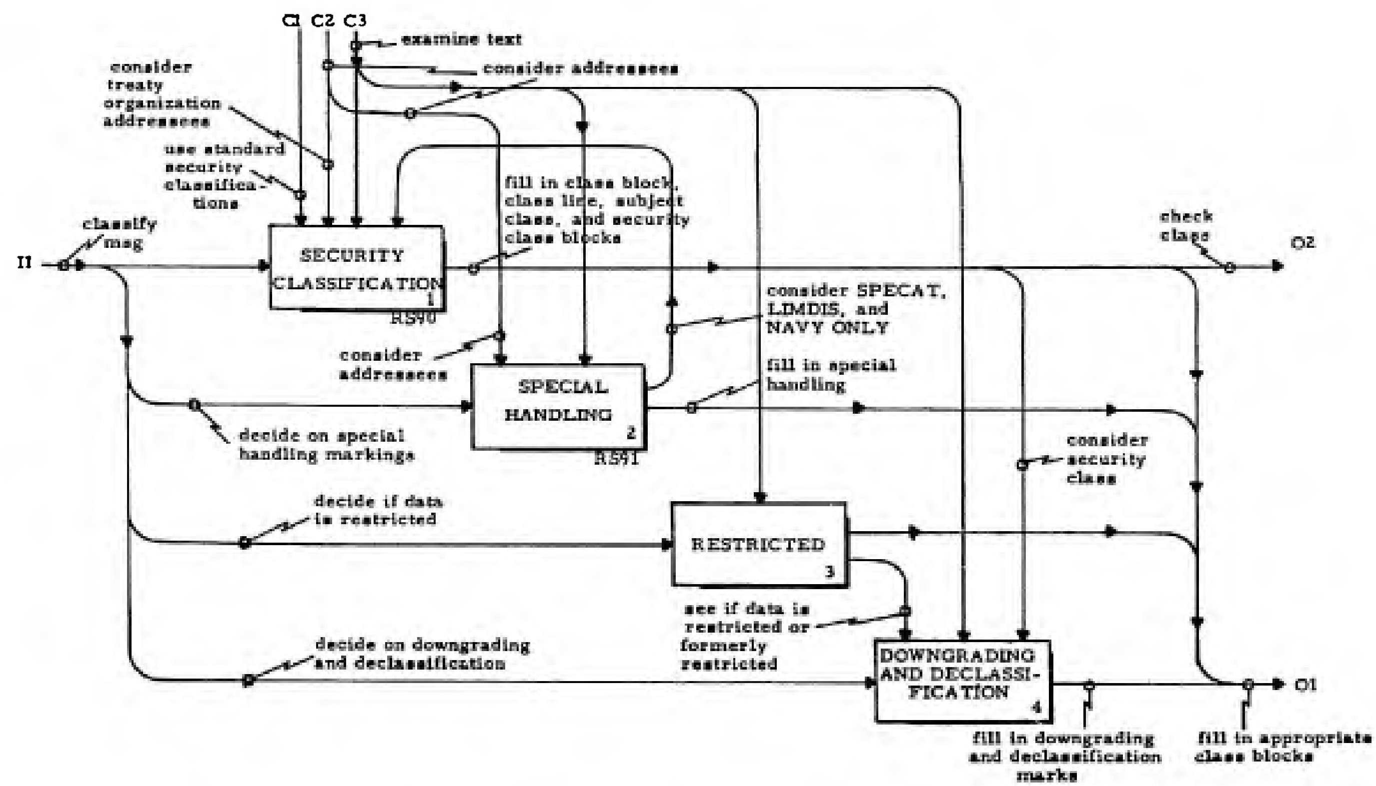
NODE: D0	TITLE: NAVAL MESSAGE	NUMBER: RS94(RS83) D0


OPERATION AND FORMAT STANDARDS
Naval Telecommunications Procedures Manual
draft message
supply checklist
fill in message form
COMPLETED MESSAGE
get OK from drafter
blank msg forms
report problems
check completed message
obtain final signatures
CORRECTNESS VERIFICATION INFORMATION
consider minimize condition
send message
NAVAL MESSAGE

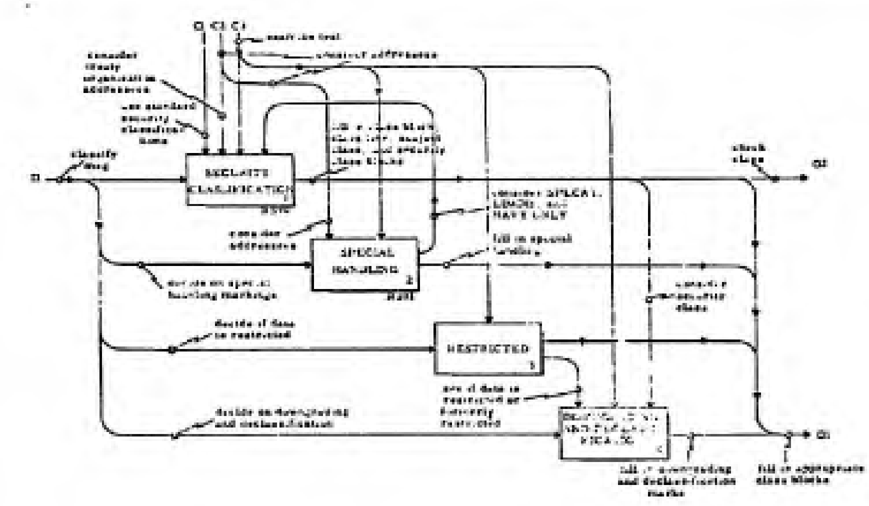


NODE: D2	TITLE: MESSAGE DRAFT	NUMBER: RS86 D2

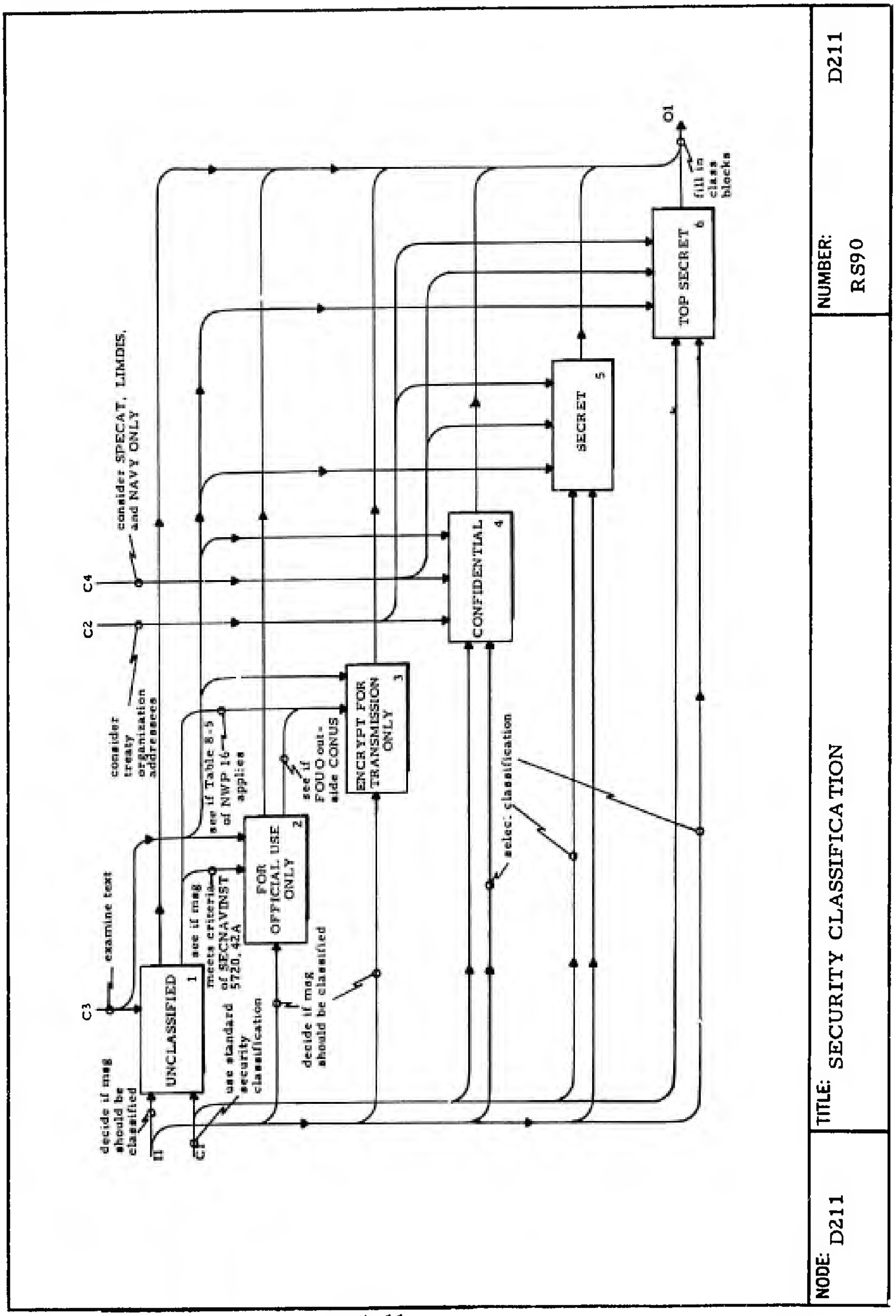

NODE
D2
TITLE
MESSAGE DRAFT
NUMBER
R586
D2

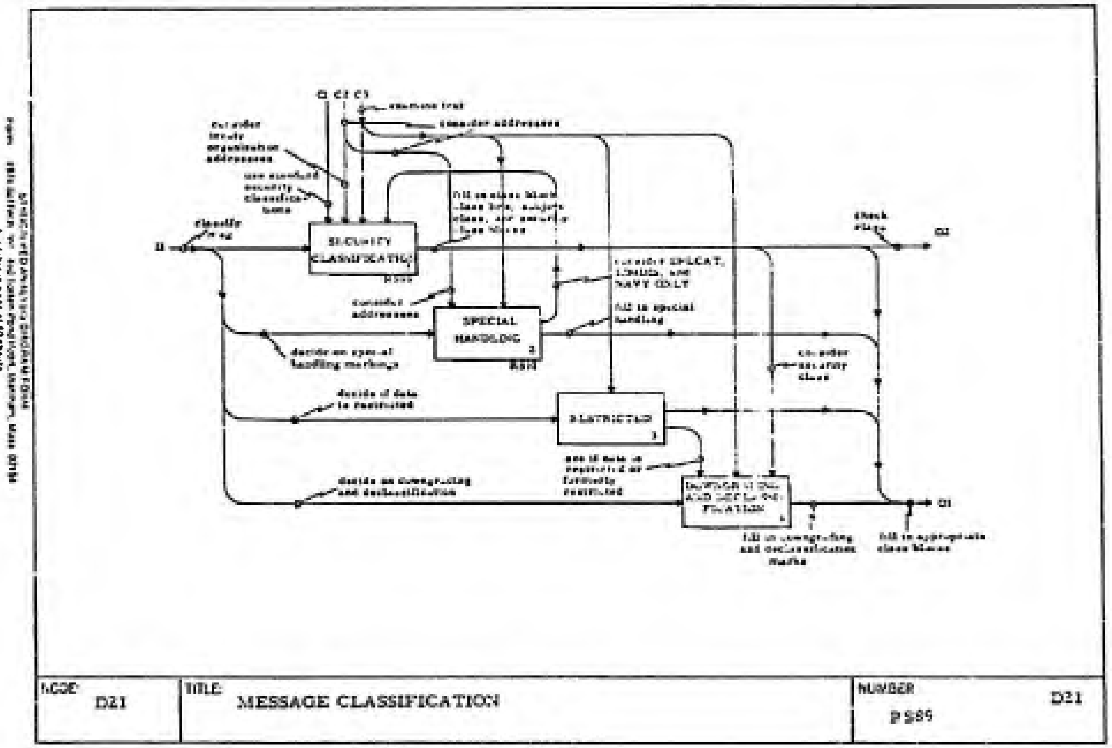


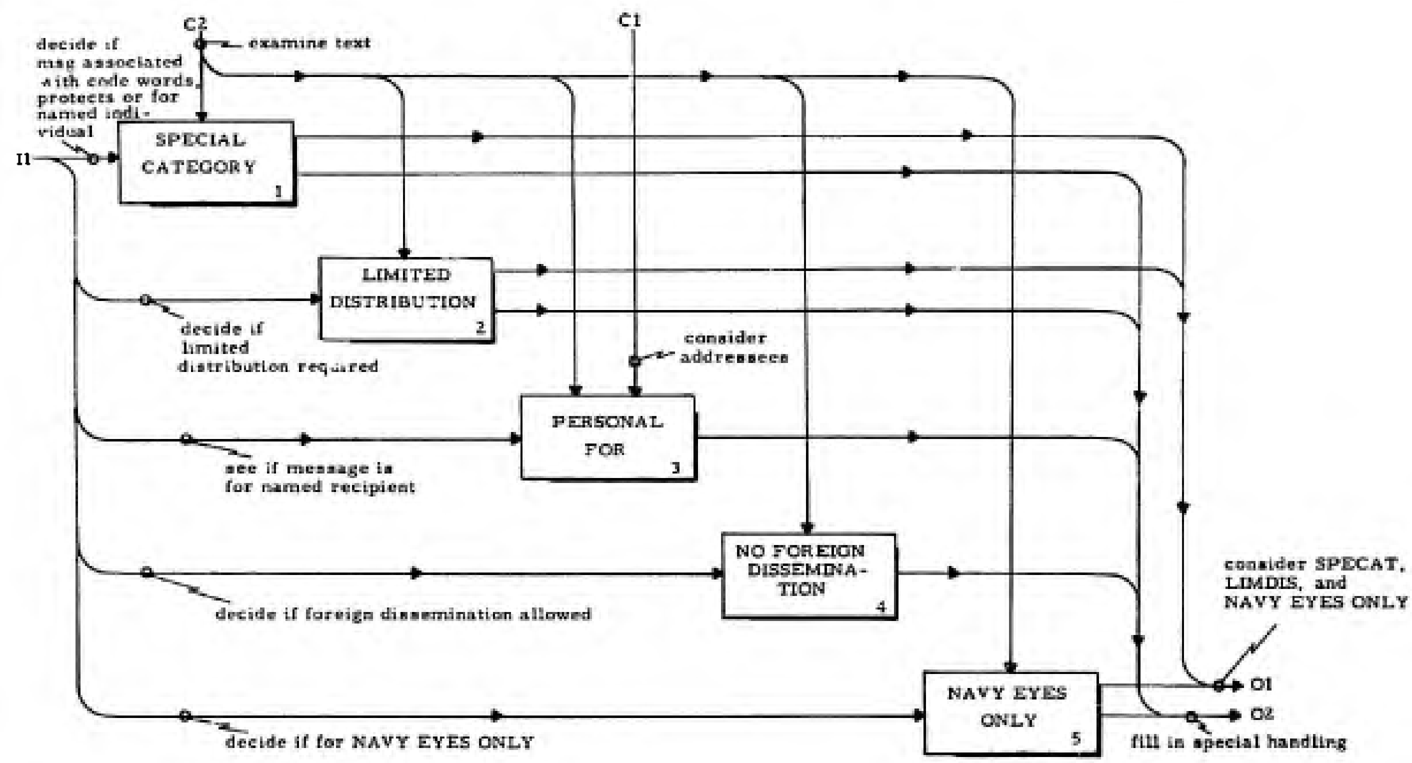
NODE: D21	TITLE: MESSAGE CLASSIFICATION	NUMBER: RS89	D21



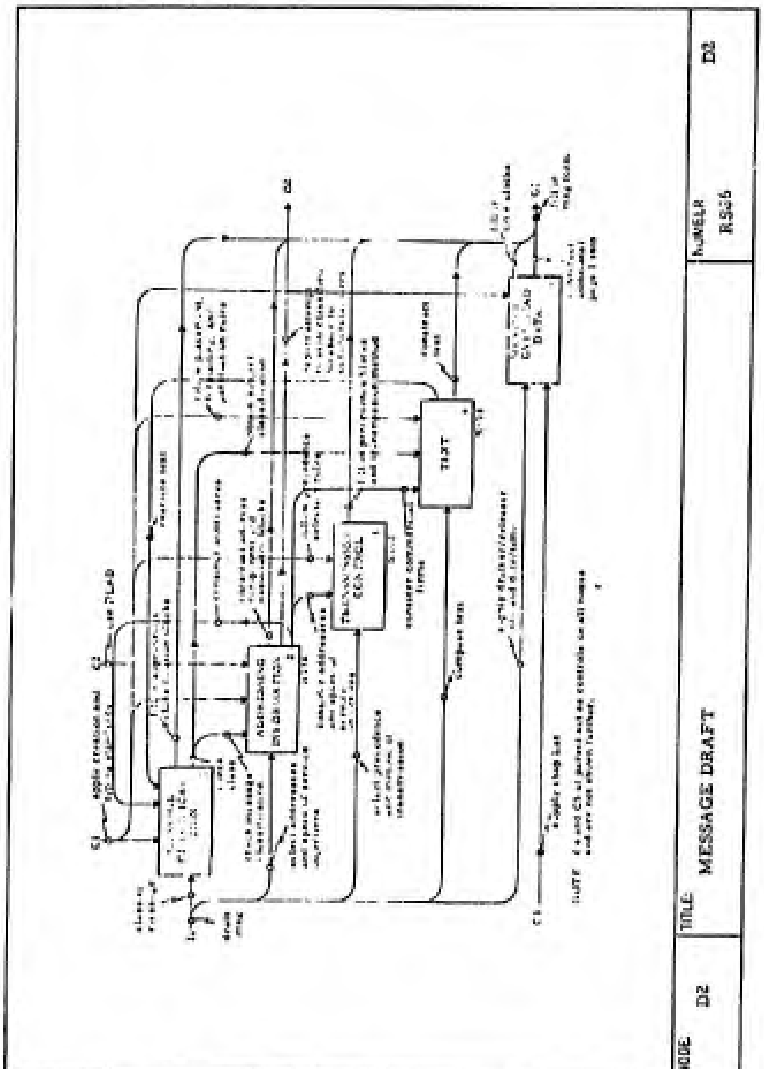
NODE: D21	TITLE: MESSAGE CLASSIFICATION	NUMBER: RS/9 D21

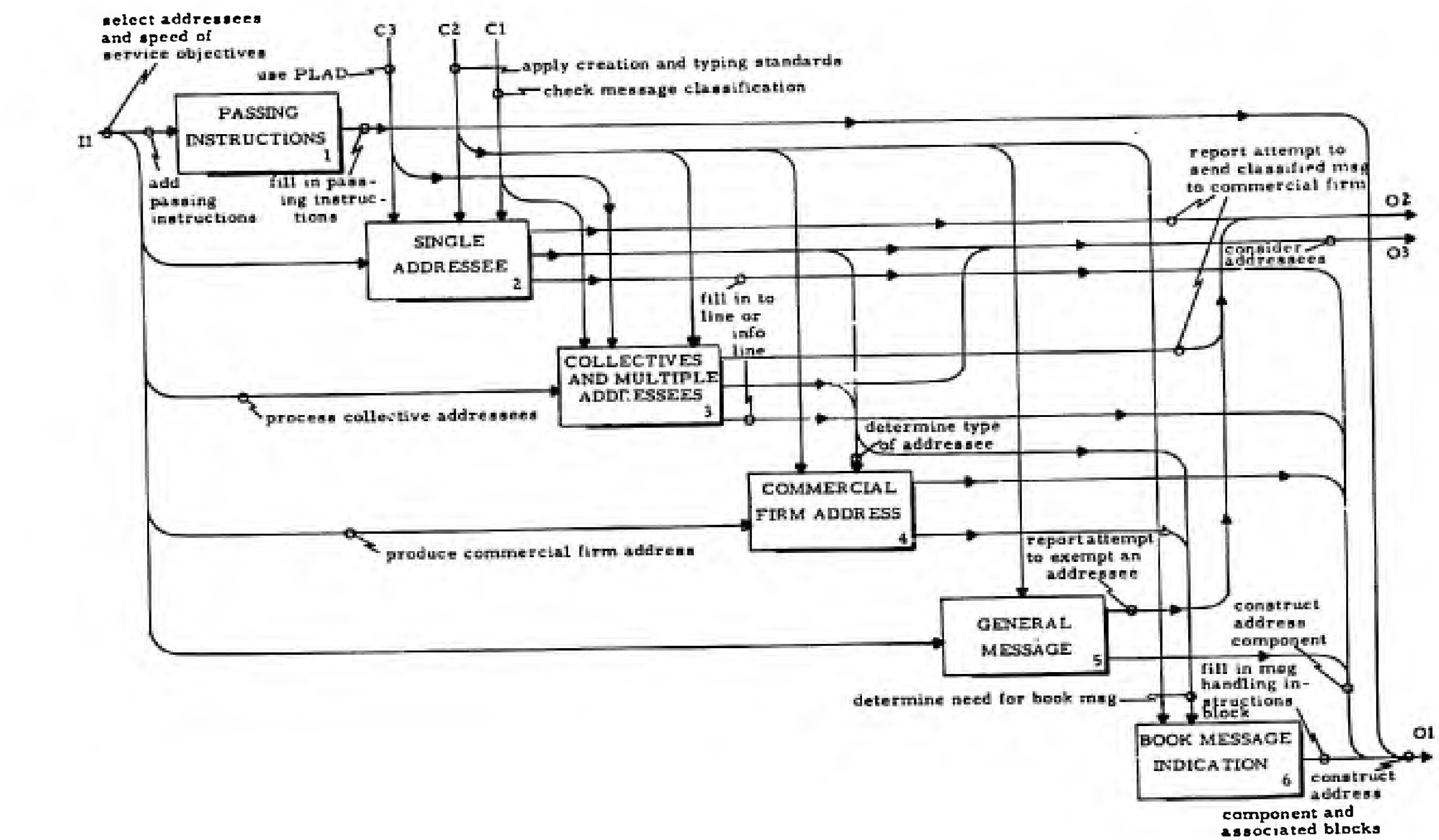

UNCLASSIFIED 1
FOR OFFICIAL USE ONLY 2
ENCRYPT FOR TRANSMISSION ONLY 3
CONFIDENTIAL 4
SECRET 5
TOP SECRET 6
I1
C1
C2
C3
C4
O1
decide if msg should be classified
use standard security classification
examine text
see if msg meets criteria of SECNAVINST 5720.42A
decide if msg should be classified
see if Table 8-5 of NWP 16 applies
see if FOUO outside CONUS
consider treaty organization addressees
consider SPECAT, LIMDIS, and NAVY ONLY
select classification
fill in class blocks
NODE: D211
TITLE: SECURITY CLASSIFICATION
NUMBER: RS90
D211


C1 C2 C3
I1
O1
O2
SECURITY CLASSIFICATION
1
SPECIAL HANDLING
2
RESTRICTED
3
DOWNGRADING AND DECLASSIFICATION
4
consider addresses
decide on special handling markings
decide if data is restricted
decide on downgrading and declassification
fill in special handling
consider security class
fill in appropriate class blocks
fill in downgrading and declassification marks
check stamp
classify msg
consider LIMDIS, and NAVY ONLY
consider security classification guide
PAGE
D21
TITLE
MESSAGE CLASSIFICATION
NUMBER
PS89
D21

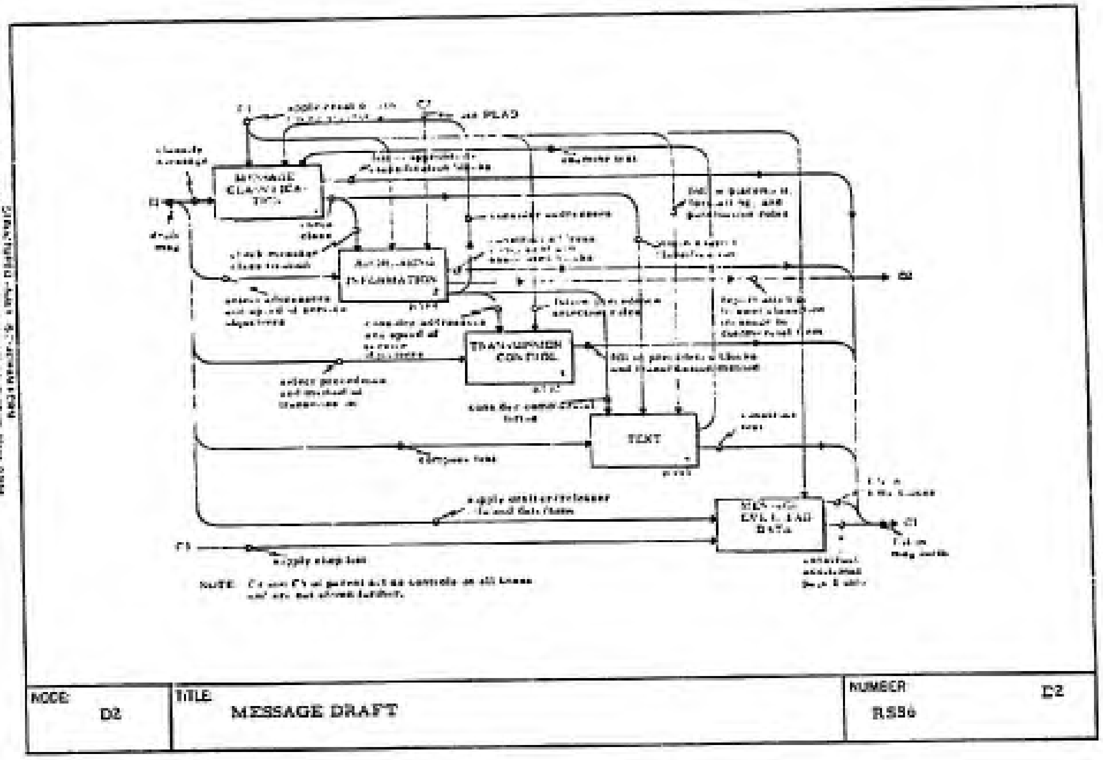


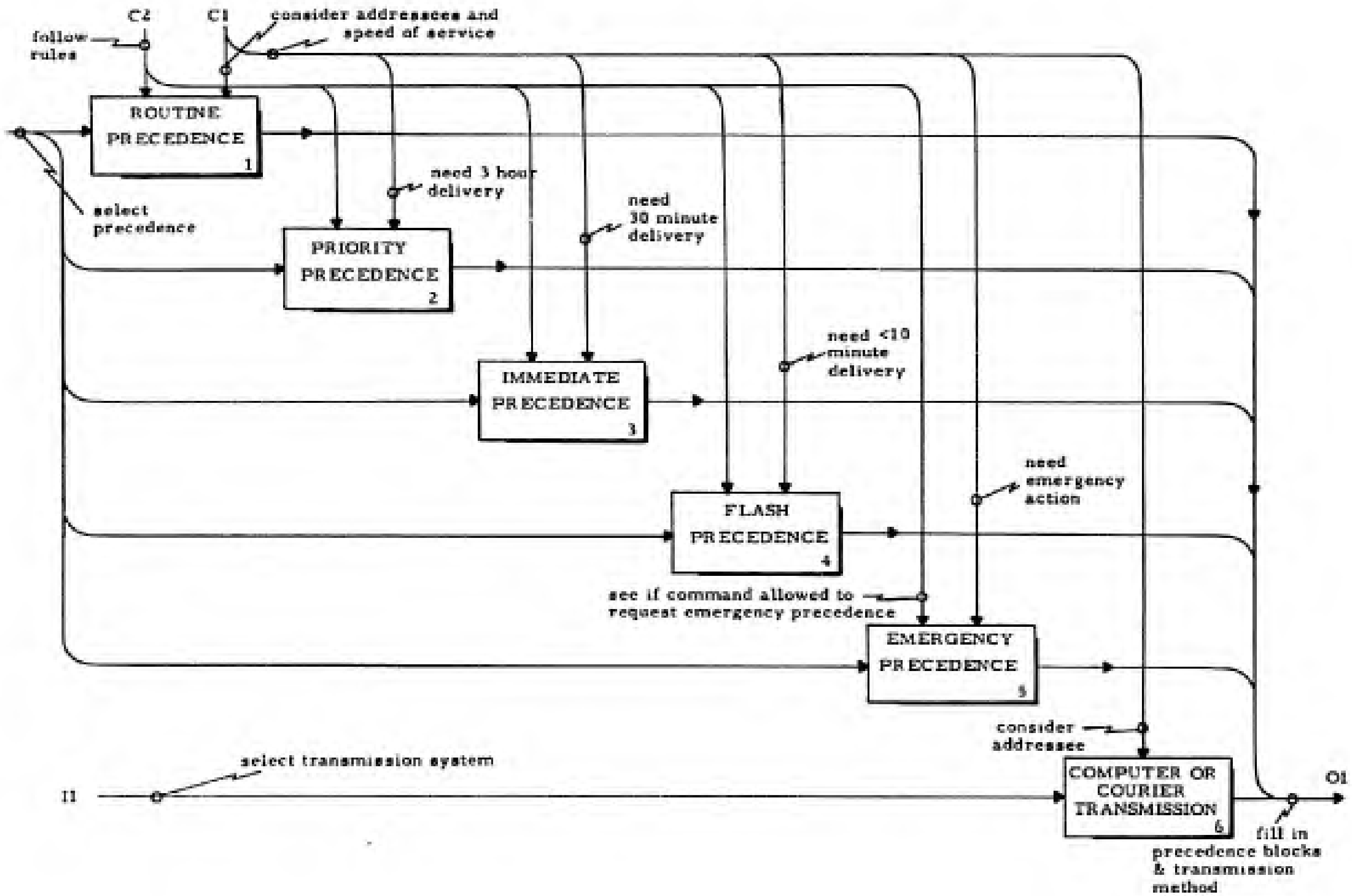
NODE: D212	TITLE: SPECIAL HANDLING	NUMBER: RS91 D212


NODE:
D2
TITLE:
MESSAGE DRAFT
NUMBER
RS35
D2
ADDRESSING INFORMATION
TRANSMISSION CONTROL
TEXT
use 7LAD
compose text
supply chop list

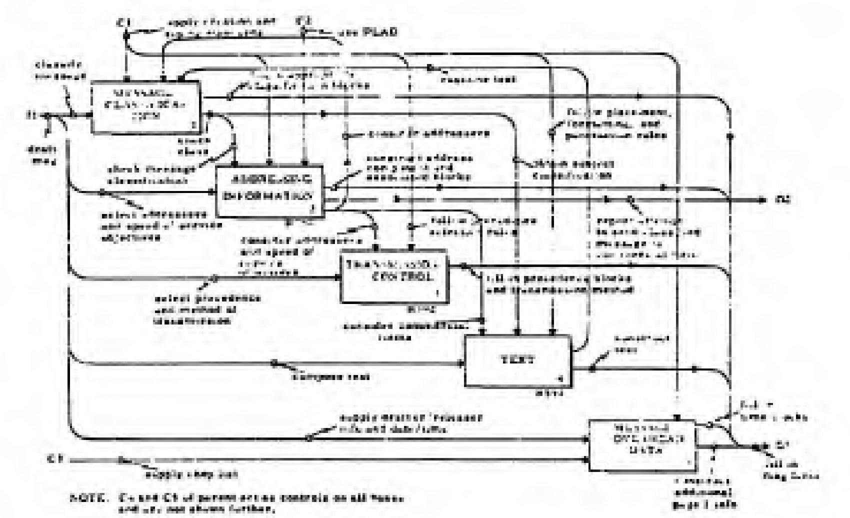


NODE: D22	TITLE: ADDRESSING INFORMATION	NUMBER: RS88 D22

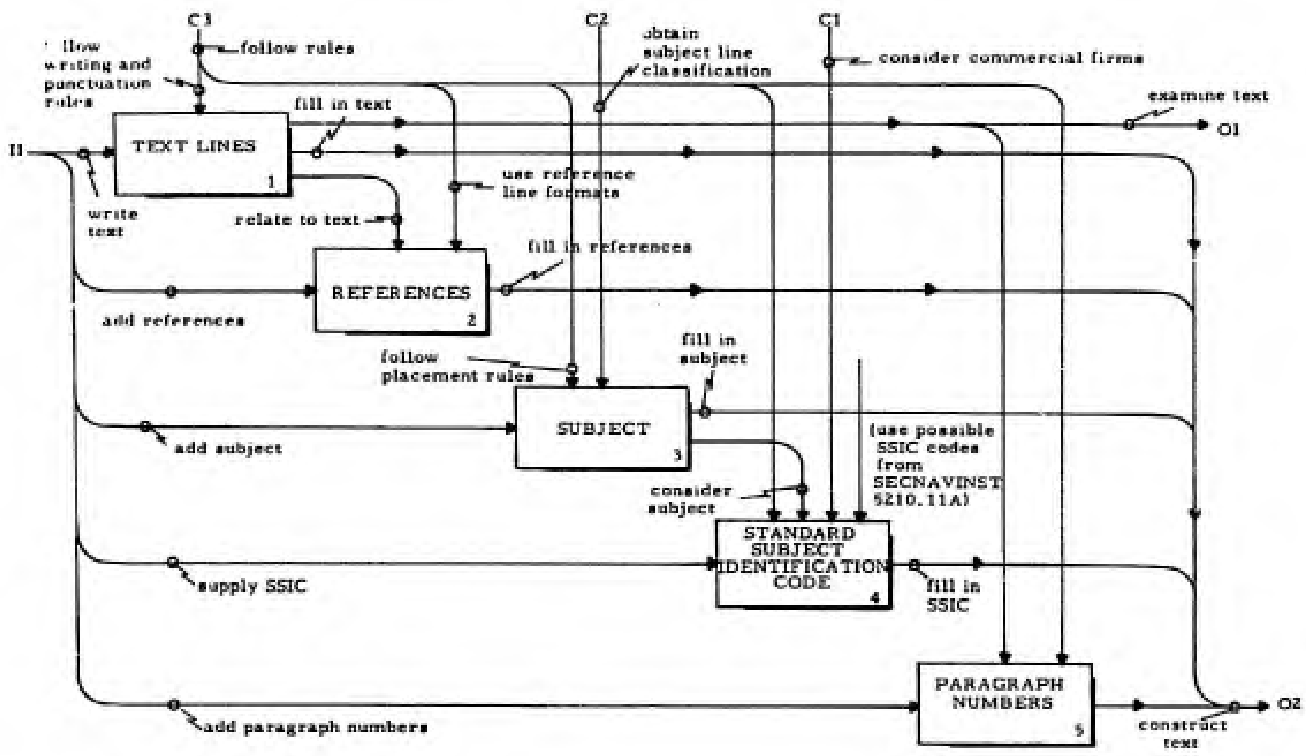

TEXT
NODE:
D2
TITLE:
MESSAGE DRAFT
NUMBER:



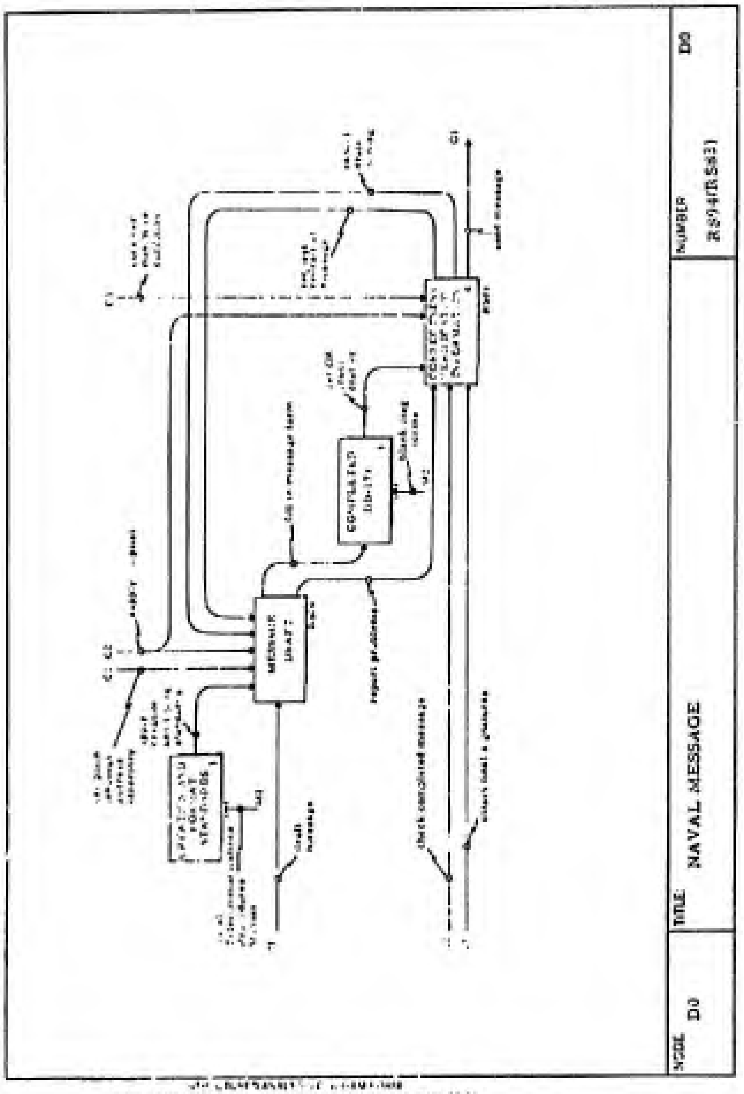
NODE: D23	TITLE: TRANSMISSION CONTROL	NUMBER: RS92 D23

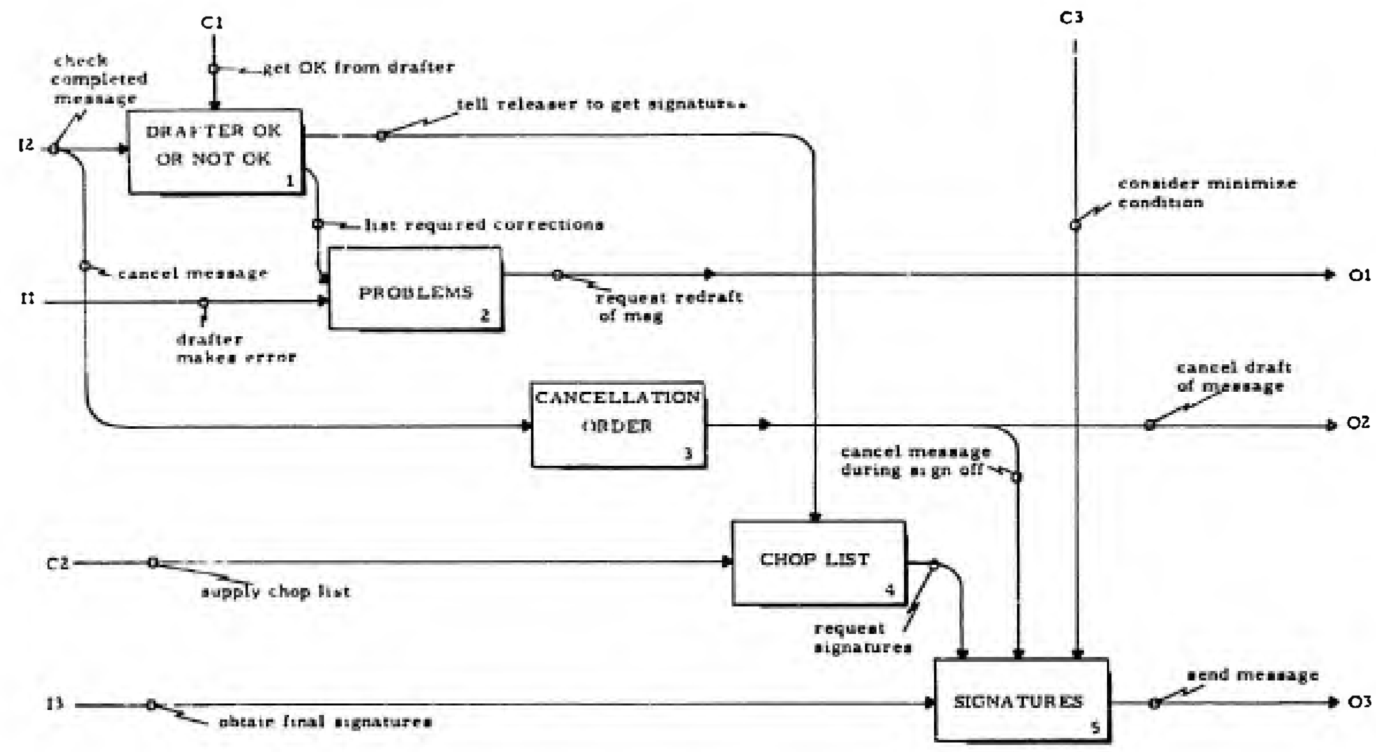


NODE:	TITLE:	NUMBER
D2	MESSAGE DRAFT	RS86 D2

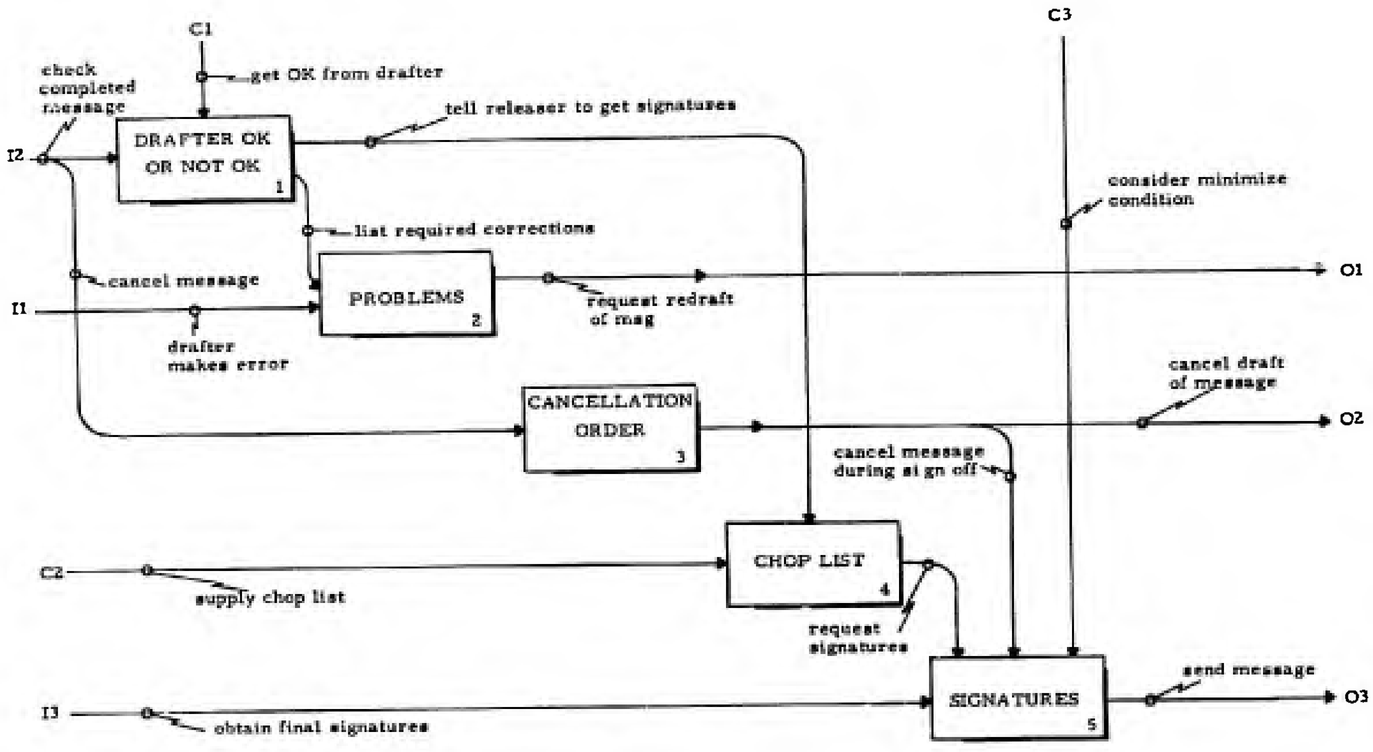


NODE: D24	TITLE: TEXT	NUMBER: RS93 D24


NODE: D0
TITLE: NAVAL MESSAGE
NUMBER:
D0



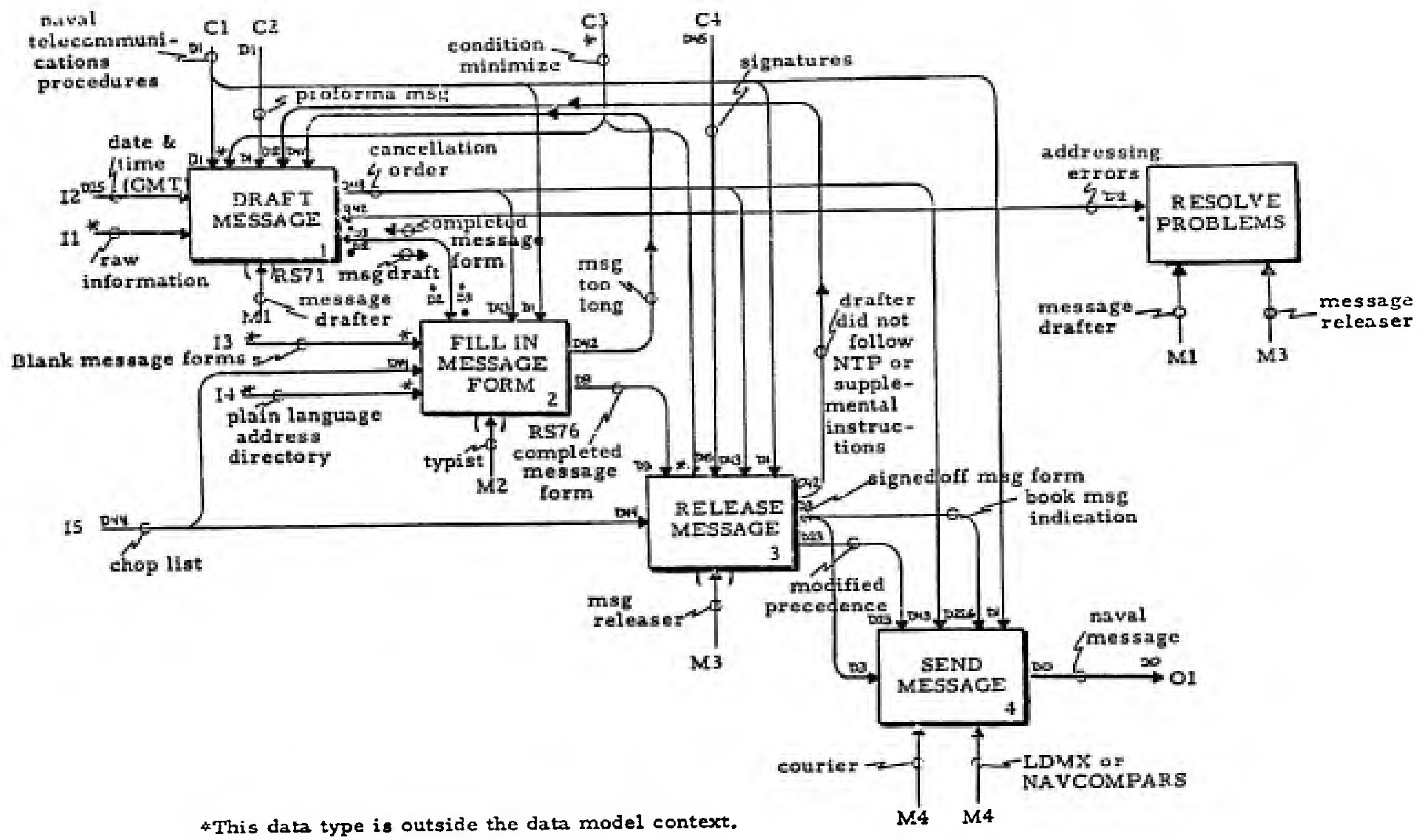
NODE: D4	TITLE: CORRECTNESS VERIFICATION INFORMATION	NUMBER: RS85 (RS84) D4



NODE: D4 | TITLE: CORRECTNESS VERIFICATION INFORMATION | NUMBER: RS85 (RS84) D4

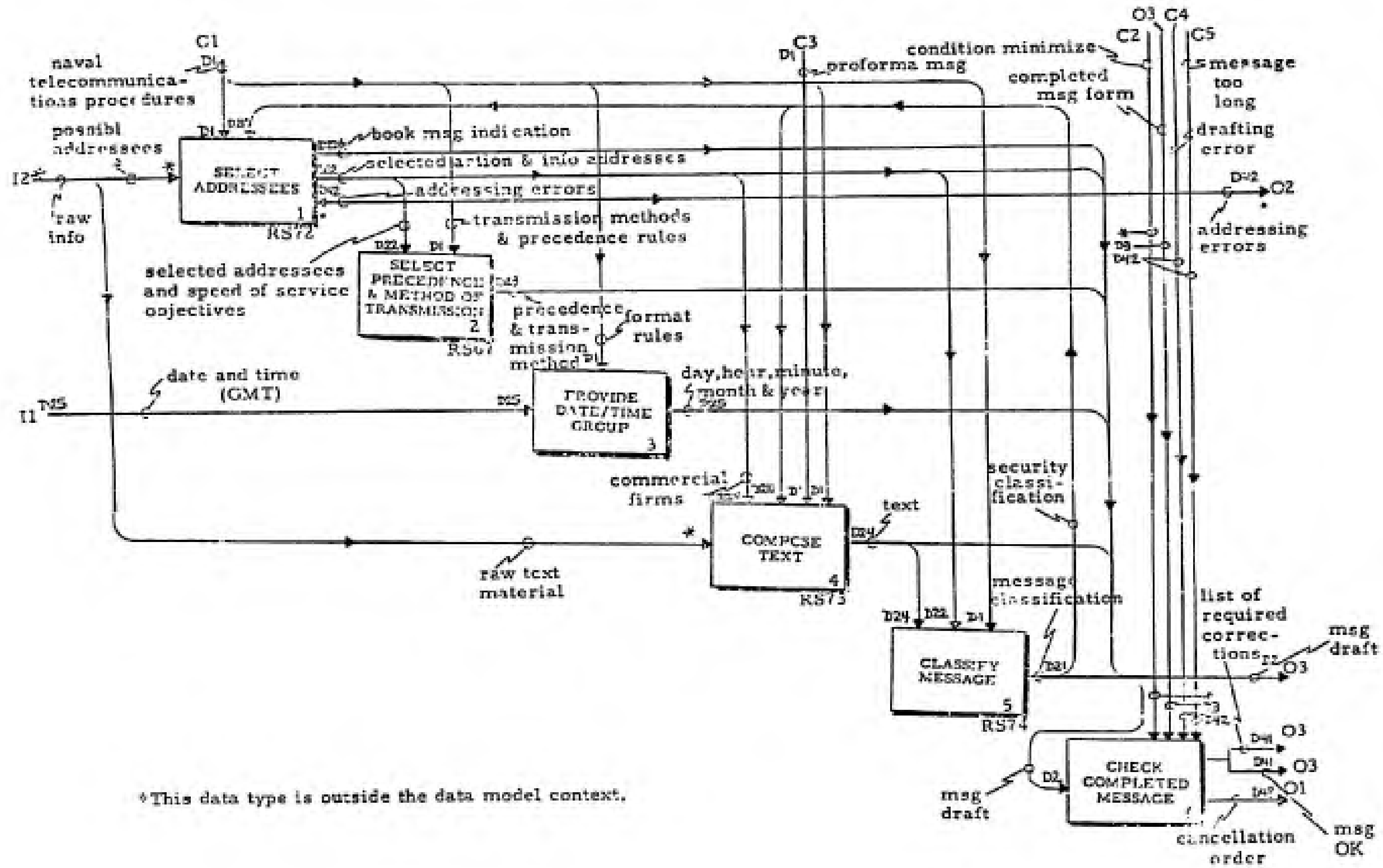
Section 5

ACTIVITY AND DATA MODEL TIE



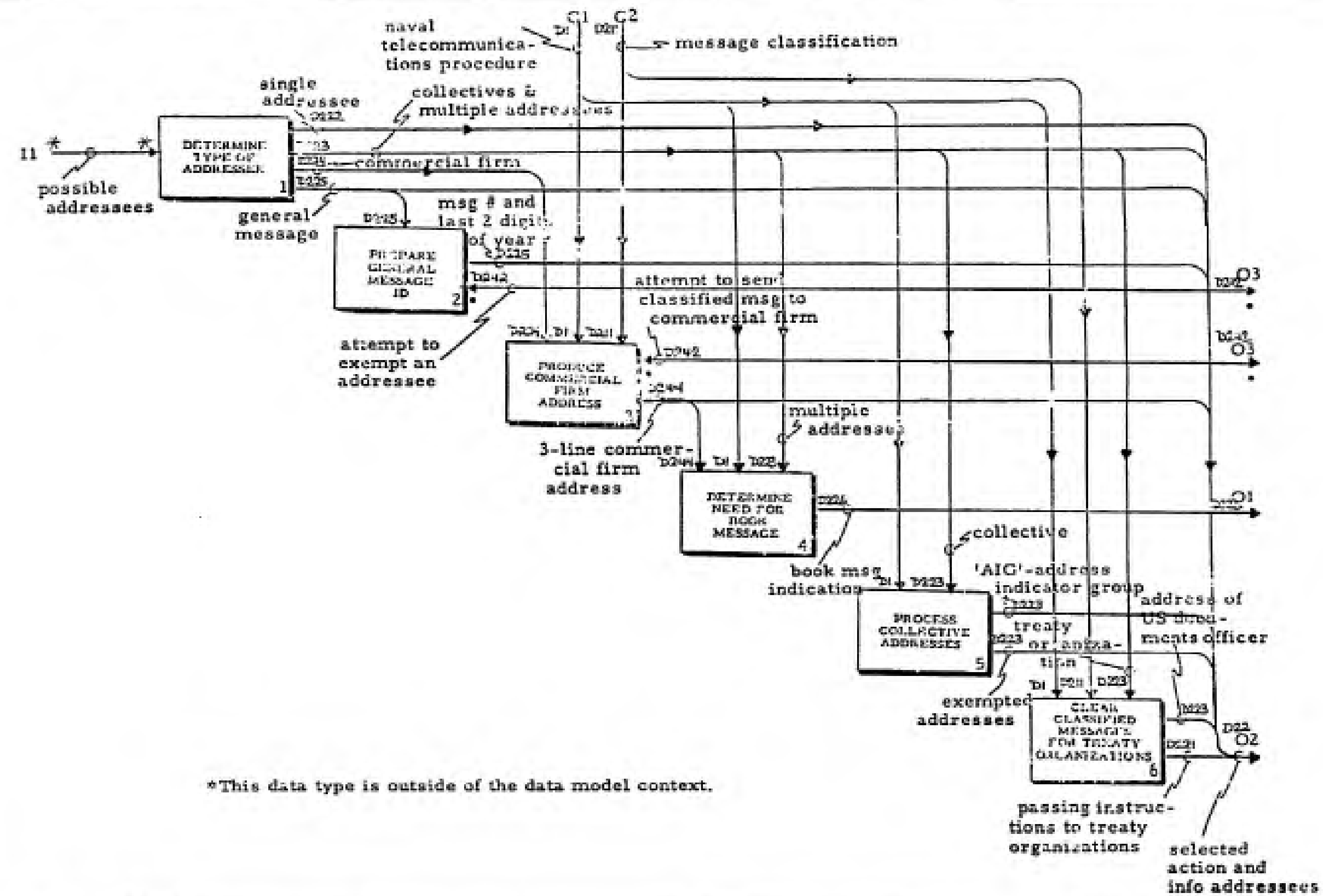
*This data type is outside the data model context.

NODE:	TITLE:	NUMBER: A0
A0	COMPOSE NAVAL MESSAGES	RS59(RS58)



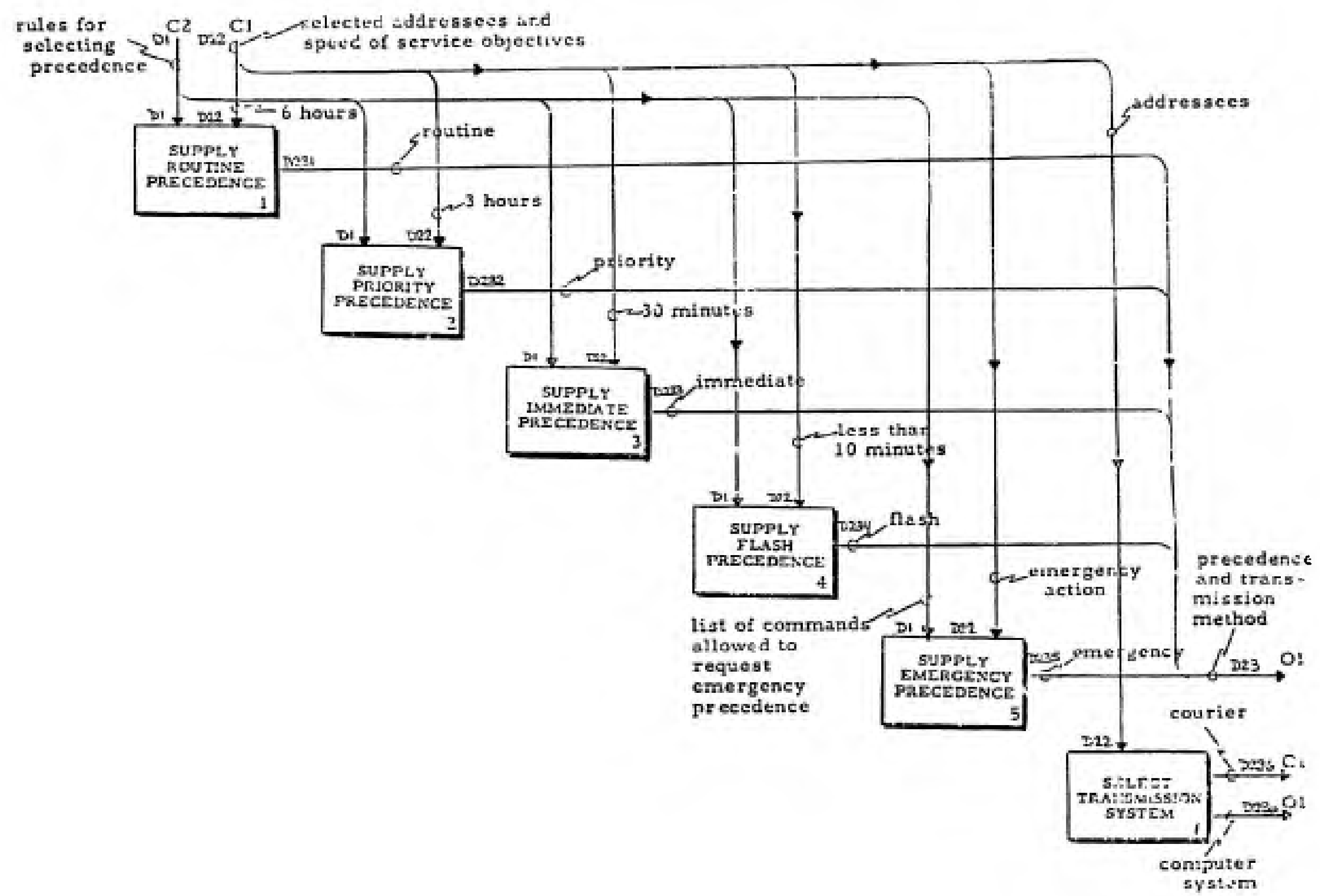
*This data type is outside the data model context.

NODE:	TITLE:	NUMBER:
A1	DRAFT MESSAGE	RS71 (RS61) A1

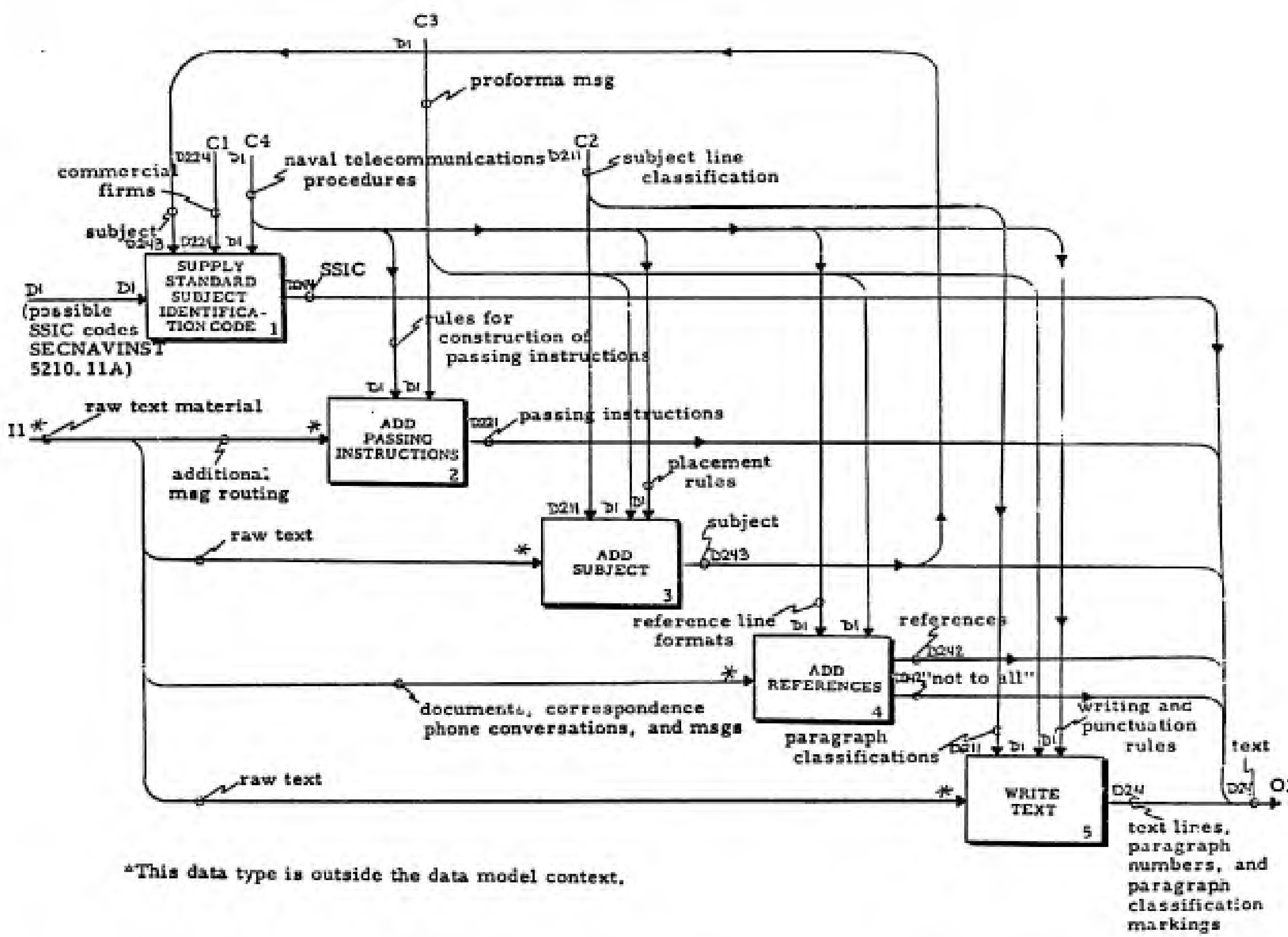


*This data type is outside of the data model context.

NODE: A11	TITLE: SELECT ADDRESSEES	NUMBER: RS72(RS64) A11

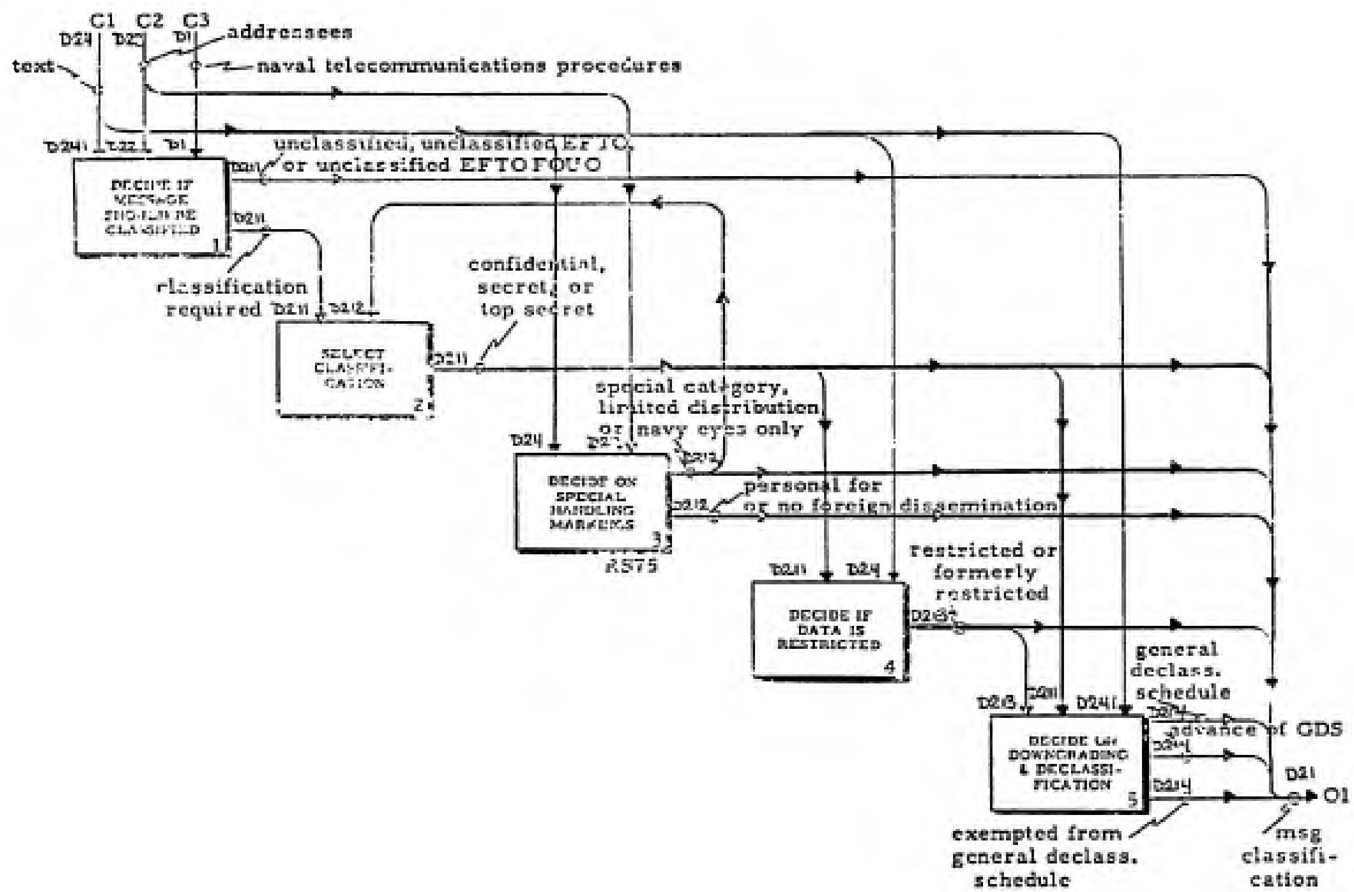


NODE:	TITLE:	NUMBER: A12
A12	SELECT PRECEDENCE AND METHOD OF TRANSMISSION	RS67(RS66)

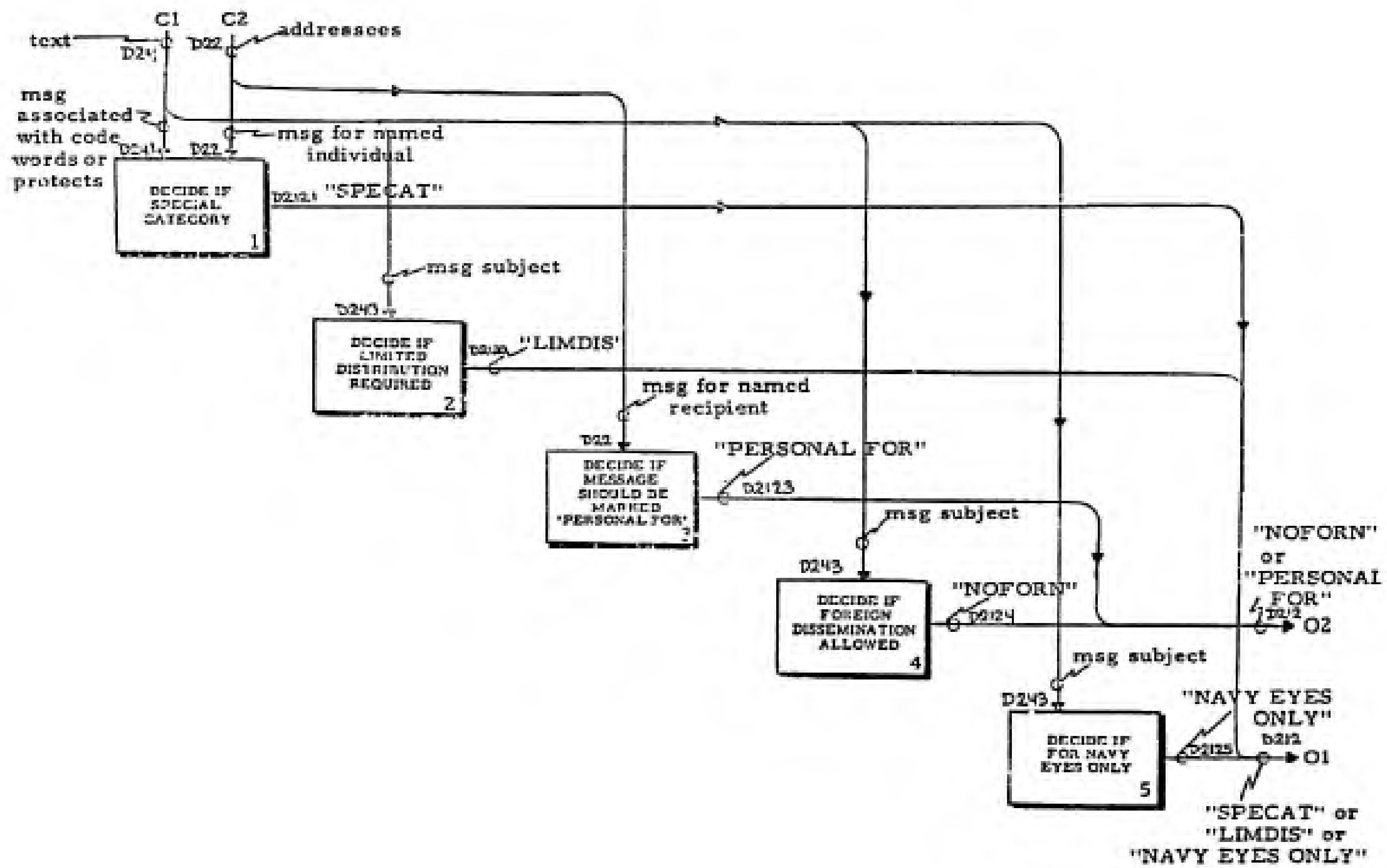


*This data type is outside the data model context.

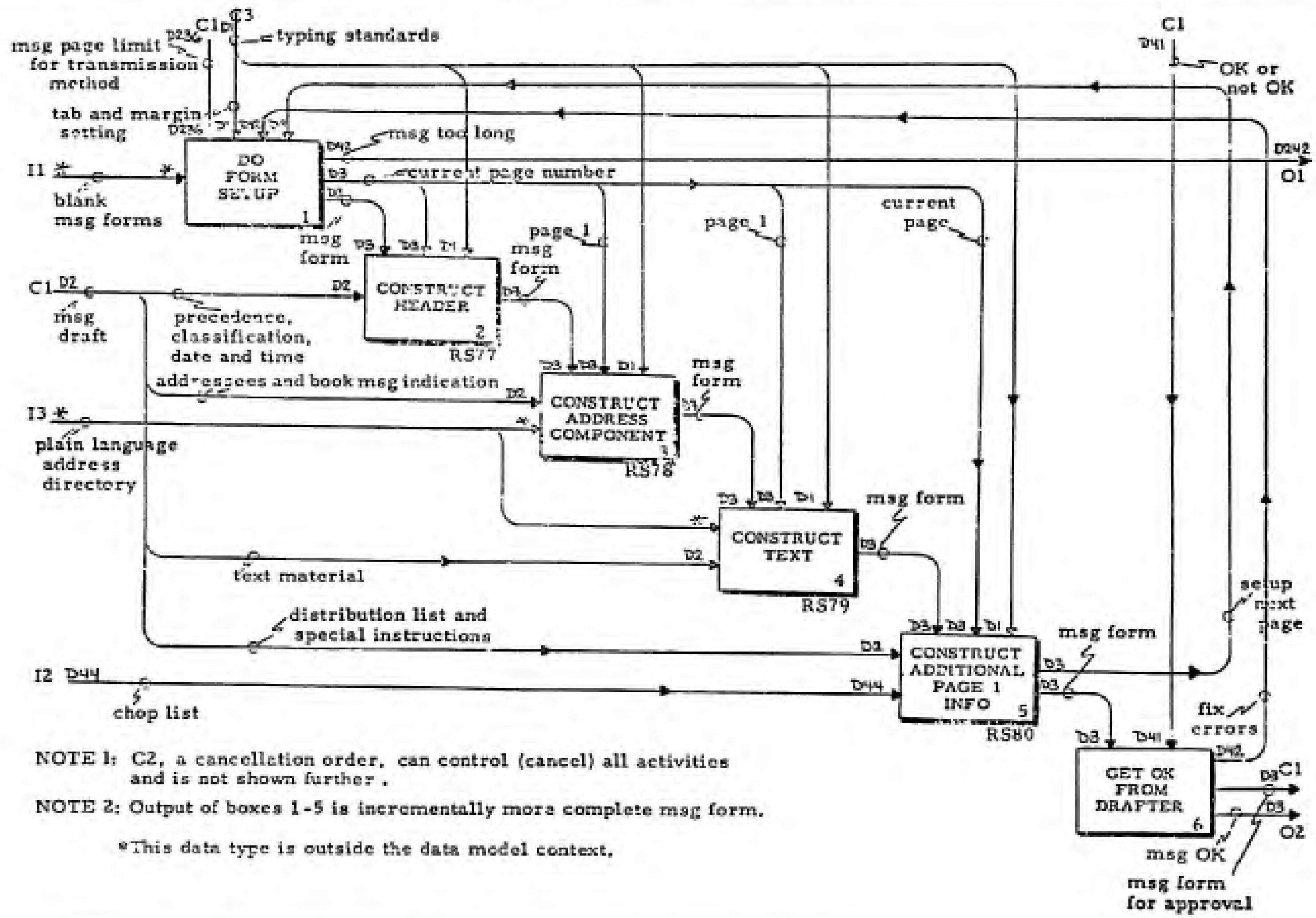
NODE:	TITLE:	NUMBER:
A14	COMPOSE TEXT	A14 RS73(RS68)



NODE: A15	TITLE: CLASSIFY MESSAGE	NUMBER: RS74(RS69) A15



NODE: A153	TITLE: DECIDE ON SPECIAL HANDLING MARKINGS	NUMBER: A153 RS75(RS70)

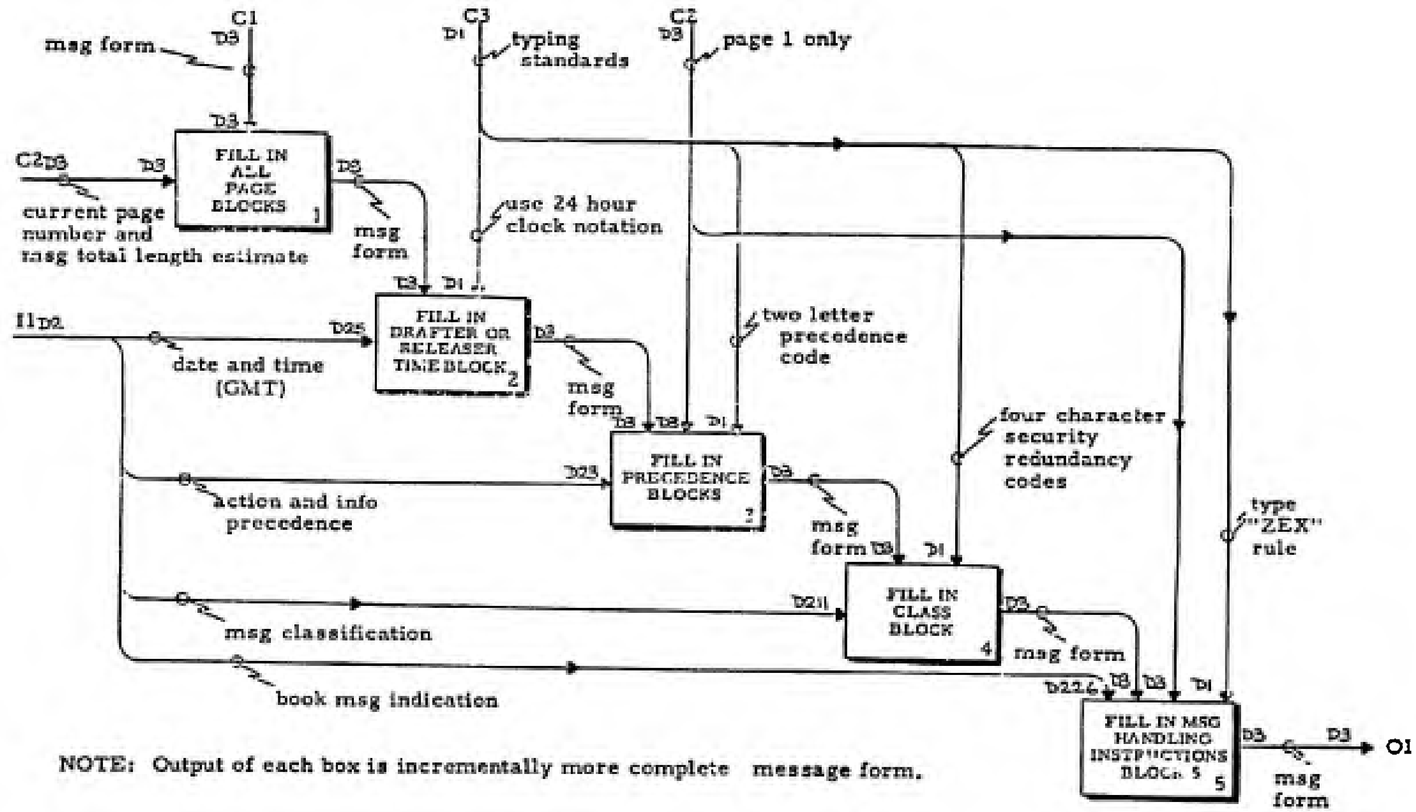


NOTE 1: C2, a cancellation order, can control (cancel) all activities and is not shown further.

NOTE 2: Output of boxes 1-5 is incrementally more complete msg form.

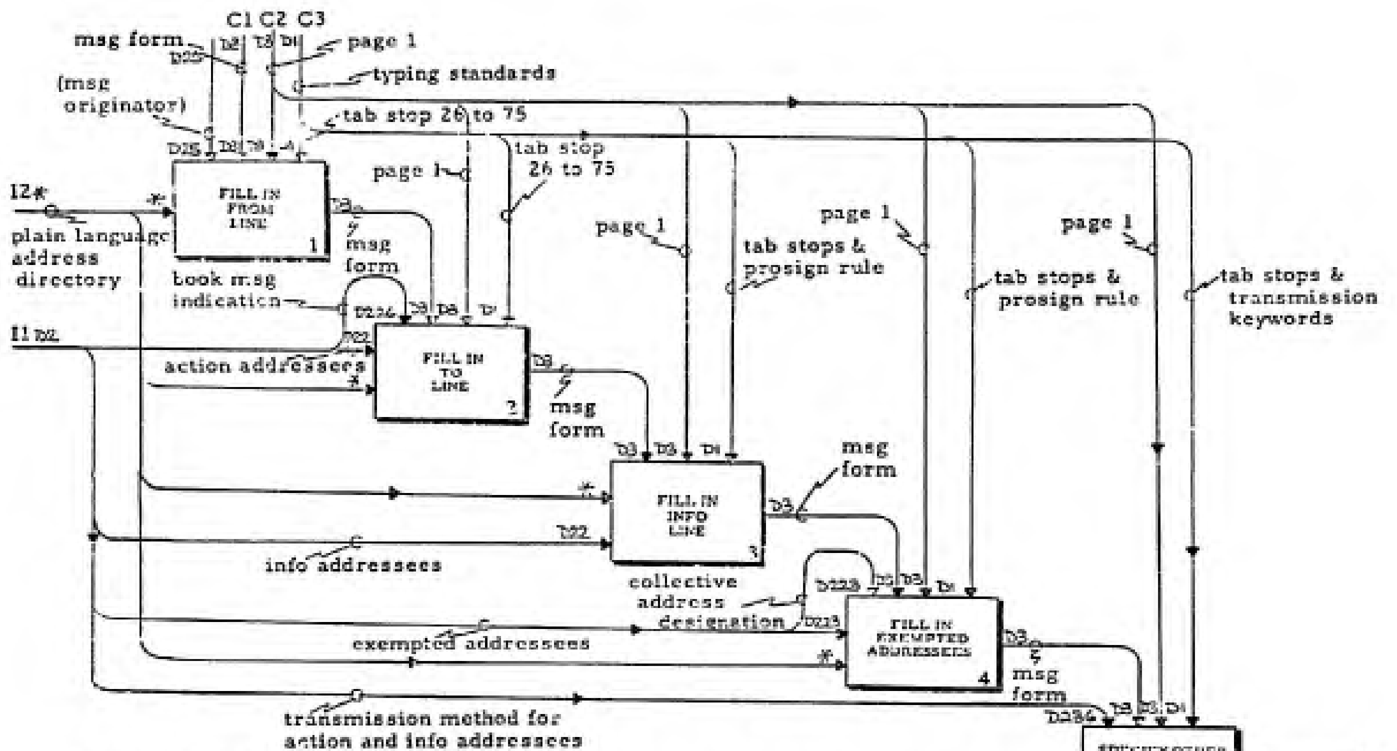
*This data type is outside the data model context.

NODE: A2	TITLE: FILL IN MESSAGE FORM	NUMBER: RS76(RS14) A2



NOTE: Output of each box is incrementally more complete message form.

NODE:	TITLE:	NUMBER:
A22	CONSTRUCT HEADER	RS77(RS17) A22

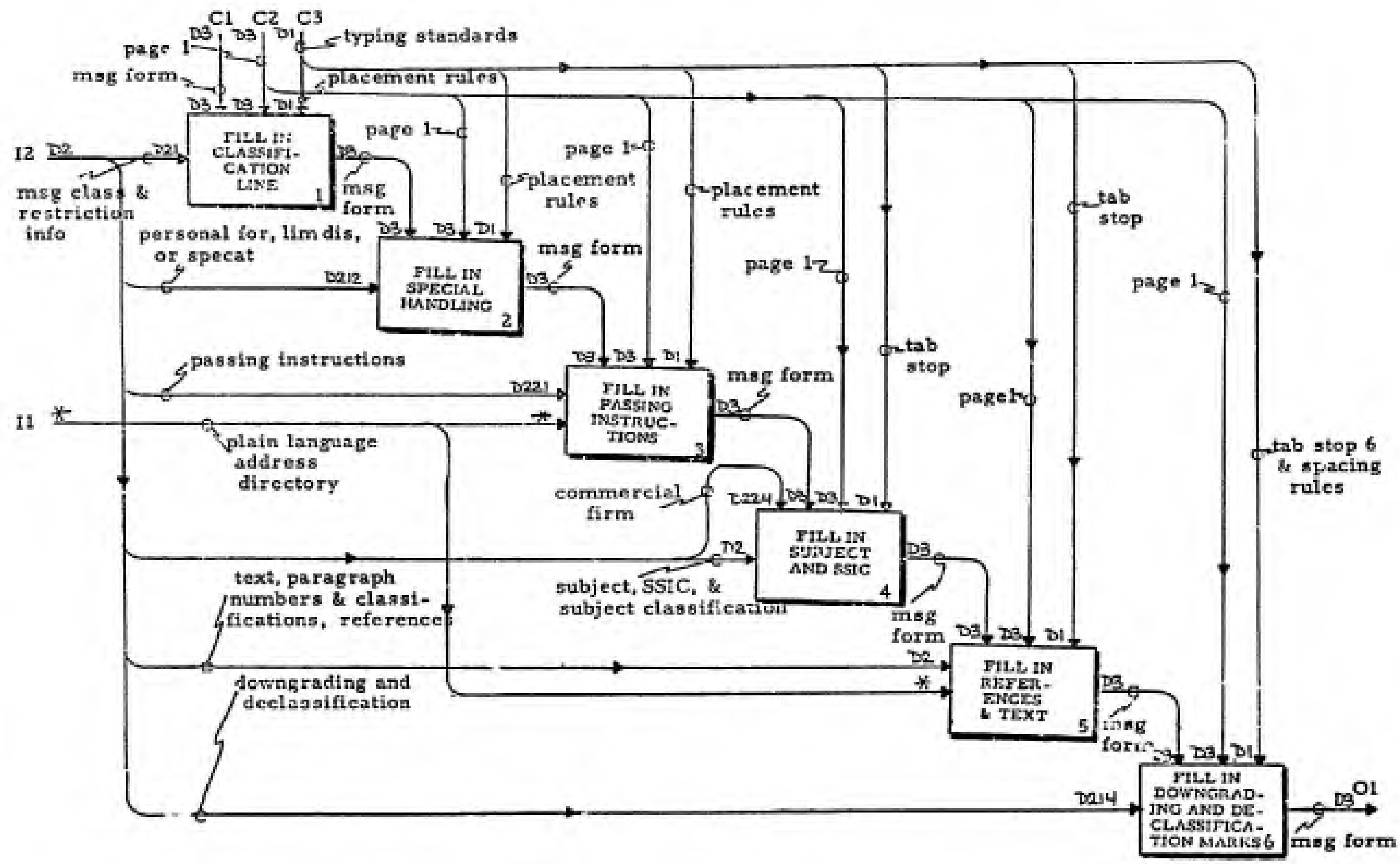


NOTE 1: Output of each box is incrementally more complete message form.

NOTE 2: No address component is constructed on any but the first page (except when additional pages are needed to continue the address component - not shown here).

*This data type is outside the data model context.

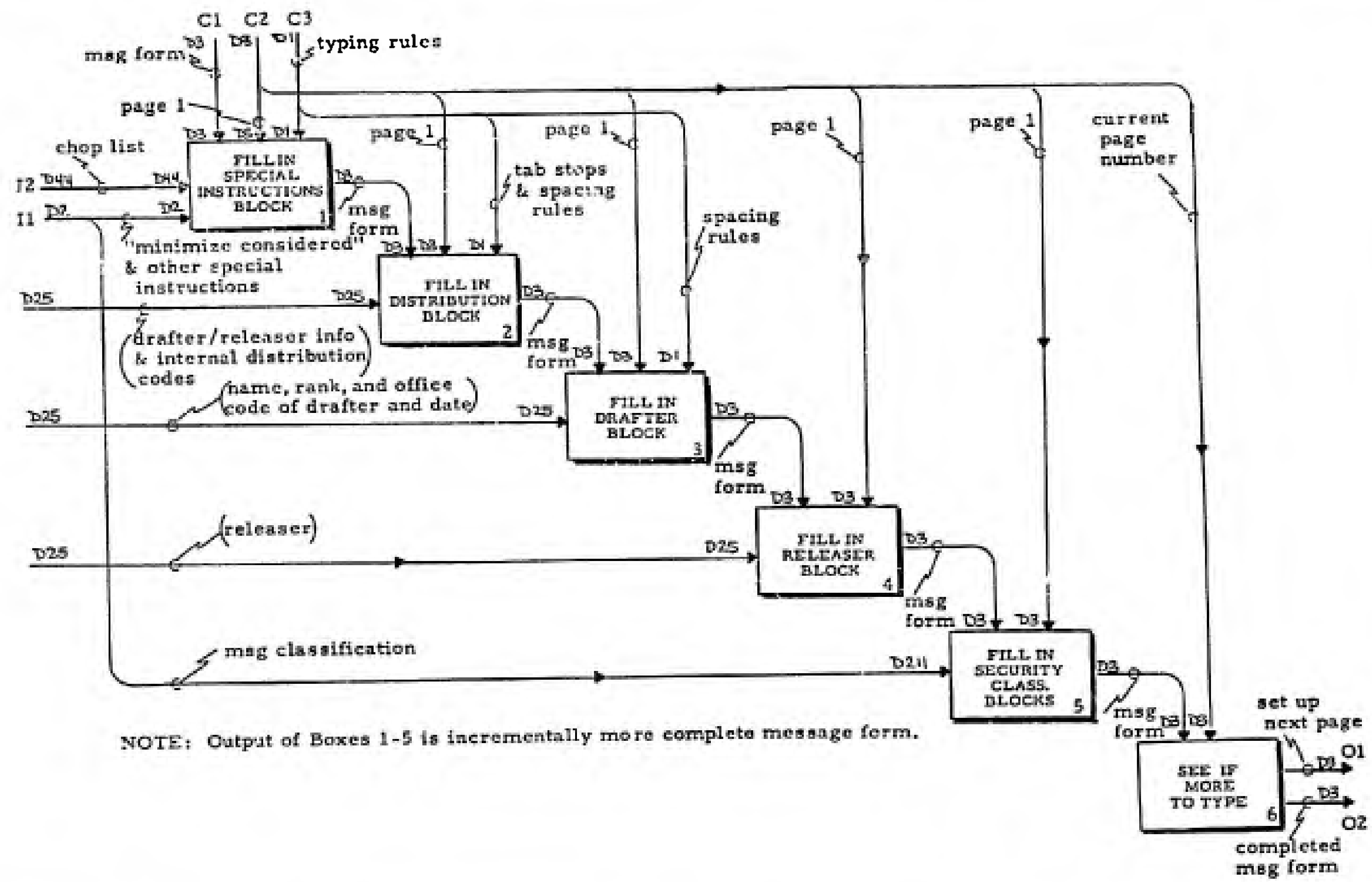
NODE:	TITLE:	NUMBER:
A23	CONSTRUCT ADDRESS COMPONENT	RS78(RS16) A23



NOTE: Output of each box is incrementally more complete message form.

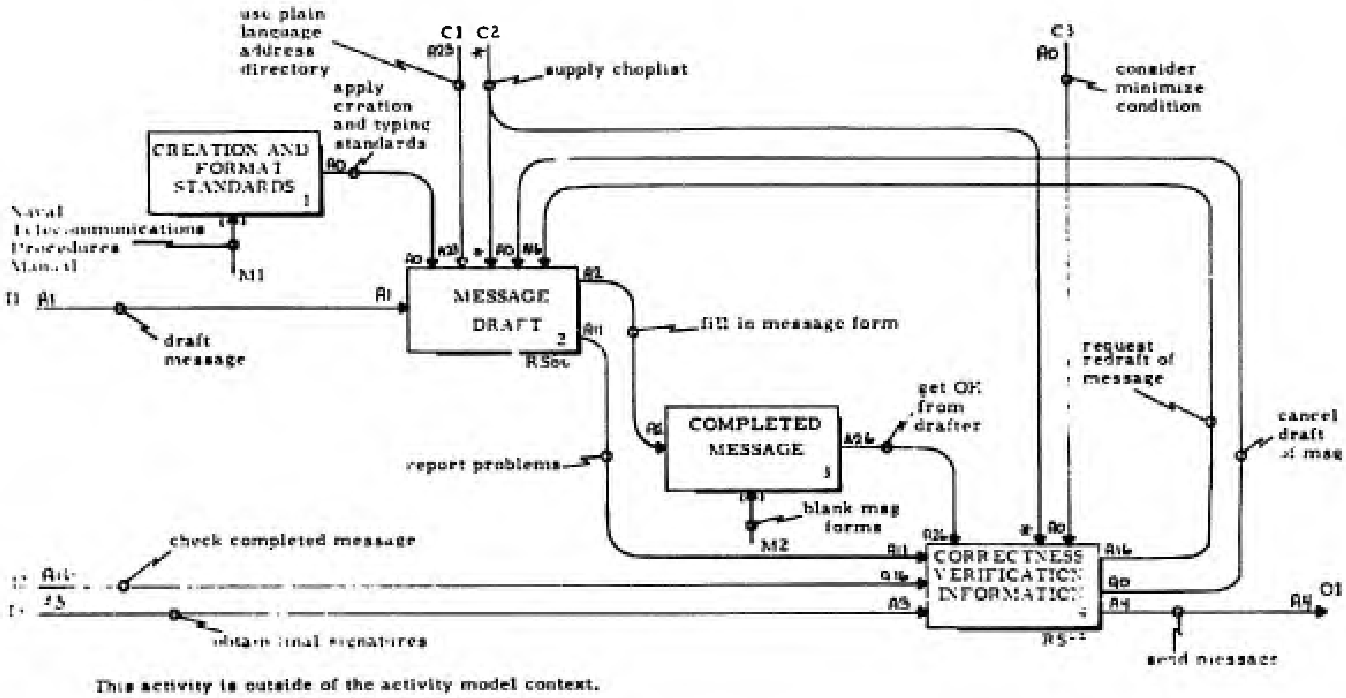
*This data type is outside the data model context.

NODE:	TITLE:	NUMBER:
A24	CONSTRUCT TEXT	RS79(RS18) A24

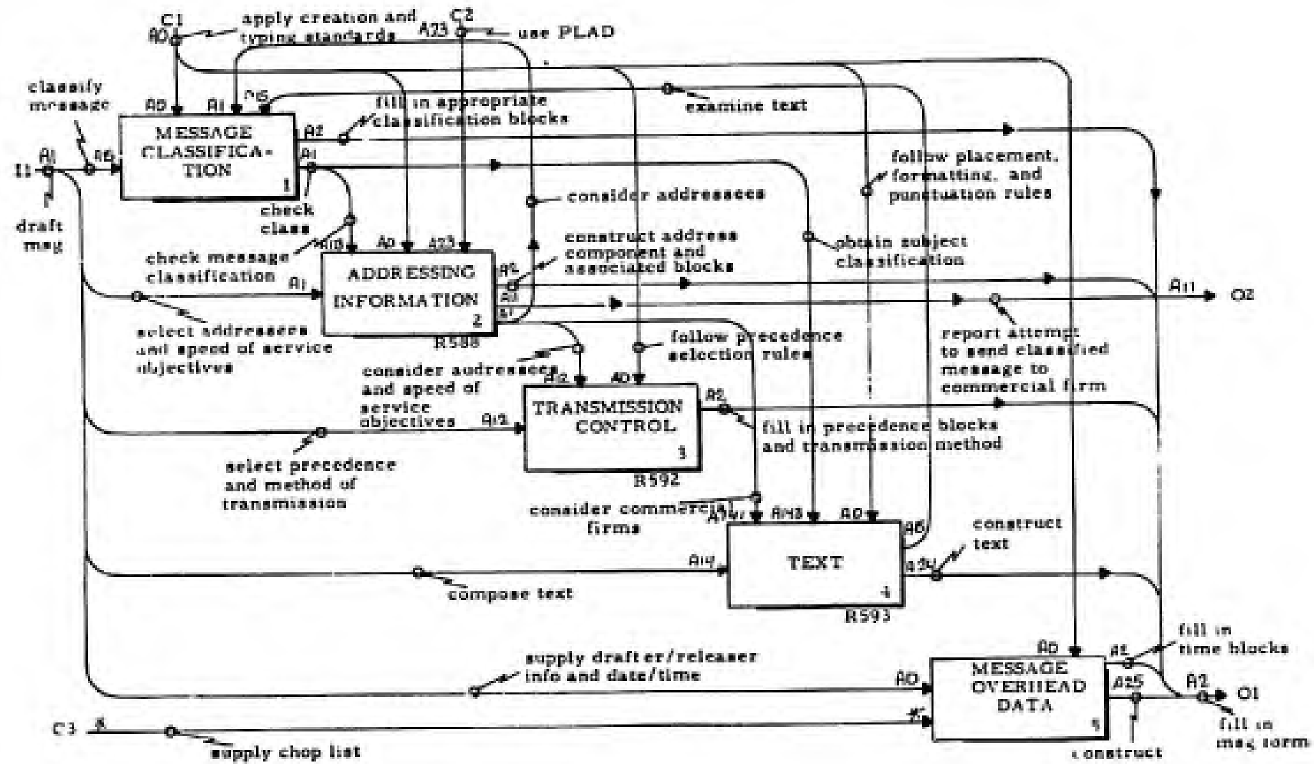


NOTE: Output of Boxes 1-5 is incrementally more complete message form.

NODE: A25	TITLE: CONSTRUCT ADDITIONAL PAGE 1 INFO	NUMBER: A25 RS80(RS19)



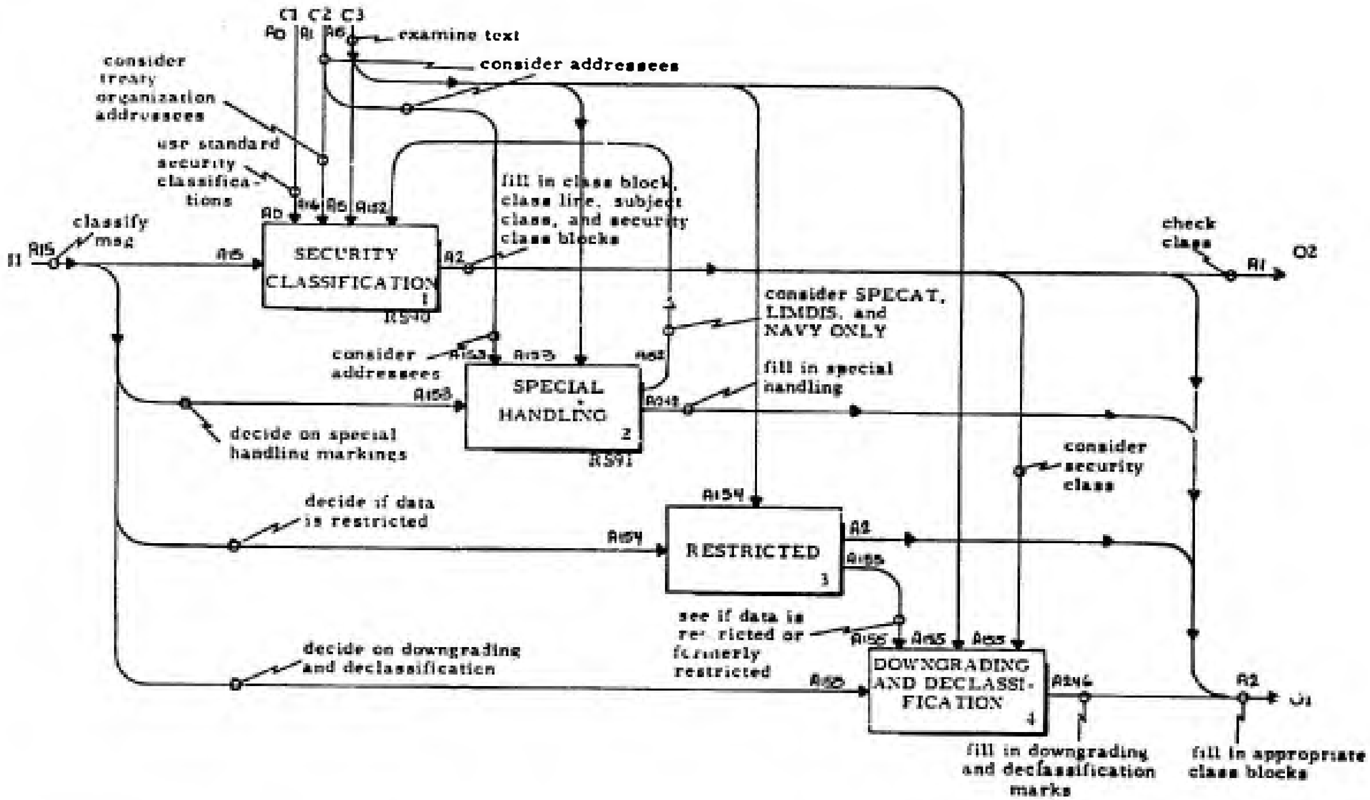
NODE: D0	TITLE: NAVAL MESSAGE	NUMBER: RS94(RS83) D0



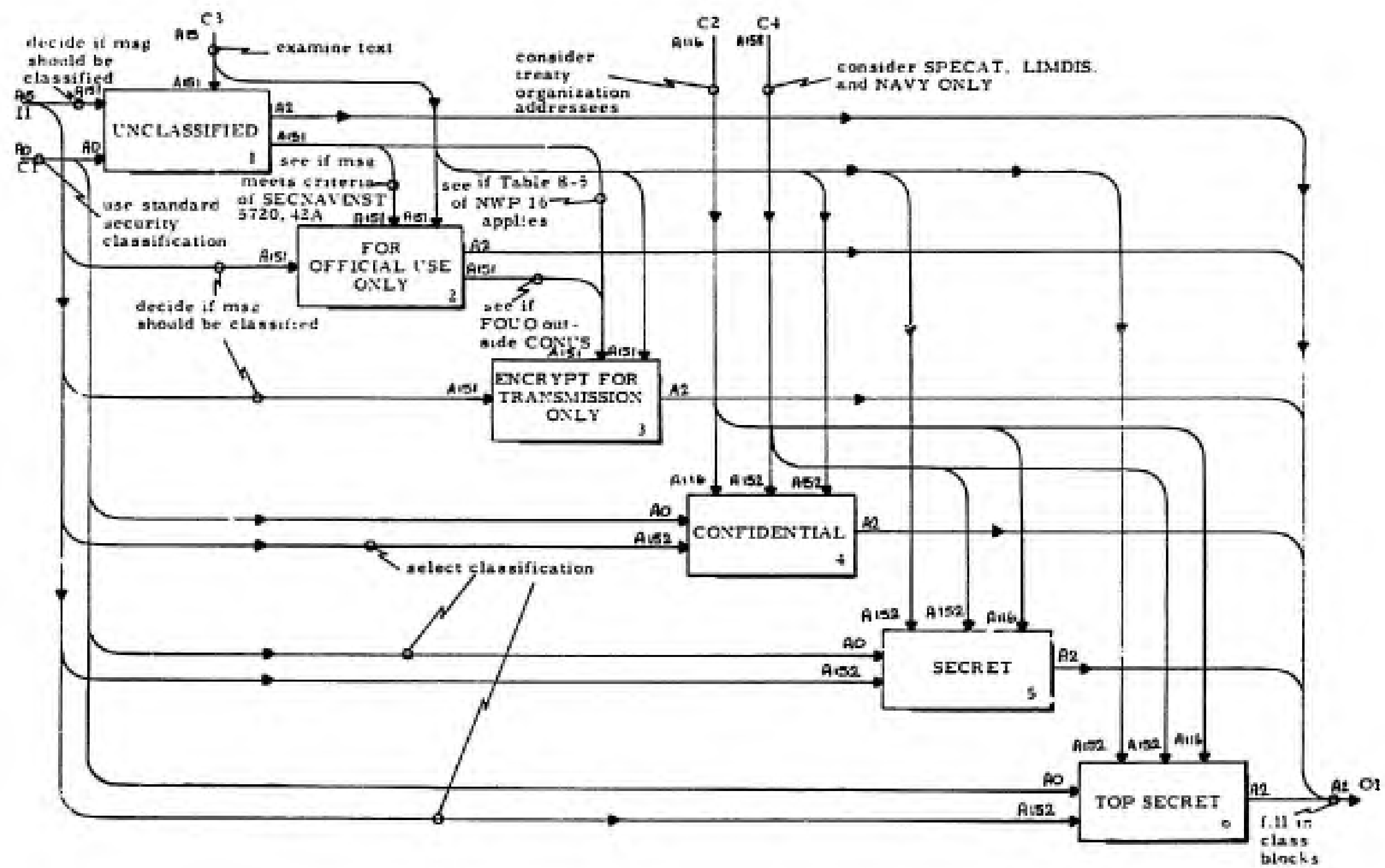
NOTE: C4 and C5 of parent act as controls on all boxes and are not shown further.

*This activity is outside of the activity model context.

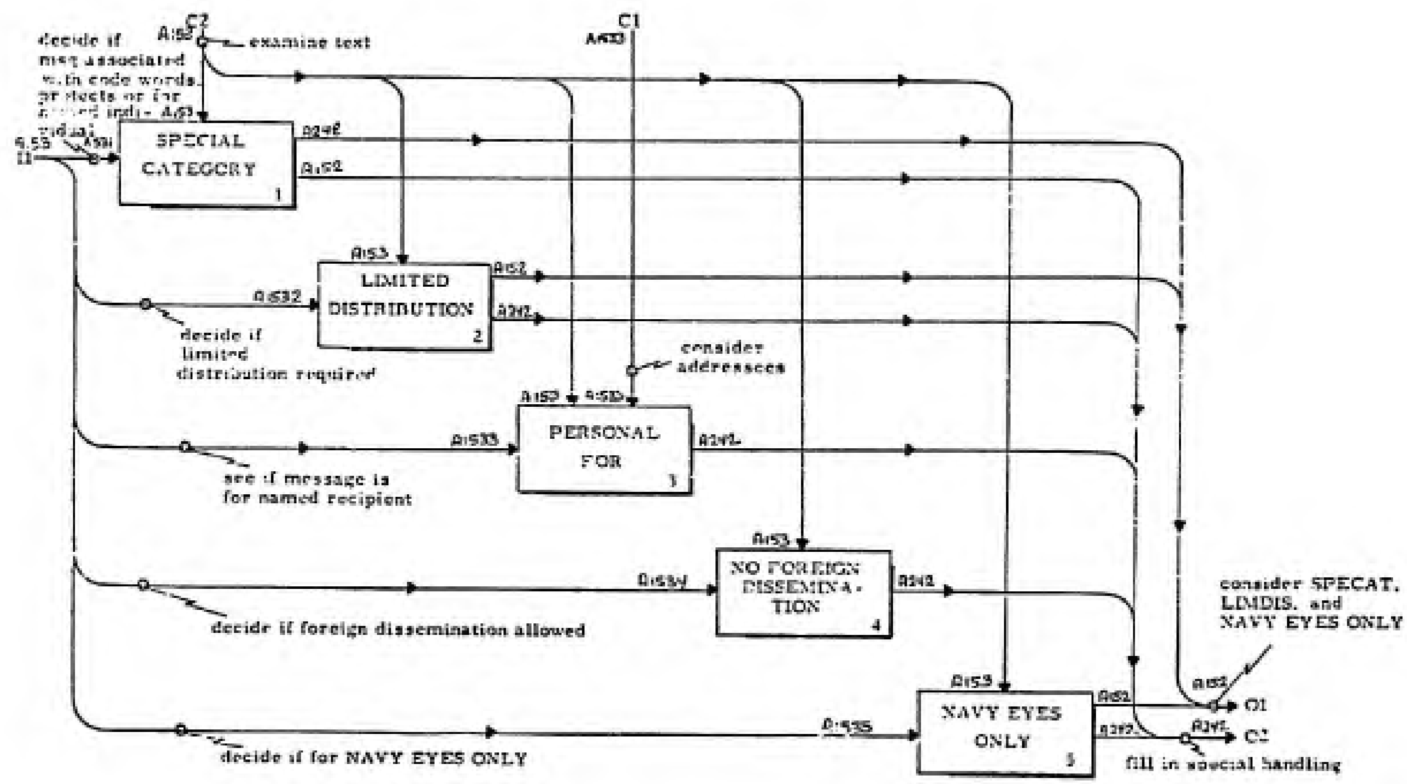
NODE: D2	TITLE: MESSAGE DRAFT	NUMBER: RS86 D2



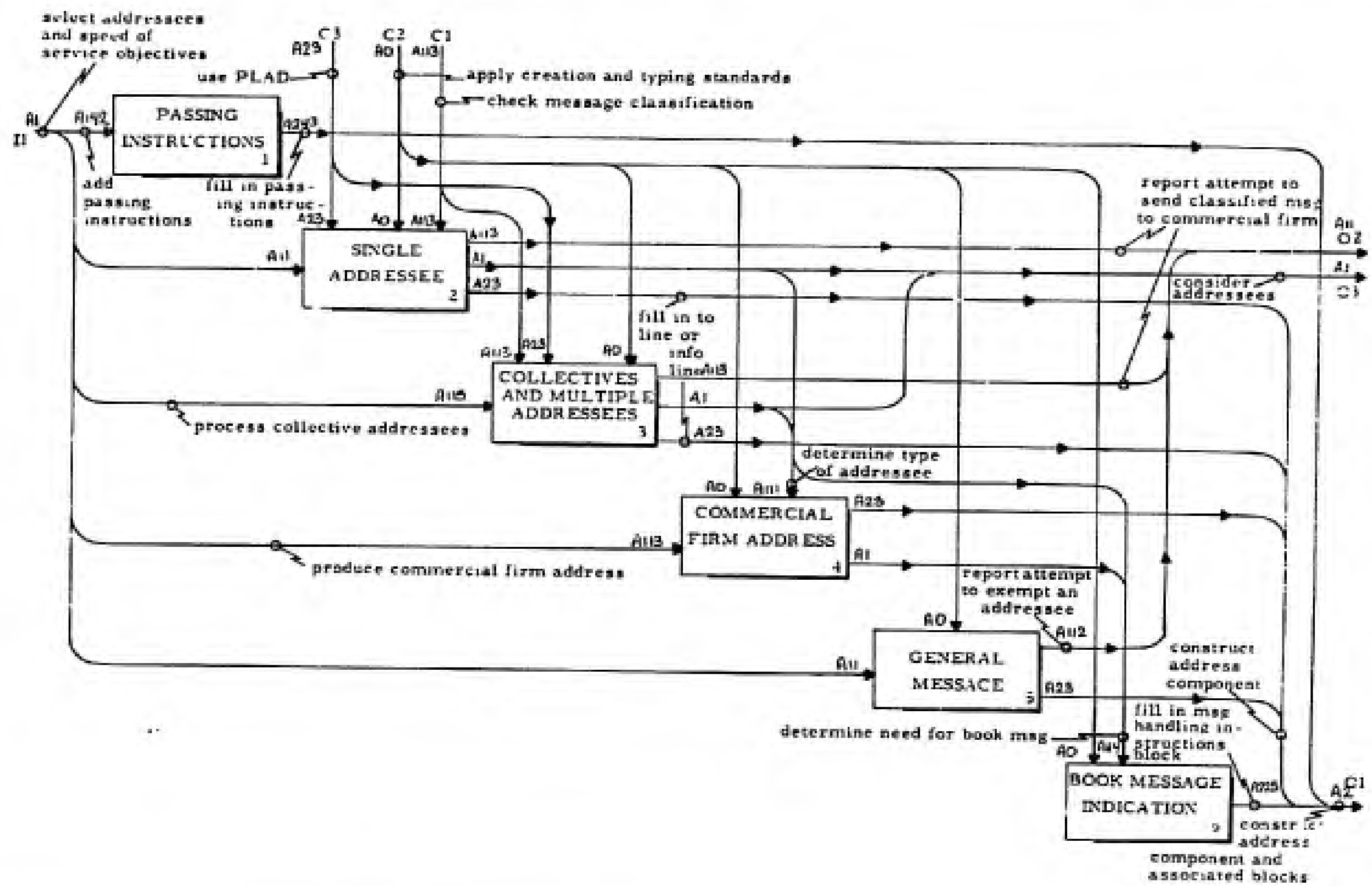
NODE: D21	TITLE: MESSAGE CLASSIFICATION	NUMBER: RS89 D21



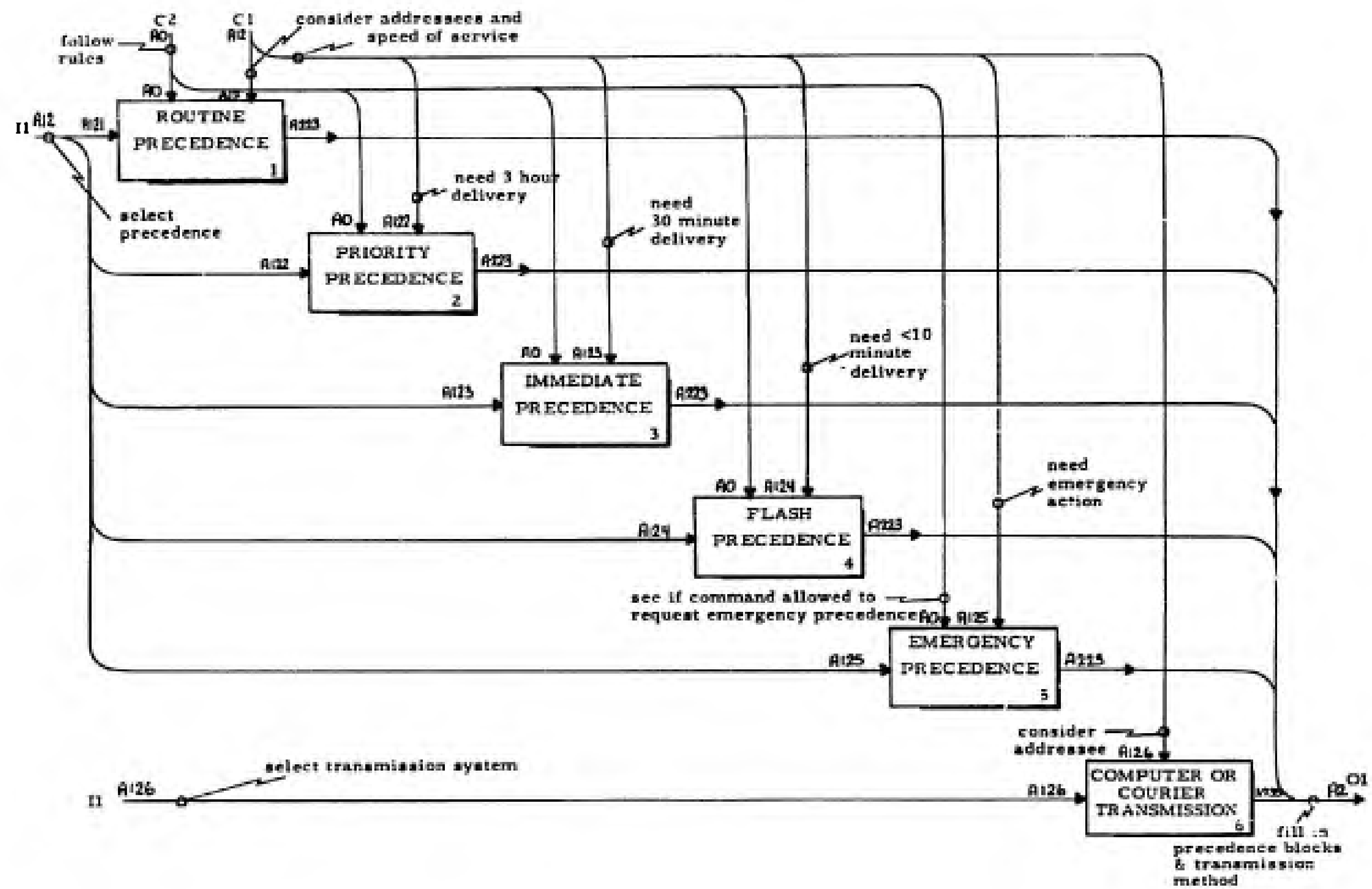
NODE: D211	TITLE: SECURITY CLASSIFICATION	NUMBER: RS90 D211



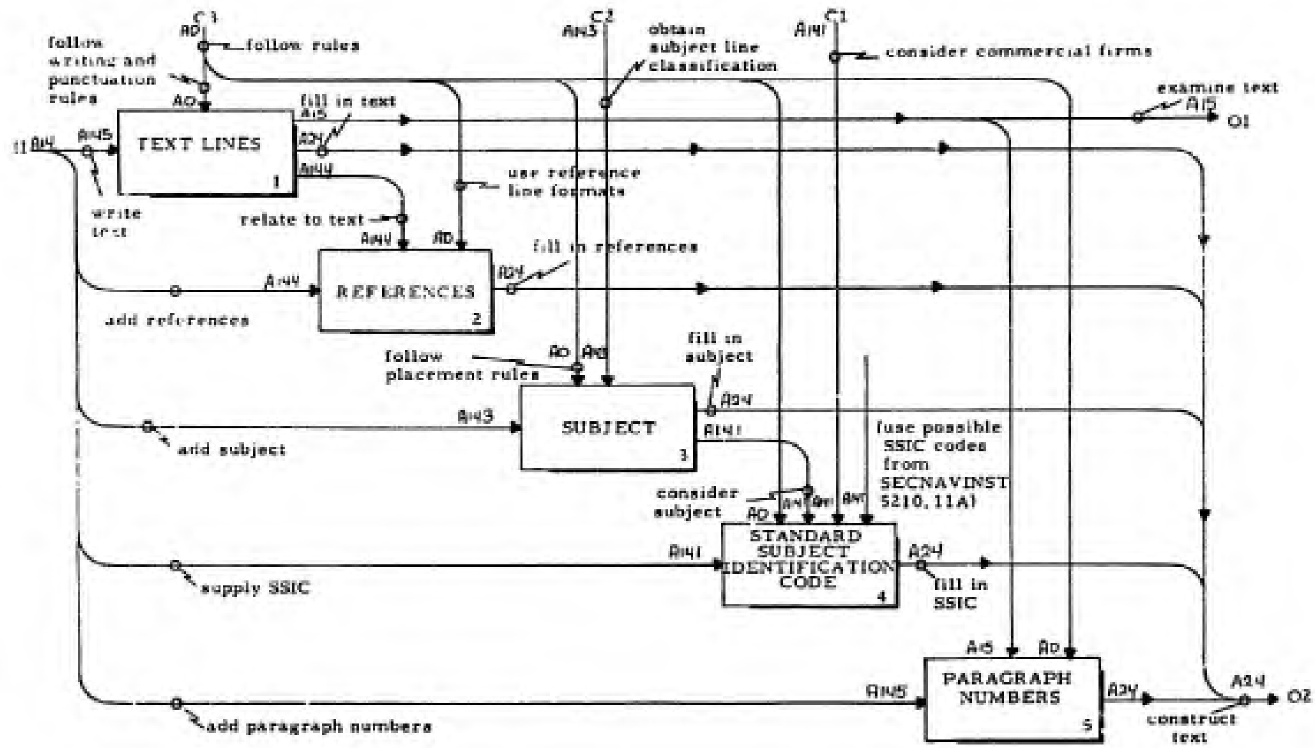
NODE: D212	TITLE: SPECIAL HANDLING	NUMBER: RS91 D212



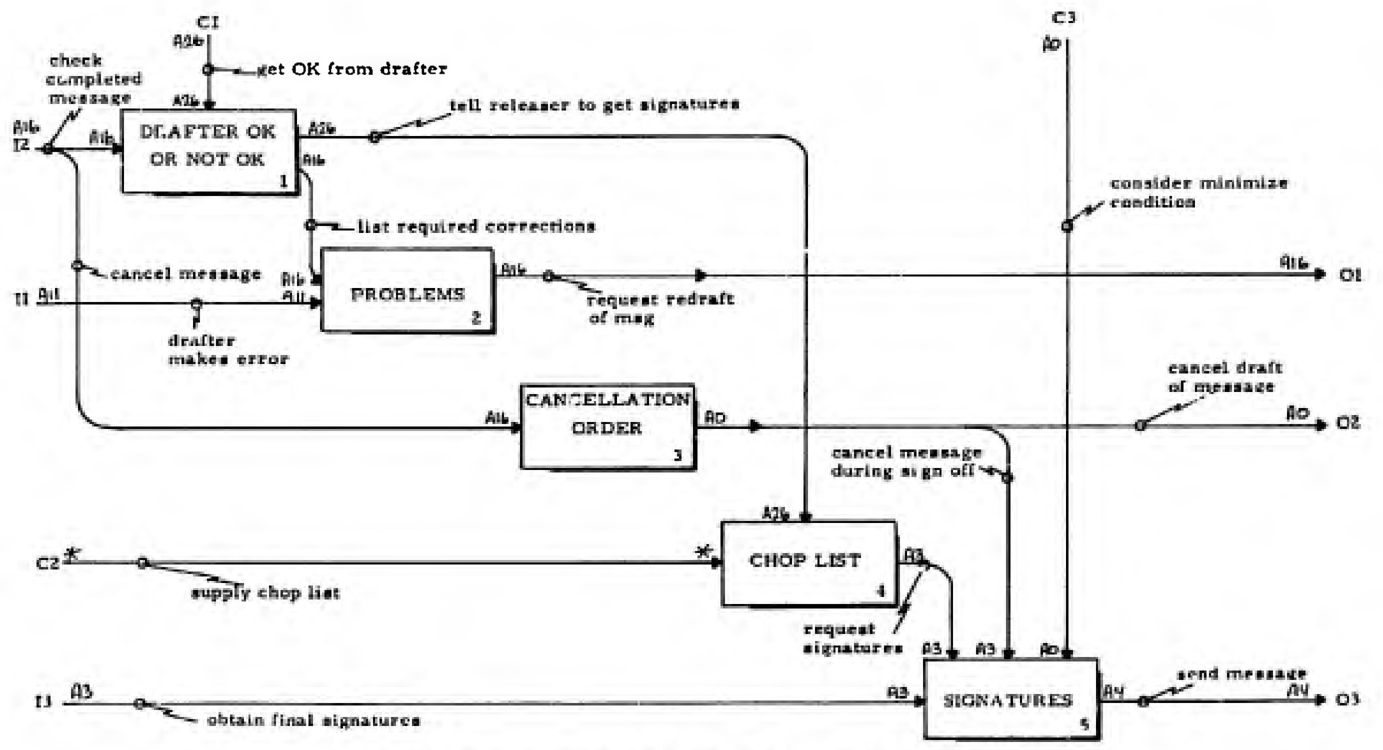
NODE:	TITLE:	NUMBER:	
D22	ADDRESSING INFORMATION	RS88	D22



NODE: D23 | TITLE: TRANSMISSION CONTROL | NUMBER: RS92 | D23



NODE: D24	TITLE: TEXT	NUMBER: RS93	D24



*This activity is outside of the activity r context.

NODE: D4	TITLE: CORRECTNESS VERIFICATION INFORMATION	NUMBER: RS85 (RS84) D4

Section 6

GLOSSARY

ADDRESS INDICATOR GROUP	A predetermined list of specific and frequently occurring action and info addressees.
BOOK MSG	A message which is destined for two or more addressees but is of such a nature that the drafter considers that no addressee should be informed of any other addressee.
CHOP LIST	A list of signatures required to be present on a message before it can be sent.
CLASSIFICATION	The standard security classification of the message and various paragraphs of the message (e. g., unclassified, secret, etc.)
COLLECTIVE ADDR. DESIGNATOR	A single plain language address designator used to address a group of activities.
CONDITION MINIMIZE	A condition imposed by a commander to reduce and control electrical message and telephone traffic within his area of authority during an emergency or exercise.
COURIER	An individual who hand carries messages.
DD-173	The standard form for drafting naval messages.
DECLASSIFICATION INFO.	Information regarding the declassification date of the naval message.
LDMX	A computerized message transmission system.
MSG DRAFTER	The individual who composes a message.
MSG ORIGINATOR	The authority (command or activity) in whose name a message is sent.
MSG RELEASER	The individual authorized to release a message for transmission in the name of the originator.
NAVAL MESSAGE	The prinicpal means by which commanders communicate.
NAVCOMPARS	A computerized message transmission system.

PASSING INSTRUCTIONS	Instructions within the text of a message naming a particular office or individual to whom the originator wants the message routed.
PLAIN LANGUAGE ADDRESS DIR.	A listing of all ordinary language spellings of command short titles and geographical locations used in message addressing.
PRECEDENCE	A system of message categories to allow a message drafter to indicate a desired writer to reader delivery time (e.g., ROUTINE, PRIORITY, IMMEDIATE, FLASH, etc.)
PROFORMA MESSAGE	A message type where the textual elements are strictly formatted.
SPECIAL HANDLING	Additional markings on messages such as Special Category, Limited Distribution, etc.
SSIC	Standard Subject Identification Code.