MITRE-Bedford Division

MTR-4718

CR160107
JSC-14481
DEC 11 1978

JSC Interactive Basic Accounting System Design Options

(NASA-CR-160107) JSC INTERACTIVE BASIC ACCOUNTING SYSTEM (Mitre Corp., Houston, Tex.) 59 p HC A04/MF A01 CSCL 05A
N79-18800
Unclas
G3/81 14679

J. F. Spitzer

SEPTEMBER 1978

MITRE

Department Approval: ______________________

MITRE Project Approval: ______________________

ABSTRACT

An Interactive Travel System (ITS) was recently designed by the Institutional Data Systems Division (IDSD) of the Johnson Space Center and was placed in production in February 1978 on the UNIVAC 1108-9 system.

The design concept for an Interactive Basic Accounting System (IBAS) is currently being selected by IDSD; the design objectives for IBAS are similar to ITS. The objective of this task is to forecast the IBAS transaction response under a variety of design options, and to select the design option which provides the best response at the lowest cost. This will be accomplished by modelling the IBAS workload and applying this workload to a U1108 EXEC 8 based system using both a simulation model and the real system.

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

LIST OF ILLUSTRATIONS

L'ST OF TABLES

SECTION I
INTRODUCTION

1.1 Background

The Institutional Data Systems Division (IDSD) provides computer systems support to Johnson Space Center's administrative and management functions, including basic accounting. This support is currently provided by UNIVAC 1108 systems under both EXEC-2 and EXEC-8 operating systems.

An Interactive Travel System (ITS) was recently designed by IDSD and was placed in production in February 1978 on the UNIVAC 1108-9 system. The concept is based on the use of indexed sequential files, using COBOL and UNIVAC's Indexed Sequential Access Method [1]. On-line update and retrieval of data from a database of three basic files by multiple users, via "form-mode"[1] transactions, is handled concurrently. Actual access to the files is serial, with the database being locked until the user transaction is completed. This design is effective due to the small number of terminals (5), the small number of user transactions per week (1500), and design simplicity which implies small resource demands per user transaction.

The design concept for an Interactive Basic Accounting System (IBAS) is currently being selected by IDSD. The design objectives are similar to ITS: system simplicity and adequate response to the users. This leads to the question: Would the ITS design concept be adequate for IBAS? The modeling of the increased IBAS workload (14 terminals and 5000 to 6000 user transactions per week) and various design concepts is the subject of this task.

[1] The user "fills-in-the-blanks" of a template (form).

1.2 Objective

The objective of this task is to forecast the IBAS transaction response under a variety of design options, and to select the design option which provides the best response at the lowest cost. This will be accomplished by modelling the IBAS workload and applying this workload to a U1108 EXEC 8-based system using both a simulation model and the real system.

1.3 Approach

This task will be accomplished in four steps:

1) Develop a workload model using current ITS accounting data as a guide. To accomplish this, performance data will be collected from the current operation of the ITS, using data generated by an ITS statistics logging program.

2) Develop an ITS workload model to be run on the U1100 Series Simulation [2], and validate the workload model and the simulation against the observed data from step 1.

3) Modify the ITS workload model to reflect the anticipated IBAS workload, define several IBAS design cases, and run them on the U1100 Series Simulation. Select the best one or two candidate cases based on these runs.

4) Configure these "preferred" candidate cases for runs on the Performance Evaluation System [3] for final selection.

This report is organized in four major sections which parallel the above steps.

1.4 Results Summary

Both simulation and synthetic program experiments indicate that individual IBAS users on the U1108-9 can expect better than 10-second mean response if the following conditions are met:

- Seven or less concurrent IBAS users are signed-on,
- IBAS transaction programs do not exceed 45K words and one SUP-second,
- Transaction interarrivals at an IBAS terminal are in the 45 second range,
- Demand workload completion is limited to four or less test-and-set GIM/PMATS users,
- Background batch workload parallels the current SVDS/MPAD useage in terms of resource utilization.

When more than seven IBAS terminals are signed-on, IBAS response will degrade due to queuing on the serially reuseable ISAM database. If fourteen terminals are concurrently in use (the planned maximum number of installed IBAS terminals), individual IBAS mean user response will exceed 10 seconds. Since planned IBAS transaction rates do not require more than seven concurrently active terminals, and because ITS experience indicates that concurrently active terminal usage does not generally exceed two terminals (five ITS terminals are installed), it seems inadviseable to abandon the simplicity of the current IBAS design approach which utilizes a serially reuseable database for update transactions.

Study results also indicate

1) that the resource utilization characteristics of the batch background workload have a major impact on IBAS response,

2) that four GIM/PMATS test-and-set runs do not impact IBAS response, and

3) that IBAS transaction program segmentation does not improve IBAS response.

SECTION 2
THE CURRENT INTERACTIVE TRAVEL SYSTEM APPLICATION

This section will provide an overview of the current ITS Application (Section 2.1), briefly explain its program design (Section 2.2) and set forth statistics on its use gathered in the April-May 1978 time period (Section 2.3).

2.1 Overview

The current ITS application supports the JSC Financial Management Division (FMD) in managing funds related to JSC travel. Requirements definition, design and coding of the application have been the responsibility of the Data Systems Development Branch of IDSD.

Five Megadata terminals, one in Building 45, two in Building 416 and two in Building 1 are used by FMD personnel in order to access the ITS application program. The application program is resident on the UNIVAC 1108-9 system in Building 12 at JSC. FMD personnel interact with the ITS via displayed "terminal templates"; any of nine possible ITS transactions can be executed by "filling in the blanks" of the proper template. The nine ITS transactions are:

- ITVPWA - Travel PWA Inquiry
- PWAE/M - PWA Establishment/Modify
- ITMAST - Travel Master Inquiry
- TVLEST - Travel Establishment
- TAREST - Travel Accounts Receivable Establishment
- TD/LIQ - Travel Disbursement/Liquidation
- TVLADJ - Travel Adjustment
- TVCORR - Travel Correction of Accounts Data
- CTOTAL - Total of Account Balances

These transactions interact with a database consisting of three basic files:

- Primary Work Authorization (PWA) File – An indexed sequential file consisting of records of 54 characters, keyed on several concatenated fields [Method of Authority (MA), Program Year (PY), Fund Source (FS), Responsible Organization (RO) and Funding Object Class (FOC)], and containing issues and receipts of the account identified by the concatenated key field;

- Travel Master File – An indexed sequential file consisting of records of 210 characters, keyed on a trip identifier field, and containing voucher-oriented data such as trip start and end dates, advance balance, received balances, and a cross-reference field to the PWA record which will fund the trip.

- Edit Master File – An indexed sequential file containing records of 72 characters which define "legal" template field values via tables.

The database is managed via the UNIVAC Indexed Sequential Access Method (ISAM), and the ITS application is largely coded in ASCII COBOL [4]. The database is locked (i.e., restricted to serial use) during update transactions; other update transactions are queued for service during this lock interval. Retrieval transactions are processed concurrently.

A log file is also maintained for recovery and journalling purposes, and as a mechanism for controlling access to the database.

2.2 Program Design

The ITS application consists of an overlayed program, one copy per user, which accesses a common database. This structure is shown in Figure 2-1, and a brief description follows. For details, the interested reader should consult the ITS program documentation [5].

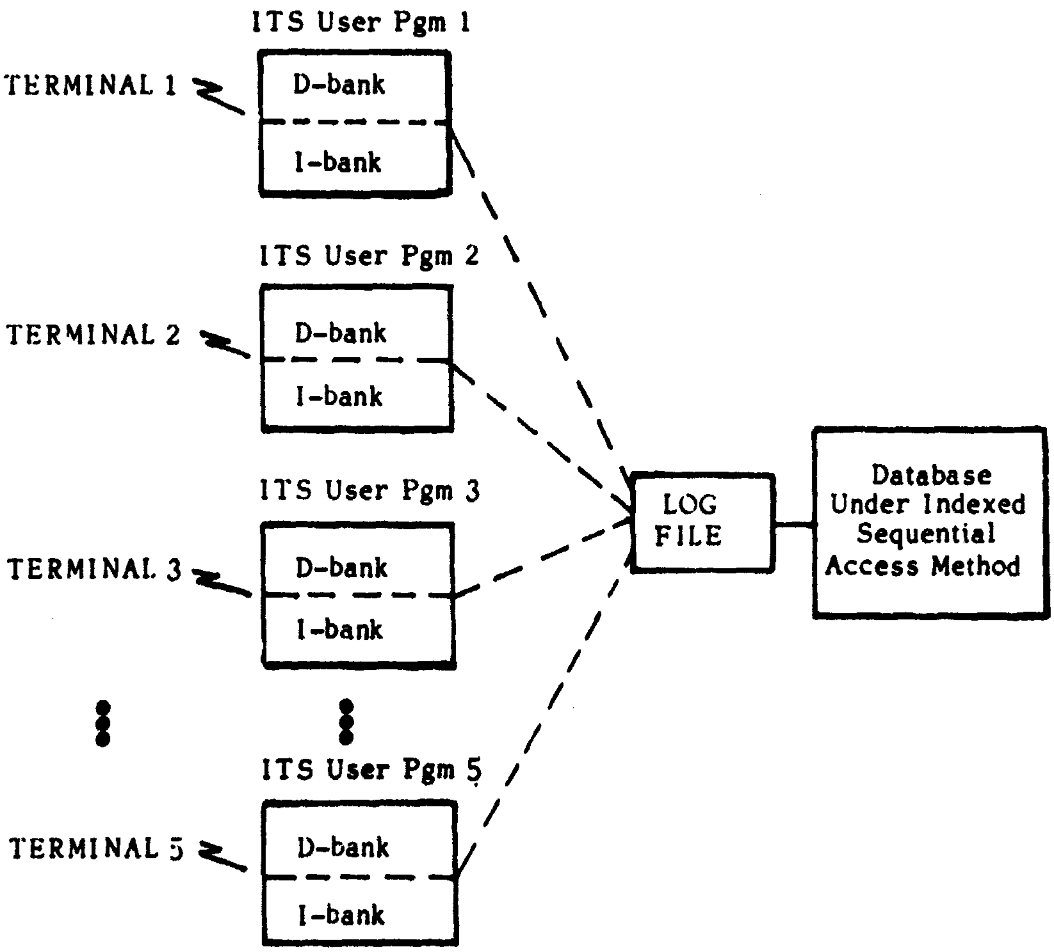


Figure 2-1. ITS Application Design Structure

Serial use of the database by multiple update users is forced by a "read and lock" routine which locks the log file, thus preventing simultaneous entry to the database files (PWA, Travel Master and Edit Master). If Update User 1 is using the database and Update User 2 tries to access it, User 2 will be suspended in the read and lock routine until User 1 finishes use of the database and frees the lock. Any modifications to the database done by User 1 are forced to the database, before it is unlocked, by a "flush" routine which causes the program buffers to be written to disk.

The internal structure of the overlayed ITS transaction program is shown in Figure 2-2.

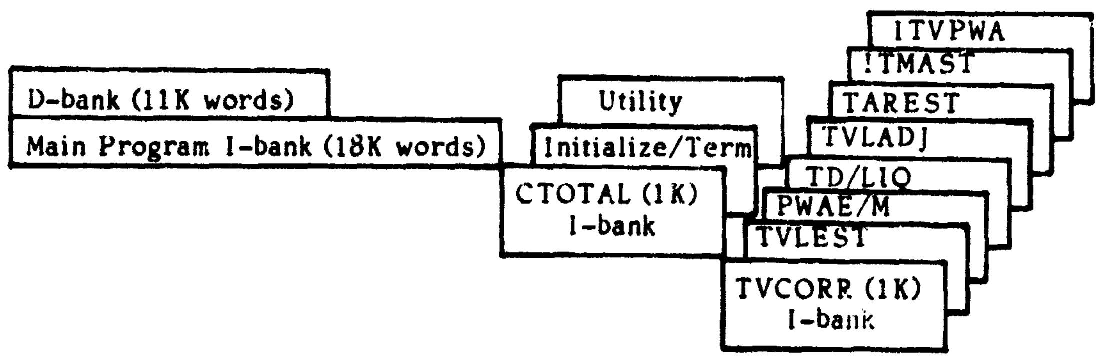


Figure 2.2 ITS Program Design

The main program consists of an ITS Control Routine (CTL) which reads and writes a Megadata terminal via a Terminal Handler routine and which calls the appropriate "process-data" module (i.e., TVCORR, TVLEST, etc.) corresponding to the "template" the terminal user selects. CTL and the process-data modules are written in COBOL; external program references from CTL include the list of routines shown in Table 2-1.

Table 2-I. External References of CTL

•	THII	TERMINAL HANDLER INITIATION
•	THI	TERMINAL HANDLER INPUT
•	THO	TERMINAL HANDLER OUTPUT
•	THT	TERMINAL HANDLER TERMINATION
•	REGCGY	REGISTERS CONTINGENCIES
•	FASGN	ASSIGNS DATABASE AND LOG FILES
•	LOCKIT	LOCKS DATABASE FILES
•	UNLOCK	UNLOCKS DATABASE FILES
•	LOGGER	PERFORMS VARIOUS LOGGING FUNCTIONS
•	DMR	
	- RRD	READS DATABASE RECORDS
	- MOD	MODIFIES EXISTING RECORDS FOR DB
	- INS	INSERTS NEW RECORDS FOR DB
	- OPENR	OPENS DATABASE INDEX SEQUENTIAL FILES
	- CLOSR	CLOSES DATABASE INDEX SEQUENTIAL FILES

These routines are written in a variety of languages (Assembler, COBOL, FORTRAN). Of special interest is the DMR routine, written in COBOL, which performs all of the indexed sequential database I/O. Isolating the COBOL I/O in DMR should ease ITS program extension and maintenance: e.g., incorporating new releases of the UNIVAC Indexed Sequential Access Method (ISAM).

In order to avoid "timeout" at active terminals where usage may be infrequent, the ITS program attached to a terminal is scheduled into main storage every 30 seconds by the Terminal I/O Handler (TH) routine; for details on this and other features of TH, the reader is referred to [6].

2.3 Current Usage Statistics

Accounting data on the FMD use of the current ITS application were gathered during the April-May 1978 period. The April data (Table 2-II) reflect the ANSI version of ISAM (no longer in use), whereas the May data (Table 2-III) reflect the current ASCII version of ISAM.

To better understand the response time statistics, consider the transaction components shown in Figure 2-3. Since initial time on the UNIVAC 1108 Core Request Queue is not included in the measured response time data, the measured responses tend to be one to two seconds lower than observed responses (depending on the competing load on the UNIVAC 1108). While the ANSI data only give visibility at a gross SUP-second level (i.e., ER/CC, I-O, and CPU charges are not listed separately) the ASCII data show individual SUP charges.

The only data that exceed goal response[1] are for the TVLEST and TVCORR transactions; since these transactions require more than one line of user data input the transaction program is sometimes "swapped-out" while awaiting the second and third lines of template data[2]. This swap-out problem has recently been remedied by a modification to the Terminal Handler (TH); a separate activity performing a "test-and-set" of a memory cell in the user program was added to TH to avoid the ITS program swap-out while awaiting second and third line "reads".

---

[1] Goal Response = 1.5 + 2S/R + 3W sec.
where; S = program size in K words,
R = swap device speed in K words per second, and
W = work time in SUP-seconds.

[2] The template data has already been typed in by the user but the terminal handler can only read 80 characters at a time.

TABLE 2-II. "RAW" ITS ACCOUNTING DATA (ANSI VERSION)

Period	CLASS	XACTION	#	FREQ. %	MEAN RESP. (Sec.)	MEAN SUP SEC
17-21 Apr. (+ some from 3-7 Apr.)	R	ITVPWA	33	1	0.768	0.245
	M	TVLADJ	654	18	2.276	0.712
	L	TVLEST	1310	36	5.404	1.639
	R	ITMAST	76	2	2.894	0.681
	S	TD/LIQ	1233	34	1.921	.629
	S	PWAE/M	18	-	1.815	.644
	L	TVCORR	70	2	4.069	1.194
	S	TAREST	221	6	2.139	0.521
	R	CTOTAL	47	1	2.917	1.103
24-28 Apr.	L	TVLEST	514	26	5.115	1.390
	S	TD/LIQ	812	42	1.725	0.636
	M	TVLADJ	501	26	3.145	1.465
	L	TVCORR	16	1	4.85	0.826
	R	CTOTAL	7	-	1.869	1.796
	R	ITMAST	76	4	2.737	0.683
	S	TAREST	12	1	1.457	0.540
	R	ITVPWA	3	-	0.471	0.287
	S	PWAE/M	3	-	1.27	0.662
			5606			

*Mean Inter-arrival (T.T. + Resp.) 40 sec. at any given terminal.

TABLE 2-III. "RAW" ITS STATISTICS (ASC II VERSION)

CLASS	XACTION	#	FREQ %	RESP (Sec)	CPU (SUP-Sec)	ER (SUP-Sec)	I/O (SUP-Sec)	TOTAL (SUP-Sec)	
M	TVLADJ	256	18	2.8	.093	.070	0.498	0.661	5/15/78 5/23/78
L	TVLEST	288	21	4.3	.135	.097	0.717	0.949	
M	TD/LIQ	645	46	1.8	.094	.073	0.515	0.682	
M	TAREST	97	7	1.8	.090	.064	0.451	0.605	
S	ITMAST	83	6	3.1	.071	.033	0.179	0.283	
R	CTOTAL	15	1	1.8	.076	.038	0.256	0.370	
L	TVCORR	6	-	5.0	0.102	0.094	0.532	0.728	
R	ITVPWA	1	-	3.2	0.046	-	-	-	
S	PWAE/M	9	1	0.7	0.056	0.052	0.331	0.439	
		1400	Mean Terminal Inter-arrival = 45 sec.						
M	TVLADJ	317	17	2.1				0.545	5/23/78 - 5/31/78
L	TVLEST	428	22	5.6				0.905	
M	TD/LIQ	1013	53	2.3				0.571	
M	TAREST	35	2	1.2				0.612	
S	ITMAST	60	3	2.5				0.287	
R	CTOTAL	31	2	2.0				0.937	
L	TVCORR	3	-	3.2				0.763	
R	ITVPWA	17	1	1.6				0.258	
S	PWAE/M	14	1	0.8				1.942	
		1918	Mean Terminal Inter-arrival = 49 sec.						

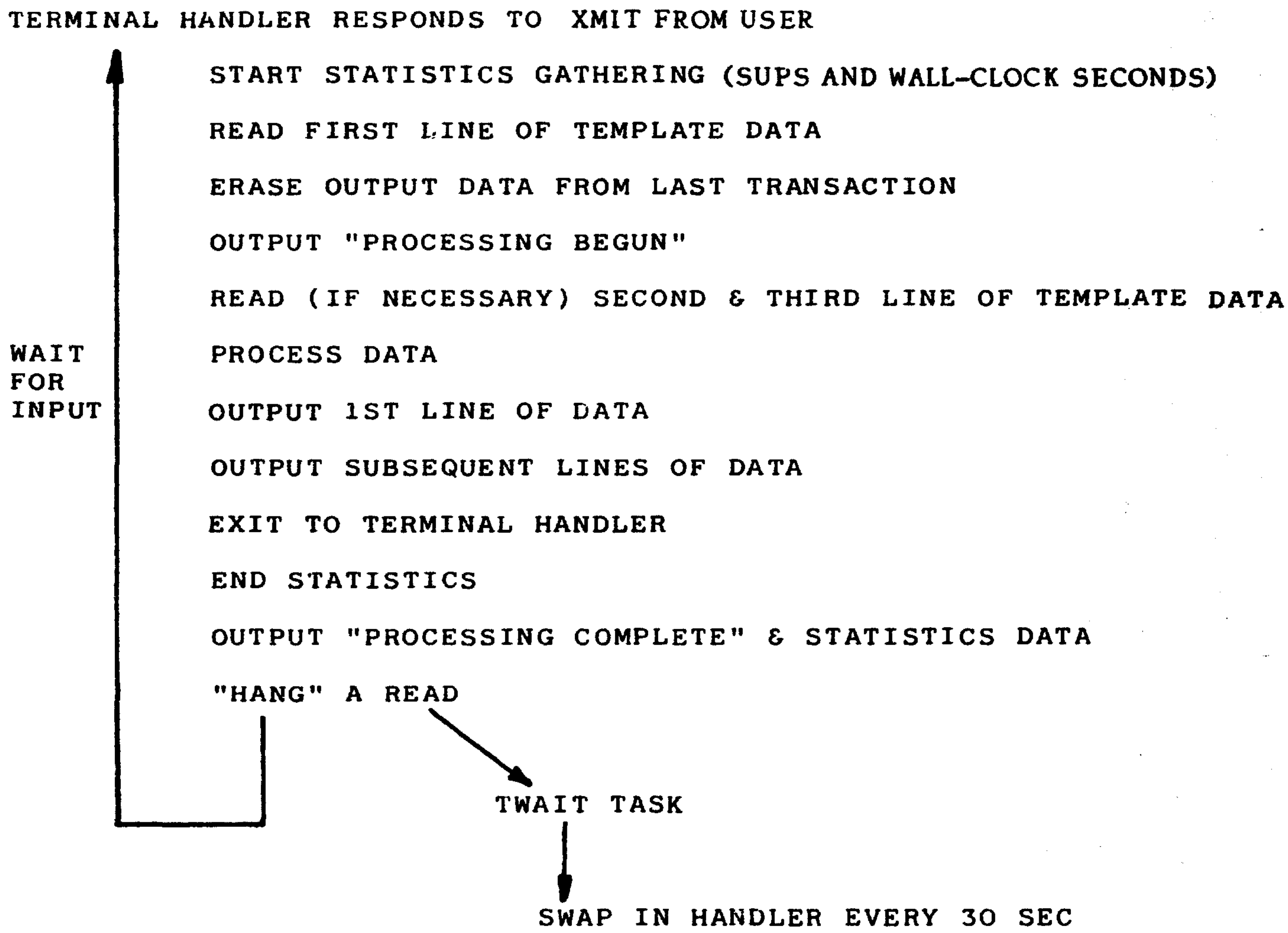


Figure 2-3. The ITS Transaction Components

The "number" and "frequency" columns of the data show that the April–May transaction load was in the 1500–1800 transaction per week range, and that retrieval transaction frequencies (class "R" in the Tables) constitute less than 10% of the load. Updates were assigned to three classes: small ("S"), medium ("M") and large ("L"), according to their relative SUP usage; the use for these classes will become clearer in Section 3 where we construct a model of the observed workload.

Transaction interarrival statistics were also gathered for each terminal during the April–May monitoring period and were found to be in the 40–50 second range (i.e., 40–50 seconds between the start of transactions "n" and n+1). The interarrival statistic includes both think-time (user thinking and entering data) and response time (computer working on data). Since ITS response times average about three seconds, ITS user think-times average about 42 seconds (assuming the observed mean 45 seconds interarrival time). Since one terminal working at a 45 second transaction interarrival rate will produce 480 transactions in a six hour day (thus 2400 transactions in a five day week), it can be seen that the mean observed weekly load (1500–1800 transactions per week) is produced by the equivalent of one terminal used continuously six hours per day. We will refer to this as a "concurrently" active average" (CAA) terminal in Section 3. In reality, of course, none of the five terminals is used continuously six hours per day. Typically the terminals are used in two to three hour sessions; during a session, transactions may occur every 10–15 seconds.

SECTION 3
THE ITS WORKLOAD MODEL

The statistics of Section 2, with one additional data source to be described shortly, allow the construction of an ITS application workload model. This model can serve as a basis for development of a model of the planned extension to ITS; i.e., the Interactive Basic Accounting System (IBAS). This section will construct the ITS workload model (Section 3.1) apply it as an input to the UNIVAC 1100 Series Simulation, (Section 3.2) and compare the ITS simulation model results to the observed April–May ITS data (section 3.3). This latter comparison is intended to "validate" the U1100 simulation model; this is a necessary step preparatory to the use of the model for forecasting the impact of a future IBAS workload on the U1108-9.

3.1 ITS Workload Model Construction

In order to accurately determine the file accesses of the various ITS transactions, IDSD personnel wrote an IO-Log routine which provided the detailed I/O information shown in Table 3-I. The table demonstrates the I-O profile of the TVLEST transaction; similar profiles were obtained for the other transaction types. This data, along with the measured CPU, ER/CC and frequency data of Table 2-III, facilitated the construction of the ITS workload model as shown in Table 3-II.

Table 3-1. Transaction I-O Data

FILE NAME	FUNCTION	# WORDS	B-ADDR	SECTOR	E-ADDR	
LOG	025	1	032602	4	032565	
LOG	020	112	055613	0	032544	
LOG	010	112	055613	0	032544	
EDITS	020	896	073020	676	017747	
EDITS	020	896	073020	708	017747	
EDITS	020	896	073020	68	017747	
EDITS	020	896	073020	36	017747	
EDITS	020	896	073020	292	017747	
EDITS	020	896	073020	772	017747	
EDITS	020	896	073020	708	017747	
TPWA	020	448	064703	100	017747	TVLEST
TPERF	020	896	067411	21572	017747	
TPERF	020	896	067411	23684	017747	
TPERF	020	896	067411	21572	017747	
TPERF	020	896	067411	23684	017747	
LOG	010	112	055613	136	032544	
LOG	010	112	055613	0	032544	
TPWA	010	448	064703	100	017747	
TPWA	020	448	064703	20	017747	
TPERF	010	896	067411	23684	017747	FLUSH
TPERF	020	896	067411	4	017747	
TPERF	020	896	067411	36	017747	
LOG	026	1	032602	4	032565	

Table 3-11. ITS Workload Model Construction

XACTION	FREQ. %	SIZE (K-Wds)	CPU (SUP-msec)	DISK ACCESSES
LARGE (TVLEST, TVCORR)	22	30	136	2(1792) 2(1) 4(112) 14(896) 4(448)
MEDIUM (TVLADJ, TD/LIQ, TAREST)	71	30	104	2(1792) 2(1) 4(12) 10(896) 4(448)
SMALL & RETRIEVE (CTOTAL, ITVPWA, ITMAST, PWAE/M, ITVPWA)	7	30	74	2(1792) 2(1) 4(12) 4(896) 2(448)

Let us deal with this table briefly. For example, the "disk access" column shows two program file accesses of 1792 words each per transaction, corresponding to the overlayed sections of the ITS program (see Section 2.2). Next, two log file accesses, one to "lock" and one to "unlock", were assumed per transaction. Four log file accesses to record data (for recovery and summary/journalling processing) were also assumed per transaction. Finally, four to 14 accesses to the Travel Master File (TPERF) and Edit File (EDITS) and two to four accesses to the PWA File (TPWA) were assumed, depending on the type of transaction and its complexity. These accesses represent averages on a fairly "clean" ISAM database; i.e., one which has been recently reorganized. CPU, program size and frequency parameters were based directly on the gathered statistics of Section 2.3.

3.2 U1100 Simulation Model Inputs

The ITS workload of Table 3-II was input to the U1100 Simulation Model (hereinafter called the "Model") using two ITS terminals as the assumed load. The two terminals are twice the observed concurrently active average (CAA) terminal load (one terminal, as demonstrated in Section 2.3) and thus represent a typical "instantaneous" load instead of a weekly average load. Four CAA GIM terminals using the recent test and set queueing design were also input to the model. The GIM test and set logic is assumed to be represented by the model shown in Table 3-III; this workload model is the same one used in the IDSD FY79 Computer System Plan [7]. Seven SVDS background batch runs, as shown in Table 3-IV, were also opened as input to the model, along with one Category 2,5 run (as shown in Table 3-V). The SVDS and Category 2,5 background batch models were also taken from the IDSD FY79 Computer System Plan.

The hardware configuration parameters of the Model were chosen to represent the current U1108-9, since ITS, GIM and SVDS workloads are currently running on that computer at IDSD. Table 3-VI shows the configuration parameters input to the Model to represent the U1108-9.

Table 3-VI. U1108-9 Model Parameters

SUBSYSTEM \ EXEC-8 SYSTEM		U1108-9	
Size of Memory:	Primary (words)	262K (163K available to user)	
:	Extended	–	
No. of Drums:	FH432 (single-ported)	2*	
	FH432 (dual-ported)	3	Hybrid string*
	FH1782 (dual-ported swap file)	1	Hybrid string*
No. of 8433 Disks (dual-ported)		4	
No. of Tapes (dual-ported)		12	

*1 to 1 interlace.

TABLE 3-III. GIM DEMAND WORKLOAD MODEL

FREQUENCY %	SIZE (K-WDS)	CAU (SUP-ms)	# ACCESS 432	# ACCESS 1782*	# ACCESS DISK	# ACCESS TAPE	I/O (SUP-ms)	ER/CC (SUP-ms)	TOTAL (SUP-ms)	GOAL RESP (SUP-ms)**	THINK TIME (SEC)	
40	45K EXEC + 15K REMOTE	400		4(1792)	7(560)		282		682		15	PROD
50		300		10(1792)	16(560)		660		960		100	PROD
10		2700		5(1792)	135(560)		4140		6840		38	PROD
AVE	60	580		7(1792)	24(560)		857		1437	11100	60	PROD
90	45K EXEC	300		5(1792)	8(560)		330		630		-	DEVEL
10		650		4(1792)	20(560)		672		1322			DEVEL
AVE	45	335		5(1792)	9(560)		364		699	3990	-	DEVEL

NOTE: NUMBERS IN PARENTHESES DENOTE RECORD SIZE IN WORDS

*FOR PRODUCTION GIM, EACH OF THE TRANSACTIONS ALSO INCLUDES THREE 112 WORD ACCESSES AND ONE 80 WORD ACCESS TO THE 1782 FOR COMMUNICATIONS AND STATUS FILE TASKS.

**BASED ON 4 USERS COMPETING FOR A SINGLE GIM EXEC COPY @ GOAL RESPONSE.

	% CAU	% I/O	I-O/CAU
PROD	40	60	1.5
DEV	48	52	1.1

TABLE 3-IV. SVDS BATCH WORKLOAD MODEL

FREQUENCY %	SIZE (K-WDS)	CAU (SUP-ms)	# ACCESS 432	# ACCESS 1782	# ACCESS DISK	# ACCESS TAPE	I/O (SUP-ms)	ER/CC (SUP-ms)	TOTAL (SUP-ms)
50	35	1000			90(224)		1990	1890	4880
34	16	6970	70(224)*	90(224)	830(224)	280(224)	26920	5480	39370
8	42	25680	820(224)		2680(224)	280(224)	70750	11720	108150
8	58	116440	3140(100)		740(100)	30(100)	30760	20410	167610
AVE	31	14239	341	31	601	120	18269	5379	37887

NOTE: NUMBERS IN PARENTHESES DENOTE RECORD SIZE IN WORDS.

% ER/CC	% CAU	% I/O	I-O/CAU
14	38	48	1.3

TABLE 3-V. CATEGORY 2,5 BATCH WORKLOAD MODEL

FREQUENCY %	SIZE (K-WDS)	CAU (SUP-ms)	# ACCESS 432	# ACCESS 1782	# ACCESS DISK	# ACCESS TAPE	I/O (SUP-ms)	ER/CC (SUP-ms)	TOTAL (SUP-ms)
12.5	4	84	25		592	1181	43337	1563	44984
12.5	7	3319	111		308	5972	158455	9623	171397
12.5	11	1795	53		763	1832	63812	13090	78697
12.5	15	3432	60	1	1199	320	35534	10458	49424
12.5	13	3958	27		4277	6133	251521	26220	281699
12.5	28	69626	39		1887	40	43831	168600	282057
12.5	32	24613	206	12	2727	263	69716	25827	120356
12.5	33	2986	98		1495	939	58067	15044	76097
AVE	18	13752	77	2	1656	2085	90534	33803	138039

NOTE: ALL I/O'S HAD RECORD SIZES OF 256 WORDS.

% ER/CC	% CAU	% I/O	I-O/CAU
24	10	66	6.6

3.3 Model Results and Analysis

The response time results of the ITS Model run, constructed as described in Section 3.2, are presented in Table 3-VII. In the table these results are compared to the actual observed ITS response time data gathered in the April-May time period. Not enough "small" transactions (ITMAST, CTOTAL, ITVPWA, PWAE/M) were processed in the three minute model run to present any statistically significant comparison. This occurs because the "small" class of transactions is only 7% of the observed weekly load. The comparisons for the "medium" and "large" transactions, while favorable, must be viewed with caution, since only a few "large" and "medium" transactions were processed in the Model run.

Table 3-VII. ITS Model Validation Results

	MODEL RESULTS	OBSERVED RESULTS*
Large Transaction Response (sec) (TVLEST, TVCORR)	6.4	6.7
Medium Transaction Response (sec) (TVLADJ, TD/LIQ, TAREST)	3.7	3.6

*1.5 sec initial core request queue delay added to observed responses.

SECTION 4
THE INTERACTIVE BASIC ACCOUNTING SYSTEM MODEL

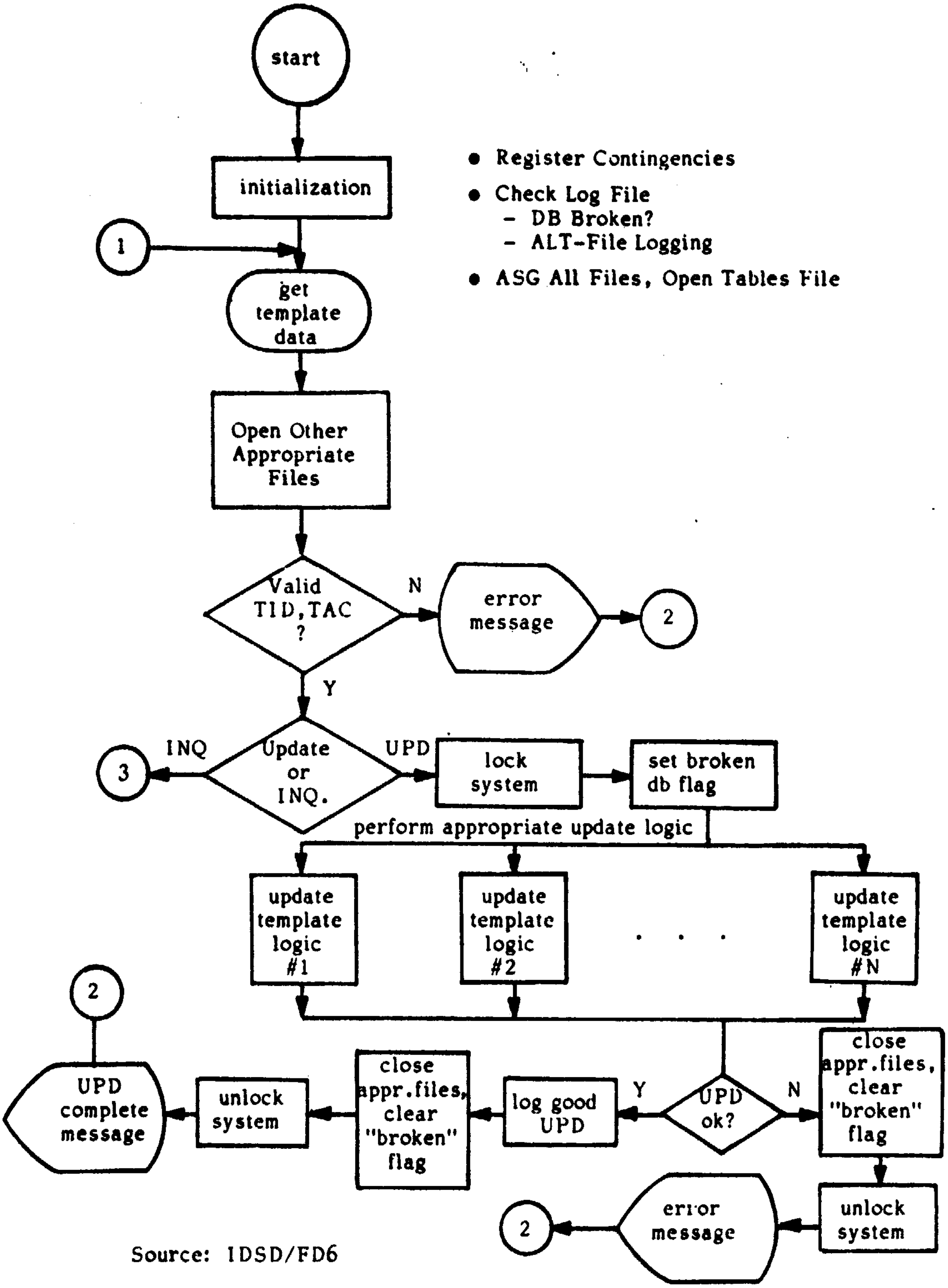
The Interactive Basic Accounting System (IBAS) is currently in the process of design by IDSD personnel. It is viewed as an extension to ITS, both in program design and in database design. Figure 4-1 and Table 4-I, respectively, illustrate the current IBAS program design and database structure. The currently planned design differs from ITS mainly in the number of files (five versus the current three - purchase request (PR) and contract (CONT) files have been added) and in the program logic for assigning, opening and closing files as needed by a given transaction, instead of having the whole database collection of files open for all transactions as is currently done in ITS.

While this approach to the IBAS design has the obvious advantage of simplicity, there is some concern that the increased IBAS terminal population (14 users) will cause database queuing (which was not the case for the five ITS terminals). Thus one of the major IBAS design issues to be investigated is the adequacy of the serially reusable database structure inherent to ITS.

Some other IBAS design issues involve:

- Tradeoffs of increasing the number of index and data buffers allocated to a program for ISAM use (at the expense of program size) against the I-O saved by having more buffers;
- Segmentation of the IBAS transaction (e.g., the terminal I-O Handler is currently "mapped" with the ITS transaction, but could be segmented, with a resultant storage savings).

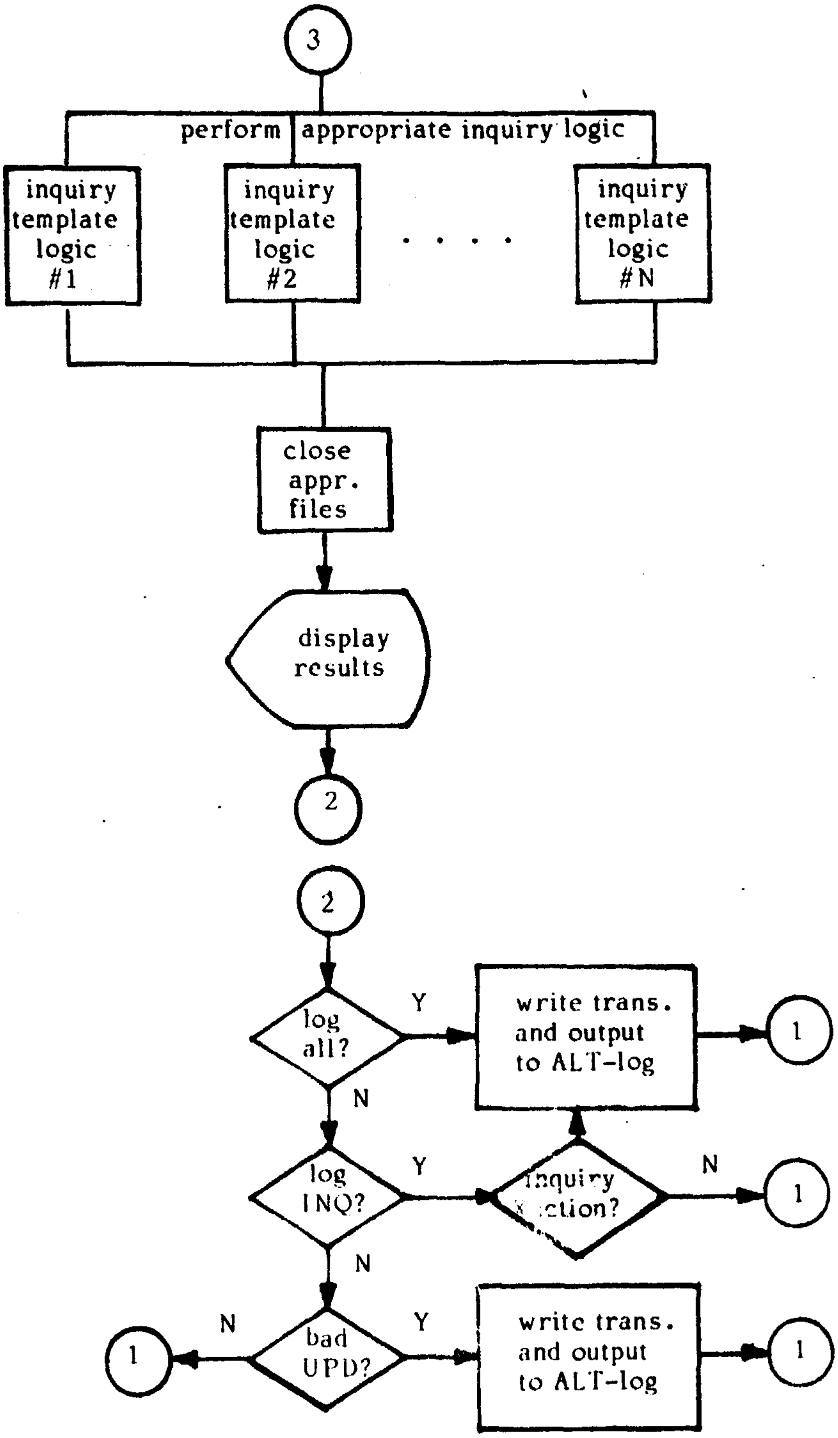
There are also U1108-9 system considerations which impact IBAS response, such as the nature and amount of the workload competing with IBAS.



- Register Contingencies
- Check Log File
  - DB Broken?
  - ALT-File Logging
- ASG All Files, Open Tables File

Source: IDSD/FD6

Figure 4-1. IBAS Program Design



Source:
IDSD/FD6

Figure 4-1. (Concluded)

TABLE 4-1. IBAS FILES

FILE NAME	NO. RECORDS MAX	RECORD LENGTH (Wds)	KEY LENGTH (Wds)
PR	50,000	41	2
CONT	25,000	21	2
TABLES	26,000	21	2
FC	7,000	25	5
TPERF	30,000	41	3

Source: IDSD/FD6

This section investigates these design issues by simulation. Test cases are defined and input to the UNIVAC 1100 Series Simulation Model (Section 4.1). Results of the test cases are set forth in Section 4.2, and conclusions regarding "preferred" cases and their impact on IBAS design presented in Section 4.3.

4.1 IBAS Test Cases and Assumptions

The following assumptions are made about the IBAS:

- There are three transaction types (Large, Medium and Small) each having three subtransactions: a polling subtransaction, a template subtransaction and a process data subtransaction.
- The IBAS will be driven by seven terminals[1], running medium, large and small transactions respectively in the ratio 71%, 22%, 7% (the same frequencies as observed in ITS), and with think time between transactions of 45 seconds at each terminal.
- Four GIM terminals, configured with the "test and set" workload shown in Table 3-III, will run as background to the seven ITS terminals.
- A serially reusable database design will be assumed for IBAS; other options will be considered only if response under this design appears unacceptable.
- The resource requirements of the transactions and subtransactions are assumed to be as shown in the test cases of Table 4-II.

[1] The assumption is made that although there are 14 physical terminals, a normal instantaneous load will consist of only seven terminals.

TABLE 4-II. IBAS TEST CASES

COMPONENTS \ CASE	CASE 1	CASE 1'	CASE 1"	CASE 1'''	CASE 2
Polling Sub-Transaction					
Memory (KW)	1I,2D	1I,2D	3I,12D	23I,12D	1I,2D
Period (Sec)	30	5	5	5	5
Template Sub-Transaction		↑	↑		↑
• "Large"					
– Memory (KW)	3I,12D			23I,12D	
– CPU (SUP-msec)	104			104	
• "Medium"					
– Memory	3I,12D			23I,12D	
– CPU	5			5	AS IN CASE 1
• "Small"					
– Memory	3I,12D	AS IN CASE 1		23I,12D	
– CPU	5			5	↓
Process Data Sub-Transaction			AS IN CASE 1	↑	
• "Large"					
– Memory	23I,12D				23I,22D
– CPU	132				132
– I/O- #(size in words)	2(1),4(112), 8(448),22(896)			AS IN CASE 1	2(1),4(112), 4(448),14(896)
• "Medium"					↑
– Memory	23I,12D				
– CPU	102				
– I/O	2(1),4(112), 4(448),10(896)				
• "Small"					
– Memory	23I,12D				AS IN CASE 1
– CPU	72				
– I/O	2(1),4(112), 2(448),4(896)	↓	↓	↓	↓

- Case 1 (and variations) represent the minimally buffered IBAS program design; Case 2 represents the maximally buffered IBAS program design.

- The batch background workload variations are as shown in Table 4-III.

Table 4-III. Batch Background Workload

Component \ Background Name	MPAD	LIGHT MPAD	MGMT
SVDS Runs	7	2	-
Management Runs	-	-	6
CAT 2,5 Runs	1	1	1

Relative to Table 4-III, SVDS and CAT 2,5 workloads were shown respectively earlier in Tables 3-IV and 3-V; the Management workload is shown in Table 4-IV (taken from the FY79 Computer System Plan).

4.2 IBAS Model Results and Analysis

The results of the IBAS model run are shown in Table 4-V. Before analyzing these data, it is instructive to study Figure 4-2 which shows the IBAS response time components in the context of the U1108 architecture. The "core request queue" row of Table 4-V accumulates swap delays attributable to 1) initial transaction load, 2) READ$ swap delays (on multiple line template inputs) and 3) swap delays because the database is locked. Actual time on the database queue is recorded in the "DBQ" row of Table 4-V. Ready-to-execute delays are caused when the transaction waits for CPU service.

TABLE 4-IV. MANAGEMENT BATCH WORKLOAD MODEL

FREQUENCY %	SIZE (K-WDS)	CAU (SUP-ms)	# ACCESS 432	# ACCESS 1782	# ACCESS DISK	# ACCESS TAPE	I/O (SUP-ms)	ER/CC (SUP-ms)	TOTAL (SUP-ms)
10	10	18156	775		273	558	25232	12717	56105
10	9	1255	10		1507	2821	105292	3842	110389
10	11	134	151		22	245	7654	5387	13175
10	14	792	66		16	60	2300	3490	6582
10	12	72	10		116	156	6620	1009	7701
10	31	12039	1005		1212		33805	19367	65211
10	31	1572	105		827	5	19459	4511	25542
10	41	249224	486		28045	8069	839562	64529	1153315
10	41	16056	71		3831	1444	123345	6219	145620
10	39	5082	62		3434	337	86379	7403	98864
AVE	24	30438	274		3928	1370	124965	12847	168250

NOTE: ALL I/O'S HAD RECORD SIZES OF 256 WORDS.

% ER/CC	% CAU	% I/O	I-O/CAU
8	18	74	4.1

TABLE 4-V. GPSS MODEL RESULTS OF IBAS DESIGN CASES

PARAMETER \ CASE	CASE 1 w/MPAD BATCH	CASE 1' w/MPAD BATCH	CASE 1' w/MGMT BATCH	CASE 1' w/LIGHT MPAD	CASE 1" w/MPAD BATCH	CASE 1''' w/MPAD BATCH	CASE 2 w/MPAD BATCH	CASE 2 w/LIGHT MPAD
Response (Sec)								
• Large Xaction	12.2	7.5	5.4	4.8	7.3	8.0	8.7	4.3
• Medium Xaction	6.7	6.0	4.1	3.1	6.7	7.3	7.8	2.8
CRQ (Sec)								
• Large Xaction	1.8	0.9	1.5	0	1.4	3.8	1.8	0
• Medium Xaction	4.0	1.7	1.5	0	2.2	4.5	3.0	0
RTEQ (Sec)								
• Large Template Sub-Xaction	6.0	0.5	0.7	0.7	1.6	0.8	1.6	0.6
GIM Response (Sec)								
• Retrieve	14.9	11.6	9.1	3.6	12.0	14.9	10.7	4.1
• Update	15.8	21.9	7.2	3.4	4.0	13.7	14.3	2.7
• Conditional TL-1	21.8	24.9	19.5	14.2	21.5	21.1	18.1	14.3
DBQ (Sec)	0.5	2.1	0.4	1.0	1.8	0.6	2.4	1.0
Memory Utilization (%)	90%	88%	85%	47%	88%	87%	88%	47%
CPU Utilization (%)	97%	95%	88%	80%	96%	98%	97%	80%

CONSTANTS (except as noted):
- 7 SVDS, 1-CAT 2,5
- 7 BAS Terminals
  - 45 Seconds Think Time
  - Large/Medium/Small Frequencies - 22/71/7

CRQ = Core Request Queue
RTEQ = Ready To Execute Queue
DBQ = Database Queue

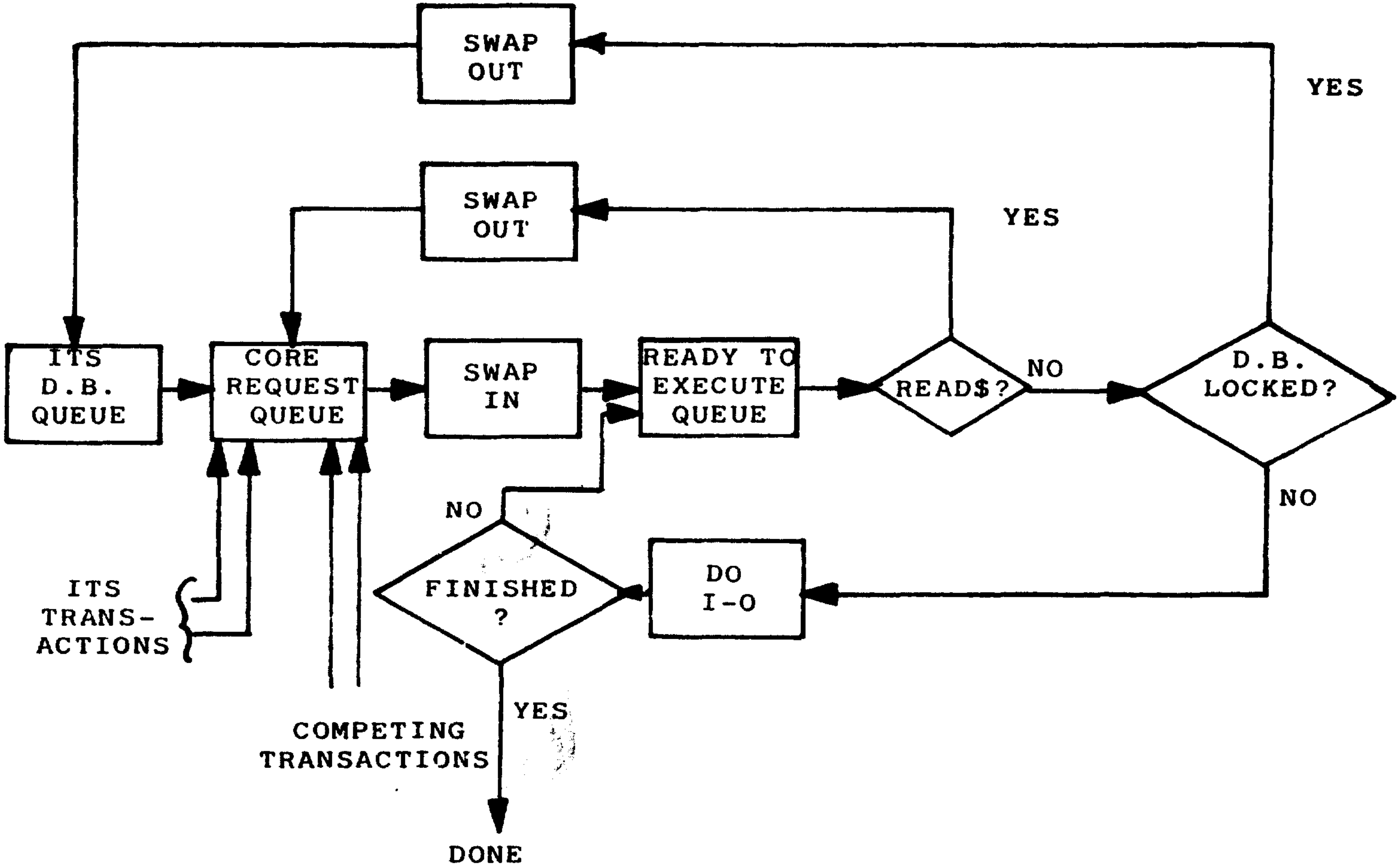


Figure 4-2. ITS Response Time Components

Results of the individual test cases, along with some analysis of these results, follow:

a) <u>Case 1 with MPAD Batch</u>

This case demonstrates that a 30 second polling subtransaction of 3K words causes unacceptably high IBAS core request queue (CRQ) and ready-to-execute queue (RTEQ) time delays. In competing with the MPAD workload, which makes heavy demands on the CPU and main memory, it is beneficial to IBAS to limit the resources given to MPAD; hypothetically more frequent IBAS polling should do this.

b) <u>Case 1' with MPAD Batch</u>

This case explores the above polling hypothesis: the IBAS response results indicate a favorable outcome (at the expense of MPAD throughput, and memory and CPU utilization).

c) <u>Case 1' with Management Batch</u>

This test shows the effect of changing the background batch workload from the current MPAD to a management batch workload mix. The IBAS response results are very much in favor of the management batch workload.

d) <u>Case 1' with Light MPAD Batch</u>

This test shows the effect on IBAS (Case 1') of removing five of the seven MPAD batch runs. IBAS mean response improves by almost three seconds per transaction: this design case is a lower limit to the expected IBAS response, which of course will increase with background workload.

e) Cases 1" and 1'" (with MPAD Batch)

These cases explore the sensitivity of the IBAS response to the subtransaction program size: there is little statistical evidence to support any beneficial aspects (to IBAS) of making the polling and template subtransaction smaller than the process data subtransaction.

f) Cases 2 (with MPAD Batch) and 2 (with Light MPAD Batch)

These two cases demonstrate the effect of changing the "large" process data subtransaction from 35K to 45K words and correspondingly lowering the I-O requirements of the 45K subtransaction. There is no statistical evidence to clearly favor or reject this approach, since the responses of Case 2 with light and heavy MPAD batch closely match those of Case 1' with light and heavy MPAD batch.

4.3 Conclusions

Several conclusions are derivative from the model runs:

- IBAS competes more favorably, from a mean response viewpoint, with a management batch background workload than with an MPAD batch background workload. This would indicate some shifting of the current MPAD workload off the U1108-9, if it could be replaced with workload components having management batch characteristics.

- There is no need to eliminate or increase the 30 second ITS Terminal Handler polling period (which currently avoids terminal timeout). There is, in fact, model evidence to favor reducing the polling period.

- IBAS response is not strongly corellated to subtransaction or transaction size. Thus, expensive efforts to reduce program size are not indicated. Indeed, they could be counter-productive (to IBAS) since making memory available to non-IBAS programs increases the RTEQ and CRQ waits for IBAS programs.

- Database queuing on the serially reuseable IBAS ISAM-managed database is not a problem as long as the mean transaction work requirements (I-O and CPU) remain in the one SUP-second range and the number of concurrently active terminals does not exceed seven. Relative to this latter point, Table 4-VI shows a 14-concurrent terminal IBAS model run. While this is a much heavier load than expected, it does indicate that with this load database queuing assumes significant proportions.

TABLE 4-VI. 14-TERMINAL IBAS MODEL RUN

PARAMETER \ CASE	CASE 1' WITH 14 TERMINALS & MPAD BATCH
Response (Sec)	
• Large Xaction	23.0
• Medium "	14.0
• Small "	6.5
CRQ (Sec)	
• Large Xaction	5.6
• Medium "	3.2
• Small "	0.1
RTEQ (Sec)	
• Large Template Sub-Xaction	2.2
GIM Response (Sec)	
• Retrieve	10.0
• Update	9.1
• Conditional TL-1	20.9
DBQ (Sec)	7.7
Memory Utilization (%)	88%
CPU Utilization (%)	95%

CONSTANT (except as noted):

- 7 SVDS, 1-Cat 2,5
- 7 BAS Terminals
  - 45 Seconds Think Time
  - Large/Medium/Small Frequencies - 22/71/7

SECTION 5

TESTING OF CANDIDATE IBAS DESIGNS ON THE U1108-9

This section describes the conversion of the IBAS workload to a synthetic program (Section 5.1) and presents the results of running the synthetic IBAS workload on the U1108-7 with an actual test-and-set GIM workload and a synthetic background workload (Section 5.2). The Performance Evaluation System (PES), a tool developed by MITRE for FD2 [3], was used in performing these tests.

5.1 Development of the Synthetic IBAS Workload

A PES workload model of ITS was developed from the observed ITS statistics shown in Table 2-III. This workload model is shown in Table 5-I.

Table 5-I. PES Workload Model

SEQ.NO.	NO. RUNS	SIZE (words)	% P*	DEL. RATE BY SIZE			TR	WORK REQUIREMENT**		
				CAU	I/O	TOTAL		CAU	I/O	TOTAL
DEM 1	7	22700	100	.014	.073	.087	MB	.094	.512	.606
							MC	.094	.484	.578
							MA	.094	.501	.595
							LA	.135	.693	.828
							LC	.135	.707	.842
							SC	.072	.186	.258
							SA	.072	.193	.265
							LB	.135	.701	.836
							SB	.072	.189	.261

*% execution from primary memory

**SUP-seconds

In this table, SA, SB and SC represent "small" ITS transactions; analogously, MA, MB, MC and LA, LB, LC represent "medium" and "large" transactions respectively. The 22.7K program size was intentionally made smaller than the 30K "actual" ITS program size in order to make allowance for the driver program used to dispatch the synthetic ITS demand runs. The model was constructed to preserve the observed 22%/71%/7% ratio of large/medium/small ITS transactions. Interarrival between transactions at a given terminal was set to 45 seconds, again to correspond to observed ITS statistics.

Extension of this ITS workload model to IBAS was accomplished by simply increasing the number of terminals from two to seven.

5.2 Background Workload and Results

As companion workloads to the seven IBAS terminals, four GIM terminals running the transactions shown in Appendix A and seven background batch runs executing the transactions shown in Table 5-II were obtained from other sources. Specifically, the GIM workload was obtained from a LEC effort which monitored actual PMATS[1] activity on GIM during a brief period in June 1978 [8]; TRW has been using this workload in testing the soon-to-be-released "test-and-set" version of GIM. The background batch workload was obtained from LEC in support of a then-active FD2/LEC task to collect statistics on existing UNIVAC workloads and convert them to synthetic programs. The background workload of Table 5-II represents a typical Mission Planning and Analysis (MPAD) workload which was generated from accounting tapes on the UNIVAC 1108-9. It is noticeably "lighter" than the MPAD workload used in the Model runs (Table 3-IV). Analogously, the PMATS/GIM workload of Appendix A is different than the workload used in the model runs (Table 3-III). Thus we cannot in any way validate the model runs; instead,

[1] Program Management and Tracking System

Table 5-11. PES Batch Background Workload Model

SEQ.NO.	NO. RUNS	SIZE (words)	% P	DEL. RATE BY SIZE			TR	WORK REQUIREMENT		
				CAU	I/O	TOTAL		CAU	I/O	TOTAL
BAT 1	7	27264	100	.050	.093	.143	FB	.186	.349	.535
							FC	.134	.305	.439
							FD	.069	.236	.305
							FE	.063	.236	.299
							FA	.287	.392	.679
							FX	2.600	.000	2.600
		24192	100	.052	.176	.228	MX	2.600	.000	2.600
							MB	.873	2.397	3.270
							MA	1.472	2.730	4.202
							ME	1.281	7.188	8.469
							MD	1.587	6.166	7.753
							MC	1.125	3.743	4.868
		16128	100	.035	.013	.048	DX	2.600	.000	2.600
							DM	1.019	.108	1.127
							DN	.068	1.117	1.185
		5760	100	.007	.002	.009	DX	2.600	.000	2.600
							DL	.009	.208	.217
							DI	.967	.037	1.004
							DK	.013	.037	.050
							DJ	.031	.065	.096
		8832	100	.004	.066	.070	PX	2.600	.000	2.600
							PA	.011	.171	.182
							PE	.034	.954	.988
							PC	.054	.835	.899
							PD	.022	.551	.573
							PB	.062	.475	.537
		1536	100	.000	.003	.003	DX	2.600	.000	2.600
							DA	.049	.065	.114
							DD	.002	.094	.096
							DB	.038	.350	.388
							DC	.001	.037	.038
		4224	100	.000	.001	.001	DX	2.600	.000	2.600
							DF	.021	.065	.086
							DG	.004	.037	.041
							DE	.052	.074	.126
							DH	.003	.037	.040

we can only get estimates of how IBAS would compete on the UNIVAC 1108-9 with "typical" background workload. With these cautionary statements in hand, we present as Table 5-III, the IBAS results of a PES experiment in which the workload consisted of the following runs:

- seven IBAS demand,
- four PMATS/GIM demand,
- seven MPAD batch.

Table 5-III. PES IBAS Results

XACT	MEAN RESP	GOAL RESP	GOAL/ MEAN	SAMPLE	MAX RESP	MIN RESP	ST.DEV. RESP	MAX TRQT	MIN TRQT	MEAN TRQT
M	2.6	3.2	1.4	27	4.3	1.2	1.1	2.3	0.4	0.9
L	3.3	4.4	1.3	5	3.9	2.1	0.7	1.7	0.5	0.7
S	1.5	3.3	3.2	3	2.3	1.0	0.7	1.4	0.3	0.7

All response and TRQT in seconds.

The results shown in Table 5-II indicate that a medium transaction (i.e., the weighted mean of transactions MA, MB and MC) exhibits 2.6 seconds response, while a mean large transaction required 3.3 seconds to execute. Since the PES does not model the database queue delay (DBQ) of an update transaction, these responses must be considered as best case estimates (e.g., see the DBQ delays observed in the simulation runs of Table 4-V). Exact comparisons are impossible, due to the fact that dissimilar GIM/PMATS and batch background workloads were run on PES and the simulation model. Additionally, the sample size (especially for large and small transactions) is small.

James F. Spitzer
Group Leader

APPENDIX A

GIM/PMATS TRANSACTIONS
(Provided by TRW Systems)

```
POLL*GIM(1).DI
???00300
2018/GIM
2018/GIM
@RUN 660DS,FD8/2018,FD6-T61796,25,500
???00020
@QUAL POLL
???00020
@XQT *GIM.GIMCOM/MDATA
???00020
LOGON 290030#
???00026
ADD CONT-FILE 'T-7715F' %1 "TEKTRONIX" %2 "780424" %3 "78092
5"
 %4 "4" %6 "FFP"#
???00355
ADD CONT-FILE 'T-7486F' %1 "EG + G" %2 "780517" %3 "780712"
 %4 "2" %6 "FFP"#
???00068
ADD CONT-FILE 'T-7483F' %1 "VALIDYNE ENG" %2 "780512" %3 "78
0626"
 %4 "1" %6 "FFP"#
???00282
ADD CONT-FILE 'T-7474F' %1 "WATKINS JOHNSON" %2 "780509" %3
"780731"
 %4 "2" %6 "FFP"#
???00025
ADD CONT-FILE 'T-8288F' %1 "HUMPHREY" %2 "780509" %3 "780707
"
 %4 "2" %6 "FFP"#
???00036
ADD CONT-FILE 'T-7843F' %1 "INTL MICROWAVE" %2 "780519" %3 "
```

```
780907"
        %4 "4" %6 "FFP"#
       ???00028
       ADD CONT-FILE 'T-8308F' %1 "SPIN PHYSICS" %2 "780519" %3 "78
0828"
        %4 "3" %6 "FFP"#
       LOGOFF#
       @FIN
```

COMES UP LATE

```
@QUAL POLL
READY
>@PRT *GIM.EW,.DS,.JM,.DH
FURPUR 27R2  PL72-8 07/07/78 14:01:00



POLL*GIM(1).EW
        2018/GIM
        2018/GIM
        @RUN 660EW,FD8/2018,FD6-T61796,25,500
        ???00020
        @QUAL POLL
        ???00020
        @XQT *GIM.GIMCOM/MDATA
        ???00020
        LOGON 290014#
        ???00412
        LIST DR-FILE 'EA*0084*T' T-DATES PR-NO B-NM B-PH#
        ???00019
        LIST DR-FILE 'EA*0084*V' T-DATES#
        ???00064
        LIST DR-FILE 'EA*0119*' T-DATES PR-NO B-NM B-PH#
        ???00048
        LIST DR-FILE 'EA*0119*' ACT-DT INIT$ COMM$ OBLG$ FUND-Y#
        ???00050
        LIST DR-FILE 'EA*0107*A' PR-NO T-DATES B-NM B-PH#
        ???00056
        LIST DR-FILE 'EA*0107*A' ACT-DT INIT$ COMM$ OBLG$ FUND-Y#
        ???00031
```

```
LIST DR-FILE 'EA*0121*' PR-NO T-DATES B-NM B-PH#
??700035
LIST DR-FILE 'EA*0121*' ACT-DT INIT$ COMM$ OBLG$ FUND-Y#
??700041
LIST DR-FILE 'EA*0119*B' PR-NO T-DATES B-NM B-PH#
??700046
LIST DR-FILE 'EA*0119*B' ACT-DT INIT$ COMM$ OBLG$ FUND-Y#
??700040
LIST DR-FILE 'EA*0119*C' ACT-DT INIT$ COMM$ OBLG$ FUND-Y#
??700018
LIST DR-FILE 'EA*0119*C' PR-NO T-DATES B-NM B-PH#
??700035
LIST DR-FILE 'EA*0084*W' PR-NO B-NM B-PH T-DATES#
??700060
EXECUTE PSR-1 '2J' VARIABLES "P=78" "V=A"#
??700055
LIST DR-FILE 'ED*1528*V' T-DATES PR-NO B-NM B-PH#
??700035
LIST DR-FILE 'ED*1528*AA' T-DATES PR-NO B-NM B-PH#
LOGOFF#
```

```
POLL*GIM(1).JM
2018/GIM
2018/GIM
@RUN 660JM,FD8/2018,FD6-T61796,25,500
??700020
@QUAL POLL
??700020
@XQT *GIM.GIMCOM/MDATA
??700020
LOGON 290024#
??700067
CHANGE DR-FILE 'EW*3216*' %44 TO "780418" %45 TO "780419"#
??700005
LIST DR-FILE 'EW*3218*A' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
 S-1 S-2 S-3 S-4 S-5 S-6 CONT#
??700023
LIST DR-FILE 'EW*4347*' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A-
D
 S-1 S-2 S-3 S-4 S-5 S-6 CONT#
??700067
LIST DR-FILE 'EW*3247*' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A-
D
 S-1 S-2 S-3 S-4 S-5 S-6 CONT#
??700005
LIST DR-FILE 'EW*3252*A' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
 S-1 S-2 S-3 S-4 S-5 S-6 CONT#
??700042
CHANGE DR-FILE 'EW*3252*' %55 TO "9BC73-27-8-54P" %43 TO "78
0330"
 %44 TO "780403" %45 TO "780403"#
??700016
LIST DR-FILE 'EW*3286*' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A-
D
```

```
29        S-1 S-2 S-3 S-4 S-5 S-6 CONT*
30       ??700040
31       CHANGE DR-FILE 'EW•3286•' %42 TO "000000" %43 TO "000000"
32        %44 TO "000000" %45 TO "000000"*
33       ??700009
34       LIST DR-FILE 'EW•3295•' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A-
D
35        S-1 S-2 S-3 S-4 S-5 S-6 CONT*
36       ??700049
37       CHANGE DR-FILE 'EW•3295•' %55 TO "9BC2D-A06-8-22P" %32 TO "9
-15579"*
38       ??700009
39       CHANGE DR-FILE 'EW•3295•' %55 TO "9BC2D-A06-8-22P"*
40       ??700010
41       LIST CONT-FILE '9-15579'*
42       ??700007
43       ADD TO CONT-FILE '9-15579'*
44       ??700039
45       ADD TO DR-FILE 'EW•3295•' %32 "9-15579"*
46       ??700027
47       LIST DR-FILE 'EX•0123•F' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
48        S-1 S-2 S-3 S-4 S-5 S-6 CONT*
49       ??700039
50       LIST DR-FILE 'EX•0143•I' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
51        S-1 S-2 S-3 S-4 S-5 S-6 CONT*
52       ??700024
53       LIST DR-FILE 'EX•0143•K' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
54        S-1 S-2 S-3 S-4 S-5 S-6 CONT*
55       ??700027
56       CHANGE DR-FILE 'EX•0143•K' %32 TO "9-13247"*
57       ??700023
58       LIST DR-FILE 'EX•0170•' PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A-
```

```
D
    59      S-1 S-2 S-3 S-4 S-5 S-6 CONT#
    60     ???00008
    61     LIST DR-FILE ´EX•0143•J´ PR-NO RFP A-1 A-2 A-3 A-4 A-5 A-6 A
-D
    62      S-1 S-2 S-3 S-4 S-5 S-6 CONT#
    63     ???00078
    64     ADD TO DR-FILE ´EX•0143•J´ %53 "78-139-022" %47 "780213" %48
 "780622"
    65      %49 "780722" %50 "780816" %51 "780910" %52 "781010" %88 "76
0503"#
    66     LOGOFF#
    67     @FIN
```

```
POLL*GIM(1).DH
2018/GIM
2018/GIM
@RUN 660DH,FD3/2018,FD6-T61796,25,500
??700020
@QUAL POLL
??700020
@XQT *GIM.GIMCOM
??700020
LOGON 290004*
??700014
ORDER DR-FILE DIV "*1" PWC "*2" INIT$-CUM COMM$-CUM OBLG$-CU
M T-AC$
 PWC-3 WITH FUND-YR1 "75" ANDD WITH PWC-3 "778" FILENAME "DH
"*
??700012
REPORT FORM4 FILENAME "DH"*
??700023
EXECUTE PSR-1 /MK/ VARIABLES "P=77" "V=R"*
??700032
ORDER DR-FILE DIV "*1" PWC "*2" INIT$-CUM COMM$-CUM OBLG$-CU
M T-AC$
 PWC-3 WITH FUND-YR1 "75" ANDD WITH PWC-3 "640" FILENAME "DH
"*
??700013
REPORT FORM4 FILENAME "DH"*
??700013
ADD FORM4 /LINE1/ NEXT /L/*
LOGOFF*
@FIN
>
```

GOES DOWN EARLY

REFERENCES

1. Sperry UNIVAC 1100 Series American National Standard COBOL (ASCII), Chapter 13, OP7923, 1975.

2. Hennigan, K. B., A GPSS Model of UNIVAC 1100 Series Computer Systems, MTR-4590, The MITRE Corporation, March 1976.

3. Fiske, et al, EXEC-8 Performance Evaluation System, MTR-4575, Rev. 1, The MITRE Corporation, June 1975.

4. , Sperry UNIVAC 1100 Series American National Standard COBOL (ASCII), UP7923, 1975.

5. , Detailed Documentation of the ITS Control Program, internal IDSD/FD6 unpublished communication, 1977.

6. , ITS Terminal I/O Handler (TH) Usage Guide, unpublished IDSD/FD6 internal communication, September 1977.

7. Jain, G. P., IDSD FY79 Computer System Plan, The MITRE Corporation, MTR-4707, September 1977.

8. Kirbie, L. R., "GIM Performance Testing", JSC Letter 78-FD64-147, 20 June 1978.