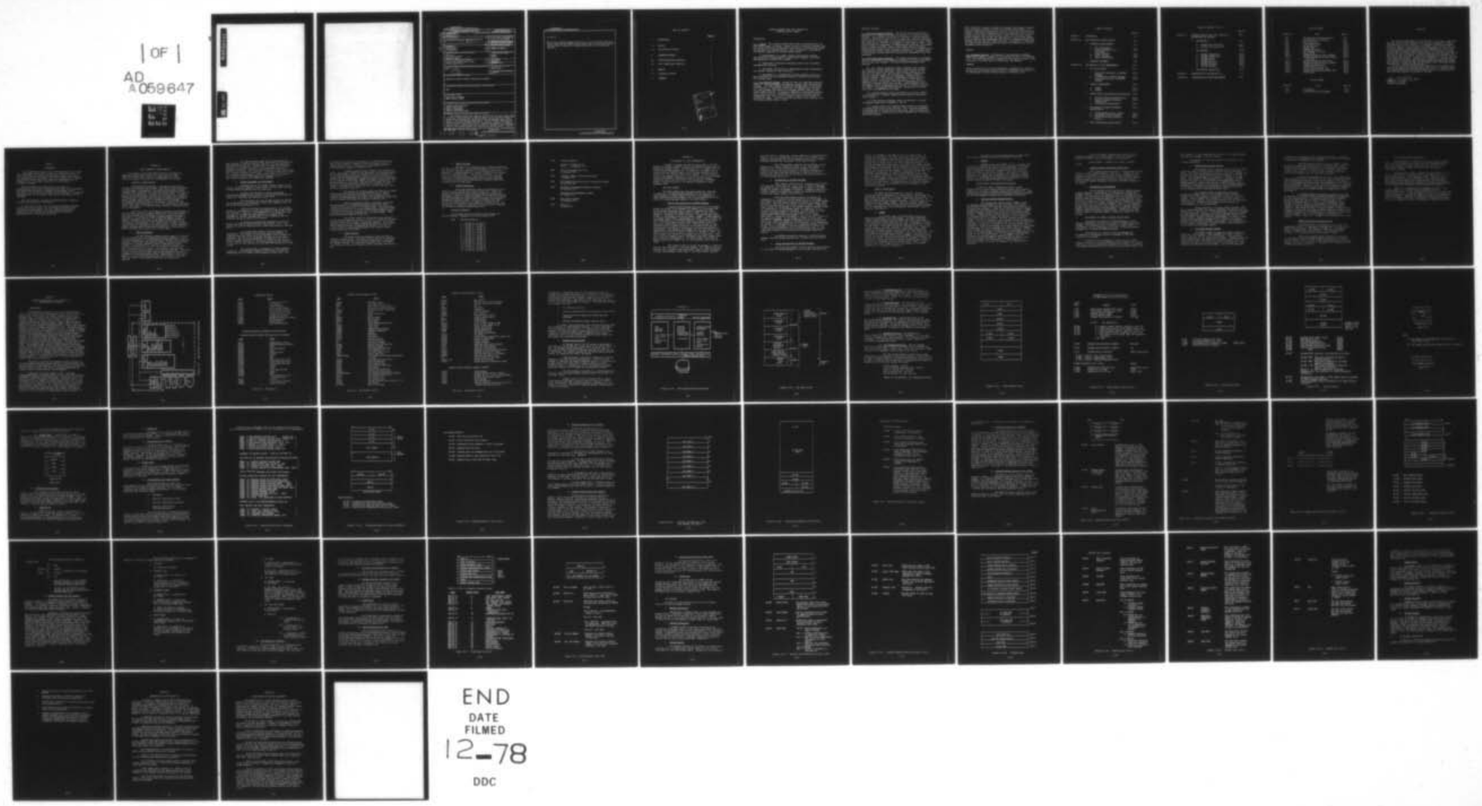
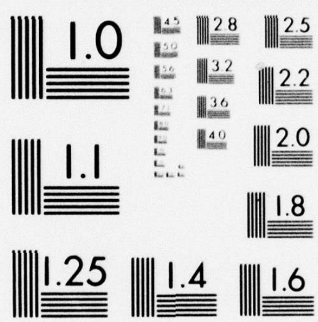
AD-A059 647 INCO INC MCLEAN VA F/G 9/2
STANDARD SOFTWARE BASE (SSB) RELEASE III.(U)
JUL 78 T TRUMP, G MORGAN F30602-77-C-0046
UNCLASSIFIED INCO/1073-378-TR-26-D(F) RADC-TR-78-167 NL

1 OF 1
AD
A059647



END
DATE
FILMED
12-78
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADC-TR-78-167	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) STANDARD SOFTWARE BASE (SSB) RELEASE III		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report 26 Aug 76 - 28 Feb 78
		6. PERFORMING ORG. REPORT NUMBER INCO/1073-378-TR-26-D(F)
7. AUTHOR(s) Thomas Trump George Morgan		8. CONTRACT OR GRANT NUMBER(s) F30602-77-C-0046
9. PERFORMING ORGANIZATION NAME AND ADDRESS INCO, Inc. 7916 Westpark Drive McLean VA 22101		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 31025F 21830135
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (IRDA) Griffiss AFB NY 13441		12. REPORT DATE July 1978
		13. NUMBER OF PAGES 72
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

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)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer:
Samuel DiCarlo (IRDA)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Automatic Data Processing
Computer Networking
Computer Programming
Computer Operating Systems

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

The Standard Software Base (SSB) was developed over the past several years to provide a common inventory of modular software tools with which AN/GYQ-21(V) system users can quickly and effectively develop and implement data systems unique to their site-specific requirements. The SSB provides common baseline software to enable the AN/GYQ-21(V) as a stand-alone, frontend, or communications processor. Release III provides a Terminal Transparent Display Language (TTDL) as the interface between user terminals and the SSB system. (Cont'd)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

78 09 25 059

20 (Cont'd)

SSB utilizes a "gateway manager" capability to route all traffic within the system, separating the communications software from the applications software, thus providing a capability to implement "gateways" to other systems and networks.

TABLE OF CONTENTS

STANDARD SOFTWARE BASE (SSB) RELEASE III
TECHNICAL REPORT SUMMARY

INTRODUCTION

1.1 Purpose. The Standard Software Base (SSB) was developed over the past several years to provide a common inventory of modular software tools with which AN/GYQ-21(V) system users could quickly and effectively develop and implement data systems unique to their site-specific requirements. The specific objectives of the SSB system include:

a. Establishment of a common standard technological software base supporting the development of applications programs, and overall implementation of AN/GYQ-21(V) systems.

b. Elimination of duplicate development efforts and shortening implementation schedules.

c. Development and delivery of comprehensive system documentation and software releases to user activities.

d. Development of a comprehensive training program to equip Air Force personnel with the knowledge required to develop mobile on-site SSB training teams.

1.2 The Technical Problem. During 1973 and 1974, AFIS/IND conducted a survey of USAF Intelligence Data Handling System (IDHS) modernization programs. The USAF programs included the implementation of the AN/GYQ-21(V) system as a stand-alone, front-end, or communications processor. All these programs used some form of systems software, many of which shared common features. To eliminate redundancy in development efforts and to realize both cost avoidances as well as cost savings, AFIS/IND commissioned INCO, INC. in 1975 to develop common system software for AN/GYQ-21(V) users. This effort evolved into what is now referred to as the Standard Software Base, or SSB.

TECHNICAL APPROACH

2.1 Interim Operating Capability. The RSX-11D Operating System was selected as the community standard and retained in its vendor-released versions. This resolved the problem of providing a standard operating system for AN/GYQ-21(V) users and at the same time, permitted concentrating resources on the design of sub-executive modules which would extend the capabilities of RSX-11D to accommodate a wider range of tasks. These tasks included the development of communications networking, and terminal device interface software. Rome Air Development Center's Terminal Oriented Support System (TOSS) was used to achieve interim communications networking and terminal device interface capabilities. Releases I and II of the Standard Software Base were installed and implemented using TOSS components.

2.2 Fully Operational Capability. The design development and implementation of SSB Release III represented significant technical improvements over the first two releases. Four major design enhancements were involved:

(1) The Terminal Transparent Display Language (TTDL) replaced the Interactive Support Capability (ISC) as the interface between user terminals and the SSB system. TTDL provides support for the IBM 3270, the UNIVAC 1652, and the Model 40 Teletype terminals as well as continued support for TTY-compatible terminals used in previous releases. TTDL permits a programmer to design simple or complex terminal screen displays without regard for the type of terminal in use. In addition, TTDL supports concurrent use of an application program by more than one user. Under Releases I and II for example, if two intelligence analysts were to use BUILD function, two copies of the programs supporting the function had to be read into core to accommodate the two cases.

(2) A Gateway Manager concept was devised to route all traffic in the system and to separate communications software from applications software.

(3) A WICS-compatible message format was developed to replace the message format used in Releases I and II.

(4) A global library of shareable common routines was developed to access system files, tables, and data. This relieved programmers of devising code to process data in the system. These routines were used extensively in rewriting applications programms and in developing additional gateways.

Other significant changes to Release III included providing an interface with the Computer Assisted Tactical Intelligence System (CATIS), and developing additional gateways with which to access the DIAOLS and COINS systems. CATIS-related enhancements included developing a two-step user access procedure which controls access to both CATIS and SSB; a CATIS gateway between CATIS and SSB; and modifications to the AUTODIN gateway which accommodate the transmission and receipt of segmented messages and messages from the DSSCS and GENSER networks. The gateways to the DIAOLS and COINS systems have been coded and laboratory tested but are not yet implemented under Release III.

RESULTS

3.1 Acceptance Testing. SSB Release III was subjected to Defense Communication Agency, Category III tests during the week of 13 February 1978. The system satisfactorily passed all tests and is currently being installed at two USAFE sites, at Schierstein, and Ramstein.

COMMENTS

Source documentation for system development, implementation, technical discussions, and operating instructions, are referenced in Section II of the SSB Release III Final Technical Report, 28 February 1978.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

LIST OF FIGURES

LIST OF TABLES

EVALUATION

The value and significance of the SSB lies in its being a common software base that provides standard system software, augmenting the RSX-11D operating system of the AN/GYQ-21(V), eliminates redundant development efforts to develop on-line interactive terminal support capabilities, and provides a means for communications between geographically remote sites via existing and planned communication lines. The SSB enables the individual sites to customize their system software by using only those modules that are necessary to operate its system, minimizing the demands on the resources of the AN/GYQ-21(V), and still being assured of reliable and centrally supported software. The SSB is the foundation or baseline software that supports Intelligence Data Handling Systems development and enhancement activities within the Rome Air Development Center Technology Program Objective R1E, "Indications and Warning."

As such, any future application developed for or by an operational user can be transferred much more easily from one site to another because of the standardization of software interfaces within the system and the ease with which modules can be added to the SSB.

SAMUEL S. DICARLO
Project Engineer

SECTION I

INTRODUCTION

This document provides the Final Technical Report for the Rome Air Development Center under Contract No. F30602-77-C-0046. The report describes the research and development conducted by INCO, INC. during the period 26 August 1976 to 28 February 1978 to enhance the Standard Software Base (SSB) capability established with SSB Releases I and II to support operation of the AN/GYQ-21(V) minicomputer system.

Major achievements are described as both technical accomplishments and contract deliverables furnished to the government. The architecture and special features of SSB Release III are discussed. The utility of the release for the AN/GYQ-21(V) system and the adaptability of the system to meet unique requirements are also addressed.

The last section of the report discusses possible technical direction to be pursued in the future.

Mr. Sam DiCarlo, RADC, was the cognizant government engineer during this contract period. Mr. Carl Compton, Directorate of Intelligence Data Management, Air Force Intelligence Service, Headquarters USAF, was the contracting officer's technical representative for contractual activities.

SECTION II

MAJOR TECHNICAL ACCOMPLISHMENTS

This section of the Final Technical Report describes the major work accomplished by INCO, INC. under contract number F30602-77-C-0046. Major technical accomplishments and major technical documents delivered to the government are discussed.

1. TECHNICAL ACCOMPLISHMENTS

SSB Release III represents a significant technical improvement over previous SSB releases. Four major design enhancements were developed and implemented. The Terminal Transparent Display Language (TTDL), replaced the Interactive Support Capability (ISC), as the interface between user terminals and the SSB system. A Gateway Manager concept was devised to control the routing of all traffic in the system and to separate communications and applications software. Several gateways were written to interface with this manager. A WICS-compatible message format was developed to replace the TISS Common Format used in Releases I and II. The fourth major enhancement was the development of a shareable global library of routines for use by all SSB modules.

After the initial SSB Release III software was developed, it was enhanced to provide increased compatibility with CATIS software. This work included the development of a CATIS interface gateway; development of a gateway to transmit GENSER messages in an offline magnetic tape mode; extensions to the existing AUTODIN gateway in support of GENSER input, sectioned messages, and expanded routing requirements; modifications to the center system to support expanded routing security requirements; and modifications to several modules to support expanded security and access capabilities. These technical efforts are described in Section III of this report.

a. TTDL Enhancements

The TTDL software package provides support for the IBM 3270, the UNIVAC 1652, and the Model 40 Teletype terminals, as well as continued support for the TTY-compatible terminals used in previous releases. TTDL permits the design of simple or complex input/output displays for a virtual terminal screen by adapting the virtual terminal's characteristics to those of the physical terminal upon which an application is to be run. Additionally, TTDL supports multi-user applications programs, e.g., reentrant programs. Under this concept only one copy of a given application program is required to accommodate several simultaneous users. Under Release II and ISC for example, if two analysts were using the BUILD function at the same time, two copies of the programs supporting the function were required.

The TTDL software package was integrated with SSB, and the ability to support the IBM 3270, UNIVAC 1652, and MOD-40 Teletype terminals as well as continued support for all TTY-compatible terminals was implemented. Concurrently, the TTDL/CATIS interface package was tested and installed at the Bunker-Ramo facility in Westlake, California. Work on the MOD II version of TTDL began in May 1977, and was finished in October 1977. This modification to the basic TTDL capability was required to reduce service time to .5 second or less to better support CATIS application programs, and was specifically directed at the UNIVAC 1652 terminal.

b. Gateway Management and Development

Implementation of the Gateway Manager concept in the central system was completed in March 1977. By this time, the basic SSB application program set was also completed.

A gateway was developed to handle unsectioned AUTODIN/DSSCS message traffic as part of the initial SSB Release III capability. This work was completed in March 1977.

The 2780 Remote Job Entry gateway between the IBM 360 and the AN/GYQ-21(V) was completed, tested, and integrated with the SSB system during March-April 1977.

Work on the design and implementation of a baseline gateway for the Community On-Line Intelligence System (COINS) network was completed in June 1977. Recommendations and a design were also developed for an expanded COINS capability to support PACAF requirements, however, implementation of this capability was not addressed during the term of this contract.

An interactive gateway was developed to access the Defense Intelligence Agency On-Line System (DIAOLS - TSS). The gateway was ready for demonstration to AFIS/IND personnel in February 1977.

An Interprocessor Gateway (IPG) was designed and implemented. This gateway contains three separate modules: a communications line interface; an SSB interface, and; an IDHS(C)-II interface. The IPG can be configured with any two of the three modules to serve as an SSB system IPG, an IDHS(C)-II IPG, or as a direct transfer interface in a processor hosting SSB and IDHS(C)-II. At the close of this contract only the SSB-to-SSB capability has been tested.

The CATIS gateway was designed to convert messages between the CATIS-produced format eveloped jointly by INCO and Bunker-Ramo and the internal TISFIL-supported format used by SSB.

The gateway also supplies the queueing and routing services necessary to allow CATIS to transmit and receive messages through SSB communications software. Integration of this gateway with the CATIS system was completed in February 1978.

A GENSER transmit/receive capability was required by CATIS. The transmit capability was implemented as a separate gateway. The GENSER gateway provides the interface with the SSB system and writes outgoing GENSER messages to magnetic tape for offline review, validation, and transmission. Review and validation are accomplished by the separate utility modules GENUTY and GENVAL, respectively. The GENSER gateway will also acept DSSCS messages in JANAP 128 tape format as a communications backup capability. Transmission is performed at a communications center and is not an SSB function. This work was completed in June 1977.

The Release III AUTODIN gateway was modified for CATIS. Numerous detailed extensions were made; the ability to support transmission and reception of multiple section messages and the reception of GENSER messages was added. This gateway was incorporated into the standard SSB Release III software in October 1977.

Security requirements made it necessary to manually review messages for all GENSER destinations before they were released for any destination in the DSSCS system. This capability required modification of the center system modules, primarily TISMDM, to route by priority and to receive positive acknowledgement before releasing lower priority transmissions of the same message. These modifications were completed in May 1977.

The access controls for SSB software were modified to allow controlled subsystem selection in order to incorporate the CATIS and SSB systems under the same set of access controls. At the same time, the access control module, SECMON, was rewritten. The security related system files were reorganized and modified to reflect new access criteria for subsystem authorization and terminal overlay (characteristics of the U-1652).

c. <u>Global Routines</u>

The global library of shareable common routines was completed in January 1977. The global library was used in rewriting the applications programs and is currently being used in developing the gateway programs. All application programs except the SBAPRT print module are written in reentrant code to facilitate simultaneous use.

d. AFAITC Training

Two weeks of training was given to AFAITC personnel on the SSB in October 1977. This included presentation of a basic course with specific training objectives for intelligence analysts, programmers, and computer operators. Current SSB manuals were provided for use as study references and guides to hands-on use of the system in a laboratory setting. The objective of this training course was to provide AFAITC personnel with sufficient knowledge of SSB Release III and CATIS modifications to prepare formal USAF training courses for SSB users.

e. System Installation

An interim operating version of SSB Release III was installed on the AFIS/IND 21(V) system in May 1977. Under this version, AFIS/IND tested basic system performance using the new capability to support the MOD-40 Teletype and the IBM 3270 terminals. The AUTODIN gateway was tested successfully using the Western Union Programmable Terminal Controller (PTC) and Analytics Telecommunications Line Controller (TLC). Category III testing and certification of SSB Release III was completed in February 1978. The SSB Release III system was installed at Air Force sites in Ramstein and Schierstein, Germany during February 1978.

2. TECHNICAL DOCUMENTS

The following technical documents were produced and delivered between 9 September 1976 and 28 February 1978:

A001 R&D Status Reports

1 - 26 Aug 76 - 25 Jan 77
2 - 26 Jan 77 - 25 Feb 77
3 - 26 Feb 77 - 25 Mar 77
4 - 26 Mar 77 - 25 Apr 77
5 - 26 Apr 77 - 25 May 77
6 - 26 May 77 - 25 Jun 77
7 - 26 Jun 77 - 25 Jul 77
8 - 26 Jul 77 - 25 Aug 77
9 - 26 Aug 77 - 26 Sep 77
10 - 26 Sep 77 - 25 Oct 77
11 - 26 Oct 77 - 25 Nov 77
12 - 26 Nov 77 - 25 Dec 77
13 - 26 Dec 77 - 25 Jan 78
14 - 26 Jan 78 - 25 Feb 78
15 - 26 Feb 78 - 28 Feb 78

A002	Technical Reports:	
	Interim - 25 August 1977	
	Final - 28 February 1978	
A004	Test and Implementation Plan	
	28 February 1978	
A005	Standard Software Base Review Document	
	1 November 1977	
A006	SSB Program Specifications and Installation Manual	
	28 February 1978	
A007	SSB User's and Computer Operator's Manuals	
	28 February 1978	
	SSB Appliction Programmer's Manual	
	19 October 1977	
A008	Test Analysis Report	
	28 February 1978	
A009	Work Plan	
	29 April 1977	

SECTION III

SSB RELEASE III CATIS ENHANCEMENTS

A number of changes and additions were made to the SSB Release III system in order to support the Computer Assisted Tactical Intelligence System (CATIS). The objectives of this support were to provide the AUTODIN gateway with the capability to transmit and to receive DSSCS and GENSER traffic, and to provide enhanced security controls for the combined CATIS and SSB systems. The five areas of technical achievement discussed in this section are the CATIS Gateway, security enhancements, modifications made to the SSB central supervisory group, modifications to the SSB Release III AUTODIN gateway, and modifications to the Terminal Transparent Display Language, TTDL.

1. THE CATIS GATEWAY

The CATIS gateway has three basic functions. They are: 1) reception of incoming AUTODIN messages and the conversion of those messages into CATIS-compatible format; 2) transmission of CATIS-originated AUTODIN messages, and; 3) providing CATIS with the status of incoming messages queued to the CATIS system.

a. Receipt and Conversion of AUTODIN Messages

Incoming AUTODIN messages are received by the AUTODIN Gateway and placed in the Message File (MSGFIL) on the system disk. If the gateway determines that a message is destined for CATIS software, the SSB Message Header Block is marked to show that the message is addressed to a CATIS recipient. The gateway then passes control to the Message Distribution Module (TISMDM). TISMDM examines the Message Header Block, acknowledges the "token" route to CATIS, and then passes control to the CATIS Gateway via a priority-structured call. the CATIS Gateway retrieves the Message Header Block and constructs a CATIS Control Block (CCB). The CCB contains the message destination, message originator, the precedence and security classification attributes of the message, and the date-time-group of transmission. The completed CCB becomes the first disk block of an RSX-11D FCS file which is used to transfer the message to CATIS. The CATIS Gateway then transfers the message text to the FCS file. When the FCS file is closed, an entry is posted in the CATIS Gateway Queue File ([102,102] CATISQUE.QUE;1). The CATIS Gateway then issues an RSX-11D wakeup macro (RQST$) call to the appropriate CATIS modules, TRIRP.

When TRIRP is read, to accept the message, it issues an RSX-11D send data macro to the CATIS Gateway. The 13-word data block contains only a function code which means, "Send to CATIS the highest priority input message on your queue." The CATIS Gateway complies with this request, passing the version number of the appropriate FCS transfer file in a 13-word RSX-11D send data block together with a function code meaning, "This is the highest priority input message." A different function code is returned if the queue is empty.

When CATIS software completes the processing of each message, it it issues a VSDR$ macro back to the CATIS Gateway, to permit deletion of the processed AUTODIN message. The macro parameters contain the version number of the AUTODIN message and a function code indicating that processing is complete. The CATIS Gateway then deletes the FCS file and deletes the associated entry in the CATIS Gateway Queue File.

b. Transmission of AUTODIN Messages

CATIS support software builds a message using FCS and a file descriptor block identical to that used in receiving messages. The message is then closed and a VSDR$ macro is issued to the CATIS Gateway to transmit the message via AUTODIN. Parameters specified in a buffer passed in the VSDR$ call include the transmission function code and the version number of the FCS file.

When the CATIS Gateway receives the request for transmission, it constructs the SSB mesage from the FCS message. It then queues the request by the priority contained in the CATIS Gateway Queue File. When the request reaches the head of the transmit queue, the CATIS Gateway marks it and issues a call to TISMDM. TISMDM then passes control of the message to the AUTODIN (DSSCS or GENSER) Gateway for transmission. If the message cannot be transmitted, or is rejected, the CATIS Gateway is informed. In such cases, the CATIS Gateway determines whether or not the message is part of a multi-segment CATIS-originated message. If it is, a further examination is conducted to determine if this is the first section of the segmented message. If the message is found to be the first section of a segmented message, the gateway advises the originating CATIS module that the message was not transmitted. If the message was the second, or a subsequent section of a segmented message, the CATIS Gateway routes the message to the SSB system console terminal for correction and retransmission via a stand-alone MCR task. In such cases CATIS software is not notified of the message transmission failure.

If AUTODIN successfully transmits a CATIS-originated message, the CATIS Gateway deletes both its queue entry and the FCS file.

c. Status and Retrieval of AUTODIN Messages

While the CATIS Gateway notifies CATIS software modules of the arrival of AUTODIN messages, it is not likely that CATIS can process the messages at the same rate at which they arrive. To resolve this difficulty, the CATIS Gateway maintains a priority ordered log of message entries for all incoming messages. These entries are maintained by this log until the gateway receives notification to delete them. If a CATIS module is engaged in processing a message when notification is received that another message has arrived, the new message may be deferred by issuing one of the RSX-11D Receive Data macros. One of these macros must be used so that RSX-11D will return the nodes changed to the CATIS Gateway when the VSDR$ macro is executed. To retrieve a message which has been deferred, CATIS software may call the CATIS Gateway using a VSDR$ macro requesting the status of the highest priority message queued. When the CATIS Gateway receives this request, it queries the CATIS Log Queue File and passes the information back to the calling module in a buffer via a VSDR$ macro. The module may then retrieve the message from the FCS file, and once finished, notify the gateway it has completed processing.

2. SECURITY ENHANCEMENTS

The CATIS and SSB systems were logically joined by considering them as subsystems of a larger system. The Release III LOGON module was rewritten to include a subsystem selection menu. The menu currently provides three options: CATIS, SSB, and EXIT. The modified LOGON module was renamed as SIGNON. In similar fashion, the LOGOFF module was renamed SIGNOFF. These two modules, SIGNON and SIGNOFF, are currently the only SSB Release III applications programs that are compatible with the CATIS-specific version TTDL, which is referred to as TTDL II.

a. SIGNON

For the most part, the SIGNON module retains the functions maintained by the LOGON module. The major difference between these two modules is that SIGNON uses a two phase system access procedure whereas LOGON uses only one. Using LOGON, the user need provide only user identification, name, and password to access the SSB system. Once these entries are verified, he is free to use any SSB function as either a general or privileged user. SIGNON imposes an additional control in that users must be previously authorized to access specific subsystems, e.g., SSB, CATIS, SARP. The user specifies the subsystem he wishes to access. The selection is checked by the security monitor module, SECMON, to determine if the user is authorized. After three unsuccessful attempts to SIGNON the terminal is disconnected from the system, thereby denying access to any system or subsystem function function except SIGNON. To gain access to any other system function, both phases of the SIGNON process must be completed successfully.

In addition to the two-phase access process, SIGNON also logs the time and the subsystem accessed by the user.

b. SIGNOFF

SIGNOFF differs from LOGOFF in that a user may sign off from one subsystem and access another subsystem to which he is authorized without the necessity of performing phase one SIGNON again. Prompts are issued to determine if the user wishes to terminate access to the entire system, or gain access to another subsystem. If termination is signified, the terminal is disconnected from the system and reinitialized for a subsequent SIGNON function. If accessing another subsystem is signified, the subsystem selection process in SIGNON is initiated.

3. CENTER SYSTEM CATIS-RELATED MODIFICATIONS

In order to support CATIS/SSB operation, several modifications to center system supervisory modules were required, primarily in the areas of message routing and accountability. These modifications included priority-ordered message delivery to gateways, and expanded message processing and accountability procedures.

a. Priority-Ordered Message Delivery

The Message Distribution Module (TISMDM) routes all messages between gateways in the SSB system. The Network Characteristics Table (NCT) maps each routing network identity to the appropriate gateway. A flag bit in the NCT is used to identify gateways which are designated to receive priority routing. TISMDM evaluates message routing tokens to identify priority-designated gateways. TISMDM performs a sequential scan of destination tokens in a message, marks each token, and compiles a list of network identifiers found in the tokens. Prior to issuing an SRB to a gateway, the list is scanned for the presence of priority-designated networks. The first one found reduces the list to one and the scan is terminated. An SRB will be issued to the priority gateway only, although all tokens in the message will remain marked for their proper destinations. The priority gateway must return status to TISMDM to relieve TISMDM of restart responsibility for the message. At this point, the priority gateway is responsible for the message until it is routed to its other destinations. If the message is successfully delivered, the priority gateway requests TISMDM to mark all destination tokens addressed to the gateway as successful, and to route the message to all destinations. TISMDM examines the destination tokens and routes the message to all remaining destinations.

If, for any reason, the priority gateway could not transmit the message, it requests TISMDM to mark all tokens as unsuccessful and to return the message to its originator.

These changes to TISMDM do not affect non-CATIS routing.

b. Expanded Message Processing Capabilities

Three capabilities were added to the SSB global routine library to support CATIS operation. They are: 1) Removal of restrictions on the number of destinations that may be specified in a message; 2) Removal of the order restriction which formerly existed between originator and destination blocks in a message, and; 3) increasing the number of message type blocks that can be handled by the system.

c. Accountability Enhancements

The Accountability Functions Module (TISAFM) provides a central place for accountability in handling disk resident TCF messages (TISFIL). Once a TCF message is built by the originating gateway, only TISAFM will make modifications to the message; all other modules access the message as read-only. The token flag byte in all destination blocks is used to determine the processing status as well as any error conditions. Using this information, TISAFM will request either TISJOR only, or TISJOR and TISMDM to return to the originator message files processed with error conditions, or message files from gateways requiring positive acknowledgement. The positive acknowledgement feature was added to support CATIS. TISAFM was also modified to support an expanded number of destination tokens in a TCF message.

4. SSB RELEASE III AUTODIN GATEWAYS MODIFICATIONS

SSB Release III employs two AUTODIN gateways; one for message transmission, the other for receipt of AUTODIN unsectioned messages. Both gateways had to be modified to support CATIS requirements for the transmission and receipt of sectioned AUTODIN messages over the DSSCS and GENSER networks.

The capability to transmit sectioned messages was accommodated by processing sections of messages as individual single-section messages.

Incoming sectioned messages are detected by the CATIS Gateway and routed to a Section-Ordering Module, SBGCOM. This module temporarily stores message section numbers on disk until all sections of a message have been received, then notifies the CATIS Gateway. The

CATIS Gateway, in turn, assembles the sections into a single message and passes the message to the CATIS system.

The operation of these two gateways is described in the following paragraphs.

a. <u>TISGTA AUTODIN Receive Gateway</u>

This gateway converts messages received from an AUTODIN Switching Center in either DSSCS or GENSER format to SSB Common Format. The converted messages are stored in the central message file maintained on disk, and are accessible for processing by the SSB system. The receive gateway can accommodate a variety of message types. These include: DSSCS messages with a precedence of Flash-Override or lower, in both narrative and Query/Response format; GENSER messages in either plain address or CODRESS format; service messages in either DSSCS or GENSER format; AUTO-CALL-BACK messages in either DSSCS or GENSER format originated at other SSB sites; and control messages from the AUTODIN switching center.

All incoming traffic is examined for errors in transmission message type, and for imbedded control messages. Whenever errors are detected, notification is output to the system console. Control messages are displayed at the system console immediately upon receipt. These include requests for initialization, cancellation of transmissions, message rejection notices, and acknowledgements of lost transmissions.

During conversion of incoming AUTODIN messages, TSGTA places network dependent and network independent information into the appropriate routing blocks in the message header. When all processing has been completed, a message containing the message sequence number and identifying the user to whom it is addressed is output to the system console. Whenever AUTO-CALL-BACK messages are received, TSGTA builds an acknowledgement message which is passed to the Message Distribution Module for subsequent transmission. When messages with a flash or higher precedence are received, TSGTA outputs notification of the arrival to the system console.

b. <u>The GENSER Transmit Gateway</u>

The GENSER Gateway distinguishes between GENSER and DSSCS (R&Y) type messages. DSSCS messages are routed to tape if a special flag has been set by the CATIS gateway. This provides a backup capability to the communications hardware. GENSER messages are constructed in JANAP 128G format for output to magnetic tape. The tapes are printed by an off-line review utility program for manual acceptance or rejection of each message. A record validation and response utility program is used to mark each message as acceptable or not, and to issue appropriate responses back to SSB so that message accountability procedures may be performed.

The SSB Message Distribution Module (TISMDM) receives all AUTODIN transmission requests from the CATIS gateway. Since GENSER traffic is always reviewed before transmission, any messages routed, either exlusively or in part, over the GENSER network are automatically routed to the GENSER Transmit Gateway for journalization and validation prior to any other processing.

During the journalization process, a tape volume directory is created on disk. Each entry in the directory cross-references SSB message sequence numbers to the tape reel number of the tape to which the messages have been written. The maintenance of the directory requires operator interaction in that the correct tape must be mounted and placed on-line. The data referenced by the directory is dumped to 9-track, 800 bpi tape in a format suitable to Communication Center requirements.

As GENSER messages are manually reviewed and accepted or rejected, the entries for the specified messages are deleted from the tape volume directory. The operator uses an off-line GENSER Review and Validation Utility (GRU) to review messages, and a Validation Response Module to accept or reject messages. When a message is accepted, GRU calls the message Accountability Module (TISAFM) to mark the GENSER destination token in the message as completed. The Message Distribution Module (TISMDM) is then called to complete routing of any remaining non-GENSER destination tokens. When a message is rejected, the GENSER Review and Validation Utility module instructs TISAFM to mark all destinations as unsuccessful and instructs TISMDM to return the message to its originator for further action.

c. <u>GENSER Review and Validation Utility</u>

The GENSER Review and Validation Utility, GRU, is a stand-alone program. It processes the message placed on tape by the GENSER Security Gateway. GRU produces a printout of messages for review by communication center personnel. GRU requires no SSB services; it is activated and run under operator control.

Operations personnel signify acceptance or rejection of a message by using the GENSER Validation Response Module (GVR). Operator inputs include the tape volume number upon which the message resides, and a yes or no response. GVR passes this information to the Message Accountability Module.

5. TTDL CATIS-RELATED MODIFICATIONS

Modifications to the Terminal Transparent Display Language (TTDL) for CATIS support were divided into two large subdivisions. TTDL I was modified to support CATIS operation, and TTDL II was specifically designed to meet the response time and paging requirements of CATIS.

Modifications to TTDL I included providing support for the IBM 3270 and UNIVAC 1652 terminals, and the implementation of several new features. These features include: light pen selection and termination functions for the UNIVAC 1652; a watchdog timer; application program suspend and resume functions (not currently in use); default initialization of retrieved data fields; user validation of input fields; and an external buffer for security purposes.

TTDL II is an adaptation and refinement of TTDL I. Its primary purpose is to improve terminal response times to a CATIS-required level. It was written specifically for the UNIVAC 1652 terminal and does not include the dynamic definition and reformatting features of TTDL I. Displays are compiled in advance and stored on disk. The reader is referred to the separate Final Technical Report for further information on TTDL II (Contract No. F30602-77-C- 0046).

SECTION IV

STANDARD SOFTWARE BASE (SSB) RELEASE III
ARCHITECTURE AND FEATURES

1. ARCHITECTURE

SSB Release III consists of four major software components. They are the Terminal Transparent Display Group (TTDL), the Applications Modules Group, the Gateway Manager Group, and the Gateways Group. The first, TTDL, provides the interface between different types of terminals and the system. It enables the designer/programmer to develop simple or complex input and output displays for a virtual terminal display screen. TTDL's software adapts the virtual terminal's characteristics to those of any physical terminal on which a terminal applications is run. The Applications Modules Group provides the I/O terminal user with a variety of message handling functions such as logging on to access the system, building and transmitting messages, and receiving and reviewing message traffic. The Gateway Manager Group interfaces all SSB applications software with non-applications software. It provides centralized traffic control for the system and separates communications, terminal, and system software from each other. This separation makes possible the interface with other site-dependent systems such as CATIS. The Gateway Manager concept protects against unauthorized access to SSB capabilities and files, and also provides for automatic journalization of incoming and outgoing system traffic. The Gateways Group provides the actual gateways, or interfaces, between SSB and external networks such as AUTODIN, CATIS, and the DIA on-line system (DIAOLS). Figure IV-1 illustrates the interaction of each subsystem in SSB Release III. Table IV-1 lists the individual modules that comprise SSB and their functions.

The SSB is an orderly and systematic system architecture providing common networking software for U.S. Air Force systems that use the AN/GYQ-21(V) minicomputer. The SSB system provides users with software which satisfies their needs for network communications, terminal support, system security, and maintenance.

Modularity is the key element of SSB. Individual commands/agencies/activities may use one or more of the SSB software subsystems or modules at their discretion. Integrated, tested, and validated software subsystems are available to assist users in the development of their respective system/capabilities and to internet or interface with other data files, data bases, and/or intelligence sources within the community. As a result, users throughout the SSB-supported community can evaluate their mission requirements, select only those SSB subsystems which will support that mission, and plan to fill the gaps with command/agency unique software. It

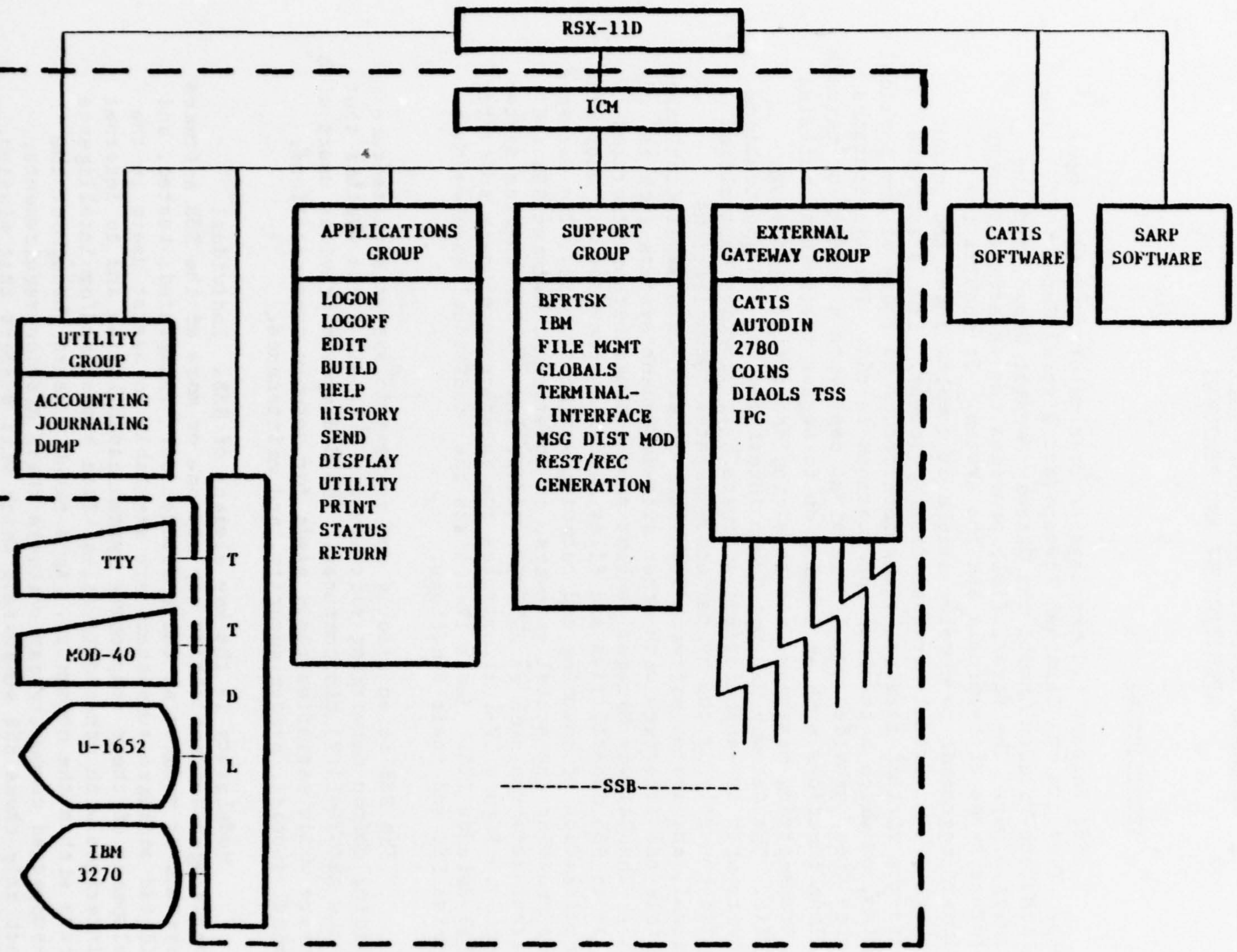


Figure IV-1. SSB Release III System/Subsystems

SUPERVISORY MODULES

NAME	TITLE
TISAFM	Accounting Function
TISFIL	File Manager
TISGBL	Global Routines
TISIBM	Input Buffer Manager
TISIMP	Input Message Processor
TISINT	System Initialization
TISJOR	Journalization Control
TISLOG	Message Accounting
TISMDM	Message Distribution Module
TISSAV	Restart/Recovery
TISSTS	Status Module
TISTAP	Tape Journalization

MESSAGE HANDLING AND TERMINAL SUPPORT MODULES

MESSAGE HANDLING MODULES (SSB Application Programs)

NAME	TITLE
GENUTY	GENSER Review Utility
GENVAL	GENSER Validation Utility
SBABLD	Build
SBABTP	Tape Unavailable
SBADMP	Dump
SBADSD	Display Driver
SBADSP	Display
SBAEDI	Edit
SBAHLP	Help
SBAHRD	History/Retrieve Disk
SBAHRQ	History/Retrieve Query
SBAHRT	History/Retrieve Tape
SBALOF	Logoff
SBALON	Logon
SBAPRT	Print Applications
SBARET	Return
SBASND	Send
SBATAP	History/Retrieve Tape Initialization
SBAUTL	Utility
SBSPRT	Print Service Modules
SECMON	Security Monitor

Table IV-1. SSB Modules

TERMINAL SUPPORT MODULES (TTDL)

NAME	TITLE
ADDNTT	Add Name to-TIF
ADDSTT	Add Screen to-TIF
AFNOT	After Flash Notification
AIM	Application Interface Module
ALIVSN	Allocate Virtual Screen Node
APPEND	Append
BLDSA	Build Screen Area
BLK	Block
BUFMAN	Buffer Manger
BYTADD	Byte Additon
CHKINP	Check Input
CLD	Class Dependent Routines
CLNSTK	Clean Stack
CLRNDE	Clear Node
CMBMGR	Command Buffer Manager
CMDINT	Command Interpreter
COMMND	Command
DATAB	Data Block
DISCON	Disconnect
DISLIB	Display Library
DKIO	Disk Input/Output
EP	External Processor
EXEXFN	Execute External Function
EXTCMD	External Command
FIELDS	Field Handling Routines
FNDCL	Find Command Line
FNDNTT	Find Next Names to TIF
FNDSA	Find Screen Area
FNDSTT/FSTSTT	Find Screen to TIF/Find Start Screen to-TIF
FNDTAB	Find Conversion Tab Node
FNDTBN	Find Tabling Node
FNDTN	Find Terminal Node
FNDTTT	Find Terminal Type Table
FRCRD	Force Read
FREEUP	Free Up TTDL Resources
FREUSN	Free Virtual Screen Node
GETPARK/GETTIF	Get Packed Address/Get TIF Address
GETS	Get Information from PCB
GETSTT	Get Screen to TIF
GETTN	Get Terminal Node
GETTNT	Set Get Terminal Node Pointer to Top
HOLDEL	Hold Delete
IBM 3270	IBM 2370 Routines

Table IV-1. SSB Modules (Cont'd)

TERMINAL SUPPORT MODULES (Cont'd)

NAME	TITLE
KLFLSH	Kill Flash
MAPPRI	Mapping and Priority Routines
MAPAPP	Map Into Application Program
MD4	Mod 40
MOVDAT	Move Data
MSOUT	Message Out
PAKTIF	Pack TIF Address
PASDSP	Pass Display
PCBDSP	Process Control Module
PIO	Primary Input/Output
PREPRO	Preprocessor
PSP	Postprocessor
PTNSTT	Point to Next Screen to TIF
PTNNTT	Point to Next Names to TIF
RESAPP	Resume Application
RTNAPP	Return Application
SETRW	Set Read/Write Bit
SIO	Secondary Input/Output
SIOCOM	SIO Common
SLLDLL	Single Linked List/Double Linked List
SRCHTK	Search for Task Name
SRCHXX	Search Listhead
SUSDSP	Suspend Display
TERM	Terminate Application Program
TDLDIO	TTDL Disk Input/Output
TINIT	Terminal Initialization
TMRACT	Terminal Action Required Routines
TMRCOM	Terminal Routines in Common
TMRNCM	Terminal Routines not in Common
TSTPCB	Test Process Control Block
TTY	Teletype Terminal Routines
UNIVAC 1652	UNIVAC 1652 Routines
UNPTIF	Unpack Terminal Independent Format

COMMUNICATIONS INTERFACE (GATEWAY) MODULES

CATISG	CATIS Gateway
TISGT6	AUTODIN General Service Gateway
TISGTA	AUTODIN Receive Gateway (DSSCS/GENSER)
TISGT1	AUTODIN Send Gateway (DSSCS)
TISGT2	COINS Gateway
TISGT4	IBM 2780 Emulator
TISTGT	Terminal Gateway (Functions within the Supervisory Modules Group)

Table IV-1. SSB Modules (Cont'd)

is important to understand that the user selects and uses only SSB-supported software subsystems that are requred for his operation, thereby eliminating excessive overhead. Figure IV-2, SSB Subsystem Relationships, describes this modular concept using existing or planned SSB subsystems to demonstrate how the subsystem can operate alone or with other subsystems, as required. The subsystems illustrated in Figure IV-2 are:

- File Handling Services
- Computer Assisted Tactical Intelligence System (CATIS)
- Communications Processing and Message Handling Subsystem
- Terminal Transparent Display Language (TTDL).

In effect, a given user may select from the SSB those modules or elements of common modules pertinent to unique requirements with the assurance that they will: (1) interact with other SSB elements, (2) serve and support unique applications, (3) not impose a burden of superfluous SSB modules or elements on his system operations, and (4) accrue benefits arising from other common user requirements and modernization programs.

a. Message Data Structures

All message data in the SSB system is maintained on disk files in a WICS-compatible format known as the Common Format, or TCF. All message traffic generated by an I/O terminal user for distribution within internal or external networks is placed into TCF by the system. All traffic entering the SSB system from an external network is translated into TCF by the system.

Outgoing traffic is initially constructed in TCF and remains in that form through journalization. A copy of the TCF message is converted into network dependent format by the appropriate gateway modules and transmitted in network-compatible format. Incoming traffic is converted into TCF by the appropriate gateway module and remains in TCF through the journalization process.

Each TCF message is recorded as an indexed-sequential disk file under a message sequence number assigned by the disk I/O processor, TISFIL.

Message traffic in TCF is recorded under a logical block structure comprised of several types of fixed and variable length data blocks. Each physical block may accommodate up to 512 bytes, or 256 words. Figure IV-3 shows the general configuration of an SSB TCF Data Block.

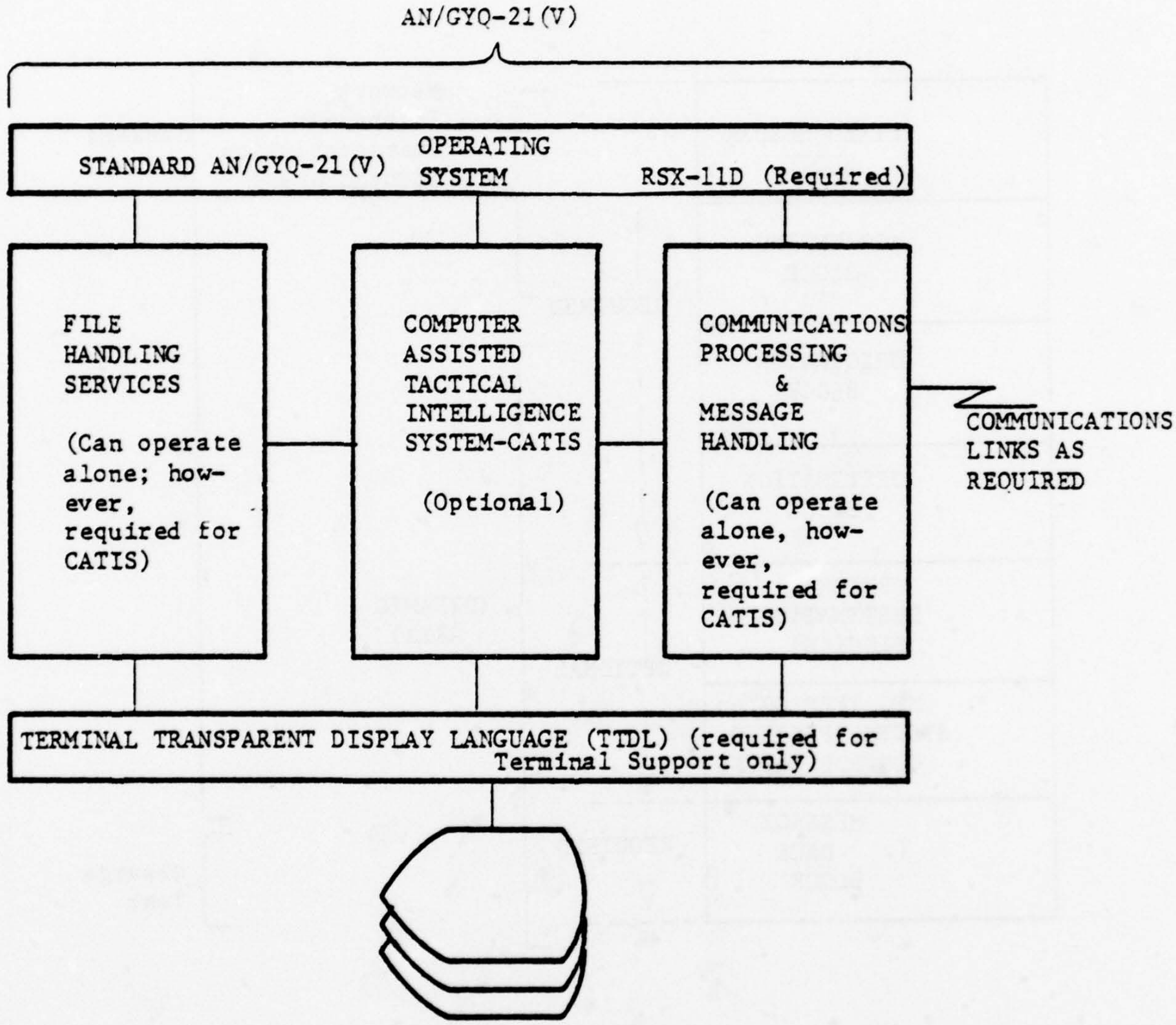


Figure IV-2. SSB Subsystem Relationships

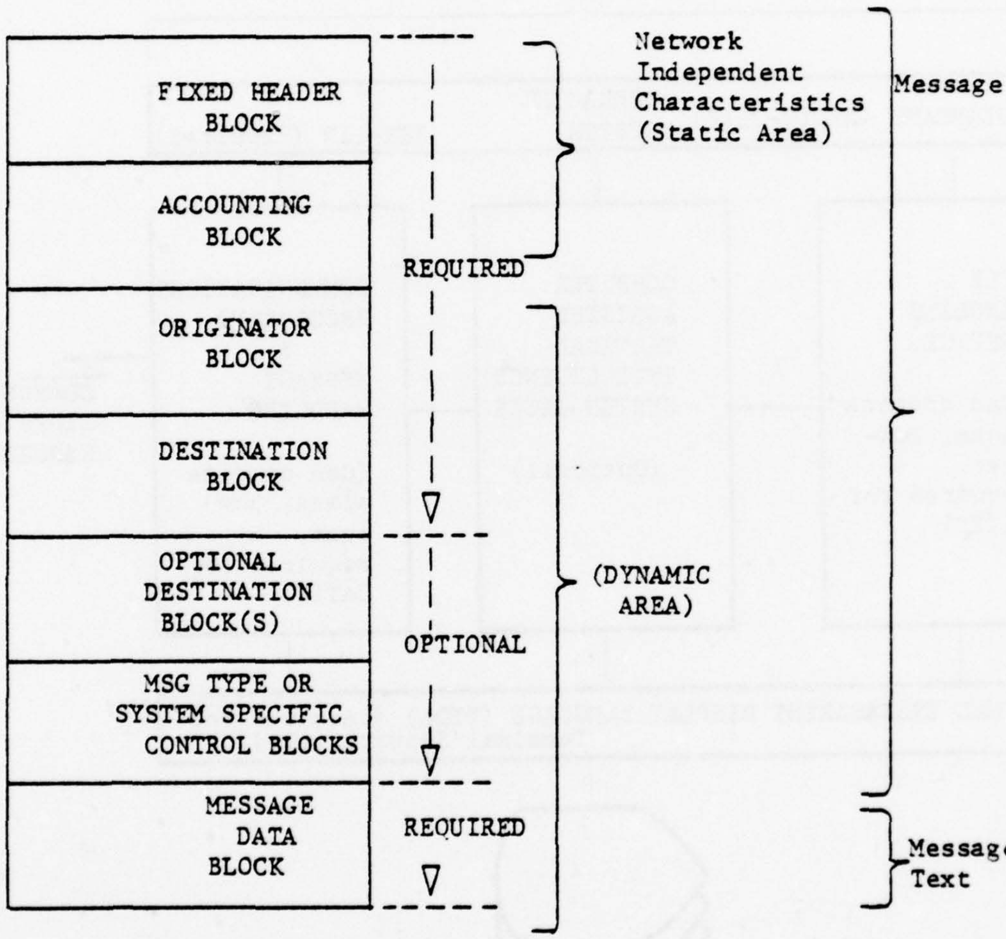


Figure IV-3. TCF Data Block

(1) Fixed Header Block. The first block for each TCF message is the Fixed Header Block. The first word in this block contains block type and message data length. The remaining words in the block contain message control data. Unused portions of the block are zero-filled. Figure IV-4 explains the structure and content of Fixed Header Blocks.

(2) Accounting Block. The first word in this block identifies its type and length. The remaining two words contain information used by the Accountability Module, TISAFM. These data include the time the message was processed by the Message Distribution Module, MDM, and the accounting function flagword used by TISAFM. Figure IV-5 illustrates the structure and content of an Accounting Block.

(3) Routing Block. The Routing Blocks is the key to the entire SSB system. Two types of Routing Blocks are used: one identifies the originator of a message, the other identifies the destination of a message. The Routing Blocks are accessed by the Gateway Manager modules to process all message traffic in the system. Figure IV-6 illustrates the structure and content of the Routing Block.

(4) Message Data Block. The message data block contains the text portion of the TCF message. It is the last block of a message and is always present. If no data is to be sent in a message, a null message data block will be used. Figure IV-7 illustrates a Message Data Block.

(5) Null Message Data Block. The null message data block contains only a 2-byte block header and the message data count in the fixed header contains zero. This block is used to signal the end of TCF messages in which no data is to be sent.

All text blocks consist of serial data strings separated by record markers and potentially marked as to the nature of the text by optional markers. The entire 256-character ASCII set may be used. In order to allow these data, all separators and markers are evaluated as two character strings as follows:

Record separator - DLE,RS
Start of Network dependent text - DLE,SOH
Start of non-NDF text - DLE,STX
End of message text - DLE,ETX
Binary DLE value in text - DLE,DLE

Figure IV-8 illustrates a Null Message Data Block.

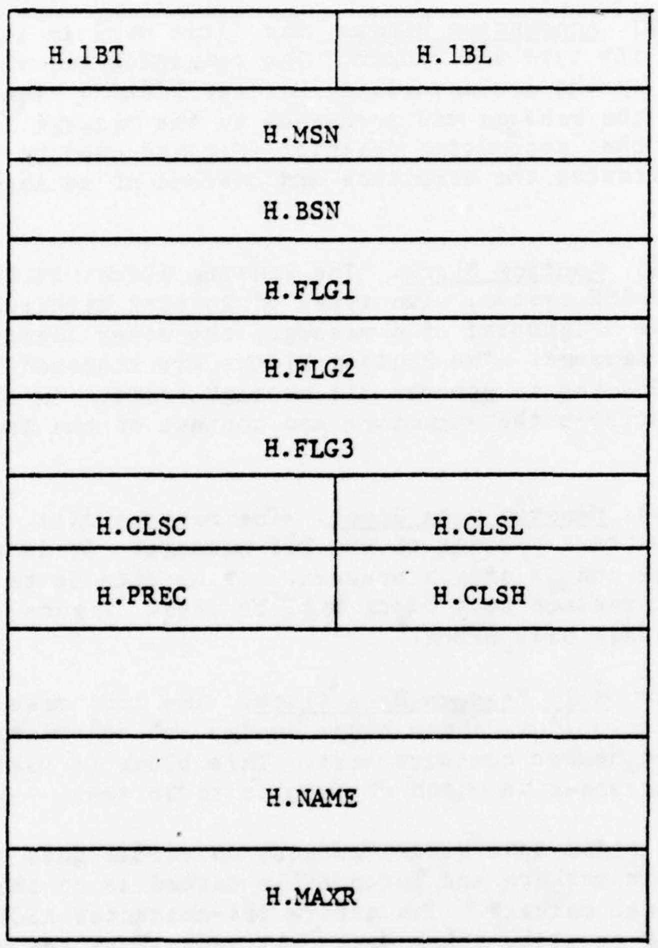


Figure IV-4. Fixed Header Block

DESCRIPTION OF STATIC AREA OFFSETS IN FIXED HEADER AND ACCOUNTING BLOCKS

OFFSET NAME	MEANING	TYPE
H.LBL	Fixed Header Message Block Length	Binary
H.LBT	Fixed Header Message Block Type	Binary
H.MSN	Message sequence number	Binary
H.BSN	Block sequence number	Binary
H.FLG1	Message characteristics flagword	Bit mask

H.FLG1 Bit Definitions

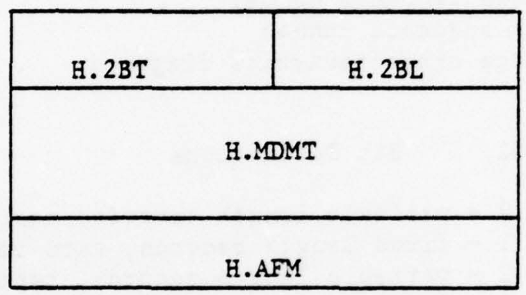
DT.NAR	Ø - variable length records, narrative (count 72)	
DT.CRD	1 - fixed length records, card image (count = 8Ø)	
DT.VAR	2 - variable length records, tape/disk (count > 72)	
DT.FIX	3 - fixed length records, tape/disk (count > 8Ø)	
DT.BIN	4 - data uses full byte binary code (not supported initially)	
	5-15 open	

H.FLG2	Message characteristics flagword	Bit mask
H.FLG3	Message characteristics flagword	Bit
H.CLS	Classification attributes	Mixed binary/bytes

H.CLSL - Classif. level, binary Ø-255
H.CLSC - Classif. compartments, Ø-255
H.CLSH - Classif. handling attributes Ø-255

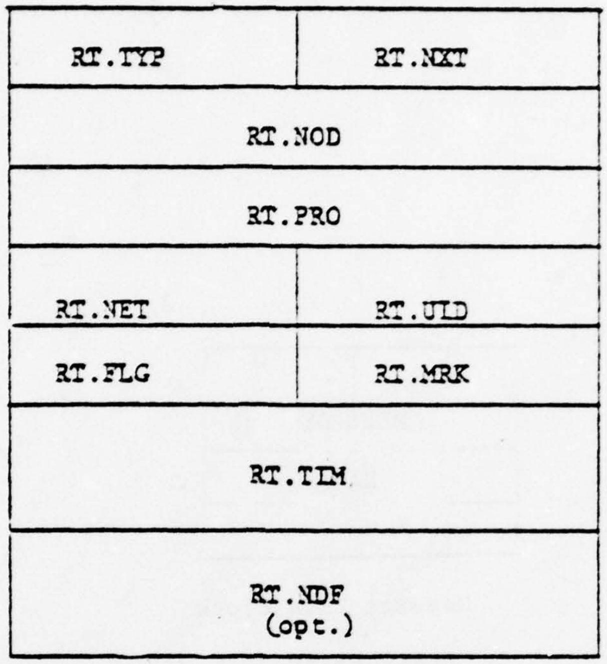
H.PREC	Precedence level, Ø-255	Binary
H.CHAR	Message data character count	Double word binary
H.NAME	Originator file name	RAD5Ø
H.MAXR	Maximum data record size	Binary value

Figure IV-4. Fixed Header Block (Cont.)



H.2BL	Accounting message block length	
H.2BT	Accounting message block type	
H.MDMT	Time message was processed by TISMDM	MIXED binary
H.AFM	TISAFM coordination flagword	

Figure IV-5. Accounting Block



variable length RT.NDF is the offset of the last byte

RT.NXT	Routing Block Length, Words	Binary
RT.TYP	Routing Block Type	Binary
RT.NOD	Node Identifying value	Binary
RT.PRO	Process Identifying value	Binary
RT.UID	User Network Identifying value	Binary
RT.NET	Network Identifying value	Binary
RT.MRK	Value	Binary

RT.FLG — Routing entry flag byte, bit positions as follows:

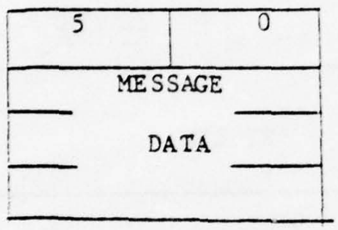
- RT.MDM =2ØØ= This entry processed by MDM for queuing purposes
- RT.TXG =1ØØ= This entry processed by indicated gateway successfully
- RT.TXE = 4Ø= This entry processed by indicated gateway unsuccessfully
- RT.LAS = 1= Last routine entry in message

Remaining four bits available for network dependent information

RT.TIM — Routing entry time stamp, 2 words mixed binary as provided by TISTIM library utilities

RT.NDF — Network dependent field as defined by the target network (optional).

Figure IV-6. Routing Block



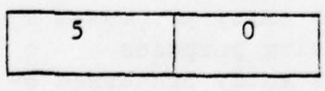
Message Data Block

Figure IV-7

NOTE:

(1) The length of the message data area is given in fixed header block.

(2) The message data area may contain any 8-bit character.

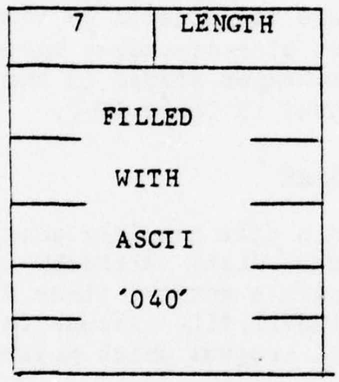


Null Message Data Block

Figure IV-8

Text will be packed (no short blocks except the last) to the length of the TCF disk blocking factor.

(6) Spacer Block. A spacer block is normally used to allow disk-resident messages to be formatted in a convenient manner. For example, the user may wish to place the message header information in one disk block and begin the message data block in a separate block. In this case, the spacer block would be used to pad the remainder of the header disk block. Subsequent references to the message would ignore the spacer block. (See Figure IV-9 below.)



Spacer Block

Figure IV-9

b. Internal Data Structures

The internal data structures of SSB are used to identify users to the system, identify internal and external routing targets, and store system information and error messages. These structures are created on disk as either MSNTOS or TISFIL files. The TISFIL files are read into memory dynamically on an "as needed" basis. The MSNTOS files are either used to maintain the system common area (TISCOM) or they are read into memory as needed.

(1) MSNTOS Files

MSNTOS files are created by the MSNTOS file handler. The MSNTOS file for TISCOM is used by TISSAV to store the updated SSB common area, TISCOM. The MSNTOS file of system error/status messages is a canned message file that is created at system generation time and thereafter accessed in read-only mode.

(a) TISCOM File

The TISCOM file is used as a storage area for the SSB system common area (TISCOM). It is core-resident and resides within the area known as BFRTSK. TISCOM is automatically overwritten every 30 seconds by the TISSAV program.

(b) System Status File (STSFIL)

The System Status File is a reserved area on disk used for storing all SSB error/status messages. Each message is assigned an ID and is referenced by the various SSB modules (by message ID) whenever an error or status report is detected during system operation. Each message is allotted up to 64 bytes of information. This file is not site-dependent but may be revised by the system programmer. The messages stored in the file and the purpose they serve are described in Table IV-2.

(2) TISFIL Files

TISFIL is a file handling program which maintains a set of SSB files on the system disk. Although each file is discussed as if it were a separate entity, these files are in reality subsets of one larger file, TISFIL.FIL. Access to the TISFIL files is gained only through the TISFIL program which maintains a complete description of each file.

(a) Precedence/Security Table (PSTFIL)

The Precedence/Security Table is a disk-resident file that contains all of the security terms used by the system for data coding and display purposes. It is a random structured file created during system generation. It is logically subdivided into four major areas:

- Precedence
- Security classification level
- Security classification handling
- Security classification compartmentalization.

This file structure is necessary because all options in these four areas are carried internally in the system as binary values which must be encoded/decoded when displayed to a user. All disk blocks in this file contain a four-word header block which identifies the starting block address of each of the four areas. The format of the PSTFIL is shown in Figure IV-10.

General error messages used by any module and displayed at the appropriate I/O device for the information of programmers:

```
STMSG  1,<ICM DIRECTIVE FAILURE LOC=$        VIRTUAL Ø=$  >
STMSG  2,<RSX-11D DIRECTIVE FAILURE LOC=$  VIRTUAL Ø=$  >
STMSG  3,<DISK I-O FAILURE--FILE PROCESSOR %  %          >
STMSG  4,<NON SPECIFIC DATA EXCEPTION  LOC=$  IO=$        >
STMSG  5,<DEVICE I-O FAILURE DEVICE ="  UNIT=&           >
STMSG  6,<BUFFER ALLOCATION FAILED  SIZE =$              >
STMSG  7,<TISS--SYSTEM INITIALIZED TISCOM VA=$           >
```

Messages for gateway modules. They are included for

the token set of messages encountering processing problems:

```
STMSG  11, < GATEWAY RECOURCE FAILURE ID=$                    >
STMSG  12, < FATAL GATEWAY DATA ILLOGICAL  ID=$               >
STMSG  15, < AUTODIN ASC REJECT (RM)   CNS=$                  >
STMSG  17, < AUTODIN MESSAGE NOT ACKNOWLEDGED (NAK)  CSN=$ >
```

Status Indicator messages used within the Terminal

Gateway system and intended for user information:

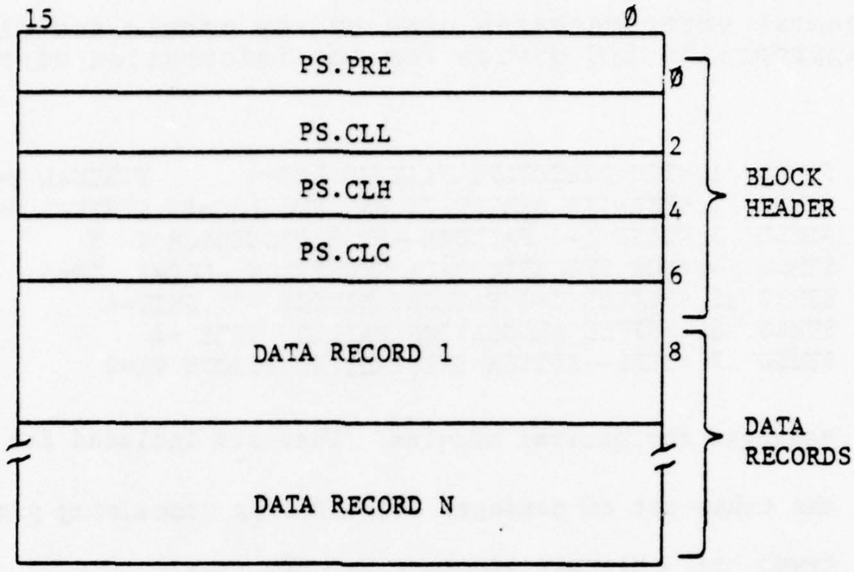
```
STMSG  2Ø, < MESSAGE PLACED ON INPUT QUEUE, MSN=$           >
STMSG  21, < MESSAGE DELETED FROM INPUT QUEUE,MSN=$         >
STMSG  22, < MESSAGE PLACED ON OUTPUT QUEUE,      MSN=$     >
STMSG  23, < MESSAGE DELETED FROM OUTPUT QUEUE, MSN=$       >
STMSG  25, < MESSAGE DELETED FROM WORK FILE QUEUE, MSN=& >
STMSG  26, < STATUS MESSAGE QUEUE EMPTY                      >
STMSG  31, < MESSAGE RETURNED, MSN=&                         >
STMSG  32, < MESSAGE RETURNED FOR UID=&   , MSN=&           >
STMSG  33, < INPUT QUEUE FULL                                >
STMSG  35, < UID=&   NOT REGISTERED IN USER DIRECTORY        >
```

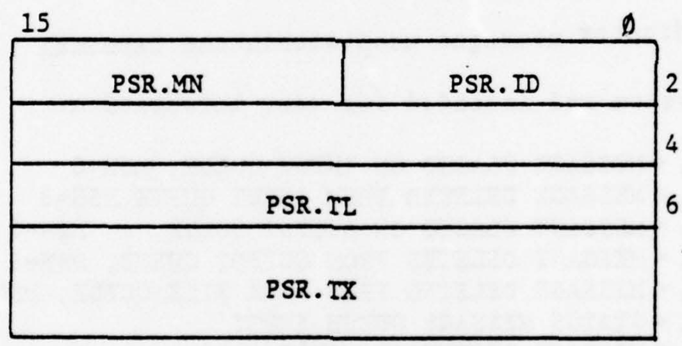
Messages used by the HISTORY/RETRIEVE module for

both operator and user information:

```
STMSG  40, < HISTORY - STATUS=$  MSN=$                       >
STMSG  42, < MOUNT TAPE CONTAINING DATE=                     >
STMSG  43, < INCORRECT TAPE MOUNTED                          >
STMSG  45, < OPERATOR INTERVENTION NEEDED, ID=$              >
```

Table IV-2. System Error/Status Messages





Data Record Layout

HEADER OFFSETS:

PS.PRE - Precedence Starting Block Number
PS.CLL - Classification Level Starting Block Number
PS.CLH - Classification Handling Starting Block Number
PS.CLC - Classification Compartments Starting Block Number

Figure IV-10. Precedence/Security Table (PSTFIL)

DATA RECORD OFFSETS:

PS.REC - Data Record Starting Point

PSR.ID - Record Identifier Value (Binary)

PSR.MN - Abbreviated Entry Mnemonics (3 ASCII characters)

PSR.TL - Message Text Byte Count

PSR.TX - Starting Point for Message Text (up to 30_8 bytes)

PS.MXR - Maximum Number of Data Records per block (16)

PSR.LN - Maximum Size of Text Data in Bytes (30_8).

Figure IV-10. Precedence/Security Table (Cont.)

(b) Routing Information File (RIFFIL)

The Routing Information File is a disk-resident random access file created by TISGEN during system generation. This file contains the information required to format routing blocks for message transmission, and to decode routing blocks for display at an I/O device. The file is used in conjunction with the Gateway Options Table File (GOTFIL), and is accessed in read-only mode by the Send, Display, and Print programs. The file is comprised of 256-word blocks, each containing 16 records, or entries. Each entry contains the name of either an SSB message destination or an originator, a routing token, and an extra word. The structure of the RIFFIL and its entries are illustrated in Figures IV-11 and IV-12.

A user may wish to send a message to an addressee not contained in the RIFFIL who nevertheless is at a node serviced by a net- work (e.g., AUTODIN).

In this case, he may interactively specify the network as his addressee. The SEND program will search the RIFFIL and find the network name (e.g., AUTODIN), and use the mask byte to further request entries to complete a route. In the AUTODIN case, OT.BTO specifies that the 'Routing Indicator' and 'To Line' must be entered by the user.

It is possible to have two or more entries identical in every respect except for name in order to include shorter mnemonics at SEND time. The first entry, in each case, will be the one displayed or printed, and should be the fully-descriptive name.

An entry named LOCAL (or anything unique that a site wishes) should be the first RIFFIL entry. It will be used to reach local addressees when they are not fully or correctly specified in a message.

(c) Gateway Options Table File (GOTFIL)

Certain external networks (AUTODIN for example) require additional information for routing traffic to specific network locations. This network-dependent information is not provided in the Routing Information File; it is provided by entries in the Gateway Options Table File. The size of the GOTFIL depends on the number of gateways included in the system and the number of entries (options) required for each gateway. Each gateway is permitted a variable number of entries. Each entry consists of three words plus a variable-length field. The size of GOTFIL is calculated by adding together the size of each gateway entry plus two words per gateway. As an example, AUTODIN, the only gateway in the current version of SSB requiring network-dependent information, occupies slightly more than

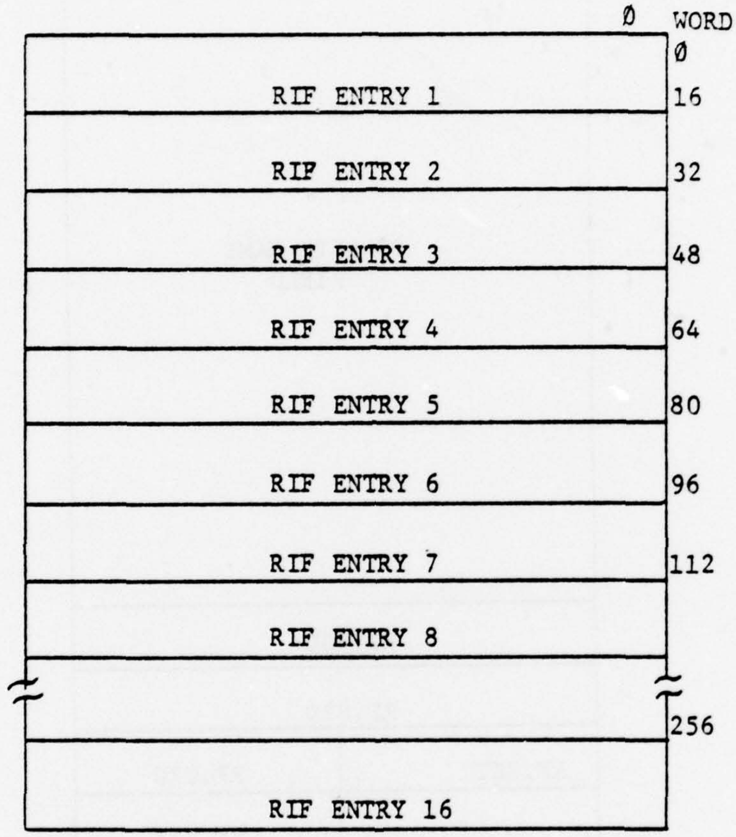


Figure IV-11. Routing Information File
RIF BLOCK (512 Byte)

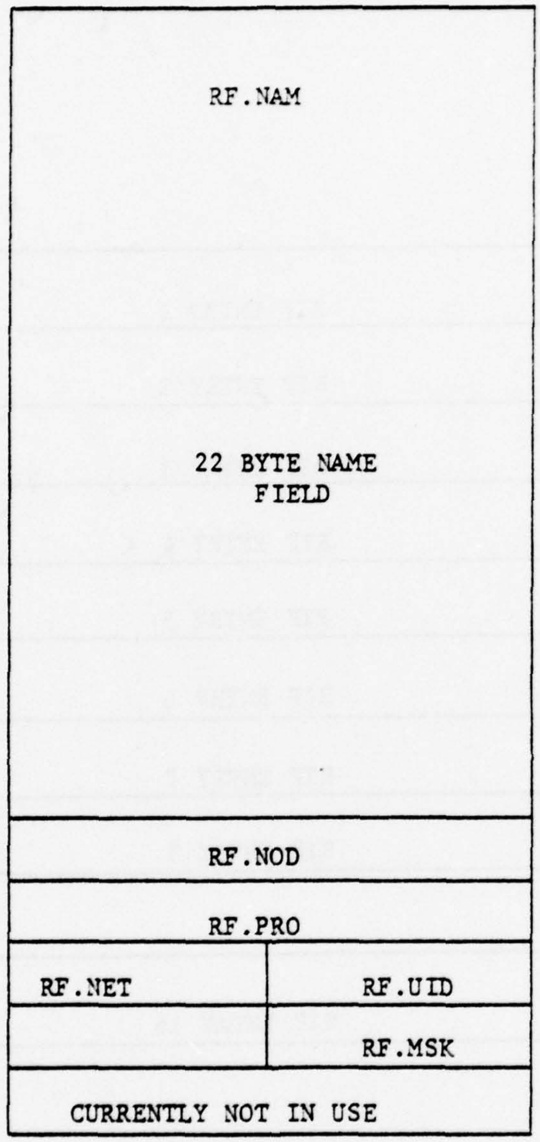


Figure IV-12. Routing Information File Entry

Description of RIFFIL Entries:

RIF Record Offsets

RF.NAM	-	22 byte ASCII name of address (destination of originator).
RF.NOD	-	16 bit binary value for node upon which addressee is located.
RF.PRO	-	16 bit binary processing value when level of address specifiction is applicable.
RF.UID	-	8 bit binary user identification value of addresse. This value must correspond to UIDs in the appropriate node's user file.
RF.NET	-	8 bit binary value for network through which addressee is reached.
RF.MSK	-	8 bit binary mask which defines processing steps required to fully specify a route. For a particular network, it specifies which GOTFIL options must be completed. In a fully specified routing entry, the mask byte should have all bits cleared. Otherwise, the network gateway will define with bit bits to set. For the AUTODIN network, the present design calls for setting bit 0 (i.e., OT.BTO) whenever a node is not fully qualified. It is necessary that such an entry exists for each gateway.

Figure IV-12. Routing Information File Entry (Cont'd) one 256-word block. The structure of the GOTFIL is illustrated in Figure IV-13.

(d) Terminal Directory File (TEDFIL)

The Terminal Directory is a disk-resident file containing the information required for each terminal defined in the system. It provides the device name, unit number, and the security classification level assigned to each terminal. The security levels defined for the TEDFIL must correspond to the levels described in the Precedence/Security Table. If these levels are in disagreement, the system will not deliver all the messages that may be authorize for the terminal. TEDFIL is a type 1 random file allocated and initialized by TISGEN. The size of the Terminal Directory is dependent upon the number of terminal entries and the accumulated sizes of all entries. Each terminal defined in the system occupies one entry in the TEDFIL. Generally, no entry is to cross a disk block boundary, and ech new entry will start at the beginning of a block. Since the directory is searched sequentially, each entry contains an offset (relative to the start of the disk block) to the next entry. If the offset is zero (TD.EOB), it indicates that this is the last entry in this block and the next terminal entry starts at the beginning of the next block. If the offset is a minus one (TD.EOF), it signifies the end of the terminal directory. The format of the Terminal Directory is shown in Figure IV-14.

(e) External Routing Indicator File (ERIFIL

The ERIFIL is a disk resident random file created by TISGEN. It contains information used by the AUTODIN gateway. It is used by AUTODIN to correlate a TCF routing token with AUTODIN's 'Routing Indicator' and 'To Line' indicator. It is a variable size file dependent on the number of entries and size of each entry. Each entry is 15 bytes plus the number of characters in the 'To Line' indicator, up to a maximum of 28 bytes. A rough count of 40 entries per disk may be used for approximate sizing.

The ERIFIL is used in read-only mode by the AUTODIN gateway exclusively. The TCF routing token must match the equivalent token in the RIFFIL.

WORD			BYTE
0	OT.INC	OT.OPT	0
1	OT.FB1	OT.DSZ	2
2	OT.FB3	OT.FB2	4
	CHARACTER-BY-CHARACTER PROMPT		

OT.OPT	Option Number	A binary value from 1 to 255 specifically ordered for each gateway. These options will be displayed in numerical order from lowest to highest. Thus, it is up to the gateway writer to arrange the displays in an appropriate order for the logical use by the gateway user. Since displays need not be numbered sequentially, space should be left for the insertion of anticipated displays.
OT.INC	Maximum Number of Characers to be Input	A binary value indicating the maximum number of characters to be read in as input. Both the Data type (e.g., numeric, alpha, etc.) and whether of not the input field must be completely filled may be specified by the contents of OT.FB1.
OT.DSZ	Display Size	A binary value indicating the total number of fixed characters in the display field. This number does not include the OT.INC value. Thus, to obtain the total number of characters in the display field = (OT.DSZ)+(OT.INC). Similarly, the size in bytes of the entry in the options table = (OT.DSZ)+ 6.(+1 if the value is odd).
OT.BF1	Input Characteristics Flag	A flag byte, describing the data type and required length of the input, to be used by "SEND" for a preliminary check of input data syntax.

Figure IV-13. Gateway Options Table File (GOTFIL)

bits 0-3	Not used.
	Required length of input field: if=0 OT.INC is the maximum number of characters accepted as input. Less will also be accepted. if=1 OT.INC contains the exact number of characters which must be input.
bits 5-7	Data type of characters to be input, and an option not to display the prompt at present if all bits are not set.
bit 5	If set, indicates acceptance of alpha characters.
bit 6	If set, indicates acceptance of numeric characters.
bit 7	If set, indicates the acceptance of special characters. Thus, if only alphanumeric characters are to be accepted, bits 5 and 6 should be set (e.g., 5-7 = 011_2).
OT.FB2	If the value is greater than zero, entry of this field is mandatory. If equal to zero, entry of the field is not mandatory.
OT.FB3	This flag byte is used to determine if any additional fields in the GOTFIL are mandatory, based on fields already entered. Accordingly, for each option or field entered, this byte, (OT:FB3), associated with that option, is 'OR'ed with the corresponding flag bytes for all of the other options that have been entered. The resultant bit pattern is then compared with OT.FB2 for all the

Figure IV-13. Gateway Options Table File (GOTFIL) (Cont'd) options in the GOTFIL. If matching bits are set in both flag bytes for any option, then that option is also prompted for, as if it had originally been a mandatory option.

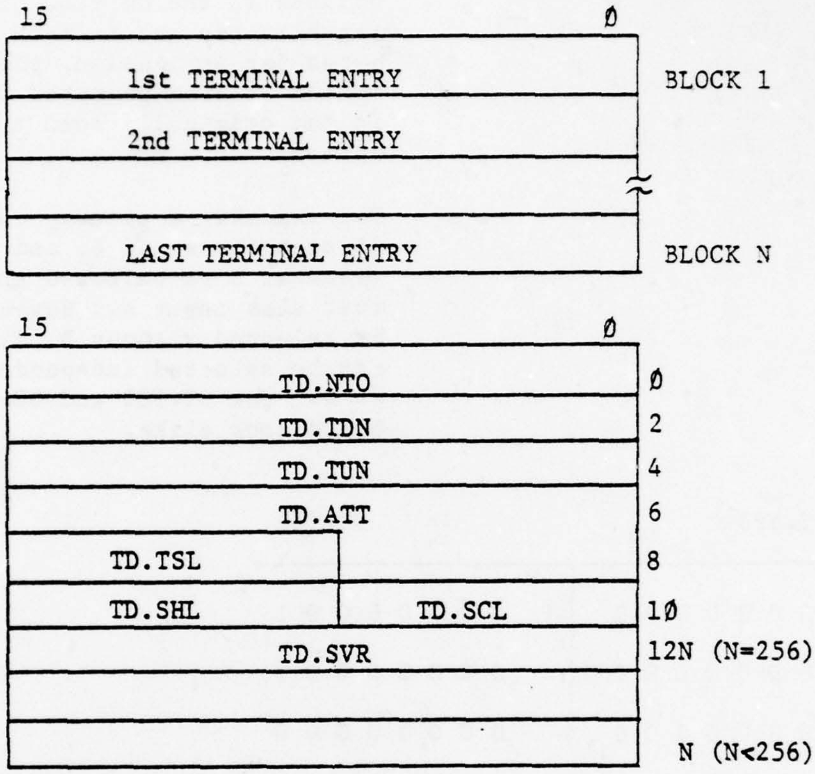
For example, a gateway user has three options, A, B, and C. Whenever B is selected the user must also input A. However, A can be selected without B. Similarly C can be selected independent of A or B. The OT.FB2 and OT.FB3 could look alike.

	OT.FB3	OT.FB2
for A	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1
for B	0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 1
for C	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0

The OT.FB3 is 'OR'ed for all options for which valid input was obtained. Then OT.FB2 is checked to be sure that all options containing a corresponding set bit contained some valid input.

Bit 7 in OT.FB2 is reserved for all input fields required if an addressee is not located in the directory.

Figure IV-13. Gateway Options Table File (GOTFIL) (Cont'd)



TD.NTO - Nest Terminal Offset

TD.TDN - Terminal Device Name

TD.TUN - Terminal Unit Number

TD.ATT - Terminal Attributes

TD.TSL - Terminal Security Level

TD.SCL - Security Compartment Level

TD.SHL - Security Handling Level

TD.SVR - Start of Variable Region

Figure IV-14. Terminal Directory File

The following information comprises a routing entry:

Routing Token	o Node		
	o Network	see RIFFIL for description	
	o Process		
	o UID		

- o Routing Indicator - as the AUTODIN protocol requires for the address specified by ROUTING TOKEN above.
- o To Line - as the AUTODIN protocol requires for the address specified by ROUTING TOKEN above.

(f) <u>AUTODIN Format Line File (AFLFIL)</u>

The AFLFIL is a risk resident random file created by TISGEN. It contains information used by the AUTODIN gateway. It is used by AUTODIN to correlate its protocol with TCF protocol. It correlates security classification levels, compartments and handling codes, and content indicator codes. AFLFIL is composed of a directory and seven subfiles (tables).

Generally, the directory occupies 28 words of each disk block. The remaining block space is assigned to each subfile in a contiguous manner. The first subfile is the AUTODIN Language Media Format (or LMF) table, occupying three words. The second is the security table, two words per entry. Each entry must match the PSTFIL classification level corresponding entry. The third subfile is the Precedence Table, two words per entry, each entry matched to the PSTFIL Precedence Table. The fourth subfile is the Op-signal code. the fifth subfile, the Transmission Control Code (TCC) table, assigns a TCC to a pair fo compartment-handling codes. It is three words per entry. The sixth subfile is the Content Indicator Code (CIC) table, four words per entry, assigning a CIC code to each ROUTING TOKEN. The seventh, and last, subfile is the SVC table, which includes a routing token plus two words per entry, used for default tokens such as for service messages. A simulated (as the actual codes are classified) AFLFIL file at INCO took less than one disk block. The AFLFIL is used in read-only mode by the AUTODIN gateway exclusively.

The following information is needed to specify an entry for each subfile in AFLFIL:

1 LMF TABLE

No information necessary

2 SECURITY TABLE

a Binary value: as specified in the PSTFIL.

b ASCII value: as AUTODIN protocol required for the corresponding classification level that matches the Binary Value in the PSTFIL.

3 PRECEDENCE TABLE

a Binary value: as specified in the PSTFIL.

b Opsignal match: a binary number matching the corresponding opsignal match in the OP-SIG TABLE next.

c Code: one letter as required by AUTODIN protocol for precedence and matching the PSTFIL for the Binary value.

4 OP-SIG TABLE

a Opsignal match: to match the corresponding opsignal match in Precedence table above.

b ASCII value: 3 characters as required for the OP-SIGNAL by the AUTODIN protocol to match corresponding values in the Precedence table above and the PSTFIL.

5 TCC TABLE

a Classification compartment and handling codes: Binary values as specified in the PSTFIL.

b TCC code: 3-character code as specified by AUTODIN protocol to match the PSTFIL for the binary values.

6 CIC TABLE

a ROUTING TOKEN: as specified in the RIFFIL.

b CIC code: One letter to specify destination. Out of the 4 CIC AUTODIN characters needed, SSB has been given the freedom of choosing 2. SSB will use 1 to specify originator-UID and 1 for destination-UID. Thus, each routing token should be assigned an alphabetic character.

7 SVC (also DFT) TABLE

a ROUTING TOKEN: as specified in the RIFFIL.

b Token ID:

ID 1 - corresponds to a default destination token.

ID 2 - corresponds to a default service destination token.

ID 3 - corresponds to TISPGT printing destination token (presently UID=1).

(g) TCF Message File (MSGFIL)

The TCF Message File is a reserved area on disk used for temporary storage of users messages, requests or inquiries. Terminal user request messages are variable in length and will require at least two disk blocks per message. The first disk block contains the message header information while subsequent blocks contain the actual message data. Each data block can contain up to 512. bytes. The size of this file is site-dependent dictated by the traffic load and long-term storage requirements.

(h) Routing Value Assignment File (TISROU)

The Routing Value Assignment module (TISROU) is an equate object file used to describe network ID values and symbolics plus the current nodal ID values and their symbolic labels.

(i) Message Retrieval Argument File (ARGFIL)

The Message Retrieval Argument File is a type 1 TISFIL record which occupies one block of disk space. ARGFIL is used for temporary storage of the retrieval request arguments/qualifiers for the currently active message retrieval from tape. Only one retrieval request may involve the tape at any given time. When additional tape retrieval requests are issued, the query processor will return the request block to the operator indicating that tape retrieval is currently in progress.

(j) TISCOM Table

The TISCOM Table is a series of words identifying the starting addresses of all the tables residing in the TISCOM area of BFRTSK. This table is the first table defined in TISCOM. It also contains the local node ID (site-dependent) and size of the TISCOM area.

The TISCOM Table is defined by a series of card inputs which identify all the table offsets plus the node ID. The node ID card is site-dependent and therefore must be checked to ensure that it agrees with the Node ID being SYSGENd. The format of the TISCOM Table is illustrated in Figure IV-15.

(k) Route Reference Table (RRT)

The Route Reference Table (RRT) is the primary routing table used by the Message Distribution Module (TISMDM) to evaluate inter-system messages. Each entry in this table is three words long and contains information such as the primary and alternate network numbers to be used in routing messages to the node identified in word 1 of the entry. There is one entry in this table for every node defined in the system. The format of the Route Reference Table is shown in Figure IV-16.

15 ... 0	
NODE ID	FIXED HEADER
SIZE OF TISCOM	
ROUTE REFERENCE TABLE	MAJOR TABLE AREA OFFSETS
NETWORK CHARACTERISTICS TABLE	
TISJOR AREA OFFSET	
TISMDM NODE POOL	
NETWORK CORE COMMUNICATION AREA	
SPARE	
REVIEW JOURNAL AREA	

Below is a list of currently acceptable nodes:

Label	Binary Value	Site Name
NOD.IN ==	1	;INCO DEVELOPMENTAL CENTER
NOD.AF ==	2	;AFIS-INDOC (SSB) CENTER
NOD.DI ==	3	;DIA ARL. HALL #1
NOD.SA ==	4	;SAC (OFFUTT) DSSCS SYSTEM
NOD.SC ==	5	;SAC (OFFUTT) I&W DEVELOPMENTAL SYSTEM
NOD.CO ==	6	;CONAD
NOD.SH ==	7	;CINCUSAFE/ACDI (VAN--SCHIERSTEIN)
NOD.US ==	10	;CINCUSAFE/ACD(USAFE I&W #2)
NOD.UA ==	11	;CINCUSAFE AFSSO (INX & 497 RTG/INT)
NOD.EC ==	12	;CINCUSEUCOM/ECADP=)
NOD.RA ==	13	;RADC-IRDA
NOD.TA ==	14	;TAC/IN
NOD.TI ==	15	;9TIS/INT (HAMPTON VA)
NOD.PA ==	16	;CINCPACP/IN (HICKAM AFB)
NOD.CN ==	17	;CINCNAVEUR/N=2 (LONDON)
NOD.CL ==	20	;CINCLANTCOM/J-2 (NORFOLK VA)
NOD.FT ==	21	;FTD/XXX
NOD.EA ==	22	;CINCUSAEUR/IDHS (HEIDELBERG)
NOD.HN ==	23	;HANCOCK ASC
NOD.SC ==	24	;AFSC/ANDREWS
NOD.PC ==	25	;CINCPACOM/J=53
NOD.NS ==	26	;NSA FT. MEADE
NOD.CS ==	27	;COINS SWITCH,AHS

Figure IV-15. TISCOM Table Directory

NODE ID		Ø
FLAG	NETWORK ID	2
ALT. NET NUMBER	PRI NET NUMBER	4

RR.NOD	Node ID Number	Node ID value, binary value of Ø - $2^{16}-1$.
RR.NET	Network ID	Byte binary value identifying a given network. This value is used in accessing the NCT table.
RR.FLG	Flag Byte	Indicates the current status to routing nodes referenced by TISMDM. Values: Bit Ø (RR.ACT) - Route/Reference entry is active Bit 1-2 - Not used Bit 3 (RR.ALT) - Alternate entry is being used. When bit is set use RR.ANN instead of PR.PNN Bits 4-7 - Not used.
RR.PPN	Pri Net Number	Contains the primary network number to be used in routing to this node.
RR.ANN	ALT. Net Number	Contains the alternate network number to be used when alternate routing to this node.

Figure IV-16. Route Reference Table (RRT)

(l) Network Characteristics Table (NCT)

The Network Characteristics Table (NCT) provides the basic correspondence of network to actual system software. Also provided is any network to device correspondence used by the network software or any additionally common service such as TISIBM and TISLPM. The NCT table resides in the TISCOM area and is periodically written to disk for restart purposes. The table format is shown in Figure IV-17.

(m) TISJOR Area

The TISJOR area of TISCOM contains information which describes the contents of the current and previous message abstract files. The TISJOR area also holds abstract file saturation information, which is used by the SSB journalization program (TISJOR) to determine when to notify the operator that the message abstract file should be journalized to type. The format of the TISJOR area is illustrated in Figure IV-18.

2. SSB FEATURES

Release III of SSB provides users with the following functions to process message traffic:

a. Message Preparation

SSB provides a standard format to build messages and prepare them for subsequent transmission to any network supported by SSB without regard for the protocol of the destination network. Message text may be input from the terminal keyboard or from any other I/O device such as disk, magnetic or paper tape, or a card reader.

b. Message Transmission

A standard format is provided which permits the transmission of message traffic to any network supported by SSB with minimum user knowledge of the protocol requirements of the destination network. Supplemental formats are automatically output to the user's terminal whenever networks requiring additional routing information are included as destinations in a message. Message pointers may be deleted from the user's output queue upon transmission.

c. Message Receipt

All messages entering the SSB system are automatically converted into a standard format without regard for the protocol requirements of the originating network. Messages may be displayed at

		Offset
DEVICE NAME		Ø
UNIT NUMBER		2
FLAG	NID	4
TKI		6
		8.
TKO		10.
		12.
SIZE		14.
# READS	TERM CHAR.	16.

NC.DEV	Device Name	Two-character ASCII value specifying device name of the physical device, if any, associated with the network.
NC.UNT	Unit Number	Used in conjunction with NC.DEV to complete physical device specification.
NC.NID	Network ID	Binary byte value corresponding to value defined in Route Reference table (RRT).
NC.FLG	Flag Byte	Bit Ø - Device specified is to read by TISIBM Bit 1 - Network read/write data is handled by TISLPM Bit 2 - When set core communication area exists for this network Bit 3 - When set core indicates system has been restarted Bit 4-6 Unused Bit 7 - Device is currently in use when set.

Figure IV-17. Network Characteristics Table (NCT)

NC.TKI	Input Task	RAD5Ø task file name of task which handles network input data.
NC.TKO	Output Task Name	RAD5Ø task file name of task designated to handle outgoing message data SRB's.
NC.SZE	Buffer Size	This word contains the maximum buffer size in words required by the device.
NC.TER	Terminal Char.	Contains the terminal character recognized by the device.
NC.RED	# Reads	Maximum number of reads to hang on a line.

Figure IV-17. Network Characteristics Table (Cont.)

	OFFSET
MSN OF PREVIOUS ABSTRACT	JOR.PA
MSN OF CURRENT ABSTRACT	JOR.CA
LOWEST MESSAGE MSN IN ABSTRACT	JOR.LM
HIGHEST MESSAGE MSN IN ABSTRACT	JOR.HM
TOTAL MESSAGES ABSTRACTED	JOR.TM
FLAGS	JOR.FG
SATURATION POINT IN FILE DATASET	JOR.SP
CURRENT POSITION IN FILE DATASET	JOR.CP
# OF BLOCKS IN DATASET SAFETY MARGIN	JOR.TB
# OF ENTRIES IN DIRECTORY SAFETY MARGIN	JOR.DB
CURRENT POSITION IN DIRECTORY	JOR.DP
SATURATION POINT IN DIRECTORY	JOR.DS

	JOR.LT
LOW MDM TIME IN ABSTRACT	
	JOR.HT
HIGH MDM TIME IN ABSTRACT	
	JOR.HT

TAPE BUSY FLAG	JOR.TP
UID FOR JOURNAL REVIEW	JOR.UI
OPEN TIME	JOR.OT
CLOSE TIME	JOR.CT

Figure IV-18. TISJOR Area

TISJOR Area Offsets:

JOR.PA	MSN of previous abstract	Word containing the Message Sequence Number (MSN) of the previous abstract file.
JOR.CA	MSN of current abstract	Word containing the MSN of the current abstract file.
JOR.LM	Low MSN	Word containing the lowest MSN in the current abstract file.
JOR.HM	High MSN	Word containing the highest MSN in the current abstract file.
JOR.TM	Total MSNs	Word containing the total number of MSNs in the current abstract file.
JOR.FG	Flag Word	Bit Ø (JOR.Ø2) Ø = journal program quiescent 1 = journal program (TISTAP) active Bit 1 (JOR.Ø3) Ø = TISTAP has run journalization in response to a request. 1 = TISTAP has not run journalization in response to a request. Bit 2 (JOR.Ø4) Ø = Message saturation point not exceeded. Bit 3 (JOR.Ø5) Ø = Directory saturation point not exceeded. 1 = Directory saturation point exceeded.

Figure IV-18. TISJOR Area (Cont.)

JOR.SP	Message Saturation Point	Word containing a number of blocks which can be used by messages. When this number is reached or exceeded, the message file is said to be saturated.
JOR.CP	Current Message Position	Word containing the total number of blocks which have been used by messages since the message file was last journalized.
JOR.TB	Message Safety Margin	Word containing the number of message blocks which can be used without checking for message file roll-over. If this value is exceeded, TISJOR must be called to check for file roll-over.
JOR.DB	Directory Safety Margin	Word containing the number of entries which can be made in the message directory without checking for directory roll-over. If this value is exceeded, TISJOR must be called to check for message directory roll-over.
JOR.DP	Current Directory Position	Word containing a pointer to the current entry the message directory.
JOR.DS	Directory Saturation Point	Word containing the total number of directory entries which can be made with impunity. When this number is reached or exceeded, the message file is said to be saturated.
JOR.LT	Low Time	Two word value holding the lowest time of a message on the current journal file.
JOR.HT	High Time	Two word value holding the highest time of a message on the current journal file.

Figure IV-18. TISJOR Area (Cont.)

JOR.TP	Tape Busy	Word indicating whether or not the journal file is currently being written to tape. Values: Ø = Journal tape is not being written. 1 = Journal tape is being written.
JOR.UI	UID	User identification (UID) of analyst requesting access to a journal tape. This is set by SBATAP and cleared by either SBAHRT or SBABTP.
JOR.OT	Open Time	Two word value showing the time the message abstract file was created.
JOR.CT	Close Time	Two word value showing the time the message abstract file was last updated.

Figure IV-18. TISJOR Area (Cont.)

a terminal or output to the line printer at the user's request. Options are provided to specify which portion of a message (header, text, network text) are to be displayed or printed. A system status feature provides the user with notification of new incoming messages addressed to him.

d. Message Review

All message traffic in the system may be reviewed in a common format regardless of the protocol of the originating network. Utility programs are available to list the contents of both the input and output queues. A status function is available to list the total number of messages contained on each user's queues. Privileged users (UID<11) may access the queues of any user and delete or transfer messages as required. (Display/Print is a primary post transmission error/status tool.)

e. Message Retention

All traffic entering or leaving the system is abstracted and journalized on tape for future retrieval and reference. Messages may be saved for subsequent transmission through the expedient of the user sending a message to himself to insure abstraction and journalization.

f. Message Retrieval

Messages stored on disk or journal tapes may be listed or retrieved at any time. The HISTORY function includes options which either create listings of messages meeting user-specified criteria, or restore user-specified messages to the user's output queue.

g. Editing Messages

All messages, irrespective of their point of origin, may be revised or edited. The message header may be retained or altered. Text may be altered, deleted from, or appended to any message. Options in the EDIT function permit single character, character string, and message record manipulation. Protocol requirements of destination or originating networks impose no restrictions or requirements upon the user. The original message remains intact; the edited message is assigned a new message sequence number.

3. SSB SUPPORT CAPABILITIES

In addition to the basic message processing functions, SSB Release III provides the following capabilities.

a. Remote job entry to the IBM 360 computer via the 2780 gateway.

b. Security monitoring of individual terminal and individual user classification categories.

c. System access restricted to previously authorized users and terminal devices.

d. Error detection and correction provisions to minimize traffic flow interruptions.

e. Automatic journalization of all message traffic to prevent potential loss of data when file saturation results in over-writing the file, and to provide site/system managers with an accurate record of throughput, and terminal and network utilization.

SECTION V

ADAPTABILITY OF SSB RELEASE III

Although the complete set of SSB baseline modules is available to the user, certain modules may be eliminated when not needed. For example, the DIAOLS gateway is valuable for analyst-to-DIAOLS data base communication within the system. If there is no need for analysts to communicate with DIAOLS, the gateway may be eliminated. As gateways are added to the SSB, only those needed needed at a specific site will be installed and run. If an application program is not of value at a particular site, it too may be eliminated.

A capability exists for a user to construct his own gateway for use with the SSB. All disk I/O is handled through global routines and with additions to SSB system tables these gateways may be used in conjunction with the SSB software.

Application programs specific to a site may be written using the SSB facilities such as disk I/O (TISFIL), external buffers (ICM and BFRTSK), TTDL and the ICM macros. The programs may be run in conjunction with SSB by inserting simple entries in the system tables. The SSB Programmer's Manual provides the information required to add applications programs to the SSB system.

Site-written application programs may be invoked from the central system to share data received over a communications line. The use of external buffers and general routing schemes inherent in the SSB design permit this capability.

With modifications to the AUTODIN gateway, the ability to handle multiple AUTODIN lines can be included.

The DL-11 and BR-1569 drivers are general purpose modules for other asynchronous communications interfaces.

It is possible to screen analysts and/or terminals based on data integrity criteria to allow only authorized personnel to access parts of the syste.

Other communication systems (i.e., IDHSC II) may be interfaced with SSB because of the common format used to store messages and the gateway oriented architecture of the system.

The use of TTDL allows the incorporation of different terminal device types without the necessity of modifying the SSB application programs.

SECTION VI

IMPLICATIONS FOR FUTURE DEVELOPMENT

SSB Release III is a stable software system of modular design which permits selection and use of site-specific capabilities only, or use of the entire system. SSB provies a wide range of networking and communications functions in a message-oriented system. These functions include message routing and accountability; journalization of incoming and outgoing traffic; a full complement of message handling functions for the intelligence analyst; and the ability, through the use of gateways, to access other networks and intelligence data handling systems independent of the data protocol requirements of such external systems.

While SSB is a stable system, it has not yet under Release III reached its full potential. A number of enhancements could be made to increase its flexibility, increase throughput times, and reduce system operating costs.

First, a need exists for interfaces to a number of networks or systems not now possessing gateways. These include IDHS(C)-II, the Defense Dissemination Program system and the ADCOM system. Further work also needs to be done on some existing gateways, including COINS, DIAOLS-TSS and the SSB gateways.

Second, the CATIS modifications which are currently unique to one version of SSB, should be merged with the basic SSB Release III system. This will lower software maintenance cost. In particular TTDL needs to be over-hauled, combining the generality and flexibility of TTDL I with the responsiveness of TTDL II.

Third, SSB installation procedures need to be simplified so that field sites will require less technical expertise to maintain their SSB installations.

Fourth, as the number of SSB installations grows, a more formal method of software configuration management control will become mandatory.

Since its inception in 1975, the Standard Software Base has been eclipsed by further reaching Air Force concepts of intelligence data handling system mangement. As it stands today, SSB is a part in a much larger data handling system now referred to as the Common User Baseline for the Intelligence Community, or CUBIC. CUBIC encompasses not only SSB, but other systems such as CATIS, (Computer Assisted Tactical Intelligence System), and SARP, (Storage and Retrieval Processor). CUBIC concepts, current capabilities, management philosophy, and indicated directions for the future development of SSB/CUBIC, are presented in the SSB Overview Document provided under separate cover.