AN IMPLEMENTATION OF A BUDGETARY SYSTEM FOR THE COLLEGE OF ARTS AND SCIENCES

by

WILLIAM PETER WEBER

B. S., Kansas State University, 1970

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1976

Approved by:

W J Hankley
Major Professor

To my Wife
Barbara

THIS BOOK CONTAINS NUMEROUS PAGES WITH DIAGRAMS THAT ARE CROOKED COMPARED TO THE REST OF THE INFORMATION ON THE PAGE.

THIS IS AS RECEIVED FROM CUSTOMER.

ILLEGIBLE

THE FOLLOWING DOCUMENT (S) IS ILLEGIBLE DUE TO THE PRINTING ON THE ORIGINAL BEING CUT OFF

ILLEGIBLE

TABLE OF CONTENTS

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I wish to take this space to acknowledge my major professor and the members of my committee. I extend to Dr. Fred J. Maryanski my deepest appreciation for his help and inspiration during the design and development of this project. A special thanks is extended to Dr. Paul S. Fisher who is responsible for providing me with this project. For their supportive efforts I extend my thanks to Dr. William J. Hankley and Dr. Virgil E. Wallentine.

LIST OF FIGURES

Figure

Figure

Figure

CHAPTER I

INTRODUCTION

1.1 OBJECTIVE:

For several years now, the College of Arts and Sciences at Kansas State University has been in need of an automated budget information system. Such a system would simplify the process of correcting and maintaining the constantly changing budgetary data (ie. tenths, dollars, annual salary, etc.). Currently this information must be kept by 1) the department, 2) the dean's office, 3) the personnel office, 4) the budget office, 5) the Vice-president for Business Affairs, and 6) the Board of Regents in the form of a computerized listing produced annually by the Data Processing Center in Anderson Hall during the month of April. Any budgetary changes made within the fiscal year must be relayed by the department to the dean's office and the personnel office. Each department maintains its own documentation on the change. The dean's office must file its portion of the documentation and record the changes either by writing on the annual budget report or as an entry into a special ledger. Upon receipt of its copy of the document, the personnel office follows its own procedures for processing the change.

The document flow requires many manual procedures to effect a change in the budget. This often results in human error. It is especially important that the information be correct at the close of a fiscal year. Each department head must know precisely the number of dollars and tenths remaining in his/her budget. As is the case in some instances; the department, the dean's office, and the personnel office have different views of the final budgetary information. Often there are discrepencies in such data as annual salaries, annual tenths, monthly basis, and source of funding.

Many of these problems could be eliminated by a computerized system which could provide a current version of the budget, either upon demand or at specified intervals. Such a report could be distributed to the departments under the College of Arts and Sciences and the personnel office. Thus, any discrepencies could be quickly corrected. The dean's office could provide immediate status of free and assigned tenths and dollars to each department. This would aid in rapid redistribution of tenths and/or dollars within a department or between departments. This would result in the maximum utilization of tenths and dollars appropriated by the Kansas legislature.

It is the intent of this master's project to provide a computerized budgetary system which will provide a consistent form of maintaining budgetary data, to be able to produce a report which will provide a current view of the annual budget, and provide a method of summarizing budgetary information within each department by fund source.

A previous study on the feasibility of the budgetary system described above[1] suggested four short range objectives. They are:

1) to provide quick and easy access to current budget information,

2) to provide for a uniform method of maintaining budget information,

3) to provide the dean's office and each of its departments with current budget information, and

4) to provide statistical information for planning purposes.

The goal of this project is to implement a system which will satisfy the suggested short range objectives of the dean's office.

CHAPTER II

PROCEDURAL ANALYSIS

2.1 CURRENT PROCEDURES:

It is necessary to analyze current procedures in order to determine the effects, if any, of an automated system upon them. The purpose here is to establish the flow of documentation and computer generated output between the dean's office and other sources. The scope of this chapter is to describe only those sources which have a direct impact upon the dean's office.

Budgeting for an upcoming fiscal year usually begins during the months of January and February, concluding during the middle of February. It is the responsibility of the department head to prepare the budget for his/her department. This procedure requires an accurate estimate of employee salaries, including raises, and other operating expenses for the upcoming fiscal year. These estimates are then sent to the dean's office. After all of the department heads in the College of Arts and Sciences have submitted their preliminary budget requests, the deans begin an in depth study of the combined budgets. This usually begins at the first part of February and lasts for several weeks. The basis for budgeting is the anticipated appropriation of funds and tenths from the Kansas legislature. In the recent past this has been approximately a ten percent increase per year. However, the main reason for making an approximation at this point in time is the fact that the legislature will not finalize its budget until April. Only at that time will the colleges and departments receive their appropriations.

During the first part of February the Data Processing Center produces a deck of data cards which contains all of the budgetary information from the previous fiscal year. This information is broken down into positions or lines each of which may represent an unclassified employee, a classified employee, or an operating expense. A line or position may be represented by one or more data cards. If the line represents a faculty or staff position, several cards may be required to describe the position and the services performed by that individual. Otherwise, if the line is an operating expense, one card will contain all data for that line.

During the latter part of February the deans will be completing the combined budget. If there are any changes in the budgetary information from the previous fiscal year, the dean's secretary notes them on cards within the deck. Additions are made by filling out data cards and inserting them into prescribed positions. Changes are made by marking fields that must be changed on designated cards. Upon completion of this task the dean's secretary returns the deck to the budget office.

Updating the budget deck is a manual process, a task performed by personnel in the budget office. Changes to lines involve the physical removal of designated cards, keypunching of replacements, and the insertion of new cards into their proper locations in the deck. Addition of lines involves keypunching new cards and their insertion into the deck. Lines may be dropped by manually removing the designated cards from the deck. After the update process has been completed, the deck of cards is sent from the budget office to the Data Processing Center.

After receiving the cards, the center goes through a procedure which loads these cards onto a tape file. This process includes copying the card in its original format and appending additional information generated from the contents of the card. This information will be used in later processing such as sorting. However, during the initial phase of the procedure, this tape is used to produce the 'proof' copy of the annual budget report, which is returned to the budget office for distribution to the colleges and departments.

In the dean's office the proof copy is examined. If there are any corrections required, they are submitted to the budget office. Any corrections will be made manually by personnel in the budget office as described previously. The updated version of the budget deck is then returned to the Data Processing Center to produce 'hearing' copies of the annual budget. These copies are so named because they are used by the President, the Vice-president for Business Affairs, the Board of Regents, and the college deans in the preparation of the final budget for the fiscal year during a special hearing. Any changes to the budget at this time are submitted to the Data Processing Center through the budget office. Upon final approval from the budget office, the final annual budget report is produced in the manner described previously. Copies are distributed to the President, the Board of Regents, and the Vice-president for Business Affairs. Each college receives two copies of its budget, and each department a single copy. After the final distribution, no further budgeting is done until the following year. At that time the above procedure is repeated.

A difficulty arises during a fiscal year when it becomes necessary to make changes to the existing budget. This happens, for example, when an employee resigns or a new graduate assistant is hired. At present, each department is responsible for maintaining its own budgetary information and informing the dean's office of these changes. The current procedure requires that the department send information to the dean's office on one of two documents, the Source of Salary document and the Graduate Student Transaction document. See sample documents in Appendix A. In either instance copies must be submitted to the Vice-president for Business Affairs and the personnel office. Information obtained from these documents is used by the dean's secretary to make manual entries into a special ledger (See sample ledger page in Appendix A) or directly onto the annual report. This tedious manual procedure may be repeated several times a week. Because of this, there are instances during which the information retained by the department, the dean's office, and the personnel office is not consistent. When this happens, special meetings must be scheduled to rectify the discrepencies, resulting in time and money lost to all concerned.

2.2 EFFECTS OF AUTOMATED SYSTEM ON CURRENT PROCEDURES:

It has been agreed that the automated budgetary system augment the existing system. The annual procedures will remain the same. The reason for this is that the existing procedure is a University standard and should not be modified for the sake of one college. Modification will occur when the Data Processing Center is requested to create a special tape containing budgetary information only for the College of Arts and Sciences. From it the data base will be recreated.

This data base will be maintained by the dean's office. From it reports can be printed and distributed to the departments as required. Any discrepencies between the dean's office and the department can be quickly noted and corrected.

Although the proposed automated budgetary system is minimal, it will greatly aid the dean's office by eliminating some of the tedious tasks now performed by the dean's secretary. Effects on the existing system will be minor.

CHAPTER III

PROBLEM SPECIFICATION

3.1 SCOPE:

Requirements for a minimal budgetary system are a group of interrelated programs which can insure complete integrity for the whole system and yet allow the user complete control over all data within that system. That is, the user is allowed to modify any data within the scope of his/her system without inadvertently creating erroneous information, incorrectly modifying existing information, or deleting information which would destroy the integrity of other dependent data within the system. In order to provide this freedom, each program within the system must contain checks and validation procedures which must be executed before any changes to the system are allowed.

Four basic data structures are necessary for the complete description of this system. They contain information related to 1) the college, 2) the department, 3) source of funding, and 4) the budget. Programs are to be specified so that each maintains only a single data structure. This not only simplifies the function of each program, but it also provides a degree of security to the system by the fact that certain restrictions can be placed on the individuals allowed access to any given program. Further, by defining each program so that it updates only one logical structure, only one program need be modified in the event that a data structure changes.

3.2 REQUIREMENTS:

The purpose of this section is to state the general function of each program within the budget system. The information to be presented here will give an overview of each program without describing specific details.

All college related data will be maintained by one program. The information contained within the college data structure has a two-fold purpose, that of providing descriptive titles for report headings and providing codes needed to validate data within the funding and department data structures. Provisions must be made for adding, changing, and deleting any or all college information within any college data structure.

Departmental data will be maintained by another program. It will be necessary to refer to the contents of the college data structure to verify departmental activity codes, the basic data field needed to uniquely identify any department within a college. In addition to activity data, this program will maintain descriptive titles for each department for reporting purposes. Provisions must also be made for adding, changing, and deleting any or all departmental information within any departmental data structure.

Fund source data is maintained by a third program. Logically fund sources are independent of college and department. But for the sake of this implementation, funding will be bound to a single college, the College of Arts and Sciences. Information contained within the funding structure will be used to enhance the fund title and to validate budgetary data during budget updates. The contents of the fund title will be used to provide descriptive headings for report purposes. Provisions must also be made for adding, changing, and deleting any or all information contained within any funding data structure.

The most important of this series of programs is the program which actually maintains the budgetary data. It must use information from departmental and funding data structures to validate, identify, and categorize budgetary data structures. This program must have the ability to add, change, and delete within specified limits any or all information contained within any specified budget data structure.

The contents of the budgetary data structure must be able to uniquely identify each line within a department and provide a reference to the source of funding. It must contain information which will indicate whether the tenths and dollars budgeted for that line or position are assigned (in use) or are unassigned (free).

The method used for updating budgetary information is through the use of transaction codes. These are one position alphabetic characters which represent an abbreviated verbal command. These are:

A An add transaction. Budgetary information for a line does not exist and is to be created for the first time.

C A change transaction. Budgetary information exists and the user wishes to alter the contents of a budgetary data structure, thus modifying the line.

D A delete transaction. The user wishes to destroy a specified line, thus removing it from the budget.

L A loan transaction. This transaction causes a transfer of tenths and/or dollars within a department or between departments by a specified fund source.

M A maintenance transaction. This transaction is provided for correction purposes. It exists in case the user correctly updates tenths and/or dollars for a specified line, but for some reason the fund source(s) for the department are incorrectly updated. An event such as this could happen during a software failure.

Another requirement for this program is that it be able to correctly identify a status change for an employee line and update the dollars and/or tenths for the department by fund source. Status codes and the actions evoked by the program are listed below.

The following codes cause a line to be budgeted and its tenths and dollars to be identified as assigned or in use.

21 filled full time continuing position,
51 filled part time continuing position,
22 a promotion,

26 filled full time new position created by Board of Regents,
56 filled part time new position created by Board of Regents,
27 filled full time new position created since previous budget,
57 filled part time new position created since previous budget,
19 complete overlaps,
59 part time overlaps,
00 students,
60 part time GRA's, GTA's, etc.

In the case of an add transaction the following codes cause a line to be budgeted and its tenths and dollars to be identified as free or unassigned. If the following represent a change in the employee's status, any unused tenths and/or dollars are returned to the department as being unassigned.

31 vacant full time continuing position,
61 vacant part time continuing position,
36 vacant full time new position created by Board of Regents,
66 vacant part time new position created by Board of Regents,
37 vacant full time new position created since previous budget,
67 vacant part time new position created since previous budget.

The following status codes represent two different types of leaves. Their effects are to cause all tenths and part or all of the dollars for a specified line to be declared as unassigned.

28 full time leave (sabbatical). Action: If the leave is to extend for over one-half budgeted period specified by the monthly basis, one-half of the dollars are specified as unassigned, the other half as assigned.

58 part time leave (leave without pay). Action: All unused dollars are specified as unassigned. Unused portion determined through calculation.

An additional requirement for the budgetary system is necessary. Initially all budgetary information must be created and loaded onto some sort of file. The program which maintains all budgetary data structures may be modified to fulfill this need. The dean's secretary has requested that as part of the system, there be a provision for recreating all budgetary information annually. The initial load program must therefore be written to perform this function as well as the initial load function.

In addition to programs which perform a maintenance function the dean's secretary has requested two print programs. The first should print a listing of all lines or positions within a department and provide totals of assigned and unassigned tenths and dollars. The second is to provide a summary of tenths and dollars assigned, unassigned, and allocated by fund source within each department. Both programs will be able to print information for all as well as selected departments.

3.3 PERFORMANCE CONSIDERATIONS:

The four data structures mentioned in the previous section must be represented as records within a file. In this section it will be necessary to consider how these structures are to be represented within one or more files. Considerations upon the choice of how the data is to be represented must be made based upon resources currently available at Kansas State University. Currently available are the access methods supplied by IBM for standard file management systems and a vendor supplied system marketed by Cullinane Corporation for data base management systems.

Access methods which could be used with the proposed budget system are the Queued Sequential Access Method (QSAM) and the Basic Indexed Sequential Access Method (BISAM). If a file management system were to be implemented, both of these methods would be used to obtain the optimum performance in data retrieval and update. It would be necessary to store all college, department, and funding structures on direct access storage devices in order to provide random retrieval of records, to validate budgetary data, and to maintain current and easily accessable information related to dollars and tenths. The budgetary data structures must be maintained on a standard sequential tape file on account of the unpredictable fluctuations in the number of records and the size of the file. The latter requirement is made because large amounts of disk space could prove cost prohibitive. It would be necessary to maintain additional data fields within the budget structure to identify the line's department and source of funding. Further, if a line contained more than one service, all information needed to identify that line would have to be duplicated for each service record.

A data base management approach to this same problem would be somewhat different. The first three data structures once again could be represented as records within a file, but in this case, the records could all reside within the same file as compared to three files. However, the structures for the department and fund source can now be reduced in length because redundant data fields may be removed. A new structure called the department-fund junction record could be created to combine overlapping data of department and fund source records.

Budgetary data in a data base management system must reside on a direct access device. However, with a data base management system, the actual data structures and the number of records would be reduced drastically because the redundant information required for a file management system would be eliminated. That is, fund and activity codes need not be retained in all structures, and redundant line information is not needed to identify services for a given line. This fact alone would be enough to rationalize using a data base management system. However, the following chart comparing the data base management system to a file management system for this implementation will shed further light on differences in the two approaches. See figure 3.1.

The file management system requires more resources, a disk drive and two tape drives, whereas the data base management system requires only one disk drive. Looking at all of the records collectively and comparing total

record type	number of occurrences	bytes per record	total bytes	device type	space required (tracks)	access method
FILE MANAGEMENT SYSTEM						
college	1	46	46	3330	38*	BISAM
department	24	61	1464	3330	38*	BISAM
fund source	27	51	1377	3330	38*	BISAM
budget	1442	85	122570	2400	--**	QSAM
TOTAL			123947		114	
DATA BASE MANAGEMENT SYSTEM						
college	1	60	60	3330	.004	BSAM
department	24	72	1728	3330	.136	BSAM
fund source	27	72	1944	3330	.154	BSAM
junction	51	104	5304	3330	.406	BSAM
page management	2	3156	6312	3330	.500	BSAM
space management	1	3156	3156	3330	.250	BSAM
budget	481	192	92352	3330	7.316	BSAM
TOTAL			110856		8.776	

* BISAM files always require allocation of tracks in units of cylinders for index and prime data areas. Thus two cylinders are required for each data set.

** two tape drives are needed at execution time, one for the old master file and one for the new master.

FIGURE 3.1 Disk space requirements for file management versus data base management systems

byte usage, the data base management system does produce a space savings due to the elimination of redundancy of budget records. The data base management approach allows service records to be combined with the budget record information for each line, whereas in the file management system all budget information must be repeated for each service record. For this reason the budget record in the data base management system will be larger than in the file management system, however, there will be fewer records. In addition, each record is increased slightly in size on account of the overhead needed to maintain pointers.

In the data base management system one should include the space required for the data dictionary because it must reside at least temporarily on direct access storage. It is, however, not required for the execution of programs. Therefore, after all programs have been compiled and link-edited, the data dictionary may be dumped onto a tape file. It may then be restored as required at some later date. The main point to be made here is that the data dictionary will be a prime cost factor if it is allowed to reside on disk storage. Once the software system begins to require minimal use of the dictionary, it may be retired to a tape file which costs much less.

The file management system requires much more disk space than the data base management system. However, if a comparison is made on the basis of utilization of that space, the file management system wastes a tremendous amount. The file management system would require three BISAM files. Each would require one cylinder of space for an index and a minimum of one cylinder for the primary data area. Thus, the total space required for this type of an application would require 114 tracks. Of this approximately six tracks would contain any information at all. On the other hand, the data base management system allows all record occurrences, including budget, to reside within a single file. All occurrences could easily reside within a space of nine tracks. This would include all overhead space required for page and space management. There is little wasted space when compared to that required by the file management system.

Comparisons between the two systems can be made on the basis of access methods. The file management system used QSAM and BISAM. In QSAM all of the records that are stored on the tape file must be accessed serially to find any given budget record for update purposes. Updating a QSAM file requires that all input records be accessed and copied onto an output QSAM file and reside there along with all added and updated records. This access method does allow very rapid access to records since queuing takes place. The three BISAM files provide access to the desired records based on the contents of a special key which is used to match its contents with those of another key located within the record itself. This too is a serial process and is not overly efficient.

The data base management system on the other hand uses BDAM, an extremely efficient access method which locates the desired record directly by accessing the desired track and the desired record within that track. With this system the data base administrator has special control over how records are physically stored on the data base. Special storage methods can be utilized to cause related records to reside physically close to each other. Therefore, if the user desired to update a budget master record, the program could cause access to the data base which would not only retrieve the desired budget record but also the department and department-fund junction record occurrences needed for the update as well. Design considerations of this sort often reduce access to the file to less than one access per updated record. A data base management system has an added advantage over the file management system in that it significantly reduces the number of input/output requests needed by the user program.

Both a file management and a data base management system can provide security to the user's data. The most basic method which can be employed by both systems at Kansas State University is limiting of access to any data set by an account number. Further, a special pass letter can be appended to the beginning of the account number to give an added degree of security. Both systems could employ IBM supplied utilities to generate pass words which would have to be validated by the computer operator before access could be granted. The programs themselves could be written in such a manner that access to the program itself would be granted only if the user could encode a special pass word or was residing at a terminal of a prespecified address. There are other forms of security which can be employed, but the above would be most realistic and least restrictive.

A data base management system provides even more security to the user than that mentioned in the previous paragraph. The system currently operative at Kansas State provides for the description of the entire data base by a schema. This schema specifies the files and areas on which data is to reside. The schema can be further subdivided into subschemas which specify the manner in which areas may be accessed, the kinds of record types and set relationships allowed access, the number of fields within a specified record type allowed access, and the types of data manipulation commands allowed against each record type and set relationship. All accesses to the data base must be through user programs. But before user programs can access the data base at all, they must have been successfully compiled and link-edited. The data base management system requires that all programs be processed by a special preprocessor before compilation takes place. If a security violation is detected at this point, all processing of the job stream is aborted. Further, if the compilation is successful, but an area is opened with an incorrect usage mode or an incorrect data manipulation (DML) command is issued, the process is aborted.

Security can be applied in another sense. That is, it can be used to protect the user from himself. With the file management system special validation procedures and invalid key conditions could be employed to prevent erroneous update of a file. The data base management system allows for these also. However, many of the validation procedures which would normally have to be coded by a programmer already exist in the data base management approach. By checking a special return code the programmer can insure that duplicate records are not added to the data base, that records do exist before they are modified or deleted. What's more, this type of security can be provided in a multiprocessing environment.

3.4 FEATURES OF IDMS:[2,3]

This section describes the basic features of the Integrated Database Management System (IDMS). IDMS is a software system marketed by the Cullinane Corporation. As implemented to date, it is a subset of the CODASYL* Database Task Group language specifications. Special features

*CODASYL (Conference on Data Systems Languages) was established as an informal and voluntary orgainzation of interested individuals, supported by their institutions, which contribute their efforts and expenses towards the ends of designing and developing techniques and languages to assist in data systems analysis, design, and implementation.[2] Through the efforts of a special task force within this committee, the Data Base Task Group (DBTG), a data definition language and a data manipulation language were developed to extend American National Standard Cobol capabilities. It is from the special reports produced by this task group that Cullinane Corporation is developing IDMS.

added by Cullinane Corporation allow IDMS to be used with any host language which processes CALL statements or their equivalent and operate under IBM 360 or 370 OS/DOS/VS or UNIVAC SPECTRA TDOS/VMOS environments.

IDMS provides data and program independence, in that, separate language facilities are provided for the description and the manipulation of data. That is, all data and data relationships are combined into a data base which is common to all application programs which it uses.

Data description within IDMS is done to provide a complete description of the data base and to describe only portions of the data base to one or more separate application programs. While there is only one complete description of the data base, there may be many sub-descriptions, each of which describes only the specific combination of records, sets, and areas that are needed by any given application program.

IDMS provides the data base administrator the ability to specify the physical placement of records on the data base in order to optimize performance. Special storage techniques place records on the data base in a manner that eliminates the need for periodic reorganization of the data base. Since IDMS is a network type data base management system, any record may be stored as an entry point into the data base.

Special utilities are provided which monitor data base performance and storage density. Rollback and recovery utilities provide added security in the event of software or hardward failure. Yet another utility provides for the listing of the contents of the Data Directory.

3.5 IDMS DEFINITIONS:[2,4]

The DATA DESCRIPTION LANGUAGE (DDL) is a language used to describe a data base. These descriptions are in terms of names and characteristics of elements included within a data base, and the relationships between occurrences of those elements.

A DATA-ITEM is the smallest unit of named data. An occurrence of a data-item is a representation of a value.

A DATA-AGGREGATE is a named collection of data-items within a record. There are two types, vectors and repeating groups. A vector is a one-dimensional. ordered collection of data-items, all of which have identical characteristics. A repeating group is a collection of data that occurs an arbitrary number of times within a record occurrence. The collection may consist of data-items, vectors, and repeating groups.

A RECORD is a named collection of zero, one, or more data-items or data-aggregates. There may be an arbitrary number of occurrences in the data base of each record type specified in the schema for that data base. One must distinguish between an actual record occurrence of a record and the type of the record. The contents of a record described by a given structure constitues the occurrence of that record. The actual description of the structure constitutes the type of the record.

A SET is a named collection of record types. As such, it establishes the characteristics of an arbitrary number of occurrences of the named set. Each set type specified in the schema must have one record type declared as its OWNER and one or more record types declared as its MEMBER records. Each occurrence of a set must contain one occurrence of its owner record type and may contain an arbitrary number of occurrences of each of its member record types.

An AREA is a named logical sub-division of storage space in the data base and may contain occurrences of records of various types. Areas may be opened by a run-unit or program with USAGE MODES which permit, or do not permit, concurrent run-units to open the same area.

The concept of AREA allows the data base administrator the ability to subdivide a data base rather than considering it as a single unit. The use of areas allows the data base administrator or the data base management system (DBMS) to control placement of an entire area to provide efficient storage and retrieval. The opening of areas by run-units also gives an opportunity to optimize access to the data base since the run-unit has narrowed the range of interest in the data base to a relatively small number of subdivisions of the entire data base. Areas are a convenient unit of recovery, as duplication or backup can be carried out selectively. Areas also provide a convenient natural subdivision for allowing certain unused portions of the data base to be saved in archival storage while the remainder of the data base is actively accessed.

A FILE is an extent of addressable secondary storage known to the operating system. A file may be equal in extent to an area, a portion of an area, or may even contain several areas.

The concept of files allcws the data base administrator to collect or distribute logical areas to physical files as desired for device allocation requirements, backup, and recovery.

A DATA BASE consists of all the record occurrences, set occurrences, and areas controlled by a specific schema. The data base resides on physical files which are divided into logical areas. If an installation has multiple data bases, there must be a separate schema for each data base.

A SCHEMA consists of DDL statements and is a complete description of a data base. It includes the names and descriptions of all of the areas, set occurrences, record occurrences, and associated data-items and data-aggregates as they exist in the data base.

A SUBSCHEMA consists of separate DDL statements from the schema. The subschema DDL need not describe the entire data base but only those areas, data-items, data-aggregates, records, and sets which are known to one or more specific programs. Further, it describes them in the form in which they are known to those specific programs.

The DEVICE-MEDIA CONTROL LANGUAGE (DMCL) consists of separate DDL statements from the schema and subschema DDL. The DMCL determines the areas of the data base and their associated files that are to be available to the data base management routines.

The DATA MANIPULATION LANGUAGE (DML) is the language which the programmer uses to cause data to be transferred between the program and the data base. The DML is not a complete language in itself, but relies on a host language to provide a framework for it and to provide the procedural capabilities required to manipulate data.

CHAPTER IV

DATA DEFINITION LANGUAGE
DESIGN SPECIFICATION

This chapter presents a solution to the problem described in chapter three in terms of data structures and data relationships. Section 4.1 describes the schema for the data base in terms of areas, records, sets, and data-items. Development of the schema is approached first from a structural point of view by describing each record type within the schema in terms of data-aggragates and data-items. A logical description is then presented by explaining the set relations which exist among records within the schema. Section 4.2 describes the six subschemas required for this application. Included is a discussion on the security provided by each subschema. Section 4.3 provides an overview of the DMCL and the characteristics of the DMCL described for this application.

4.1 SCHEMA DESCRIPTION:

The schema (See figures 4.1 and 4.2) describes a data base that will store all occurrences for college (C-COLLEGE-REC), department (D-DEPT-REC), fund (F-FUND-REC), department-fund junction (DF-DEPT-FUND-REC), and budget (B-BUDGET-REC) record types. Occurrences of college, fund, and department record types are stored using the CALC location mode. This option stores and retrieves records based upon the contents of a special key (data-item or group-item) contained within the record occurrence itself. The DBMS performs a mathematical transformation upon this key to produce a logical storage position within the area. The DBMS has been designed so that records using this location mode are distributed uniformly throughout the data base area. The CALC location mode makes each occurrence of these three record types an entry into the data base. This is highly desirable from the point of updating and retrieving

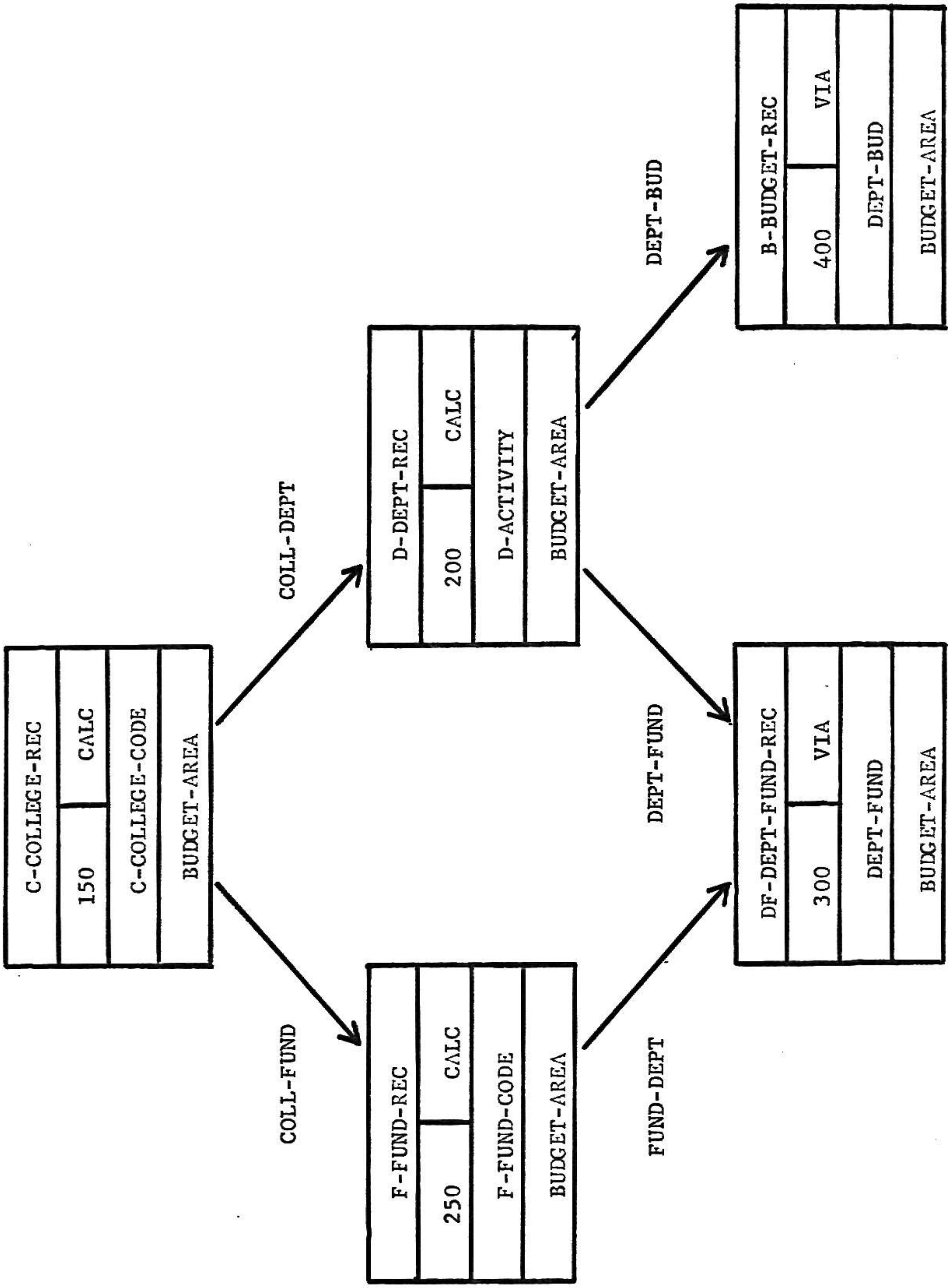


FIGURE 4.1 Schema Data Structures

```
SCHEMA DESCRIPTION.
SCHEMA NAME IS BU01SCHM.
AUTHOR. BILL WEBER.
DATE. 09/16/75.
INSTALLATION.  KANSAS STATE UNIVERSITY
               DEPARTMENT OF COMPUTER SCIENCE.

FILE DESCRIPTION.
    FILE NAME IS BUDGET-FILE
        ASSIGN TO BU01BDGT.
    FILE NAME IS JOURNAL
        ASSIGN TO BU01JRNL.

AREA DESCRIPTION.
    AREA NAME IS BUDGET-AREA
        RANGE IS 1001 THRU 1076
        WITHIN FILE BUDGET-FILE
        FROM 1 THRU 76.

RECORD DESCRIPTION.

    RECORD NAME IS C-COLLEGE-REC
    RECORD ID IS 150.
    LOCATION MODE IS CALC
        USING C-COLLEGE-CODE
        DUPLICATES ARE NOT ALLOWED.
    WITHIN BUDGET-AREA AREA.
    03  C-COLLEGE-CODE              PIC XXX.
    03  C-COLLEGE-TITLE             PIC X(40).

    RECORD NAME IS D-DEPT-REC.
    RECORD ID IS 200.
    LOCATION MODE IS CALC
        USING D-ACTIVITY
        DUPLICATES ARE NOT ALLOWED.
    WITHIN BUDGET-AREA AREA.
    03  D-ACTIVITY.
        05  D-COLLEGE-CODE          PIC XXX.
        05  D-DEPT-CODE             PIC XX.
    03  D-DEPT-TITLE                PIC X(40).

    RECORD NAME IS F-FUND-REC.
    RECORD ID IS 250.
    LOCATION MODE IS CALC
        USING F-FUND-CODE
        DUPLICATES ARE NOT ALLOWED.
    WITHIN BUDGET-AREA AREA.
    03  F-FUND-CODE.
        05  F-FUND-ACTIVITY         PIC XX.
        05  F-FUND-TYPE-ACCT        PIC X.
        05  F-FUND                  PIC XX.
    03  F-FUND-TITLE                PIC X(40).
```

FIGURE 4.2 Schema Description Statements.

```
RECORD NAME IS DF-DEPT-FUND-FEC.
RECORD ID IS 300.
LOCATION MODE IS VIA DEPT-FUND SET.
WITHIN BUDGET-AREA AREA.
03  DF-FUND-CODE                PIC X(5).
03  DF-ACTIVITY                 PIC X(5).
03  DF-USED-DOLLARS             PIC S9(9)    COMP-3.
03  DF-FREE-DOLLARS             PIC S9(9)    COMP-3.
03  DF-LOAN-DOLLARS             PIC S9(9)    COMP-3.
03  DF-BORR-DOLLARS             PIC S9(9)    COMP-3.
03  DF-09-USED-TNTHS            PIC S9(5)    COMP-3.
03  DF-09-FREE-TNTHS            PIC S9(5)    COMP-3.
03  DF-09-LOAN-TNTHS            PIC S9(5)    COMP-3.
03  DF-09-BORR-TNTHS            PIC S9(5)    COMP-3.
03  DF-12-USED-TNTHS            PIC S9(5)    COMP-3.
03  DF-12-FREE-TNTHS            PIC S9(5)    COMP-3.
03  DF-12-LOAN-TNTHS            PIC S9(5)    COMP-3.
03  DF-12-BORR-TNTHS            PIC S9(5)    COMP-3.

RECORD NAME IS B-BUDGET-REC.
RECORD ID IS 400.
LOCATION MODE IS VIA DEPT-BUD SET.
WITHIN BUDGET-AREA AREA.
03  B-BUDGET-KEY.
    05  B-OBJECT-CODE           PIC X.
    05  B-RANK-CODE             PIC X(4).
    05  B-LINE-NO               PIC X(3).
03  B-STATUS-CODE               PIC XX.
03  B-MONTHLY-BASIS             PIC 99       COMP-3.
03  B-PAY-PLAN                  PIC X.
03  B-NAME                      PIC X(20).
03  B-SOC-SEC-NO                PIC 9(9)     COMP-3.
03  B-ANNUAL-TENTHS             PIC 99       COMP-3.
03  B-ANNUAL-SALARY             PIC S9(5)    COMP-3.
03  B-BUDGET-DOLLARS            PIC S9(5)    COMP-3.
03  B-BEGIN-DATE                PIC 9(6).
03  FILLER REDEFINES B-BEGIN-DATE.
    05  B-BEGIN-MO              PIC 99.
    05  B-BEGIN-DA              PIC 99.
    05  B-BEGIN-YR              PIC 99.
03  B-END-DATE                  PIC 9(6).
03  FILLER REDEFINES B-END-DATE.
    05  B-END-MO                PIC 99.
    05  B-END-DA                PIC 99.
    05  B-END-YR                PIC 99.
03  B-SERVICE-INFO
    OCCURS 10 TIMES
    INDEXED BY B-INDEX.
    05  B-SERVICE-KEY.
        07  B-SERVICE-CODE      PIC X.
        07  B-FUND-CODE         PIC X(5).
    05  B-USED-TNTHS            PIC S99      COMP-3.
    05  B-FREE-TNTHS            PIC S99      COMP-3.
    05  B-USED-DOLLARS          PIC S9(5)    COMP-3.
    05  B-FREE-DOLLARS          PIC S9(5)    COMP-3.
```

FIGURE 4.2 Schema Description Statements (continued)

```
        05  B-SOURCE-KEY.
            07  B-SOURCE-OBJECT       PIC X.
            07  B-SOURCE-RANK         PIC X(4).
            07  B-SOURCE-LINE         PIC XXX.

SET DESCRIPTION.

    SET NAME IS COLL-DEPT.
    ORDER IS SORTED.
    MODE IS CHAIN LINKED TO PRIOR.
    OWNER IS C-COLLEGE-REC
        NEXT POSITION IS 1
        PRIOR POSITION IS 2.
    MEMBER IS D-DEPT-REC
        MANDATORY AUTOMATIC
        NEXT POSITION IS 1
        PRIOR POSITION IS 2
        ASCENDING KEY IS D-ACTIVITY
        DUPLICATES ARE NOT ALLOWED.

    SET NAME IS COLL-FUND.
    ORDER IS SORTED.
    MODE IS CHAIN LINKED TO PRIOR.
    OWNER IS C-COLLEGE-REC
        NEXT POSITION IS 3
        PRIOR POSITION IS 4.
    MEMBER IS F-FUND-REC
        MANDATORY AUTOMATIC
        NEXT POSITION IS 1
        PRIOR POSITION IS 2
        ASCENDING KEY IS F-FUND-CODE
        DUPLICATES ARE NOT ALLOWED.

    SET NAME IS DEPT-BUD.
    ORDER IS SORTED.
    MODE IS CHAIN LINKED TO PRIOR.
    OWNER IS D-DEPT-REC
        NEXT POSITION IS 3
        PRIOR POSITION IS 4.
    MEMBER IS B-BUDGET-REC
        MANDATORY AUTOMATIC
        NEXT POSITION IS 1
        PRIOR POSITION IS 2
        ASCENDING KEY IS B-BUDGET-KEY
        DUPLICATES ARE NOT ALLOWED.

    SET NAME IS DEPT-FUND.
    ORDER IS SORTED.
    MODE IS CHAIN LINKED TO PRIOR.
    OWNER IS D-DEPT-REC
        NEXT POSITION IS 5
        PRIOR POSITION IS 6.
    MEMBER IS DF-DEPT-FUND-REC
```

FIGURE 4.2 Schema Description Statements (continued)

```
        MANDATORY AUTOMATIC
        NEXT POSITION IS 1
        PRIOR POSITION IS 2
        LINKED TO OWNER
            OWNER POSITION IS 3
        ASCENDING KEY IS DF-FUND-CODE
        DUPLICATES ARE NOT ALLOWED.

    SET NAME IS FUND-DEPT.
    ORDER IS SORTED.
    MODE IS CHAIN LINKED TO PRIOR.
    OWNER IS F-FUND-REC
        NEXT POSITION IS 3
        PRIOR POSITION IS 4.
    MEMBER IS DF-DEPT-FUND-REC
        MANDATORY AUTOMATIC
        NEXT POSITION IS 4
        PRIOR POSITION IS 5
        LINKED TO OWNER
            OWNER POSITION IS 6
        ASCENDING KEY IS DF-ACTIVITY
        DUPLICATES ARE NOT ALLOWED.
```

FIGURE 4.2 Schema Description Statements (continued)

budgetary data for selected departments and funds. It also reduces the disk and file accesses required to find and retrieve any record occurrence.

Budget and department-fund junction records are located VIA sets which have the department record as the owner record. These records are stored physically close to the owner records for each set occurrence. Having been stored, a retrieval of a specified owner department record either for update or reporting purposes would also retrieve several budget and junction records. Procedures needed to update a budget record occurrence require that the department-fund junction record be updated as well. Retrieval of both records simultaneously eliminates one disk access which results in a time and cost savings. Another added feature which results from storing records using the VIA storage mode. Member records of a set occurrence are also stored physically close to each other. Thus, when retrieving member records serially for reports a group of already ordered records are brought into main storage with one access to the data base. In summary the VIA access mode provides features of random and serial access at the same time.

Each record type described in the schema must have a record id. This id must be an unsigned integer of three to four digits which ranges in value from 100 to 9999. They must be unique for each record type described. Their primary function is to provide a code to the data base management routines which can be used in place of the actual record name. When a program is compiled, the code is inserted into the working-storage section and is initialized to the value of the record name to which it corresponds. In this manner the record name is linked to linkage storage areas in the data base management routines.

Approaching the description of record types from a top-down point of view requires that the college record type (C-COLLEGE-REC) be described first. Each record occurrence of this type acts as a primary entry point into the data base. Its main purpose is to provide a description of the college for report headings and to provide the first three characters needed to edit and identify all departmental activity codes for the college.

The departmental record (D-DEPT-REC) plays much the same role as the college record type. It too provides a description for each department within the college for report headings. A five position activity code is part of the record structure. The first three characters identify the college, and the second two the department. The activity code is used for editing (ie. checking for errors) during the budget update, to identify each line within the budget by department, and to establish the record as a member of the correct set occurrence.

The fund record type (F-FUND-REC) provides a title for each fund source and a fund code which is used during the budget update to identify and edit fund codes associated with each line of the budget. The fund code is a five position group-item. The first two characters form a code which describes a category of funding activities. The third character describes the type of funding account. The last two characters represent a description of the fund itself. These codes have special meaning to the Kansas legislature and the accounting office, and as such are of significance to this master's project only in that they occupy five bytes of space per record occurrence.

The dpeartment-fund junction record type (DF-DEPT-FUND) describes a multipurpose record. It has been created to eliminate redundant data and to satisfy a many-to-many relationship between the twenty-four (24) departments and the twenty-seven (27) funds to be distributed among those departments. For the department the junction record provides current budgetary information for total tenths and dollars which are allocated.

assigned, and unassigned. Reports which use allocated dollars and tenths distributed by fund require an average access of two junction records per department, whereas an average of twenty accesses to the budget record occurrences per department plus many calculations would be required to provide the same information. For funding information these records provide information by fund distributed by tenths and dollars across departmental boundaries within a given college.

Data-items described within this record include fund and activity codes. These must be provided for ordering of record occurrences and to reduce the number of accesses to owner records just to obtain fund and activity codes.

Each department requires knowledge of tenths and dollars which are assigned and unassigned. Information is required for tenths for both nine and twelve month basis. Data fields must be reserved for dollars, nine month basis tenths, and twelve month basis tenths accounting for amounts and quantities which are assigned, unassigned, loaned, and borrowed. Note that provisions must be made for departments to loan and borrow tenths and dollars.

The budget record (B-BUDGET-REC) contains information related to a line in the annual budget. In order to uniquely identify each line within a department, several fields are grouped together to form a budget key. These are the object code, rank code, and line number. The object code specifies whether the line is classified, unclassified, or an operating expense. The rank code can be translated into the type of expenditure or an employee title within a given department. The line number represents the actual slot or position within the budget.

Associated with each line is a name used to identify an employee or to describe the type of expense. Unclassified and classified employees usually work ten (10) tenths time, whereas part time employees work fewer tenths time. Special lines group large numbers of tenths together. This type of line is used by departments to budget for graduate students. However, the number of tenths will never exceed ninety-nine (99) tenths.

Budgeting is usually based on an annual amount of dollars such as the employee's salary. A pay plan field is also provided for those employees who work on a nine month basis, but wish their annual salary distributed across twelve months. Another field must be included to specify the current budgeted dollars. It is provided for lines which represent a change in annual salary, annual tenths, status, etc. The current budgeted amount will be computed as the product of the annual salary and the number of months employed as computed from the effective dates. Fractional parts of the month are computed as the number of days worked in that month divided by the number of working days. This computation involves a quite lengthy algorithm, so the field has been reserved to eliminate repetitive calculations and to reduce code in report programs.

Certain status codes cause tenths and dollars to be budgeted but not assigned. This occurs when a line has been created and the corresponding position is vacant, when dollars and tenths have been allocated for graduate students collectively, or when a line becomes vacant due to the termination of an employee. For this reason additional fields have been described at the service level to indicate the number of budgeted tenths and dollars actually assigned and unassigned. Note that budgeting is done on an effective date basis, that is, through a beginning and an ending date.

Employees may provide one or more services to the university. For instance, it is possible for a person to work seven tenths time as a counselor and three tenths time as an instructor for a total of ten tenths time. Further, it is possible for each of these services to be funded by different sources. A logical limitation of ten different types of services has been placed on each line. Each service is identified by a key which consists of a service code and a fund code. The service code indicates the type of service being provided.

There are five set relationships described in the schema. See figure 4.2. The COLL-DEPT set provides a relationship between the owner college record and the member department records. This set provides added flexibility to the application in that all department records can be accessed either serially through the college record or individually through the CALC location mode.

The COLL-FUND set joins all member fund records to the owner college record. Again this set allows all of the fund records within a college to be accessed either serially or ramdomly. Even though funds are independent of college, it is logically convenient to divide funds by college on account of the fact that it is the college which will be viewing the activity of its departments.

The DEPT-BUD set is the junction of the owner department record and its member budget records. Budget records contain all information budgeted for a given department.

The DEPT-FUND set is a relationship between the owner department record and the member department-fund junction record. This particular relation provides funding information to each department which may be subdivided into tenths and dollars.

The FUND-DEPT set is a relationship between the owner fund record and the member department-fund junction record. This type of relation will provide departmental information to the college by fund source.

After all record types and set relationships are described, the final details of the schema may be filled in. Of interest here are the statements within the file and area descriptions. Before page ranges can be assigned, the actual space required for the data base must be calculated. See figure 4.3. Requirements for the calculations are the lengths of record types including pointer overhead. All set relationships have both forward and backward pointers. The FUND-DEPT and DEPT-FUND sets both have an additional owner pointer. Page indexing and space management pages must be included in the calculation.

Once the calculations are complete, a convenient page size must be selected. One requirement is that the page must be some multiple of four. It must be large enough to contain a significant number of record occurrences which will be stored using the VIA location mode. Lastly, it must be able to reside on an IBM 3330 disk track without producing an excess of wasted storage. The IDMS users' manual suggests 3156 bytes as a practical page length. It is a practical size for storing a significant number of records. Further, it is divisible by four and exactly four pages will reside on one track without any wasted storage. The calculations show that thirty-seven (37) pages of this size are required. This translates into about nine tracks. There are nineteen tracks on each disk cylinder, so if the remaining ten tracks were to be included in the space allocation, the data set or file would have sufficient room to grow. The small amount of space required for this application will easily allow all records to be stored within one file since no disk head movement is required once the first record has been accessed.

CALCULATION OF AREA SIZE IN PAGES

Step 1: Calculate the number of pages necessary to hold the anticipated number of record occurrences for each record type assigned to the BUDGET-AREA area.

$$N = \frac{(L_1O_1 + L_2O_2 + \ldots + L_5O_5)}{P} \quad \text{ROUNDED UP}$$

Where L = Length of record including pointers
O = Number of occurrences of the record
P = Page size

$$N = \frac{60 \text{ X } 1 + 72 \text{ X } 24 + 72 \text{ X } 27 + 104 \text{ X } 51 + 192 \text{ X } 481}{3156}$$

$$= 33$$

Step 2: Calculate the number of pages necessary to hold the required PAGE indexing (line space inventory).

$$I = \frac{8X(O_1 + O_2 + O_3 + O_4 + O_5) + 32N}{P} \quad \text{ROUNDED UP}$$

Where O = Number of occurrences of the record
N = Result of Step 1
P = Page size

$$I = \frac{8 \text{ X } (1 + 24 + 27 + 51 + 481) + 32 \text{ X } 34}{3156}$$

$$= 2$$

Step 3: Calculate the number of required space management pages.

$$S = \frac{2 \text{ X } (N + I)}{P - 32} \quad \text{ROUNDED UP}$$

Where N = Result from Step 1
I = Result from Step 2
P = Page size

$$S = \frac{2 \text{ X } (33 - 2)}{3156 - 32}$$

$$= 1$$

Step 4: Calculate the total area page requirement.

$$A = N + I + S$$

Where N = Result from Step 1
I = Result from Step 2
S = Result from Step 3

$$A = 33 + 2 + 1$$

$$= 36$$

FIGURE 4.3

4.2 SUBSCHEMA DESCRIPTIONS:

Six different subschemas will be required for the implementation of this master's project. Each subschema will be discussed in terms of record types, set relations, and security.

Subschema BU01SS01 (see figure 4.4). is used for college record (C-COLLEGE-REC) updates only. Any program invoking this subschema has exclusive access to the data base. There are no set relationships included in the description to insure that the entire data base will not be destroyed by a malicious or inadvertant DELETE DML command.

Subschema BU01SS02 (See figure 4.5) is used for programs which update department records only. This subschema allows only the program which invokes it exclusive update rights. Access to the college record is allowed, but no information may be changed or deleted. If department records have member occurrences, this subschema will prevent those records from being deleted. The set required by this subschema is the COLL-DEPT set.

Subschema BU01SS03 (See figure 4.6) is similar to BU01SS02 except that it allows update to fund record (F-FUND-REC) occurrences only. Exclusive update of the data base is allowed to the program invoking this subschema. Access to college records is allowed, but no changes or deletions may be made. Fund record occurrences that have members may not be deleted. One set relation, COLL-FUND, is described.

Subschema BU01SS04 (See figure 4.1) is involed by programs which access the data base exclusively to update budgetary information. All record types described in the schema may be accessed. However, only the budget (B-BUDGET-REC) and department-junction (DF-DEPT-FUND-REC) records may be updated by transactions. All others may be accessed

Data Structure for BU01SS01

C-COLLEGE-REC	
150	CALC
C-COLLEGE-CODE	
BUDGET-AREA	

FIGURE 4.4

Data Structure for BU01SS02

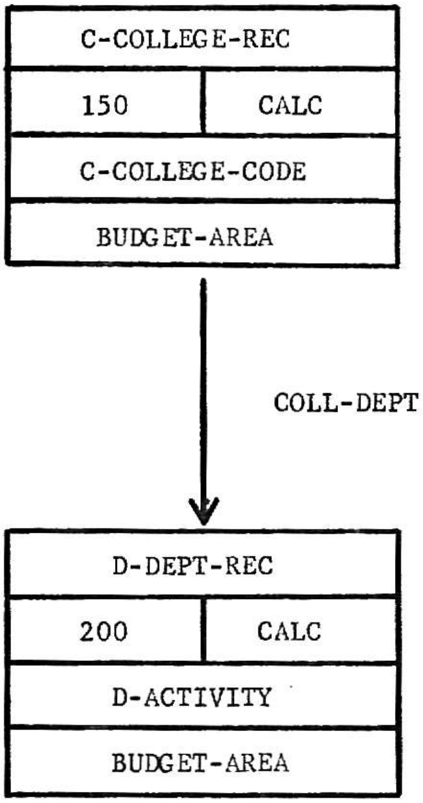


FIGURE 4.5

Data Structure for BU01SS03

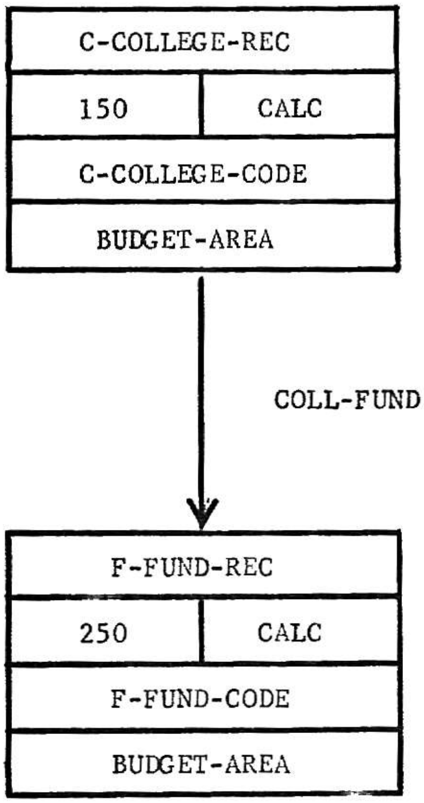


FIGURE 4.6

to provide titles for headings and to validate budgetary data, but their contents may not be changed or deleted. All set relationships described in the schema are required by this subschema.

Subschema BU01SS05 (See figure 4.7) may be invoked by one or programs concurrently to access the data base. Programs which invoke this subschema are allowed to retrieve only college (C-COLLEGE-REC), department (D-DEPT-REC), and budget (B-BUDGET-REC) record types. Information retrieved from the data base is used to produce detail budget reports. Two set relationships, COLL-DEPT and DEPT-BUD, are described.

Subschema BU01SS06 (See figure 4.8) allows access to the data base by one or more programs for retrieval of data. The college (C-COLLEGE-REC), department (D-DEPT-REC), and department-fund junction (DF-DEPT-FUND-REC) record types are allowed access. Information retrieved from the data base through this subschema is useful for producing summary reports by fund source within and department. Two sets, COLL-DEPT and DEPT-FUND, are described. In both BU01SS05 and BU01SS06 changes and deletions of any record occurrences are prohibited.

4.3 DEVICE-MEDIA CONTROL LANGUAGE DESCRIPTION:[3]

The DMCL which when preprocessed, assembled, and link-edited contains the data control blocks (DCB's) necessary to access the area (BUDGET-AREA) copied in the syntax, the necessary mapping tables to determine the physical relative block number corresponding to each logical page number, and the actual buffer pool. Since only one area has been described in the schema (See figure 4.2), one DMCL has been described.

Of primary importance to the DMCL is the page size. Larger page sizes will lower the number of input/output requests. However, too large a page will increase the size of the buffer pool and therefore increase

Data Structure for BU01SS05

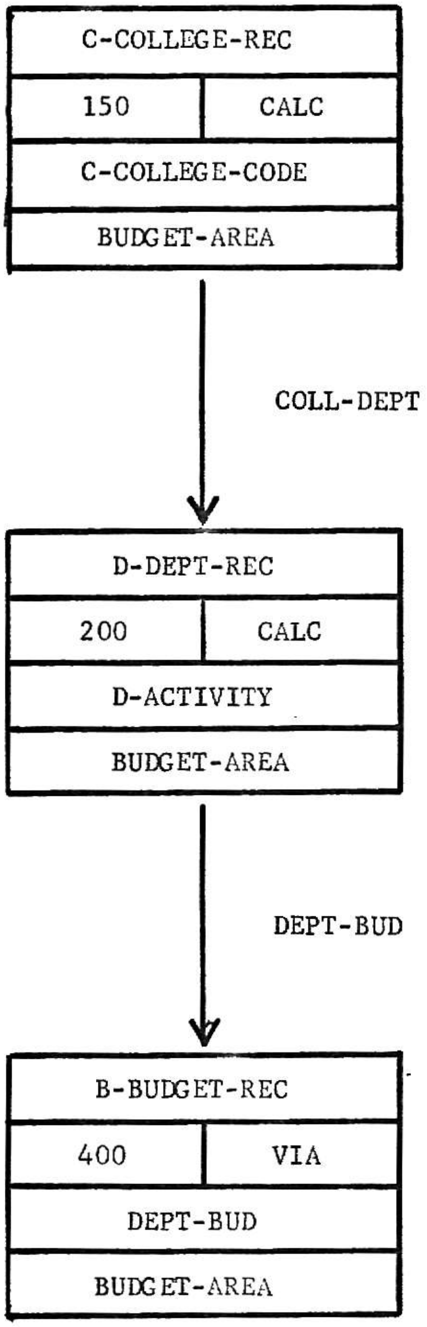


FIGURE 4.7

Data Structure for BU01SS06

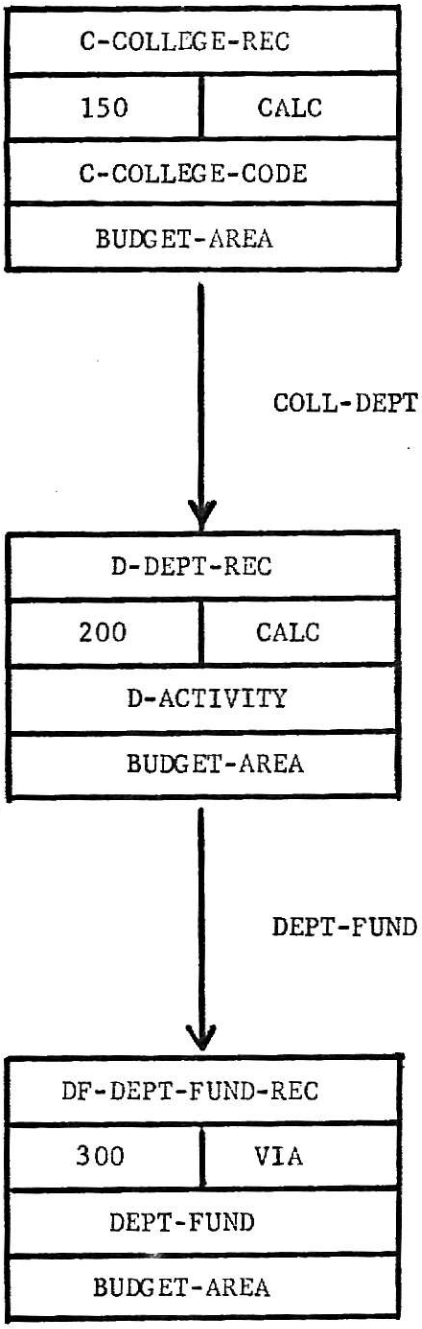


FIGURE 4.8

CPU time because of the services needed to manage main memory.

For this application, the page size was chosen to be 3156 bytes allowing four pages to fit exactly on one IBM 3330 disk track without wasted space. In addition, space for five pages is reserved for the buffer pool. The user manual states that three to eight pages provide an optimum range of choices for pool size. A number larger than eight will degrade system performance due to added memory overhead.

The reason for the choice of five pages is to insure that all record occurrences which are stored by the VIA location mode have a better chance of residing in the buffer pool simultaneously. This is especially desirable during a budget update when there is an exchange of information between budget and department-fund junction record occurrences within a department.

CHAPTER V

DBMS PROCEDURAL REQUIREMENTS

Chapter five specifies the minimal data base management procedures required to implement this master's project. The most important part of IDMS is the data dictionary (section 5.1) which is used for control and documentation purposes. The schema compiler (section 5.2), the CLUE utility (section 5.3), the DMCL processor (section 5.4), the subschema processor (section 5.5), and the DML processor (section 5.6) all post information into the data dictionary. Before the data dictionary and the user data base are allowed access, they must be initialized which is the topic of section 5.7. Sections 5.8 and 5.9 describe the utility backup and restore procedures respectively.

5.1 DATA DICTIONARY:

The data dictionary is an IDMS data base which consists of forty (40) record types, fifty-three (53) set relationships, and one area (DDLDML). The page range for DDLDML is from 1 to 1000 logical pages. For this implementation the page size is the vendor supplied size of 3156 bytes. Note that this page size may be different from that of the user's data base.

One requirement for the data dictionary is that its area exist as a separate file (See figure 5.1). The volume containing this file need be mounted only when one of the IDMS programs is posting to it. As implied, it need not be mounted for the execution of application programs. This is especially critical since the dictionary requires over thirteen cylinders of disk storage. Storage requirements of this magnitude are quite costly. So for this master's project the data dictionary will be needed up to the time that all application programs have been written

INITIALIZE DATA DICTIONARY

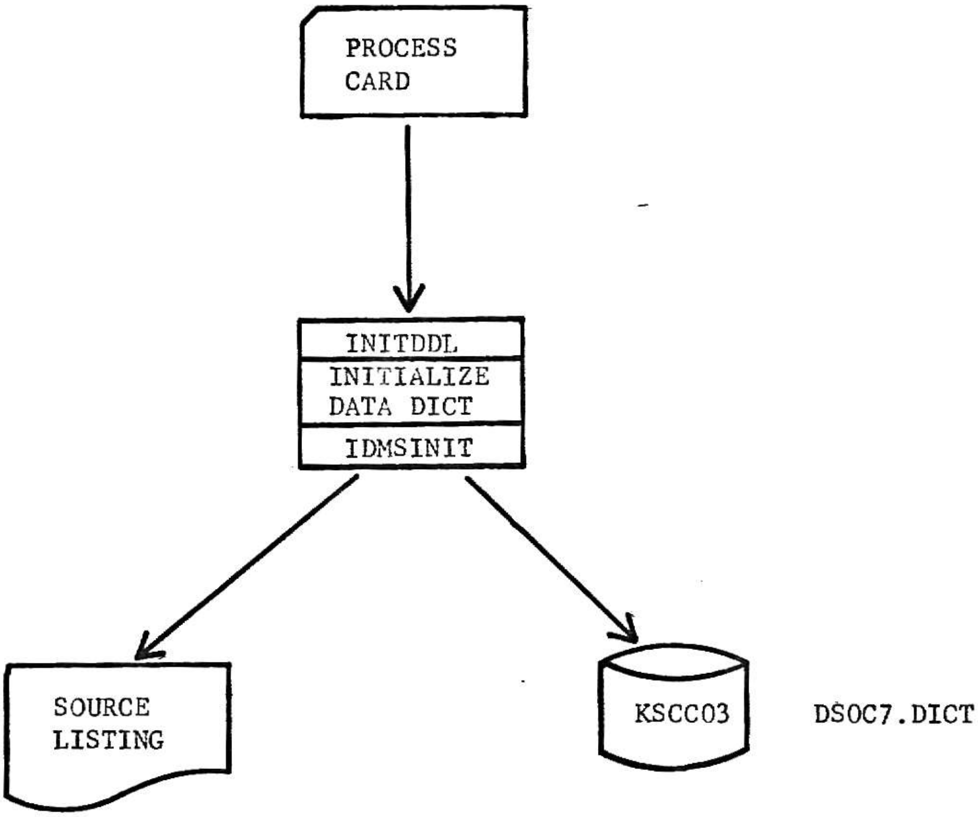


FIGURE 5.1

and debugged. At that time the dictionary will be copied onto a backup tape and the disk file deleted. If program changes are required at a later date the data dictionary can be reinitialized and restored.

5.2 THE SCHEMA COMPILER:

Before the schema compiler may execute, a newly initialized DDLDML area must exist. For this application there will be five record types and five set relationships in the schema description. The page range is from 1 to 76 logical pages of size 3156 bytes per page.

Only one schema description is ever present on a data dictionary. Figure 5.2 illustrates the I/O flow required for the successful schema compilation. Note that all schema DDL statements must be valid before any posting is done to the data dictionary.

5.3 THE CLUE UTILITY:

The DML processor must validate each DML statement for proper syntax. From special keywords in the DML syntax, the DML processor creates a special key called a CLUE which is composed of unique character created from each of the keywords in the order in which they appear in a DML statement. The processor attempts to find that CLUE in the data dictionary. If the CLUE exists, the syntax is correct, otherwise it is in error. When an error condition arises, control can be transferred to a special routine. This routine which is in the form of Cobol source statements is input into this program. At compilation time the routine is copied into the body of the Cobol program if the keywords GO TO MAIN are coded.

CLUE's must be loaded into the data dictionary by the utility program IDMSCLUE after the schema compiler run. Figure 5.3 illustrates the CLUE procedure.

SCHEMA COMPILER RUN

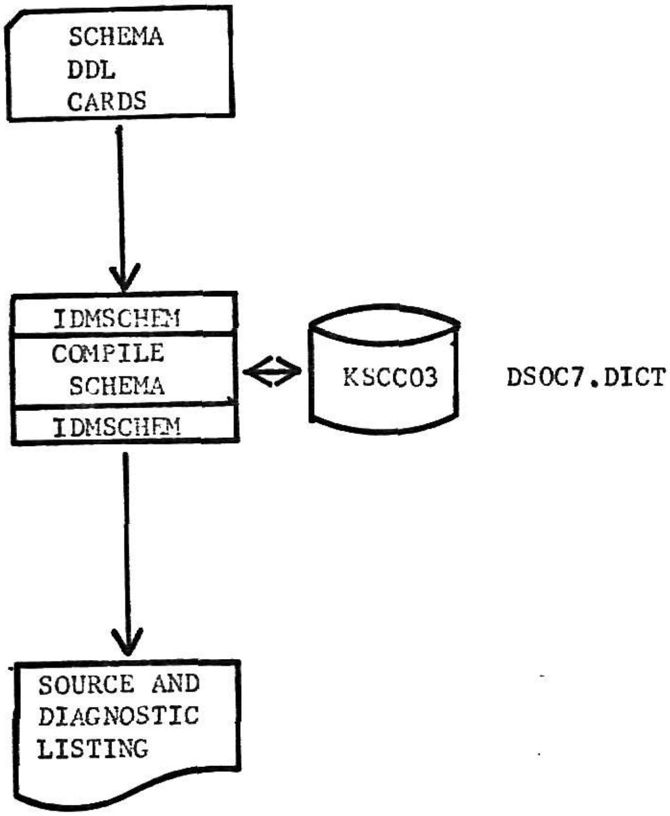


FIGURE 5.2

CLUE UTILITY UPDATE

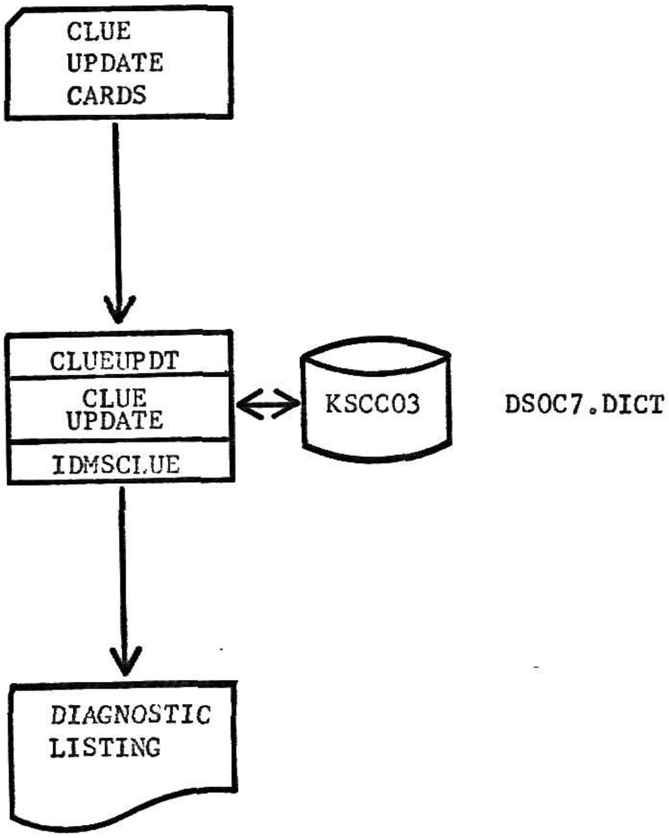


FIGURE 5.3

5.4 THE DMCL PROCESSOR:

The DMCL processor uses the contents of the data dictionary to validate the DMCL syntax against the schema specifications. All DMCL specifications are posted to the data dictionary upon successful execution of the DMCL processor.

The DMCL syntax is used by the DMCL processor to generate a BAL source module which is then assembled and link-edited for use at execution time by the data base management routines. The DMCL module contains all Data Control Blocks (DCB's) necessary to access the BUDGET-AREA area, the necessary mapping tables needed to determine the relative block number corresponding to each logical page, anu the actual buffer pool. Figure 5.4 illustrates the procedure flow chart used in the creation of the DMCL module.

5.5 THE SUBSCHEMA PROCESSOR:

The subschema processor expects that the schema has been specified, and that at least one DMCL module exists on the data dictionary before it may be successfully executed. The specifications of the subschema are validated against the specifications of the schema and the specified DMCL module. All specifications for each subschema compile are posted to the data dictionary upon successful compilation.

The subschema syntax is read by the subschema processor which in turn generates a BAL source module. Figure 5.5 illustrates the procedure whereby this module is processed, assembled, and link-edited for use at execution time by the IDMS and DBMS routines. The generated subschema module contains the area, set, record, and data-item entries required to satisfy and restrict DML function calls.

DMCL UPDATE PROCEDURE

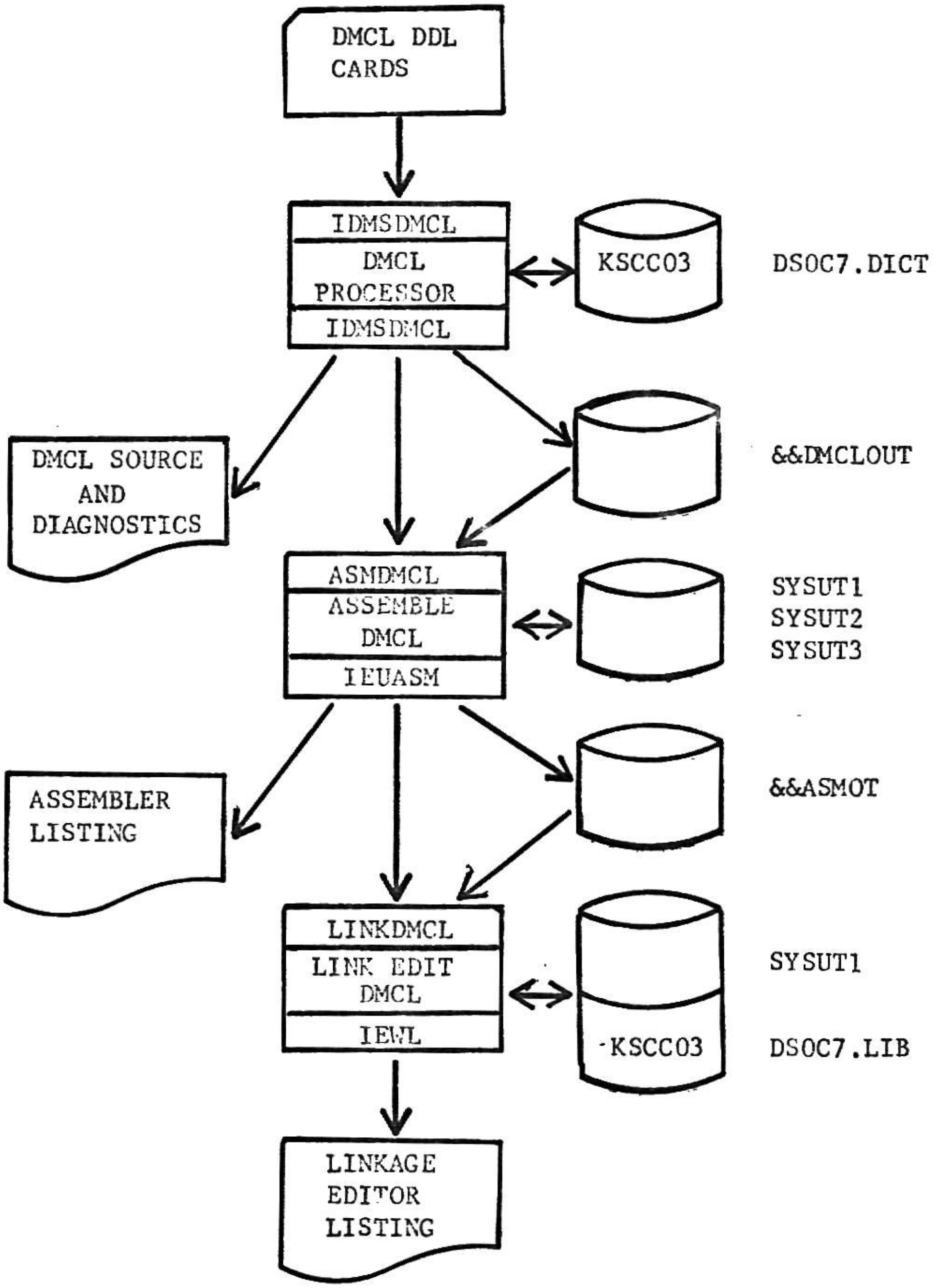


FIGURE 5.4

SUBSCHEMA UPDATE PROCEDURE

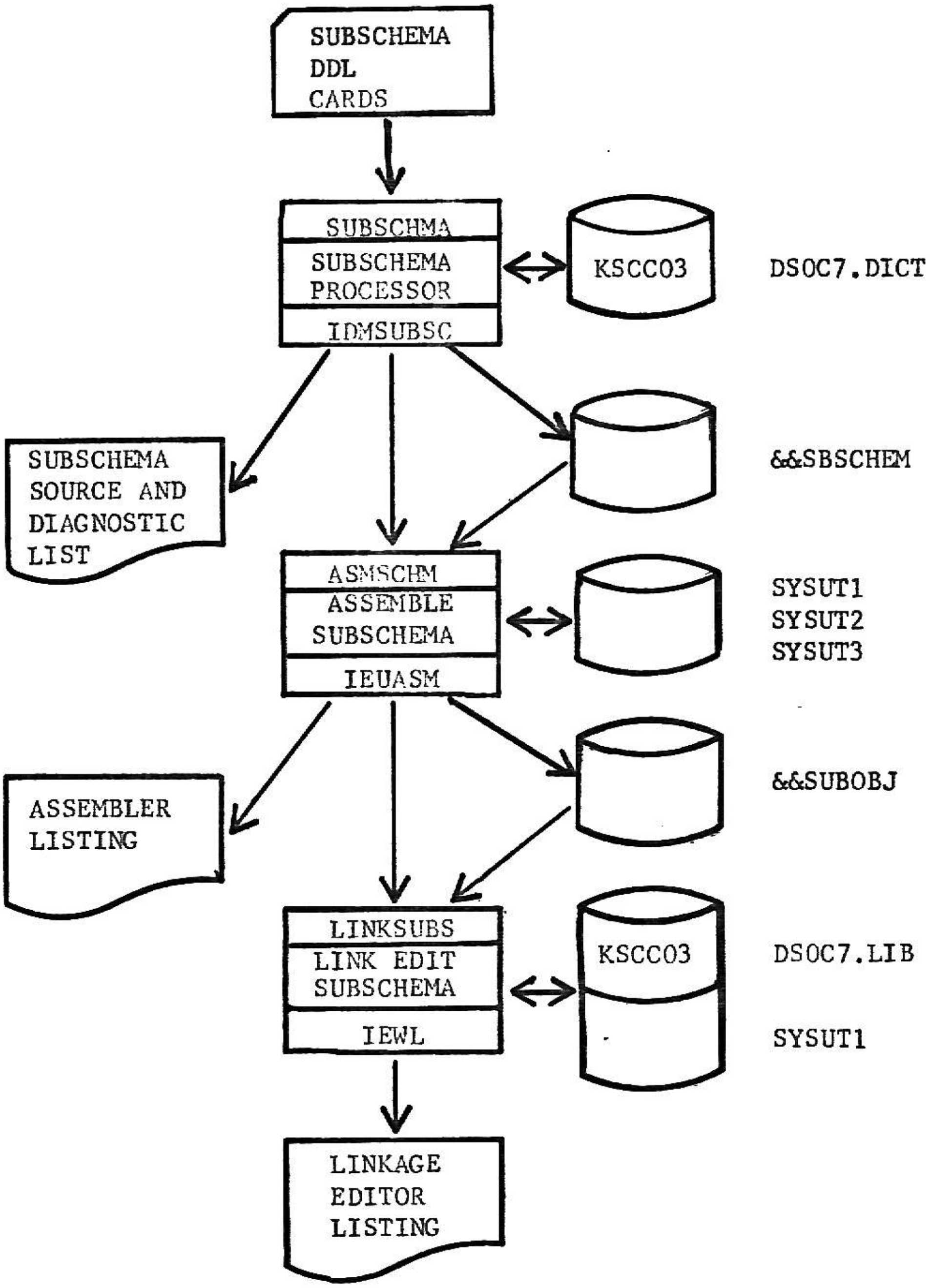


FIGURE 5.5

5.6 THE DML PROCESSOR:

The DML processor expects that at least one schema, one DMCL, and one subschema are specified in the data dictionary before it will process any Cobol/DML program. The DML syntax within a Cobol program refers to a specific subschema module in the data dictionary. The DML processor then uses contents from the data dictionary to build communication areas and IDMS record areas in the Data Division, to insert bind calls and the IDMS-STATUS section into the Procedure Division, and to validate and translate the DML statements into function calls within the Procedure Division.

The IDMS DML processor (IDMSDML5) reads the Cobol/DML source statements and converts all DML statements into standard Cobol statements. See figure 5.6. It also copies all Cobol statements as they are encountered in the input stream. If the processor run is successful, the generated source program is passed to the Cobol compiler and an entry is posted to the data dictionary. Upon successful compilation, the object module is link-edited and loaded into a user library.

5.7 DATA DICTIONARY AND USER DATA BASE INITIALIZATION:

The data dictionary and user data base are initialized by the same system program (IDMSINIT). The factor that distinguishes the type of initialization is DMCL specified on the control card. The bootstrap DMCL IDMSBASE is specified to initialize the data dictionary, whereas BU01DMCL is specified to initialize the user data base. The procedure (See figure 5.7) initializes all of the pages for the DDLDML and BUDGET-AREA areas respectively so that DBMS routines may access the storage within the respective data bases.

DML COMPILE AND LINK-EDIT PROCEDURE

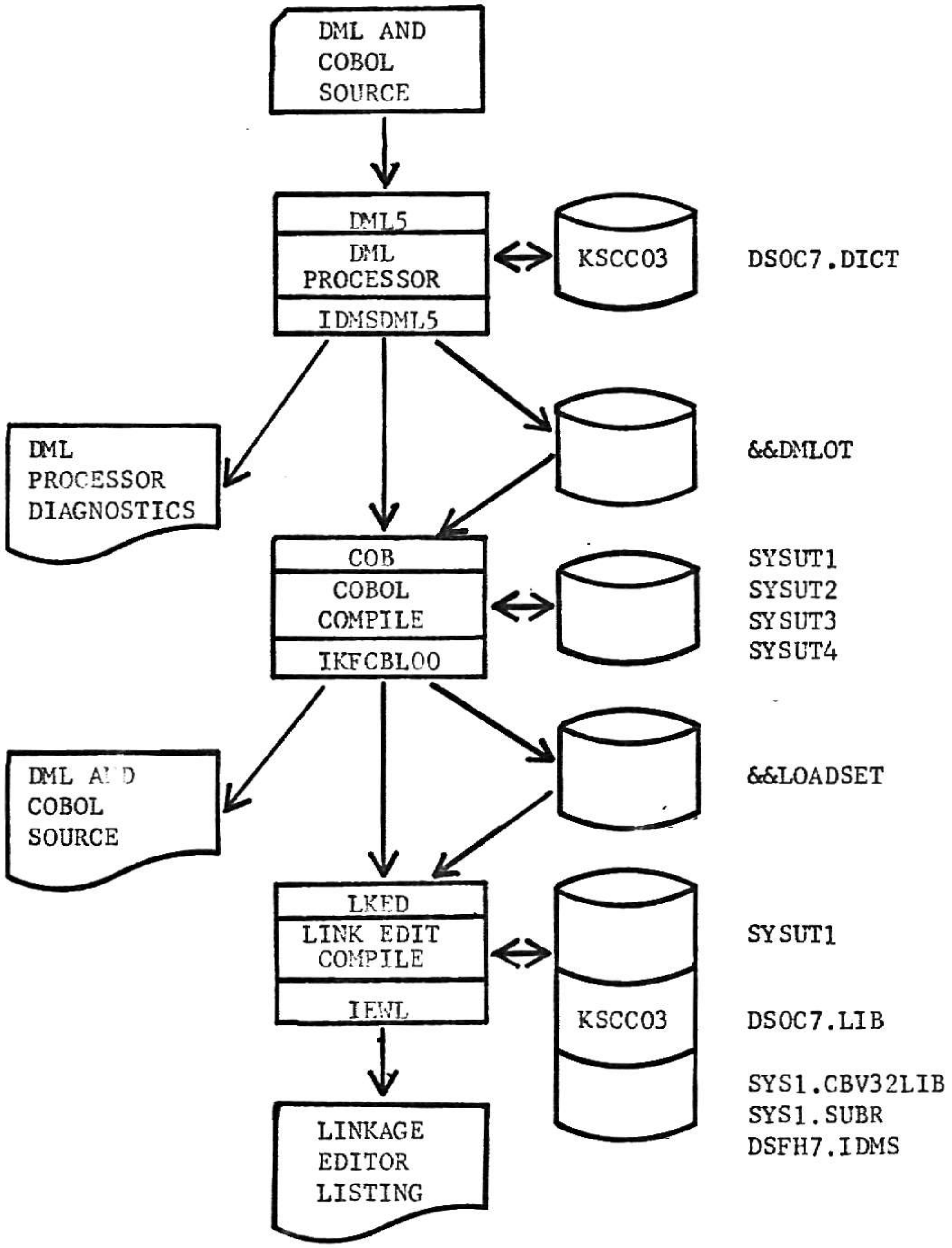


FIGURE 5.6

INITIALIZE DATA BASE

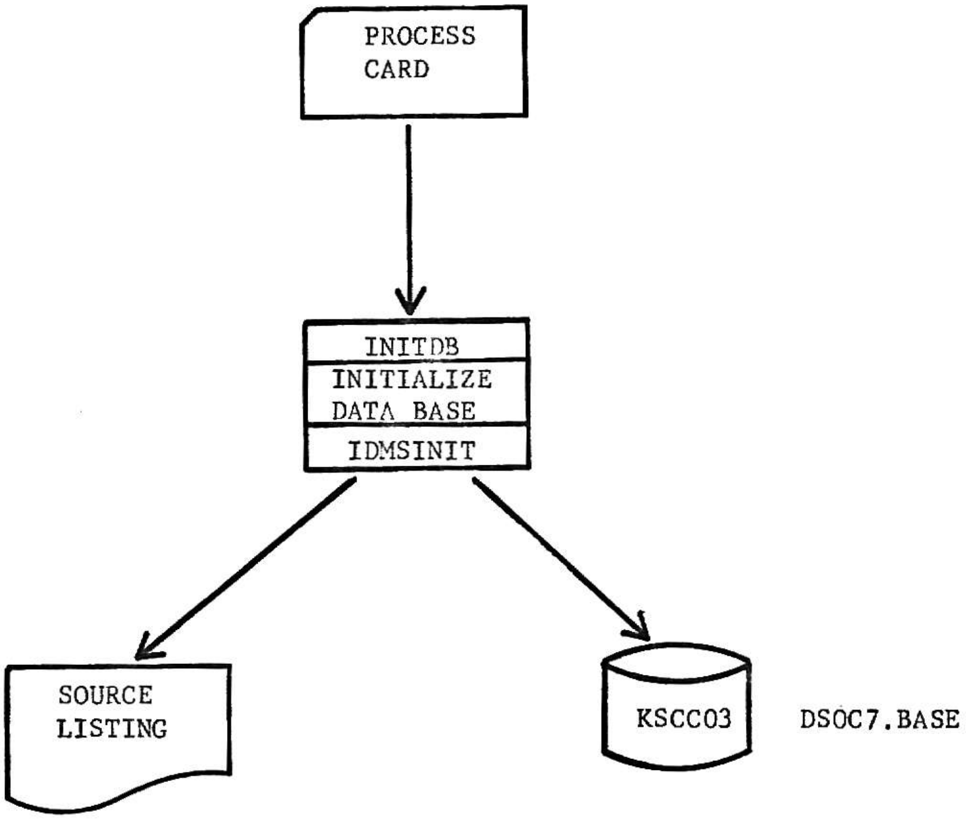


FIGURE 5.7

5.8 IDMS SECURITY DUMP:

The program (IDMSDUMP) is used to copy all or individual areas of any IDMS data base to a tape file. The primary reason for using this utility is to provide a backup tape for the user data base and to provide a less expensive means of storage for the data dictionary once the user's system is implemented. As in the case of initialization, this program requires a control or PROCESS card which specifies the DMCL used in the creation of the data base. See figure 5.8.

5.9 IDMS SECURITY RESTORE:

The program (IDMSRSTR) is used to restore all or individual areas of an IDMS data base from a security dump tape file created by IDMSDUMP. A PROCESS control card which specifies the DMCL is also required. If the area has been destroyed, as is the case of the data dictionary, the area will have to be reinitialized before it can be restored. See figure 5.9.

IDMS SECURITY DUMP

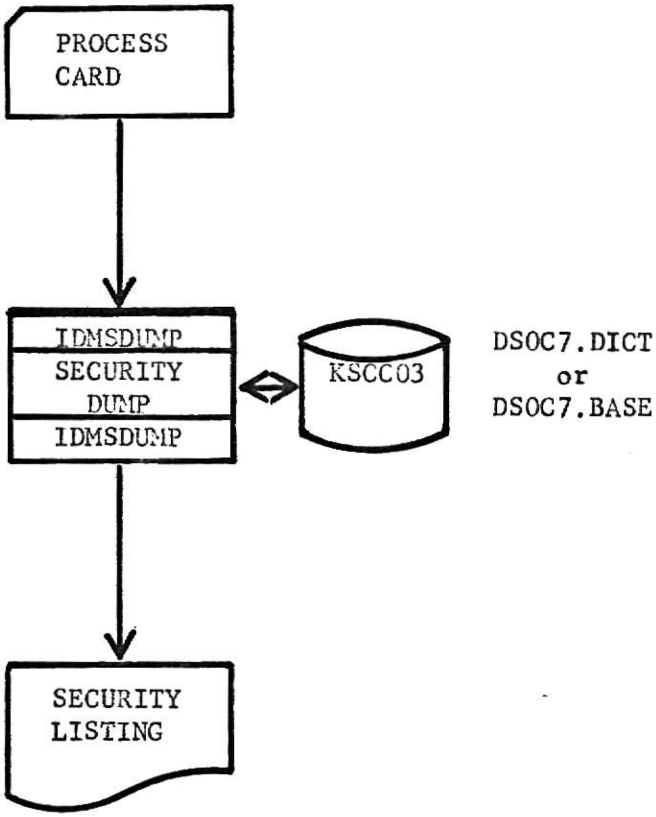


FIGURE 5.8

IDMS SECURITY RESTORE

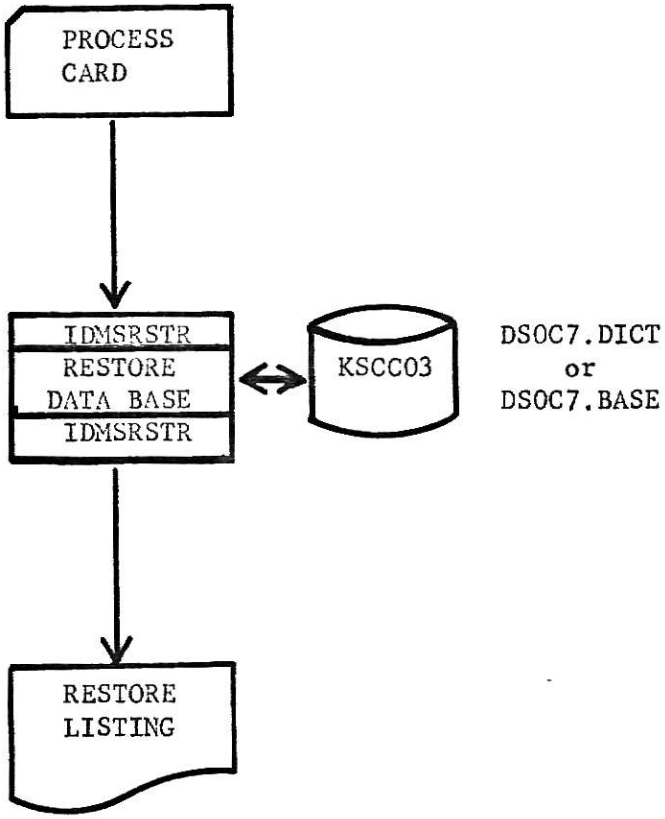


FIGURE 5.9

CHAPTER VI

SOFTWARE DESIGN SPECIFICATIONS

Chapter six describes the system of user programs needed to implement this master's project. Seven programs are required to accomplish this goal. The intent here is that each program will update a data structure described in the schema. In the event that any one of these structures change only one program is affected. Further, more security can be provided by allowing selective access to these programs. For example, department and fund record occurrences can be established only after the college record occurrence has been created. If access is not granted to the program that creates the college records, a potential user is prevented from storing undesireable records on the data base. Furthermore, only after department and fund record occurrences exist may the budget record occurrences be added to the data base.

Below is a listing of the update programs and the structure(s) each maintains.

PROGRAM	DATA STRUCTURE
PBU01AAA	COLLEGE
PBU01AAB	DEPARTMENT
PBU01AAC	FUND
PBU01AAD	BUDGET and DEPARTMENT-FUND JUNCTION
PBU01AAG	BUDGET and DEPARTMENT-FUND JUNCTION

The last two maintain two structures on account of the interdependent relationships of data in the two record structures. The budget record occurrence provides detail information, whereas the department-fund junction record provides summary information for either funds or departments.

Two programs are needed to produce reports. PBU01AAE prints a detailed budget listing, and PBU01AAF prints a summary for all funds within a department.

6.1 THE COLLEGE UPDATE PROGRAM:

Program PBU01AAA is used to create, maintain, and delete college record (C-COLLEGE-REC) occurrences. Subschema BU01SS01 is invoked to allow exclusive update of this record type. Security is provided here because the subschema prevents college records from being deleted which have member records. Figure 6.1 illustrates input and output requirements for this program. Input data resides on cards which have been keypunched. Figure 6.2 illustrates the document which may be used to create the transaction. Figure 6.3 contains the instructions necessary to accurately keypunch data cards. Figures 6.4 and 6.5 are record descriptions of the data cards and printed report.

Three types of update transactions are allowed in this program. They are adds, changes, and deletes. An add causes a new record occurrence to be established on the data base. The college code and college title on the data card are accepted as keypunched without any error checking. Currently changes are allowed to one field, the title field. This is done by identifying the correct college by code and keypunching the new version of the title on the card. Deletions are accomplished by identifying the college by code. If the college record occurrence has no member record occurrences, it is deleted from the data base.

For added security the user must keypunch the pass code, COLL, in columns one through four of the transaction before this program will accept it.

COLLEGE UPDATE PROCEDURE

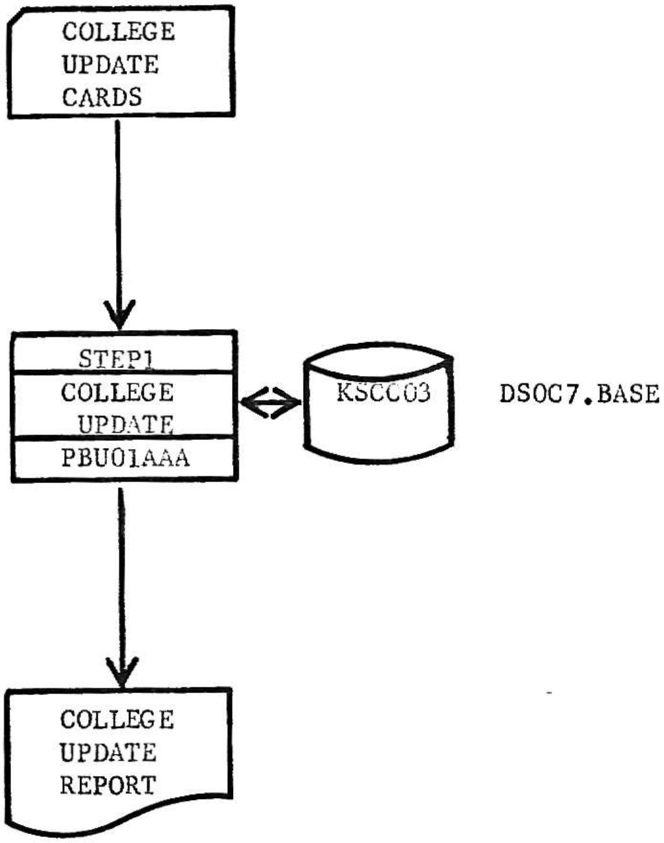


FIGURE 6.1

BU01-001

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

COLLEGE UPDATE FORM

PASS [1] CODE	UPDATE [2] CODE	COLLEGE [3] CODE	COLLEGE [4] TITLE
COLL	____	____	____
COLL	____	____	____
COLL	____	____	____
COLL	____	____	____
COLL	____	____	____

1. The pass code is a four position code which is required to insure that the update cards are processed by the correct program.

2. There are three update codes. They are: 'A' for an add transaction, 'C' for a change, and 'D' for a delete.

3. The college code is a three position code which uniquely identifies a college, usually the first three digits of the activity code.

4. The college title field consists of forty (40) alphanumeric positions which are used to provide a description of the college.

FIGURE 6.2

KEYPUNCH INSTRUCTIONS
FOR
COLLEGE UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
PASS CODE	ALPHABETIC	1 - 4	4	Duplicate pass code on all data cards.
UPDATE CODE	ALPHABETIC	5	1	Must be an A, C, or D.
COLLEGE CODE	ALPHANUMERIC	6 - 8	3	
COLLEGE TITLE	ALPHANUMERIC	9 - 48	40	Left justify this field.

FIGURE 6.3

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U. S. A.

Company ____________

Application BUDGET SYSTEM by ____________ Date ____________ Job No. PBU01AAA Sheet No 1 of 1

PASS CODE	TR CODE	COLL CODE	COLLEGE TITLE	
9 9 9 9 1 2 3 4	9 5	9 9 9 6 7 8	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

FIGURE 6.4

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

50/10/6 PRINT CHART PROG. ID ____________ PAGE ______
ACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

OGRAM TITLE ____________

OGRAMMER OR DOCUMENTALIST: ____________

HART TITLE ____________

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

CARRIAGE CONTROL TAPE CHAN

```
1  PS1131AAA                               K A N S A S   S T A T E   U N I V E R S I T Y                      DATE XX/XX/XX
3  RFE1                                                   COURSE UPDATE REPORT                                 PAGE XXX
6                                 TRANS     COURSE                          TITLE
7                                   X         XXX         X---------------------------------------X
8                                   X         XXX         X---------------------------------------X
9                                   X         XXX         X---------------------------------------X
```

FIGURE 6.5

*Number of forms per pad may vary slightly

Fold back at dotted line.

Fold in at dotted line.

6.2 THE DEPARTMENT UPDATE PROGRAM:

Figure 6.6 illustrates the input and output requirements for program PBU01AAB. The document used to initiate this procedure is illustrated in figure 6.7 and the corresponding keypunch instructions in figure 6.8. Record descriptions for both input and output are shown in figure 6.9 and 6.10 respectively.

Program PBU01AAB is used to create, maintain, and delete departmental record (D-DEPT-REC) occurrences. Subschema BU01SS02 is invoked to allow exclusive update of the data base. Two record types, C-COLLEGE-REC and D-DEPT-REC and one set, COLL-DEPT, are described in the subschema. College occurrences may be accessed to obtain data for report headings, but may not be modified or deleted.

Update codes allowed by this program are adds, changes, and deletes. An add causes a department occurrence to be established as a member of the college occurrence designated by the college code on the add card. If the code on the card does not correspond to any code for an existing college occurrence, an error occurs and the user is notified of the condition by an appropriate error message. Changes and deletions can be made only to existing department occurrences. The department occurrence is identified by the college and department codes on the change card. Restrictions on field changes are such that only the contents of the department title may be changed. Any attempt to change or delete a record which does not exist, results in the issuance of an error message. Any attempt to delete a department occurrence which has members will cause an error message to be printed.

Before any transaction may be processed the pass code must be verified. Valid cards must contain the code DEPT in columns one to four.

DEPARTMENT UPDATE PROCEDURE

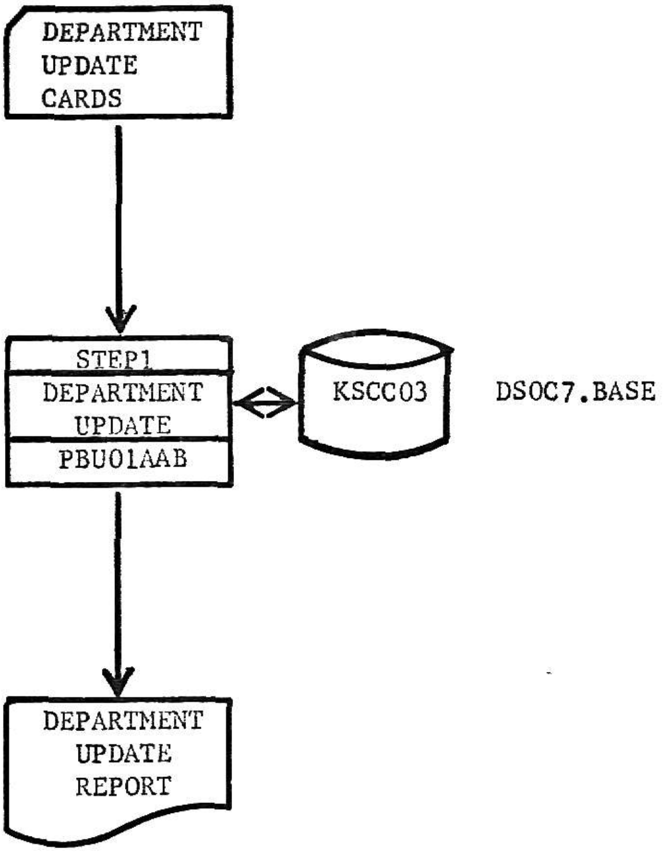


FIGURE 6.6

BU01-002

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

DEPARTMENT UPDATE FORM

PASS CODE	UPDATE CODE	COLLEGE CODE	DEPT CODE	DEPARTMENT TITLE
DEPT	____	____	____	____
DEPT	____	____	____	____
DEPT	____	____	____	____
DEPT	____	____	____	____
DEPT	____	____	____	____
DEPT	____	____	____	____

1. The pass code is a four position code which is required to insure that the update cards are processed by the correct program.

2. There are three update codes. They are: 'A' for an add transaction, 'C' for a change, and 'D' for a delete.

3. The college code is a three position code which uniquely identifies a college, usually the first three digits of the activity code.

4. The department code is a two position code which uniquely identifies a department within a college, usually the last two digits of the activity code.

5. The department title field consist of forty (40) alphanumeric positions which are used to provide a description for the department.

FIGURE 6.7

KEYPUNCH INSTRUCTIONS
FOR
DEPARTMENT UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
PASS CODE	ALPHABETIC	1 - 4	4	Duplicate pass code on all data cards.
UPDATE CODE	ALPHABETIC	5	1	Must be an A, C, or D.
COLLEGE CODE	ALPHANUMERIC	6 - 8	3	
DEPT CODE	ALPHANUMERIC	9 - 10	2	
DEPARTMENT TITLE	ALPHANUMERIC	11 - 50	40	

FIGURE 6.8

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

Form X24-6599-0
Printed in U. S. A.

MULTIPLE-CARD LAYOUT FORM

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAB Sheet No. 1 of 1

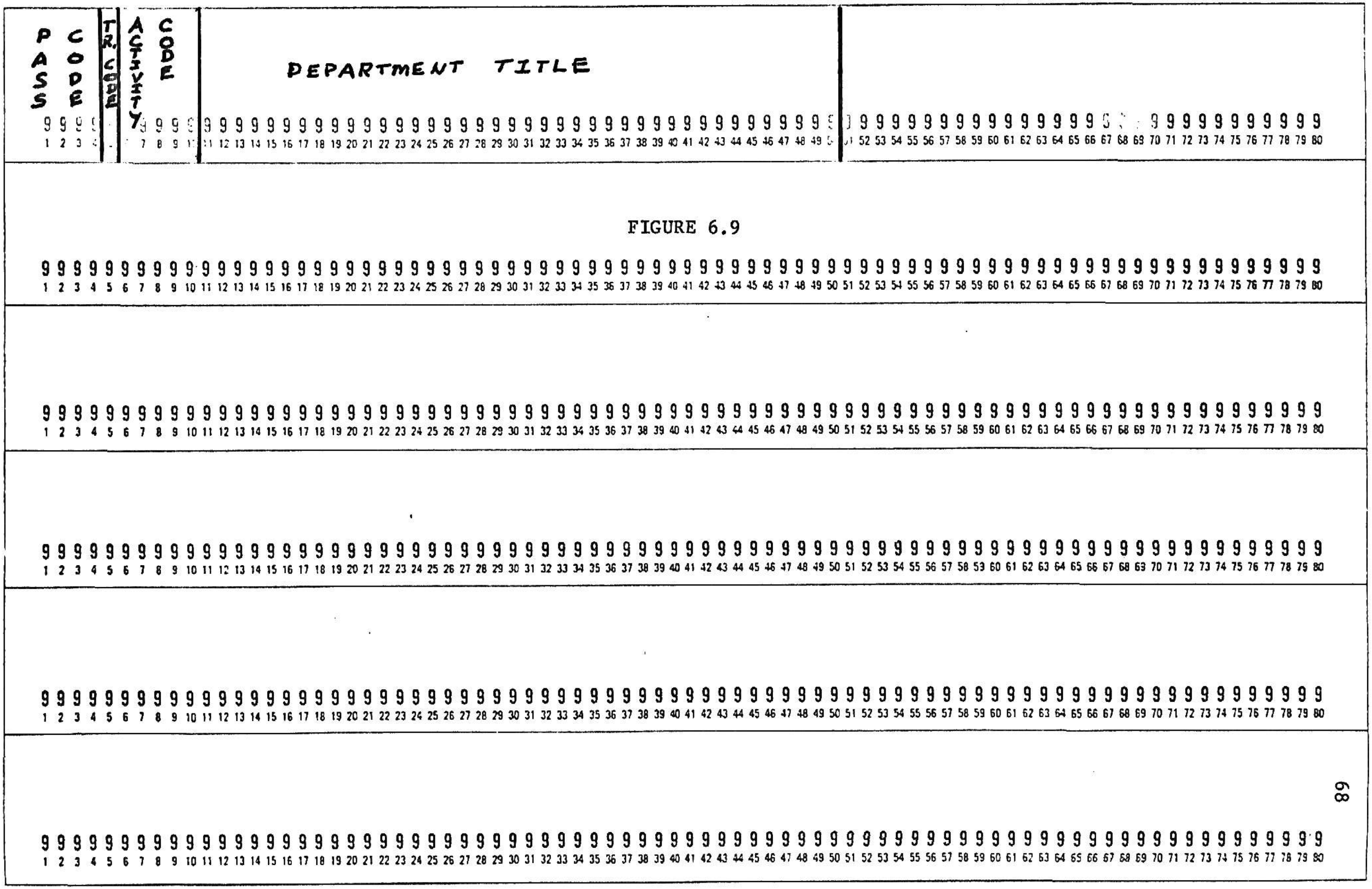
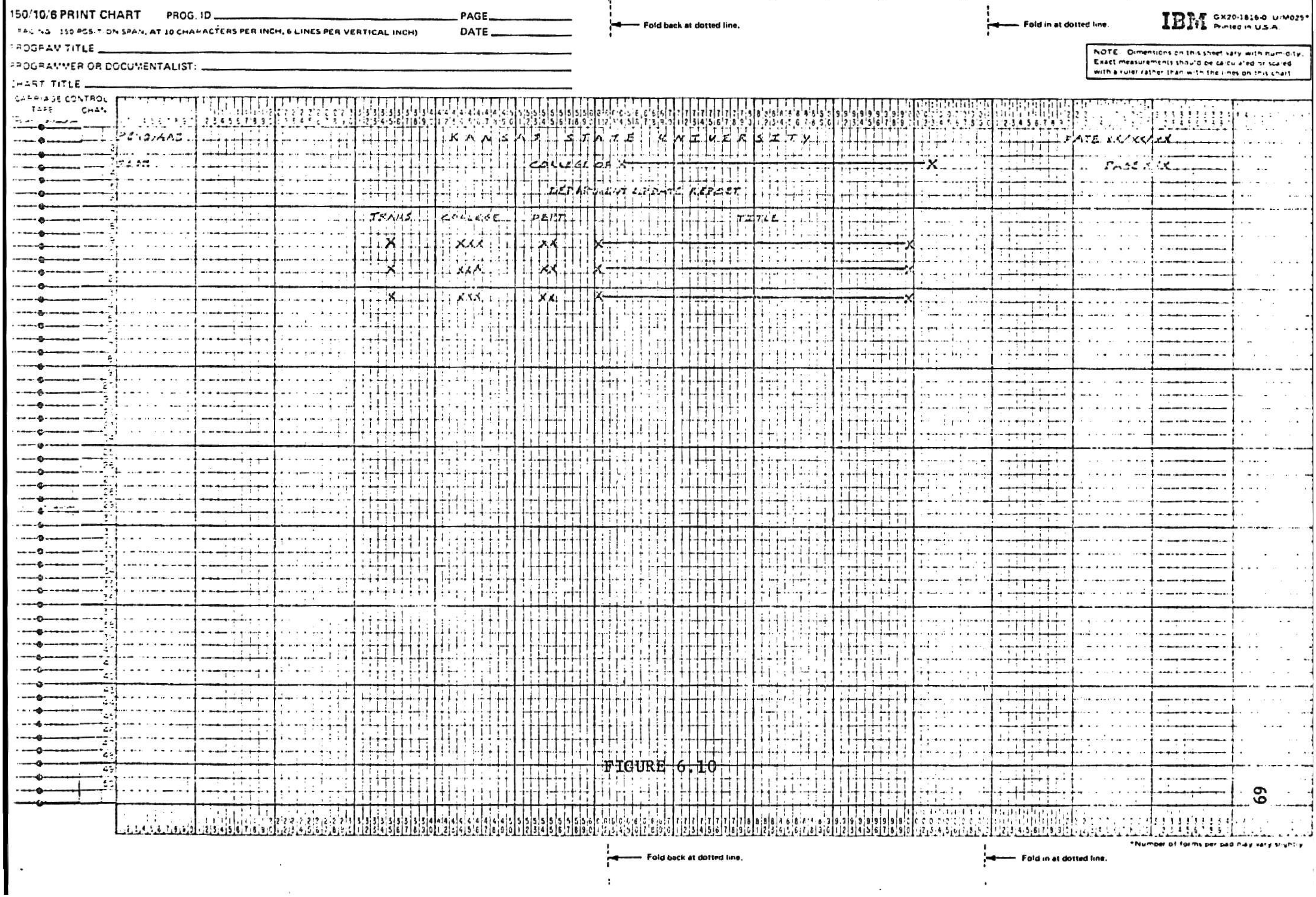


FIGURE 6.9



150/10/6 PRINT CHART PROG. ID ______ PAGE ______

(SPACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST: ______

CHART TITLE ______

CARRIAGE CONTROL TAPE CHAN.

← Fold back at dotted line.

← Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

```
PROGRAM                           K A N S A S   S T A T E   U N I V E R S I T Y                 DATE XX/XX/XX
[illegible]                                  COLLEGE OF X------------------------X                   PAGE XXX
                                               DEPARTMENT UPDATE REPORT

                          TRANS    COLLEGE    DEPT                  TITLE
                            X        XXX       XX    X------------------------------X
                            X        XXX       XX    X------------------------------X

                            X        XXX       XX    X------------------------------X
```

FIGURE 6.10

*Number of forms per pad may vary slightly

6.3 FUND UPDATE PROGRAM:

Figure 6.11 shows the input and output requirements for program PBU01AAC. The printed report lists all of the updates and error messages. Figure 6.12 illustrates the input source document which initiates updates. From this document data cards may be keypunched according to instructions in figure 6.13. Record descriptions for the input cards and printed report are illustrated in figures 6.14 and 6.15 respectively.

This program is written to create, maintain, and delete fund (F-FUND-REC) occurrences. Subschema BU01SS03 is invoked to allow exclusive update to the data base. The C-COLLEGE-REC and F-FUND-REC record types and COLL-FUND set are included in the subschema description. College occurrences may be retrieved for the purpose of providing titles for headings, but may not be modified or deleted.

Update codes allowed by this program are adds, changes, and deletes. An add causes a fund occurrence to be established on the data base as a member occurrence of a college occurrence whose code has been specified on the add card. If the college code on the add card does not correspond to a code on an existing college occurrence, an error condition causes an error message to be printed. Changes and deletions may be made only to existing fund occurrences. Restrictions are such that only changes to the fund title are allowed. An attempt to modify or delete any occurrence which does not exist will cause an error message to be printed. Fund occurrences which have members may not be deleted.

Before any fund transaction may be processed, it must be validated. Valid transactions contain the code FUND in columns one through four.

FUND UPDATE PROCEDURE

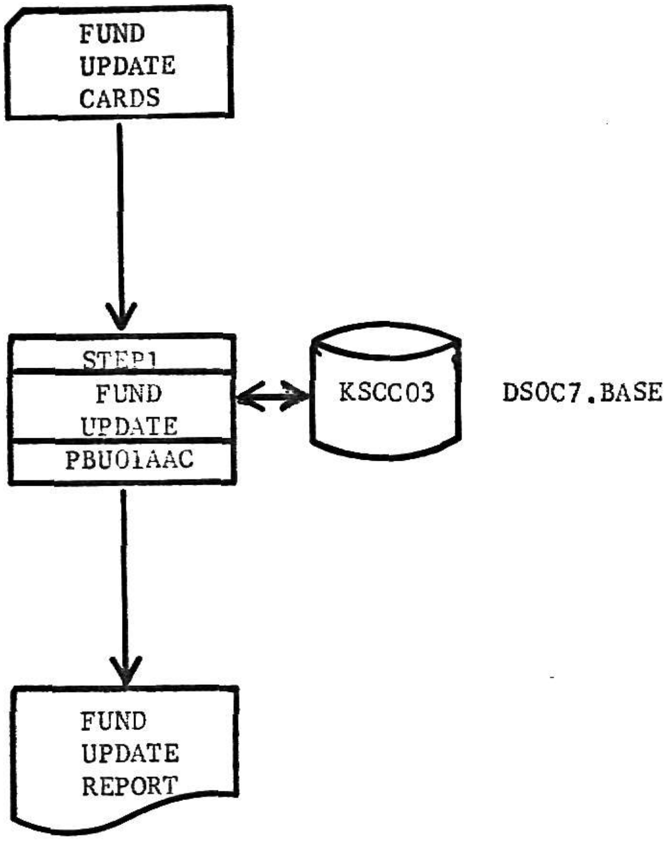


FIGURE 6.11

BU01-003

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

FUND UPDATE FORM

PASS CODE [1]	UPDATE CODE [2]	COLLEGE CODE [3]	FUND CODE [4]	FUND TITLE [5]
FUND	____	____	____	________________________________
FUND	____	____	____	________________________________
FUND	____	____	____	________________________________
FUND	____	____	____	________________________________
FUND	____	____	____	________________________________
FUND	____	____	____	________________________________

1. The pass code is a four position code which is required to insure that the update cards are processed by the correct program.

2. There are three update codes. They are: 'A' for an add transaction, 'C' for a change, and a 'D' for a delete.

3. The college code is a three position code which uniquely identifies a college, usually the first three digits of the activity code.

4. The fund code is a five position code which uniquely identifies a fund which is to be used by the college.

5. The fund title field consists of forty (40) alphanumeric positions which are used to describe the fund source.

FIGURE 6.12

KEYPUNCH INSTRUCTIONS FOR FUND UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LFNGTH	REMARKS
PASS CODE	ALPHABETIC	1 - 4	4	Duplicate pass code on all cards.
UPDATE CODE	ALPHABETIC	5	1	Must be an A, C, or D.
COLLEGE CODE	ALPHANUMERIC	6 - 8	3	
FUND CODE	ALPHANUMERIC	9 - 13	5	
FUND TITLE	ALPHANUMERIC	14 - 53	40	

FIGURE 6.13

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

Form X24-6599-0
Printed in U. S. A.

MULTIPLE-CARD LAYOUT FORM

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAC Sheet No. 1 of 1

PASS CODE	TR CODE	FUND CODE	FUND TITLE	
9 9 9 9	9	9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4	5	6 7 8 9 10	11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50	51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

FIGURE 6.14

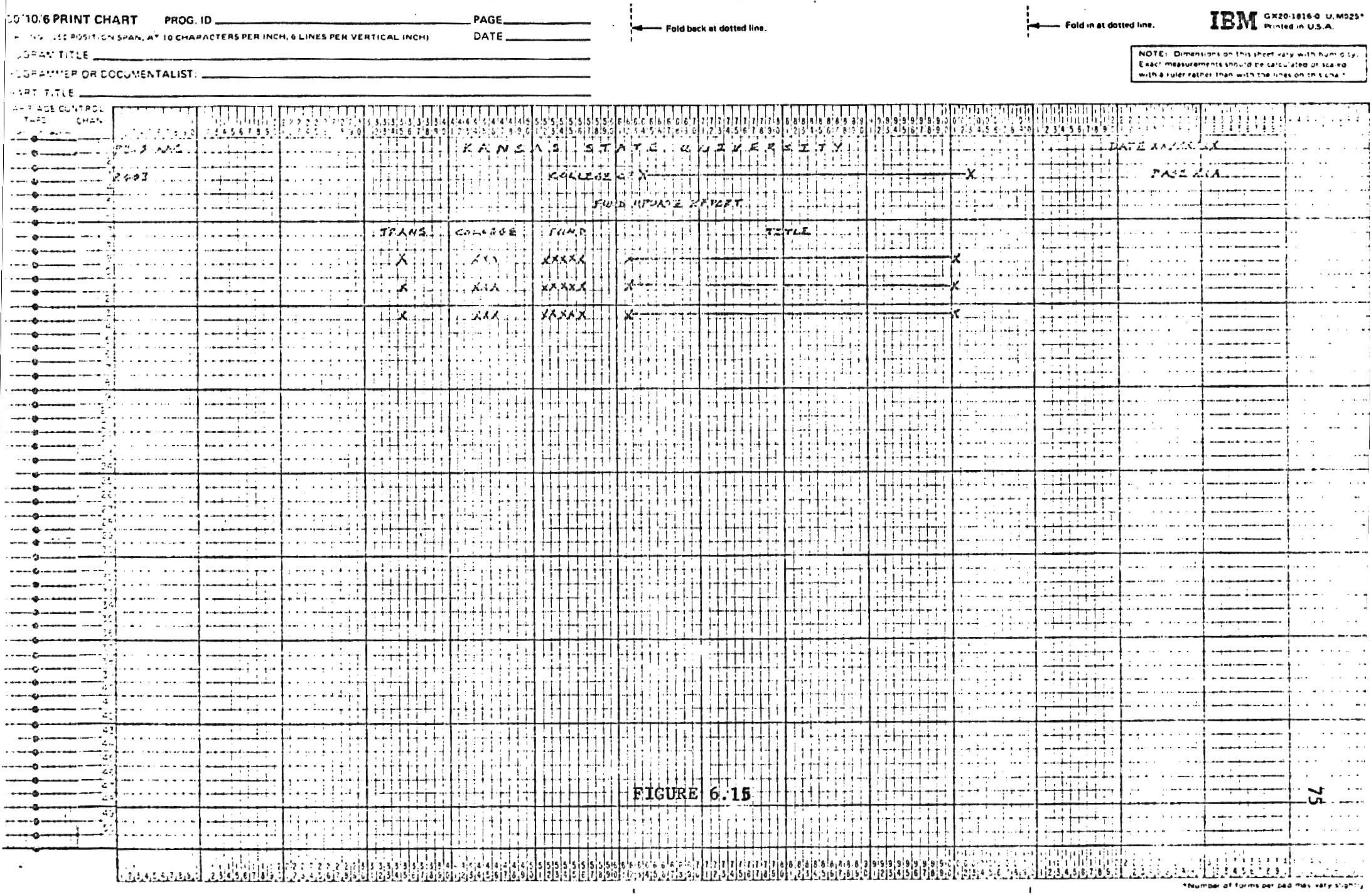
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80



0'10/6 PRINT CHART PROG. ID ____ PAGE ____

150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ____

GRAM TITLE ____

GRAMMER OR DOCUMENTALIST: ____

ART TITLE ____

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

CARRIAGE CONTROL TAPE CHAN.

```
PUBS NAC                    KANSAS STATE UNIVERSITY                        DATE XX/XX/XX
R403                              COLLEGE OF X--------------------X             PAGE XXX
                                       FUND UPDATE REPORT

                  TRANS    COLLEGE    FUND                 TITLE
                    X        XXX      XXXXX    X--------------------------X
                    X        XXX      XXXXX    X--------------------------X

                    X        XXX      XXXXX    X--------------------------X
```

FIGURE 6.15

*Number of forms per pad may vary slightly

6.4 THE BUDGET UPDATE PROGRAM:

Input and output requirements for program PBU01AAD are illustrated in figure 6.16. They are similar in nature to all previous input and output flowcharts. However, the transactions and processing involved is extremely more complex. All updates to the data base are through card input and the results are a printed listing of those transactions and any appropriate messages. The dean's office receives information from its departments related to the budget on two forms. They are the 'SOURCE OF SALARY' document and the 'GRADUATE STUDENT TRANSACTION'. See Appendix A. Information obtained from these documents can be transcribed from either document onto the BUDGET UPDATE FORM (See figure 6.17). Additional information is needed on this document which is not contained on either of the initial documents. Once this document has been completed, it may be keypunched according to directions illustrated in figure 6.18.

Two other documents initiated by the dean's office are the LOAN UPDATE FORM (figure 6.19) and the MAINTENANCE UPDATE FORM (figure 6.20). The loan form is to be used when a department within the College of Arts and Sciences wishes to borrow tenths or dollars from another department. Initialization of such a transaction occurs when a representative of a department verbally communicates the need to borrow tenths or dollars to the dean's office. The loan document is then filled out with the appropriate information. Data cards may be keypunched according to the directions shown in figure 6.21. The maintenance transaction is required because two record types, B-BUDGET-REC and DF-DEPT-FUND-REC, are being maintained by one program. All budget occurrences are maintained directly by the user. However, PBU01AAD maintains the department-fund junction occurrences. The maintenance transaction is provided for a program or system failure which could cause an out-of-balance condition to occur.

BUDGET UPDATE PROCEDURE

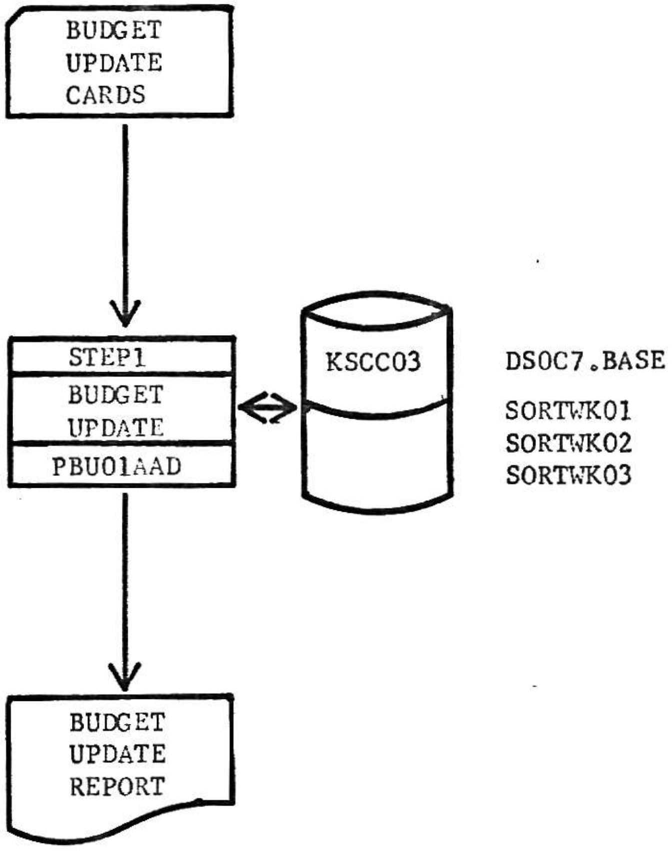
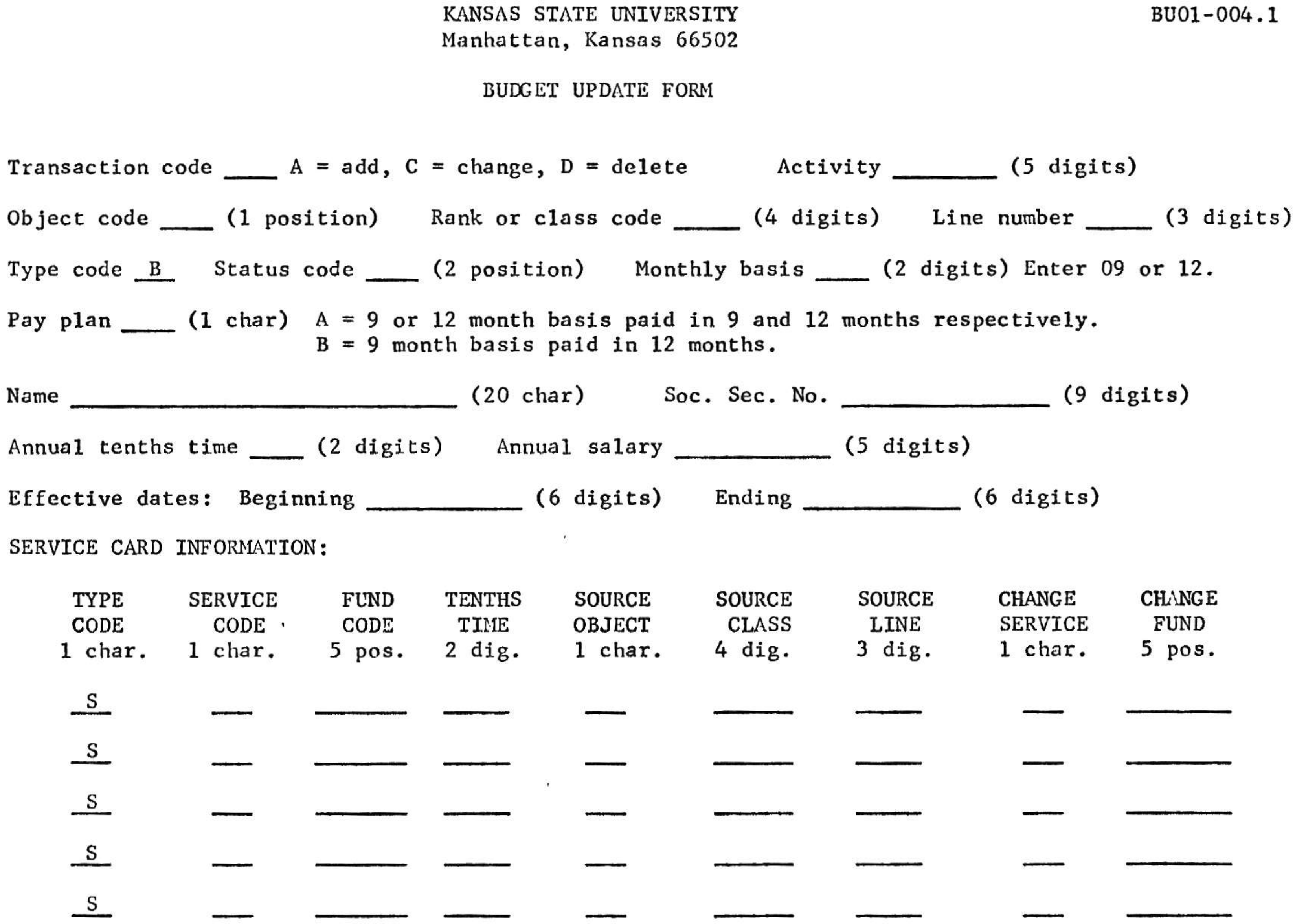


FIGURE 6.16



KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

BU01-004.1

BUDGET UPDATE FORM

Transaction code ____ A = add, C = change, D = delete   Activity ________ (5 digits)

Object code ____ (1 position)   Rank or class code _____ (4 digits)   Line number _____ (3 digits)

Type code B   Status code ____ (2 position)   Monthly basis ____ (2 digits) Enter 09 or 12.

Pay plan ____ (1 char) A = 9 or 12 month basis paid in 9 and 12 months respectively.
B = 9 month basis paid in 12 months.

Name ______________________________ (20 char)   Soc. Sec. No. ________________ (9 digits)

Annual tenths time ____ (2 digits)   Annual salary ____________ (5 digits)

Effective dates: Beginning ____________ (6 digits)   Ending ____________ (6 digits)

SERVICE CARD INFORMATION:

TYPE	SERVICE	FUND	TENTHS	SOURCE	SOURCE	SOURCE	CHANGE	CHANGE
CODE	CODE	CODE	TIME	OBJECT	CLASS	LINE	SERVICE	FUND
1 char.	1 char.	5 pos.	2 dig.	1 char.	4 dig.	3 dig.	1 char.	5 pos.
S	___	_______	_____	___	______	_____	___	________
S	___	_______	_____	___	______	_____	___	________
S	___	_______	_____	___	______	_____	___	________
S	___	_______	_____	___	______	_____	___	________
S	___	_______	_____	___	______	_____	___	________

FIGURE 6.17

KEYPUNCH INSTRUCTIONS
FOR
BUDGET UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
		BUDGET CARD		
TRANS CODE	ALPHABETIC	1	1	Must be an A, C, or D.
ACTIVITY CODE	NUMERIC	2 - 6	5	
OBJECT CODE	ALPHANUMERIC	7	1	
RANK CODE	NUMERIC	8 - 11	4	
LINE NUMBER	NUMERIC	12 - 14	3	
TYPE CODE	ALPHABETIC	15	1	Must be a 'B' for the budget card.
STATUS CODE	ALPHANUMERIC	16 - 17	2	
MONTHLY BASIS	NUMERIC	18 - 19	2	Must be 09 or 12
PAY PLAN	ALPHABETIC	20	1	Must be an 'A' or 'B'.
NAME	ALPHANUMERIC	21 - 40	20	
SOC. SEC. NO.	NUMERIC	41 - 49	9	
ANNUAL TENTHS	NUMERIC	50 - 51	2	
ANNUAL SALARY	NUMERIC	52 - 56	5	Left zero fill.
BEGIN DATE	NUMERIC	57 - 62	6	
END DATE	NUMERIC	63 - 68		
		SERVICE CARDS		
DUPLICATE		1 - 14	14	Duplicate from budget record.
TYPE CODE	ALPHABETIC	15	1	Must be an 'S' for each service card.
SERVICE CODE	ALPHANUMERIC	16	1	
FUND CODE	ALPHANUMERIC	17 - 21	5	
SERVICE TENTHS	NUMERIC	22 - 23	2	
OBJECT SOURCE	ALPHANUMERIC	24	1	
RANK SOURCE	NUMERIC	25 - 28	4	
LINE SOURCE	NUMERIC	29 - 31	3	
CHG SERVICE	ALPHANUMERIC	32	1	
CHG FUND CODE	ALPHANUMERIC	33 - 37	5	

FIGURE 6.18

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

BU001-004.2

LOAN UPDATE FORM

		FROM[1]			TO[2]		TENTHS		
TR	ACTIVITY	FUND	KEY[3]	ACTIVITY	FUND	KEY	9 MONTH	12 MONTH	DOLLARS
L									
L									
L									
L									
L									

1. Enter codes for the department, fund, and key of the line loaning tenths and/or dollars.

2. Enter codes for the department, fund, and key of the line receiving tenths and/or dollars.

3. The budget key is the combination of object, rank, and position (line) codes.

FIGURE 6.19

BU01-004.3

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

MAINTENANCE UPDATE FORM

			9 MO TENTHS		12 MO TENTHS		DOLLARS	
TR	ACTIVITY	FUND	FREE	USED	FREE	USED	FREE	USED
M								
M								
M								
M								
M								
M								

FIGURE 6.20

KEYPUNCH INSTRUCTIONS
FOR
LOAN UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
TRANS CODE	ALPHABETIC	1	1	Must always be an 'L'
FROM ACTIVITY	ALPHANUMERIC	2 - 6	5	
FROM FUND	ALPHANUMERIC	7 - 11	5	
FROM KEY	ALPHANUMERIC	12 - 19	8	
TO ACTIVITY	ALPHANUMERIC	20 - 24	5	
TO FUND	ALPHANUMERIC	25 - 29	5	
TO KEY	ALPHANUMERIC	30 - 37	8	
NINE MO. TENTHS	NUMERIC	38 - 42	5	Left zero fill.
TWELVE MO. TENTHS	NUMERIC	43 - 47	5	Left zero fill.
DOLLARS	NUMERIC	48 - 56	9	Left zero fill.

FIGURE 6.21

Discrepencies between any budget and junction occurrences may be rectified by the maintenance transaction. Note that if the system is used properly, this transaction will never need to be used. Instructions for keypunching from this document are provided in figure 6.22.

Record descriptions for the input data cards are provided in figures 6.23 and 6.24. There are three types of printed output associated with this program. All add, change, and delete transactions are printed according to the specifications outlined in figure 6.25. All loan and maintenance transactions print as specified by the record description illustrated in figures 6.26 and 6.27 respectively.

All record types and sets invoked by this program are described in subschema BU01SS04. College, department, and fund record occurrences are used to verify input data and to establish correct set membership for budget occurrences. However, they may not be modified or deleted.

The main function of this program is to process the five types of transactions described above. The highest level of the program is divided into three parts: the initialization, the processing of transactions, and the conclusion. Initialization is done in a first-time-routine which initializes the date for reporting and checking purposes. It also initializes month tables which are to be used for verification of transaction dates and for computing julian dates. The final part of the initialization process is to correct positioning of sorted file.

All transactions are processed by the main-processing routine. This routine controls the printing of appropriate error messages as error conditions arise. Only when it has been determined that the transaction will not have a detrimental effect on the data base is any update allowed as specified by the transaction. This program is written so that a

KEYPUNCH INSTRUCTIONS
FOR
MAINTENANCE UPDATE FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
TRANS CODE	ALPHABETIC	1	1	Must always be an 'M'. Duplicate.
ACTIVITY	ALPHANUMERIC	2 - 6	5	
FUND CODE	ALPHANUMERIC	7 - 11	5	
FREE NINE MO. TENTHS	NUMERIC	12 - 16	5	Left zero fill.
USED NINE MO. TENTHS	NUMERIC	17 - 21	5	Left zero fill.
FREE TWELVE MO. TENTHS	NUMERIC	22 - 26	5	Left zero fill.
USED TWELVE MO. TENTHS	NUMERIC	27 - 31	5	Left zero fill.
FREE DOLLARS	NUMERIC	32 - 40	9	Left zero fill.
USED DOLLARS	NUMERIC	42 - 49	9	Left zero fill.

FIGURE 6.22

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U.S.A.

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAD Sheet No. 1 of 2

TR 'A'	ACT		BUDGET KEY			TYPE B	STATUS	MO BASIS	PLAN	NAME	SOC SEC NO	ANNUAL TE	ANNUAL SALARY	BEGIN DATE	END DATE	
	COLL	DPT	OBJ	RANK	LINE											
9	9 9 9	9 9	9	9 9 9 9	9 9 9	9	9 9	9 9	9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	9 9 9 9 9 9 9 9 9	9 9	9 9 9 9 9	9 9 9 9 9 9	9 9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4	5 6	7	8 9 10 11	12 13 14	15	16 17	18 19	20	21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40	41 42 43 44 45 46 47 48 49	50 51	52 53 54 55 56	57 58 59 60 61 62	63 64 65 66 67 68	69 70 71 72 73 74 75 76 77 78 79 80

TR 'A'	ACT		BUDGET KEY			TYPE 'S'	SEQ	FUND CODE	SEQ TE	S OBJ	SOURCE RANK	SOURCE LINE	
	COLL	DPT	OBJ	RANK	LINE								
9	9 9 9	9 9	9	9 9 9 9	9 9 9	9	9	9 9 9 9 9	9 9	9	9 9 9 9	9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4	5 6	7	8 9 10 11	12 13 14	15	16	17 18 19 20 21	22 23	24	25 26 27 28	29 30 31	32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

TR 'C'	ACT		BUDGET KEY			TYPE 'B'	STATUS	MO BASIS	PLAN	NAME	SOC SEC NO	ANNUAL TE	ANNUAL SALARY	BEGIN DATE	END DATE	CHG OBJ	CHG RANK	CHG LINE	
9	9 9 9	9 9	9	9 9 9 9	9 9 9	9	9 9	9 9	9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	9 9 9 9 9 9 9 9 9	9 9	9 9 9 9 9	9 9 9 9 9 9	9 9 9 9 9 9	9	9 9 9 9	9 9 9	9 9 9 9
1	2 3 4	5 6	7	8 9 10 11	12 13 14	15	16 17	18 19	20	21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40	41 42 43 44 45 46 47 48 49	50 51	52 53 54 55 56	57 58 59 60 61 62	63 64 65 66 67 68	69	70 71 72 73	74 75 76	77 78 79 80

TR 'C'	ACT		BUDGET KEY			TYPE 'S'	SEQ	FUND CODE	SEQ TE	S OBJ	SOURCE RANK	SOURCE LINE	CHG SEQ	CHG FUND CODE	
	COLL	DPT	OBJ	RANK	LINE										
9	9 9 9	9 9	9	9 9 9 9	9 9 9	9	9	9 9 9 9 9	9 9	9	9 9 9 9	9 9 9	9	9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4	5 6	7	8 9 10 11	12 13 14	15	16	17 18 19 20 21	22 23	24	25 26 27 28	29 30 31	32	33 34 35 36 37	38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

TR 'D'	ACT		BUDGET KEY			
	COLL	DPT	OBJ	RANK	LINE	
9	9 9 9	9 9	9	9 9 9 9	9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4	5 6	7	8 9 10 11	12 13 14	15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

FIGURE 6.23

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U. S. A.

Company ______________

Application BUDGET SYSTEM by ______________ Date ______________ Job No. PBU01AAD Sheet No. 2 of 2

T R 'L'	FROM ACTIVITY	FROM FUND	FROM BUDGET KEY OBJ	 RANK	 LINE	TO ACTIVITY	TO FUND	TO BUDGET KEY OBJ	 RANK	 LINE	9 MO TE	12 MO TE	DOLLARS	
9	9 9 9 9 9	9 9 9 9 9	9	9 9 9 9	9 9 9	9 9 9 9 9	9 9 9 9 9	9	9 9 9 9	9 9 9	9 9	9 9	9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4 5 6	7 8 9 10 11	12	13 14 15 16	17 18 19	20 21 22 23 24	25 26 27 28 29	30	31 32 33 34	35 36 37	38 39	40 41	42 43 44 45 46	47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

T R 'M'	ACTIVITY	FUND	FREE 9 MO TE	USED 9 MO TE	FREE 12 MO TE	USED 12 MO TE	FREE DOLLARS	USED DOLLARS	
9	9 9 9 9 9	9 9 9 9 9	9 9 9 9 9	9 9 9 9 9	9 9 9 9 9	9 9 9 9 9	9 9 9 9 9 9 9 9 9	9 9 9 9 9 9 9 9 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1	2 3 4 5 6	7 8 9 10 11	12 13 14 15 16	17 18 19 20 21	22 23 24 25 26	27 28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44 45 46 47 48 49	50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

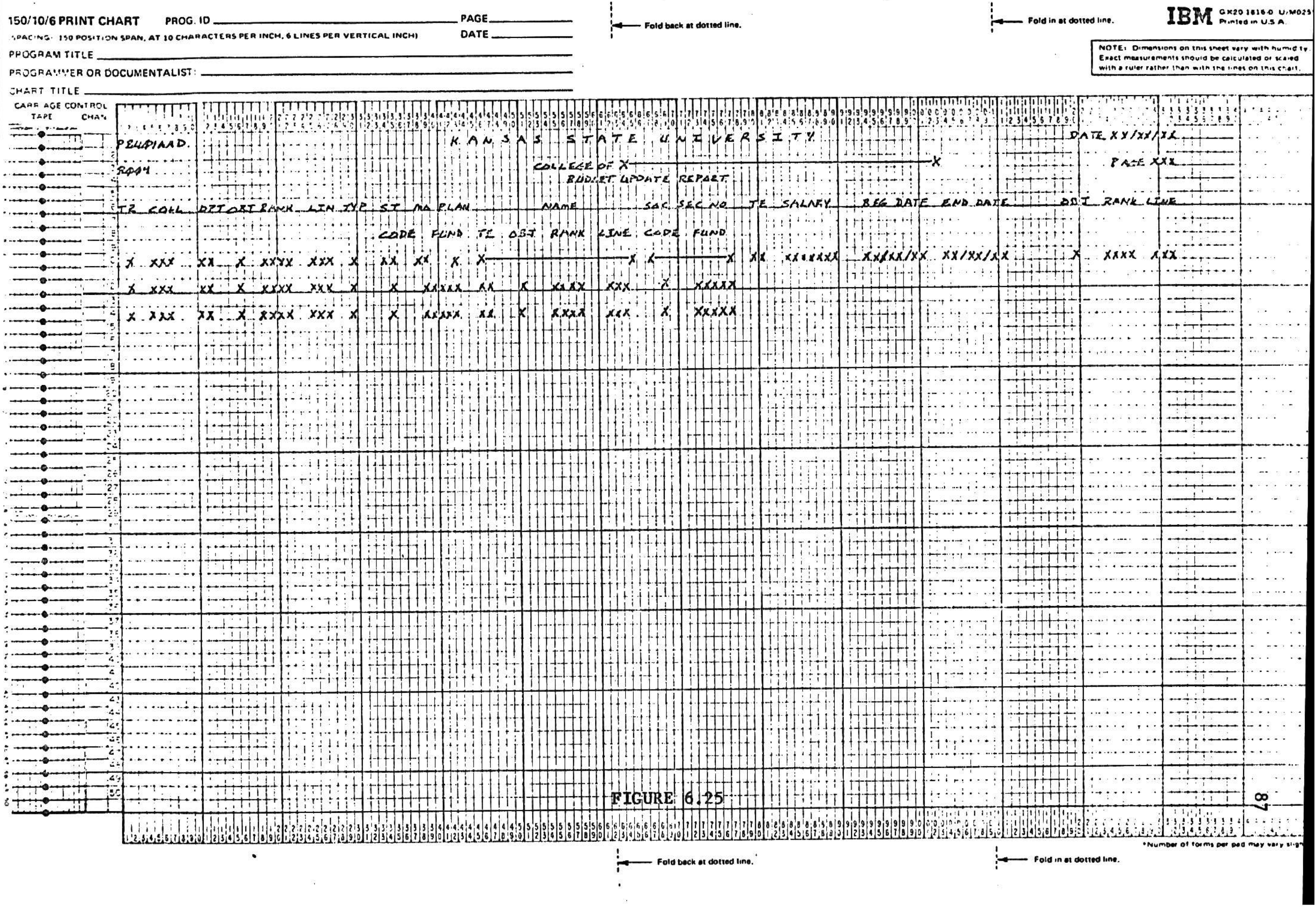
FIGURE 6.24

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80



150/10/6 PRINT CHART PROG. ID ______ PAGE ______

(SPACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST: ______

CHART TITLE ______

← Fold back at dotted line.

← Fold in at dotted line.

IBM GX20 1816-0 U/M025 Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

CARRIAGE CONTROL TAPE CHAN

```
PEUPIAAD.                                  K A N S A S   S T A T E   U N I V E R S I T Y                          DATE XY/XX/XX
R004                                                  COLLEGE OF X----------------------------------X                 PAGE XXX
                                                           BUDGET UPDATE REPORT

TR COLL DPT OBT RANK LIN TYP ST NA PLAN          NAME          SOC SEC NO  TE SALARY   BEG DATE END DATE         OBJ RANK LINE
                                 CODE  FUND  TE  OBJ  RANK  LINE  CODE  FUND

X XXX  XX  X XXXX XXX X   XX  XX   X X------------X X------X XX XXXXXXX  XX/XX/XX XX/XX/XX          X  XXXX XXX
X XXX  XX  X XXXX XXX X    X  XXXXX  XX  X  XXXX  XXX   X  XXXXX
X XXX  XX  X XXXX XXX X    X  XXXXX  XX  X  XXXX  XXX   X  XXXXX
```

FIGURE 6.25

87

← Fold back at dotted line.

← Fold in at dotted line.

*Number of forms per pad may vary slightly

150/10/6 PRINT CHART PROG. ID ____________ PAGE ______

(SPACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ____________

PROGRAMMER OR DOCUMENTALIST: ____________

CHART TITLE ____________

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M Printed in U.S.A.

NOTE: Dimensions on this sheet vary with h
Exact measurements should be calculated or sca
with a ruler rather than with the lines on this ch

CARRIAGE CONTROL TAPE CHAN

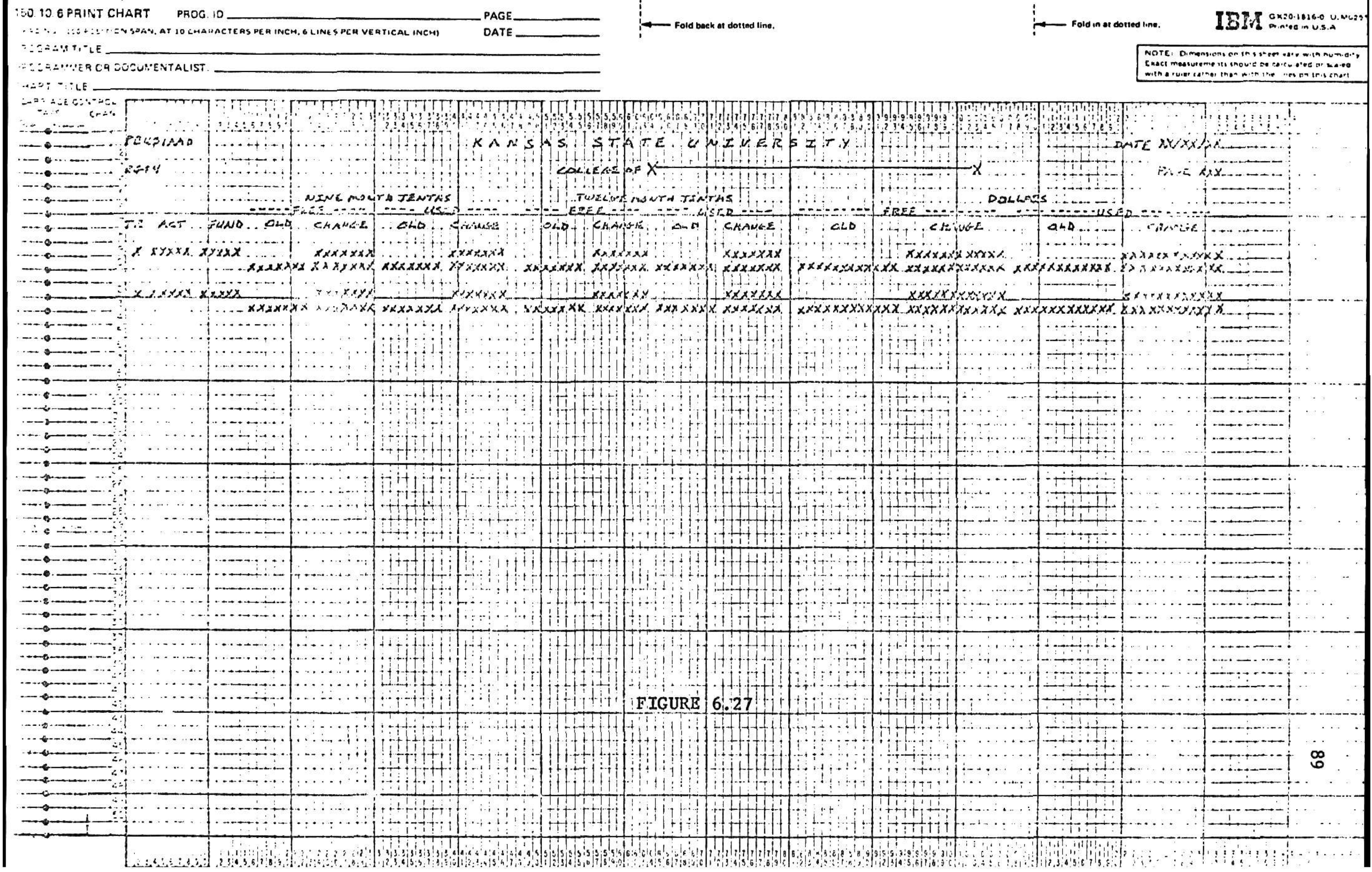
```
PBU41AAD                                    KANSAS  STATE  UNIVERSITY                                          DATE XX/XX/XX

                                                       COLLEGE OF X------------------------------------X                  PAGE XXX
R4Ø4                                                        BUDGET UPDATE REPORT

       ------------LOAN FROM--------------       -----------LOAN TO--------------   NINE MONTH     TWELVE MONTH
TR     ACT      FUND     OBJ    RANK    LINE           ACT      FUND    OBJ    RANK     LINE      TENTHS          TENTHS          DOLLARS

X      XXXXX    XXXXX     X     XXXX    XXX            XXXXX    XXXXX    X     XXXX     XXX        XXX             XXX            XXXXXXX

X      XXXXX    XXXXX     X     XXXX    XXX            XXXXX    XXXXX    X     XXXX     XXX        XXX             XXX            XXXXXXX
```

FIGURE 6.26

*Number of forms per pad may var

Fold back at dotted line.

Fold in at dotted line.



150 10 6 PRINT CHART PROG. ID ______ PAGE ______

(SPACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST ______

CHART TITLE ______

CARRIAGE CONTROL TAPE CHAN.

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M025 Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

```
PERSIAAD                         K A N S A S   S T A T E   U N I V E R S I T Y                         DATE XX/XX/XX
R544                                         COLLEGE OF X--------------------------------X                PAGE XXX
                  NINE MONTH TENTHS                 TWELVE MONTH TENTHS                      DOLLARS
              -----FREE----   ----USED----      ----FREE----  ----USED----   ---------FREE---------   ---------USED---------
T.C ACT   FUND  OLD   CHANGE     OLD   CHANGE    OLD   CHANGE    OLD   CHANGE       OLD          CHANGE         OLD         CHANGE

X XXXXX XXXXX       XXXXXXX        XXXXXXX         XXXXXXX         XXXXXXX            XXXXXXXXXXXXX              XXXXXXXXXXXXX
             XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX

X XXXXX XXXXX       XXXXXXX        XXXXXXX         XXXXXXX         XXXXXXX            XXXXXXXXXXXXX              XXXXXXXXXXXXX
             XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX
```

FIGURE 6.27

transaction causes actions based on the presence or absence of data contained in designated card fields. Further discussion will center around each transaction and the actions evoked by certain fields in the transaction.

The add transaction causes budget occurrences to be created and stored in the data base. Initially all data related to the budget is checked for errors. All numeric fields are verified. In some instances range checking is done. The number of days in the month must be correct. The years on the effective dates must be within the specified bounds of the fiscal year. All of this checking is necessary because budgeted dollars are computed on the actual number of working days that an employee works and not on monthly basis.

Once all line information has been checked, the service records for the budgeted line are read in and verified. Fund codes are compared with those allowed within a college. If the line number field has been coded it is the user's intent to draw funding for this line from another line. Verification of the existence of that line must be made, however, existence of a corresponding fund is assumed.

Dollars and tenths for any given line are broken down into those which are currently assigned (in use) and those which are currently unassigned (free). The method of determining whether tenths and dollars are in use is through the contents of the status code. If the status code for a line currently budgeted for an employee indicates that the position is vacant or the employee is on leave without pay, all tenths and dollars are designated free. If a line budgeted for an employee is currently filled, the tenths and dollars are designated used. If the line represents an employee who is taking a sabbatical leave, the tenths are free.

Further, if the period for which the linc is budgeted is greater than one-half of the monthly basis, one-half of the dollars are designated as free, the other half used. Otherwise all budgeted dollars are declared used.

Once it has been determined that the budget occurrence may be correctly stored on the data base, the department-fund junction occurrences are updated reflecting the free and used dollars and tenths by funding source. If there are no errors at this point, the budget occurrence is stored on the data base.

Change transactions are similar to add transactions in that error checking is performed by the same routines for both budget and service information. The user indicat the desire to make a change by first identifying the record by its activity, object code, rank, and line number. In order to change information on budget occurrences, the fields on the data card must contain non-blank values which correspond to fields on the budget occurrence. In all cases the field on the data card will overlay the field on the budget occurrence.

Certain changes will evoke actions to be taken by the program. Any change in the data base key of an occurrence will cause IDMS to reorder data base pointers. A change in status may cause the budget for that line to be recalculated depending upon the new status. If the line represents a filled position, and the new status represents a filled position, the new status overlays the old status. If the old status represents a filled position, and the new status depicts a vacant position, it has the effect of a termination. The actions are to redesignate the tenths as free. The budgeted dollars are recalculated and that portion which have not been used are designated as free. If the new status represents a change from a filled position to one reflecting a sabbatical leave the following actions are evoked. All tenths become free. If the period remaining after date calculations have been made is greater than one-half of the monthly basis, then one-half of the dollars become free, the other half remain used. If the new status represents a termination or a leave without pay all the tenths are flagged as free. The budget is recalculated and any unused dollars become free.

Changes in effective dates, annual salary, and annual tenths cause a line to be recalculated. Certain changes in service information such as service tenths and source of funding also evoke a similar action.

If a change to a line is acceptable to the program, department-fund junction occurrences corresponding to sources of funding must be modified successfully. Only then is the budget occurrence modified.

The delete transaction is used only when a severe mistake has been made or when removing a line from the budget. The actions evoked by this transaction are to find the budget occurrence on the data base corresponding to a key on the data card. If located, all budgeted dollars and tenths are effectively zeroed out. This must be reflected back to the department-fund junction occurrences which correspond to the sources of funding in the budget occurrence.

The loan transaction is used when a department wants to borrow tenths and/or dollars from another department. The 'from' department is loaning the dollars and/or tenths. The 'to' department is borrowing the tenths and/or dollars. Tenths are maintained by monthly basis within their respective departments.

The action evoked by this transaction is to retrieve the department-fund junction occurrence corresponding to the fund of the 'from' department. The dollars and/or tenths are added to the from fields in the junction occurrence. The junction occurrence which corresponds to the fund of the 'to' department is retrieved and the dollars and/or tenths are added to the borrowed fields of the junction occurrence. In both instances the junction is modified and replaced on the data base.

The maintenance transaction has been provided to correct errors in department-fund junction occurrences. As with all new software systems certain oversights will be made. It is entirely possible for certain types of add or change transactions to be perfectly valid, but the combination of add or change requests will evoke certain actions which will cause an erroneous update of junction occurrences. Accidents such as these will happen until the actions specified by certain transactions are completely specified or limitations on their use are established.

This transaction gives the user the ability to modify any junction occurrence within a department by fund which relates to free and used dollars, nine month basis tenths, and twelve month basis tenths. When the proper occurrence is located fields are changed by adding the contents of fields which have non-zero amounts on the data card to the contents of the junction occurrence. For each transaction the report will show the contents of the card and the contents of the junction occurrence fields before and after the change.

Certain routines performed through add and change transactions are worth mentioning. The first is the routine which calculates the number of working days. Part of the requirement for this routine is that it be able to calculate for any line the number of months plus the fractional part of a month which is to be computed by dividing the number of days worked by the number of working days in the month. Part of the requirement here is that this routine accepts calendar dates as input and computes the number of months plus the fractional part accurate to five positions to the right of the decimal point. Once this calculation has been performed the result is multiplied by the annual salary to obtain the budgeted dollars for that line.

The routine discussed in the previous paragraph is derived from the following formula.[5]

$$\text{Day-of-week} = (\ (2.6M - .2) + D + Y + (Y/4) + (C/4) - 2C\ ) \bmod 7$$

Where M = month number (January = 11, February = 12, March = 1, ... , December = 10)
D = day of the month
Y = year number (last two digits)
C = century number.

Expressions enclosed within parentheses mean that the result is truncated to the integer value. The result of this calculation gives the day of the week as a number where 0 is Sunday, 1 is Monday, etc.

A special routine which converts calendar dates to julian dates is also incorporated into the program because it is necessary to compare dates. Comparing calendar dates on the basis of month, day, and year is almost impossible, especially if the fiscal year overlaps two calendar years.

As noted previously, all junction occurrences are maintained by the program. If new funds are needed for a department, the program automatically creates a new junction occurrence. If at any time all fields within any junction occurrence are zeroed out, the record is deleted. Maintenance transactions can be used to zero fields so that these records can be deleted.

6.5 THE BUDGET REPORT PROGRAM:

Program PBU01AAE prints the budget report for selected or all departments. Figure 6.28 illustrates the input and output requirements. Selection cards are input to the program. The document which initiates the procedure by which they are keypunched is shown in figure 6.29 and the corresponding keypunch instructions are shown in figure 6.30. Actual record descriptions for the card record and print record are illustrated in figures 6.31 and 6.32 respectively.

Subschema BU01SS05 is invoked by this program for retrieval of records only. The three record types described therein are C-COLLEGE-REC, D-DEPT-REC, AND B-BUDGET-REC. Two set relationships are also required, the COLL-DEPT and DEPT-BUD sets.

This program has been written so that if the word 'ALL' is entered as the department code for a given college, all departmental budgets are listed for that college. Otherwise, the user may specify which departments are to be printed by entering the department code in the proper field.

If all departments within a college are desired, the program uses the college occurrence as the primary entry point into the data base. Then each department in order is selected until the last is reached. Meanwhile, as each department is selected all budget occurrences for that department are serially selected and their contents printed. Running totals are kept on assigned and unassigned tenths and dollars. When the last budget occurrence has been reached these totals are printed to show the number of free dollars and tenths for the department.

If selected departments are requested, the department occurrence is used as the primary entry point into the data base. Budget occurrences are retrieved and totals printed as described in the previous paragraph.

BUDGET LISTING PROCEDURE

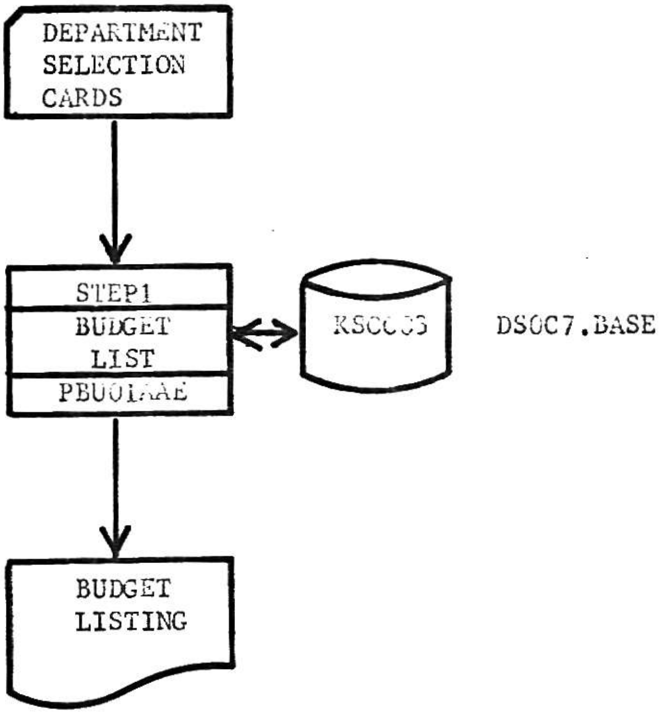


FIGURE 6.28

BU01-005

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

BUDGET LISTING
SELECTION FORM

PASS[1] CODE	COLLEGE[2] CODE	DEPT[3] CODE	PASS CODE	COLLEGE CODE	DEPT CODE	PASS CODE	COLLEGE CODE	DEPT CODE
R005	____	____	R005	____	____	R005	____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____

1. The pass code is a four position code which is required to insure that the selection cards are processed by the correct program.

2. The college code is the first three digits of the activity code for any department.

3. This program will print a budget listing for all or selected departments. To obtain all departments within a given college enter the college code followed by the word 'ALL' in the department field. For selected departments enter the department codes of the desired departments.

FIGURE 6.29

KEYPUNCH INSTRUCTIONS FOR BUDGET LISTING SELECTION FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
PASS CODE	ALPHANUMERIC	1 - 4	4	Duplicate pass code on all data cards.
COLLEGE CODE	ALPHANUMERIC	5 - 7	3	
DEPT CODE	ALPHANUMERIC	8 - 10	3	If two digit department code is entered, left justify in the field. If the word 'ALL' is entered, keypunch it in this field.

FIGURE 6.30

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U. S. A.

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAE Sheet No. 1 of 1

PASS CODE 'ROOS'	COLL	'ALL'	
9 9 9 9 1 2 3 4	9 9 9 5 6 7	9 9 9 8 9 10	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

PASS CODE 'ROOS'	COLL	DPT	
9 9 9 9 1 2 3 4	9 9 9 5 6 7	9 9 8 9	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

FIGURE 6.31

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

150/10/6 PRINT CHART PROG. ID ______ PAGE ______

(SPACING: 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST: ______

CHART TITLE ______

← Fold back at dotted line. ← Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

CARRIAGE CONTROL TAPE CHAN

```
PRUGIAAE                          K A N S A S   S T A T E   U N I V E R S I T Y                       DATE XX/XX/XX
                                                                                                         PAGE XXX
R005                                                    COLLEGE OF X---------------------------X
                                                            BUDGET LISTING



                                                            EFFECTIVE DATES   NO  MONTHLY ANNUAL BUDGETED   USED      FREE
OBJ RANK  LINE        NAME         SOC SEC NO STAT  TE   BEG DATE END DATE SER SALARY SALARY DOLLARS  TE DOLLARS TE DOLLARS CD FUND
          NO
 X  XXXX  XXX  X-------------------X X---------X  XX  XX  XX/XX/XX XX/XX/XX XX  XXXXXX- XXXXXX-  XXXXXX-
 X  XXXX  XXX                                     XX                                  XXXXXX-  XXXXXX- XX-XXXXXX- XX-XXXXXX- X XXXXX
 X  XXXX  XXX                                     XX                                  XXXXXX-  XXXXXX- XX-XXXXXX- XX-XXXXXX- X XXXXX





                     ---- NINE MONTH TENTHS ----      --- TWELVE MONTH TENTHS ---      ----------------- DOLLARS ------------------
                     USED       FREE       ALLOC      USED      FREE      ALLOC          USED           FREE            ALLOC
***** DEPARTMENT TOTALS  XXXXXXX  XXXXXXX  XXXXXXX   XXXXXXX   XXXXXXX   XXXXXXX    XXXXXXXXXXXXX  XXXXXXXXXXXXX  XXXXXXXXXXXXX
```

FIGURE6.32

*Number of forms per pad may vary slightly

6.6 THE DEPARTMENTAL ANALYSIS PROGRAM:

Program PBU01AAF prints the departmental analysis by fund report for selected or all departments. Figure 6.33 illustrates the input and output flow requirements. Again the contents of the input selection cards determine whether one, several, or all departments are to be analyzed. The document which is used to initiate the procedure and the creation of these cards is shown in figure 6.34 and the corresponding keypunch instructions in figure 6.35. Record descriptions for the card record and the print record are shown in figures 6.36 and 6.37 respectively.

Subschema BU01SS06 is invoked by this program for retrieval of records only. Three record types (C-COLLEGE-REC, D-DEPT-REC, and DF-DEPT-FUND-REC) and two set relationships (COLL-DEPT and DEPT-FUND) are described.

If the word 'ALL' is entered as the department code in one of the selection cards, all departments will be listed for the selected college. When this occurs the college occurrence acts as the primary entry point into the data base. All member department occurrences are selected in order. As each is selected all member department-fund junction occurrences are serially selected and their contents printed. Running totals for allocated dollars and tenths are kept until the last junction occurrence has been processed whereupon the totals are printed.

If the actual department code is entered in the selection card, the department occurrence acts as the primary entry point into the data base. All of its member junction occurrences are processed in the manner described in the previous paragraph.

DEPARTMENT ANALYSIS PROCEDURE

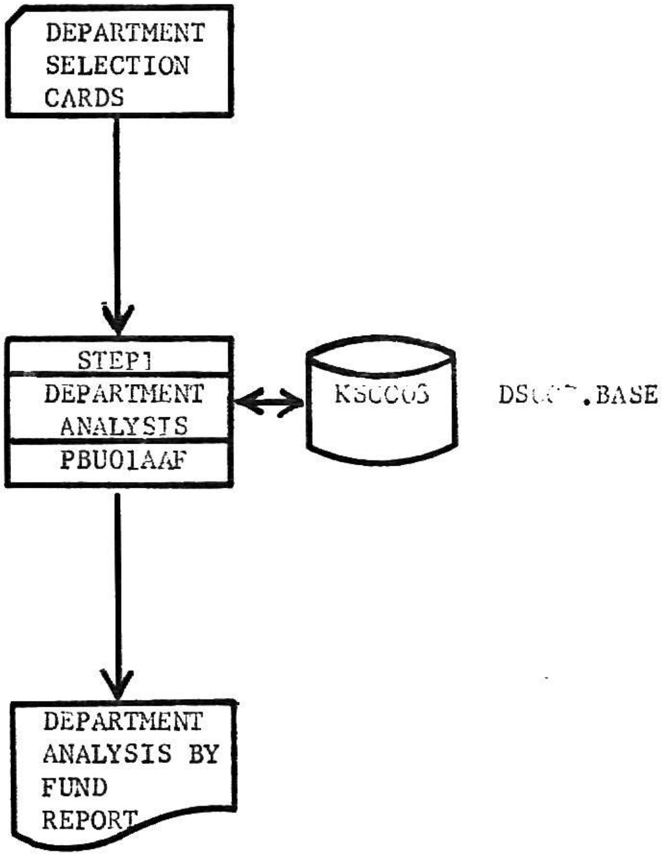


FIGURE 6.33

BU01-006

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66502

DEPARTMENT ANALYSIS
SELECTION FORM

PASS CODE[1]	COLLEGE CODE[2]	DEPT CODE[3]	PASS CODE	COLLEGE CODE	DEPT CODE	PASS CODE	COLLEGE CODE	DEPT CODE
R006	____	____	R006	____	____	R006	____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____
	____	____		____	____		____	____

1. The pass code is a four position code which is required to insure that the selection cards are processed by the correct program.

2. The college code is the first three digits of the activity code for the department.

3. The program will print a departmental analysis by fund source for all or selected departments. To obtain all departments within a single college enter the college code and follow it by the word 'ALL' in the department field. To print selected departments enter the desired department codes.

FIGURE 6.34

KEYPUNCH INSTRUCTIONS
FOR
DEPARTMENT ANALYSIS
SELECTION FORM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
PASS CODE	ALPHANUMERIC	1 - 4	4	Duplicate pass code on all data cards.
COLLEGE CODE	ALPHANUMERIC	5 - 7	3	
DEPT CODE	ALPHANUMERIC	8 - 10	3	If two digit department code is entered, left justify it in the field. If the word 'ALL' is entered, keypunch it in this field.

FIGURE 6.35

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U. S. A.

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAF Sheet No. 1 of 1

Columns 1–4	Columns 5–7	Columns 8–10	Columns 11–80
PASS CODE 'ROO6'	COLL	'ALL'	

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

Columns 1–4	Columns 5–7	Columns 8–9	Columns 10–80
PASS CODE 'ROO6'	COLL	DPT	

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

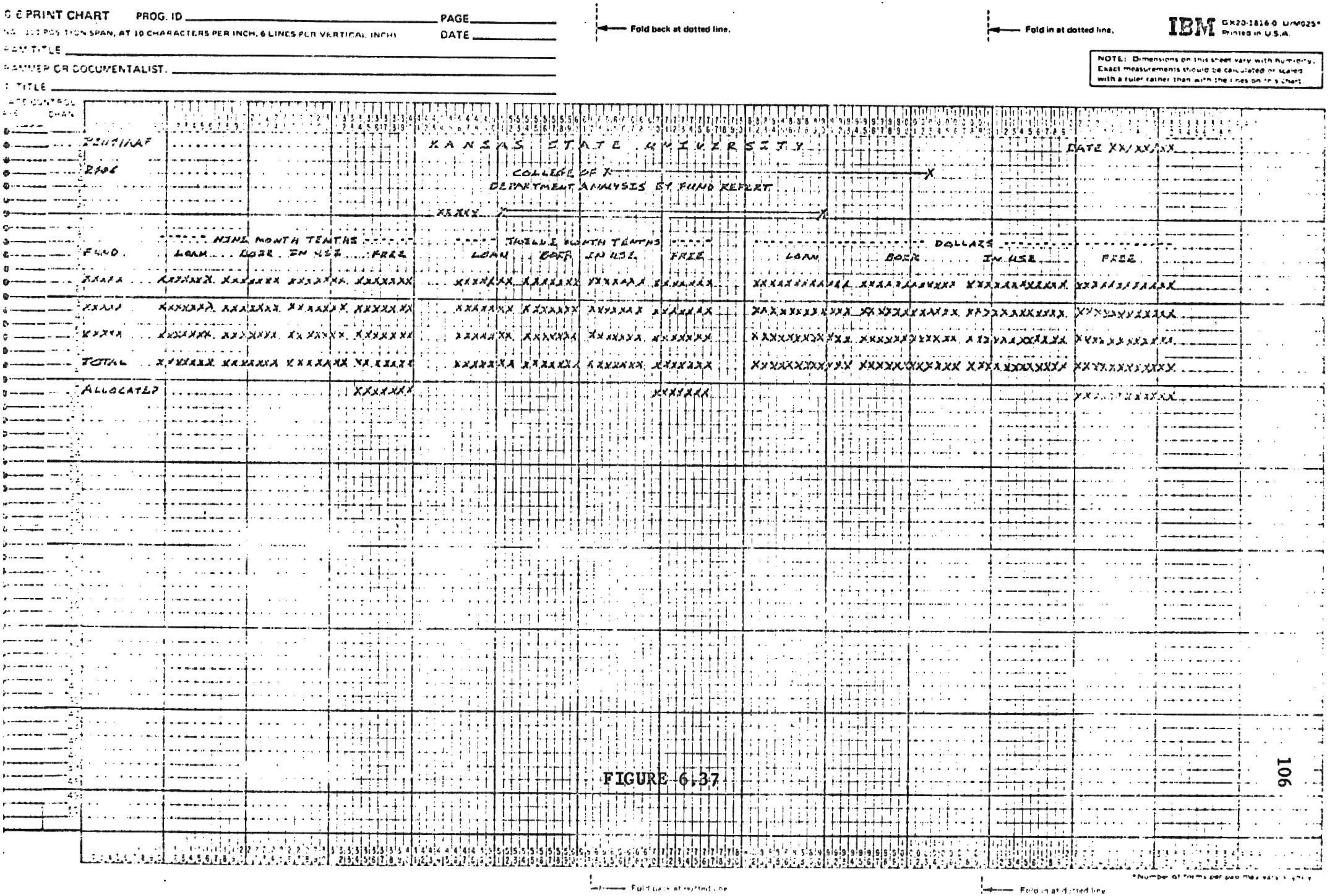
FIGURE 6.36

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```



6 E PRINT CHART PROG. ID ______________ PAGE ______

(150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

AM TITLE ______________

AMMER OR DOCUMENTALIST ______________

TITLE ______________

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

```
PLUTMAF                                  K A N S A S   S T A T E   U N I V E R S I T Y                          DATE XX/XX/XX

2906                                           COLLEGE OF X------------------------------X
                                          DEPARTMENT ANALYSIS BY FUND REPORT

                                     XXXXX  X-------------------------------X

        ------ NINE MONTH TENTHS ------      ----- TWELVE MONTH TENTHS -----    ---------------------- DOLLARS ----------------------
FUND.   LOAN     BORR     IN USE   FREE      LOAN     BORR     IN USE   FREE     LOAN          BORR          IN USE        FREE

XXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX  XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX

XXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX  XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX
XXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX  XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX
TOTAL   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX   XXXXXXX. XXXXXXX. XXXXXXX. XXXXXXX  XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX. XXXXXXXXXXXX

ALLOCATED                          XXXXXXX                              XXXXXXX                                            XXXXXXXXXXXX
```

FIGURE 6.37

*Number of forms per pad may vary slightly

6.7 THE INITIAL LOAD PROGRAM:

Program PBU01AAG is a subset of program PBU01AAD. As shown if figure 6.38, the primary difference is the input into the program. A tape file obtained from the Data Processing Center is used to initially load and then to reload the data base each fiscal year. A record description for this tape file is shown in figure 6.39. In addition there are two types of control cards which are input to this program. The first is a fiscal year card which specifies the beginning and ending fiscal years. If this card is absent, the program will abort with a return code of twelve (12). The second control card specifies the college code of the budgetary data that is to be deleted before the reinitialization process begins. See figures 6.40 and 6.41 for the keypunching instructions and the record description of the cards. The printed report description is shown in figure 6.42. It is almost identical to the report produced by PBU01AAD. Sbuschema BU01SS04 is invoked for the purpose of exclusive update. This subschema contains all records and set relationships described in the schema.

Initially all records on the tape file are to be accessed and examined. Only unclassified employees, object code of '2', are to be selected. As records are selected only the pertinent information is removed from the tape record and copied into records similar to the card records used in program PBU01AAD. The new records are then sorted into the desired sequence for processing.

The initialization routine has been abbreviated to positioning of the sorted file and the deletion of all records for the college specified on the input control card. If there are no control cards, the initialization routines are ignored. If a control card specifies a college code for a college record which does not exist on the data base, it is also ignored.

INITIAL LOAD PROCEDURE

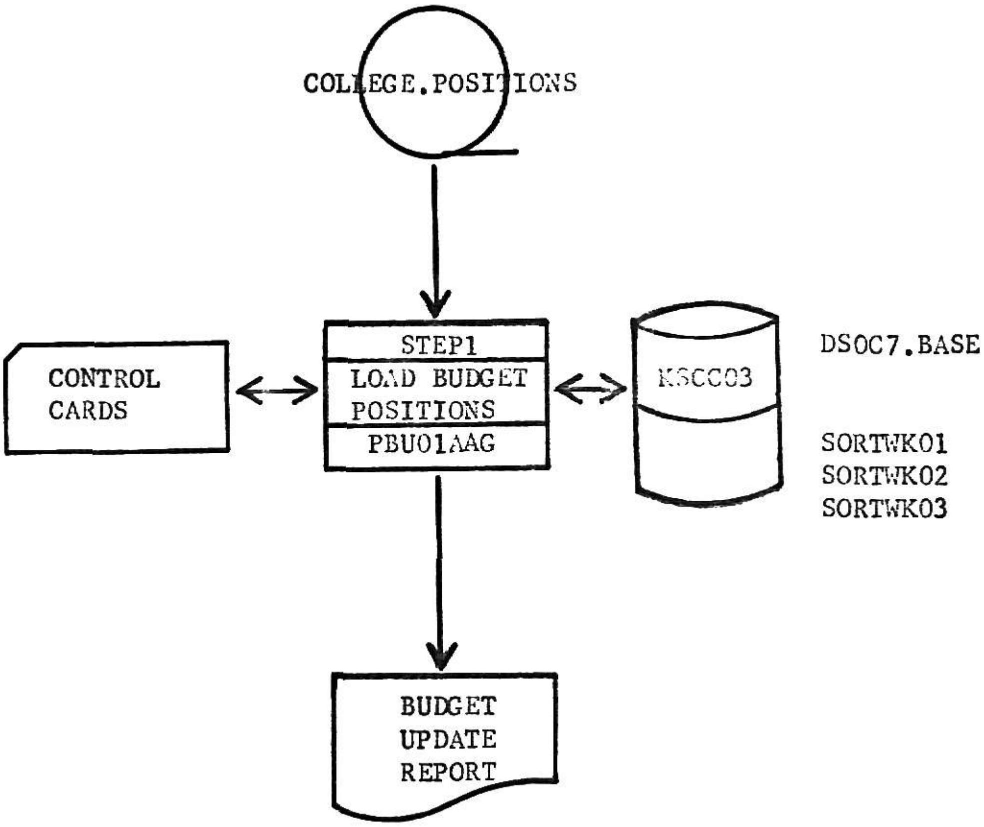


FIGURE 6.38

IBM RECORD FORMAT

INTERNATIONAL BUSINESS MACHINES CORPORATION

GX20-1702-1 UM/025 1
Printed in U.S.A.

APPLICATION . BUDGET . . . SYSTEM . . . . RECORD NAME . CURRENT YEAR BUDGET . . . . BY . . . . DATE . . . . PAGE . 1 . . OF . 1 . . . . . .

Field Name	ACTIVITY COLL	 DEPT	[illegible]	RANK CODE	LINE NO	STAT	MTHLY BASIS	FILL	NAME	FILLER	ANN. TE.	ANNUAL SALARY	FILLER	[illegible]	STFE. F.	FILLER
Characteristics*	C	C	C	C	C	C	Z		C		Z	Z		C	Z	
Position**																

FILLER	FUND CODE	FILLER
	C	

**POSITION
Hexadecimal / Decimal
Numbering
from
00 to FF / 0 to 255

HEXADECIMAL
DECIMAL

HEX
DEC

HEX
DEC

HEX
DEC

File Description STANDARD SEQUENTIAL
Recording Mode FIXED
Records per Block 10
Record Size 120
Label Records are STANDARD
File Identification COLLEGE-POSITION
File Serial Number
Retention Cycle
Organization Type

fold to here

*CHARACTERISTICS

Check the box that corresponds to the characteristics used:

☒ System/360 Characteristic Codes

A - address value, full word
B - binary
C - character, 8-bit code
D - floating point, double word
E - floating point, full word
F - fixed point, full word
H - fixed-point, halfword
P - packed decimal
S - address, base displacement
V - address, external symbol
X - hexadecimal, 4-bit code
Y - address value, halfword
Z - zoned decimal

☐ General Characteristics

A - alphabetic or blank
X - alphanumeric
9 - numeric
V - assumed decimal point

Examples of Signed Fields:
X9999 999X
X999V99 9999V9X

FIGURE 6.39

SORTING FIELDS (Major to Minor)	
1	7
2	8
3	9
4	10
5	11
6	12

WHERE USED	
Input From	Output To

Date	Revisions By

REMARKS

† The number of forms per pad may vary slightly.

KEYPUNCH INSTRUCTIONS FOR INITIAL LOAD PROGRAM

FIELD NAME	CHARACTERISTICS	POSITION	LENGTH	REMARKS
	DATE CONTROL CARD			
OLD FISCAL YEAR	NUMERIC	1 - 2	2	
FILLER		3	1	Always a '-' or a space.
NEW FISCAL YEAR	NUMERIC	3 - 5	2	
	COLLEGE CONTROL CARD			
COLLEGE CODE	ALPHANUMERIC	1 - 3	3	

FIGURE 6.40

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U. S. A.

Company ______

Application BUDGET SYSTEM by ______ Date ______ Job No. PBU01AAG Sheet No. 1 of 1

OLD YR. | FILLER | NEW YR.

DATE CONTROL CARD

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

COLL CODE

COLLEGE CONTROL CARD

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

FIGURE 6.41

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

```
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
```

150/10/6 PRINT CHART PROG. ID ______ PAGE ______

(SPACING 150 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST: ______

CHART TITLE ______

Fold back at dotted line.

Fold in at dotted line.

IBM GX20-1816-0 U/M025* Printed in U.S.A.

NOTE: Dimensions on this sheet vary with humidity. Exact measurements should be calculated or scaled with a ruler rather than with the lines on this chart.

CARRIAGE CONTROL TAPE CHAN

```
PELIQIANG                                  K A N S A S   S T A T E   U N I V E R S I T Y                        DATE XX/XX/XX

R447                                                COLLEGE OF X------------------------------------------X                   PAGE XXX
                                                         INITIAL BUDGET LOAD

TR COLL DPT OBJ RANK LIN TYP ST MO PLAN          NAME            SOC SEC NO  TE  SALARY  BEG DATE END DATE

                                 CODE FUND TE

X XXX XX X XXXX XXX X XX XX X X--------------------X X----------X XX  XXXXXXX XX/XX/XX XX/XX/XX

X XXX XX X XXXX XXX X  X XXXXX XX

X XXX XX X XXXX XXX X  X XXXXX XX
```

FIGURE 6.42

*Number of forms per pad may vary slightly

Fold back at dotted line.

Fold in at dotted line.

No initialization of date tables is required here because budgeting dates are known and do not need to be calculated.

The only transactions to be processed by this program are additions. The status code determines whether tenths and dollars are to be designated as free or in use. Budgeted dollars are computed on the basis of nine or twelve month periods only. This reduces the number of routines needed to perform calculations by eliminating all date computations.

The process of creating and updating the department-fund junction occurrences has been simplified. A temporary table is created which maintains budget totals by fund until the budget record can correctly be stored in the data base. Once this is accomplished, the junction records are updated, then the budget occurrence is stored.

CHAPTER VII

CONCLUSION

7.1 DISCUSSION:

This master's report has covered the basic aspects for the development and implementation of a budgetary system for the dean's office. It includes a system of programs that gives the user complete control of all data and provides backup and recovery procedures. All features pertinent to the development of the system have been discussed.

This system has developed out of a need by the dean's secretary to provide a more efficient method of maintaining a budget for unclassified employees. It should provide a basis for learning and development which will lead to further improvements in that system. At present, limitations exist in the budget update program on certain transactions. However, a proper combination of transactions can in most instances produce the desired result.

Even though the implemented system satisfies the minimal requirements for the dean's office, several extentions to it may be made immediately. Currently only unclassified employees are being maintained on the data base. There is no restriction on the types of budgetary lines that may be added to the data base. The current version of the budget listing program prints only final totals for each department. If additional budgetary information, such as lines for classified employees, or operating expenses, are added to the data base the current budget listing program can be modified slightly to provide subtotals by object code.

This system can be extended to other colleges and departments at Kansas State University through normal update procedures. The budgetary data must be processed in a manner similar to that of the College of Arts and Sciences.

At present one of the report programs provides a summary of funding information for each department. The schema has been designed to allow the immediate development of a program which will provide departmental summary information for each fund within a college. One requirement includes the description of another subschema to allow retrieval of college, fund, and departmental-fund junction occurrences. A slight modification of the existing departmental summary program could be used to produce a fund summary program.

A program which would require some further analysis, could be developed to use information from the existing data base. It would be used to predict the budget for the upcoming fiscal year. Input to this program would be a series of data cards each of which would specify percentage increases groups of lines designated by object and class codes. Provisions could be included to simulate the addition or deletion of lines as specified on input cards. The program would then print a predictive report that would be similar to the annual budget listing for each department.

University procedures required for the production of the annual budget listing include the manual update of a deck of cards distributed to the dean's office by the budget office. A program could be written which would analyze these cards. It would read each card, list it, analyze it field by field, and print any discrepencies between the card and the current information for the corresponding line on the data base. If a card were present, but no corresponding line existed on the data base, it would be flagged for removal. If an occurrence existed on the data base, but no corresponding card existed, a message would be printed to indicate that another card needed to be keypunched.

If the system developed as a result of this master's project were to be redesigned, the following features would be included. A modified version of the current schema (See figure 7.1) would include the description of two new record types and three set relationships. Originally service information for each line was very small. The creation of an additional record type could not be rationalized because the pointer overhead exceeded the record length. Subsequently, this information was stored as a repeating group-item occurring ten times. As the item increased in length by the end of the third refinement, the overhead became tremendous. The typical line has an average of two services up to a maximum of four. If only four group-items were used, and ten allowed, 144 bytes of space per record would be unused. The development of a service record is warranted because the space required for pointers is now much less than the unused or wasted space. This would result in a considerable storage space savings since the budget and service records occur most frequently on the data base.

A service junction record could be developed to join those lines which are deriving tenths and/or dollars from other lines within a department. This, however, would add additional pointer overhead to each service record, but would be offset by the removal of the space required to maintain the source key which is currently required. This junction record would provide added flexibility to the system in the event that the key of the source line changed. Under the current version of this system, the source key of the source line is stored within each group-item occurrence. A change in the key of the source line would cause a loss of data base integrity. However, this should never occur because tenths and dollars are normally borrowed from lines that are budgeted for the entire fiscal year.

MODIFIED SCHEMA

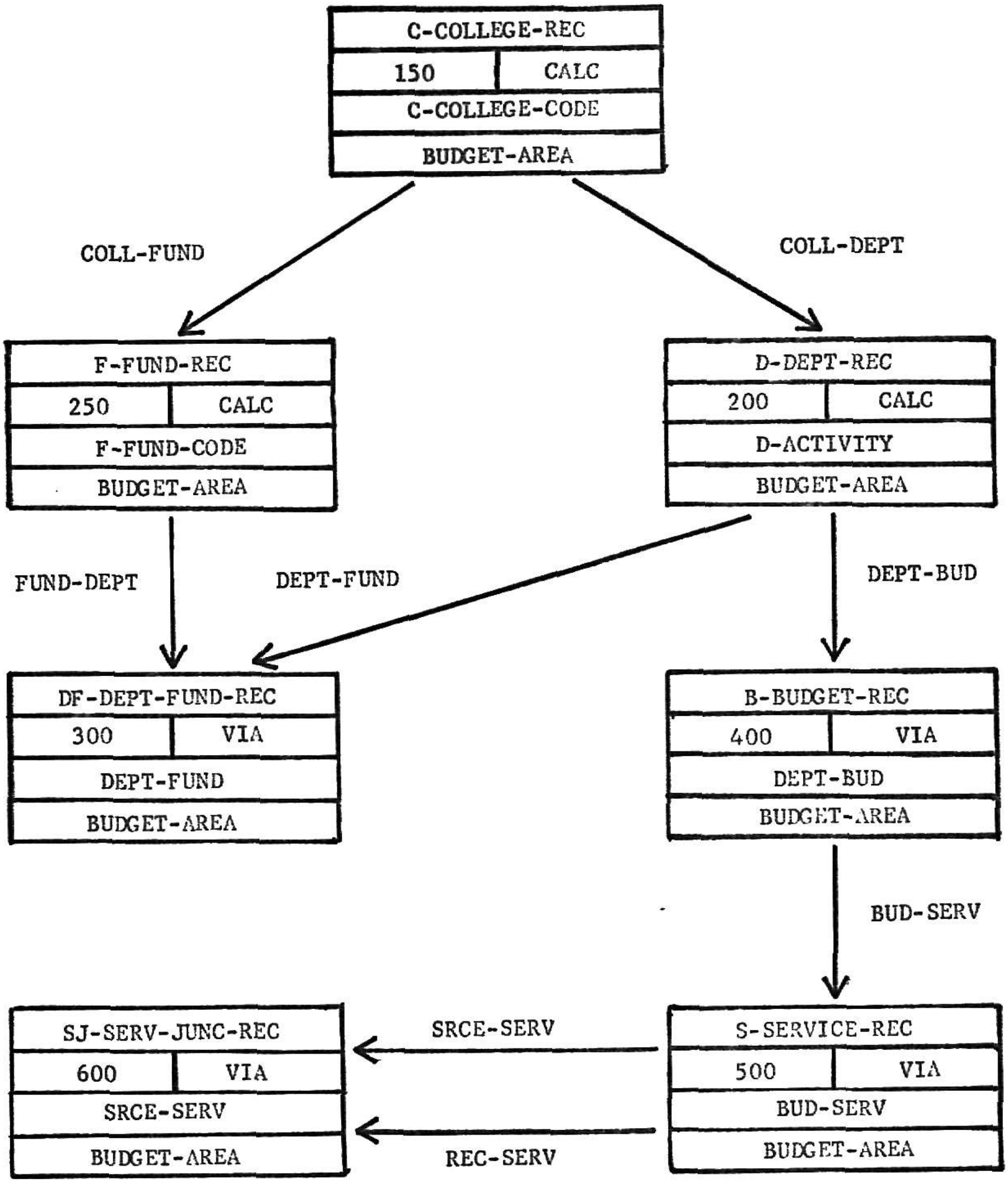


FIGURE 7.1

New set relationships required by the modified schema are listed as follows. A relation must exist between the owner budget record and the service record. Two other set relations would require the service record as the owner and the service junction record as the member. One service record would be the source of tenths and/or dollars, the other the receiver of the tenths and/or dollars.

Development of a system of software programs using a data base management system requires a significant amount of time and money. The cost factor prohibits the actual development of the system by interactive means. However, once the system has been developed, CMS may be used to communicate to the data base from a remote terminal. The system as developed currently would require that the terminal replace a card reader and a printer. Data would have to be entered on a terminal field by field as though it were coming from a card. This is necessary because Cobol requires record oriented input. Interactivity must be viewed in terms described above for the current implementation. The current budget update program could be rewritten as a truly interactive program, but it would increase somewhat in size.

The development of the budgetary system for the College of Arts and Sciences has provided its designer with an appreciation of data base management systems. Initial phases of the design required a considerable amount of learning. As the system grew, further knowledge was gained by experimenting with different techniques of combining data into meaningful structures in the form of records and set relations. In addition, this project has reinforced the idea that sound software systems are developed with much time being devoted to their analysis and design. A data base management system in a sense enforces this concept because a schema must exist before much of the developmental work can begin.

BIBLIOGRAPHY

1. Ljundahl, David Joe; Feasability of a Data Base Management System for the College of Arts and Sciences, A Master's Report, Department of Computer Science, Kansas State University, Manhattan, Kansas.

2. Data Definition Languages, Utilities and GCI Reference Guide; Release 3.1, Revision 1, Cullinane Corporation, Boston, Mass. April 1975.

3. Data Manipulation Language Programmer's Reference Guide; Release 3.1, Cullinane Corporation, Boston, Mass., April 1975

4. Codasyl Data Base Task Group Report; Association for Computing Machinery, 1133 Avenue of Americas, New York, N.Y., April 1971.

5. Scott, Sandak; PL/1 for Beginners; Addison-Wesley, Reading, Mass., 1970, p. 151.

APPENDIX A

BUDGETARY SOURCE DOCUMENTS

KANSAS STATE UNIVERSITY
Manhattan, Kansas 66506

PER-1
(4-75-10M)

SOURCE OF SALARY

COMPLETE AND SUBMIT AS FOLLOWS

Unclassified appointments—two copies, attached to DA-220.*

Classified appointments—one copy, attached to DA-216*

Changes in source of salary: (submit prior to effective date).

1. Unclassified—original to Personnel Services and duplicate to Comptroller's Office.*
2. Classified—one copy to Comptroller's Office.*

Please submit this form in duplicate in lieu of an appointment paper in those cases where a civil service form is not required but a minute must be submitted to the Board of Regents to correct the budget.

ne copy to Director, if joint appointment, and additional copies as required by Dean.

ame________________________ Social Security No.______________ Budget or Position No.______________

epartment________________________ Rank or Classification________________________

ppointment □ Source of Salary Change Only □ Delete from Annual Budget □

UNCLASSIFIED EMPLOYEE:

Effective Date: Beginning________________________ Ending #________________________

Will be included on a regular line in the next annual budget? # Yes □ No □

Mo. Basis: Check One: 9 □ 12 □ 9 paid in 12 □ Monthly Rate (Ten Tenths Basis) $__________

Annual Salary	Monthly Salary by Source	Tenths Time by Source	Fund Name**	Fund Number**	Account Number**	Project Number**
	$					
	$					
	$					

For appointment only. (9 month appointments end each May 31 and 12 month appointments end each June 30)

When paid from Faculty Salaries, use this space to insert "Faculty Salaries" and the budget line reference to the source of funds and tenths of time.

CLASSIFIED EMPLOYEE:

Effective Date________________________

Monthly Salary or Hourly Rate

____ $__________
Departmental Account Name and Number

______________ ________________________ $__________
Fund Name — Account Name and Number

______________ ________________________ $__________
Fund Name — Account Name and Number

tal .... $__________

______________ ______________ ______________
Date — Department Head — Academic Dean or Vice President

______________ ______________
Director — Vice President for Business Affairs

2. Blue—V. P. Business Affairs
3. Yellow—Dean after approval
4. Pink—2nd Dean after approval
5. Green—Dean's hold copy, becomes Department approval
6. Orange—Dept. hold copy, becomes Employee's notice

FORM GST
(9-74-1M)

GRADUATE STUDENT TRANSACTION

KANSAS STATE UNIVERSITY

Manhattan, Kansas

Social Security Number ____________ Name ____________
Last First Initial

Effective date ____________ Ending Date* ____________

Department of ____________ FLSA STATUS: GA—Covered; GTA, GRA & Fellow-Exempt

Type of Transaction:

□ Appointment □ Rank Change □ Salary Change □ Source of Salary Change □ Tenths Time Change
□ Termination □ Other Change (Specify) ____________

Rank:

□ Graduate Assistant □ Graduate Teaching Assistant □ Graduate Research Assistant □ Fellow†

Previously or presently employed (other than as an hourly student) at K. S. U. during current fiscal year: Yes____; No____.

If yes, show: Budget Line No. ____________ Department ____________

Budget Line No.	Mo. Basis 9	Mo. Basis 12	Annual Salary	Mo. Salary by Source	Tenths by Source‡	Fund Name #	Fund Number #	Account Number #	Project Number #
	□	□							

Appointee is a bona fide K. S. U. graduate student and, during period of appointment, will be enrolled in at least six semester credit hours during the regular academic year and at least three semester credit hours during the regular summer session. (Enrollment minimum not required for "Fellow.")

Recommended:

Date ____________ ____________ ____________ ____________
Department Head Director or 2nd Dean, if any Academic Dean or Vice President

Date Approved ____________ ____________
Vice President for Business Affairs

Personal Data: For FIRST Appointment EACH FISCAL YEAR

Birthdate ____________ Sex ______ Marital Status ____________ U. S. Citizen ______
Degrees ____________ ____________ Race Code ______
Local Address ____________

FOR APPOINTMENTS AND NAME CHANGES, ATTACH A W-4 FORM WITH EMPLOYEE'S OATH TO THE WHITE COPY OF THIS FORM.

Submit the first four copies to the Personnel Services Office two weeks before effective date. Those reaching the Comptroller's Office *after the 17th will* be placed on the supplemental payroll.

* For appointments only. All 9 month appointments must end no later than each May 31 and all 12 month appointments must end no later than each June 30.

† Fellows will be appointed only from funds provided specifically for this purpose and not to exceed five-tenths time.

‡ Appointments for four-tenths time or more are eligible for staff fees. (ref. Schedule of Fees.)

When paid from Faculty Salaries, use this space to insert "Faculty Salaries" and the budget-line reference to the source of funds and tenths of time.

Dept. ______________________

Line No.	Rank and Name	Tenths Time	Effective Dates Beginning - End	Mo. Ser.	Monthly Salary	Annual Salary	Source of Payment	Remarks

APPENDIX B

REQUIRED JOB CONTROL LANGUAGE

DATA DICTIONARY INITIALIZATION

```
//INITDDL         EXEC PGM=IDMSINIT
//STEPLIB         DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSUDUMP        DD   SYSOUT=A
//SYSLST          DD   SYSOUT=A
//SYSDDL          DD   DSN=DSOC7.DICT,
//                DISP=(NEW,CATLG,DELETE),
//                SPACE=(3156,1000),
//                VOL=SER=KSCC03,
//                UNIT=SYSDA
//SYSIPT          DD   *
PROCESS=TOTAL,DMCL=IDMSBASE
/*
```

SCHEMA COMPILER RUN

```
//IDMSCHEM        EXEC PGM=IDMSCHEM
//STEPLIB         DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSDDL          DD   DSN=DSOC7.DICT,DISP=OLD
//SYSJRNL         DD   DUMMY
//SYSOUT          DD   SYSOUT=A
//SYSLST          DD   SYSOUT=A
//SYSIPT          DD   *
/*                     SCHEMA DDL CARDS PRECEDE THIS CARD
```

CLUE UTILITY RUN

```
//CLUEUPDT        EXEC PGM=IDMSCLUE
//STEPLIB         DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSDDL          DD   DSN=DSOC7.DICT,DISP=OLD
//SYSJRNL         DD   DUMMY
//SYSOUT          DD   SYSOUT=A
//SYSIPT          DD   *
            SKIP2
       ******************************************************************
       *    THE STATUS SECTION IS PERFORMED IN RESPONSE TO AN ABNORMAL *
       *    RETURN CODE FROM A DML CALL STATEMENT.  IT CAUSES THE      *
       *    RUN UNIT TO BE ABORTED.                                    *
       ******************************************************************
            SKIP2
A-100-IDMS-STATUS SECTION.
            SKIP2
            PERFORM A-140-ABORT.
            DISPLAY '*****************************'
                    ' ABORTING - ' PROGRAM-NAME
                    ', '           ERROR-STATUS
                    ', '           ERROR-RECORD
                    ' **** RECOVER IDMS ****'
                    UPON CONSOLE.
            DISPLAY 'PROGRAM NAME ------' PROGRAM-NAME.
            DISPLAY 'ERROR STATUS ------' ERROR-STATUS.
            DISPLAY 'ERROR RECORD ------' ERROR-RECORD.
            DISPLAY 'ERROR SET ---------' ERROR-SET.
            DISPLAY 'ERROR AREA --------' ERROR-AREA.
            DISPLAY 'LAST GOOD RECORD --' RECORD-NAME.
            DISPLAY 'LAST GOOD AREA ----' AREA-NAME.
            CALL 'ABORT'.
        A-120-IDMS-STATUS-EXIT.
            EXIT.
            SKIP2
/*              CLUE UPDATE CARDS PRECEDE THIS CARD
```

DMCL PROCESSOR RUN

```
//IDMSDMCL      EXEC PGM=IDMSDMCL
//STEPLIB       DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSDDL        DD   DSN=DSOC7.DICT,DISP=OLD
//SYSJRNL       DD   DUMMY
//SYSOUT        DD   SYSOUT=A
//SYSLST        DD   SYSOUT=A
//SYSUDUMP      DD   SYSOUT=A
//SYSPCH        DD   DSN=&&DMCLOUT,
//              DISP=(NEW,PASS,DELETE),
//              DCB=BLKSIZE=80,
//              SPACE=(80,(400,40)),
//              UNIT=SYSDA
//SYSIPT        DD   *
/*                   DMCL DDL CARDS PRECEDE THIS CARD
//ASMDMCL       EXEC PGM=IEUASM,PARM='NOLOAD,DECK,NOLIST,NOXREF'
//SYSPRINT      DD   SYSOUT=A
//SYSLIB        DD   DSN=SYS1.MACLIB,DISP=SHR
//SYSUT1        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSUT2        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSUT3        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSPUNCH      DD   DSN=&&ASMOT,
//              DISP=(NEW,PASS,DELETE),
//              SPACE=(80,(400,40)),
//              UNIT=SYSDA
//SYSIN         DD   DSN=&&DMCLOUT,DISP=(OLD,DELETE)
//LINKDMCL      EXEC PGM=IEWL,PARM='XREF,LIST,LET'
//SYSPRINT      DD   SYSOUT=A
//LIB           DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSLIN        DD   DSN=&&ASMOT,DISP=(OLD,DELETE)
//SYSUT1        DD   SPACE=(TRK,(20,5)),UNIT=SYSDA
//SYSLIB        DD   DUMMY
//SYSLMOD       DD   DSN=DSOC7.LIB(BU01DMCL),DISP=SHR
```

SUBSCHEMA COMPILE PROCEDURE

```
//SUBSCHMA      PROC
//SUBSCHMA      EXEC PGM=IDMSUBSC
//STEPLIB       DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSDDL        DSN=DSOC7.DICT,DISP=OLD
//SYSJRNL       DD   DUMMY
//SYSLST        DD   SYSOUT=A,DCB=BLKSIZE=133
//SYSOUT        DD   SYSOUT=A
//SYSPCH        DD   DSN=&&SBSCHEM,
//              DISP=(NEW,PASS,DELETE),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600),
//              SPACE=(CYL,(2,1)),
//              UNIT=SYSDA
//ASMSSCHM      EXEC PGM=IEUASM,PARM='NOLOAD,DECK,NOLIST,NOXREF'
//SYSPRINT      DD   SYSOUT=A
//SYSLIB        DD   DSN=SYS1.MACLIB,DISP=SHR
//SYSUT1        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSUT2        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSUT3        DD   SPACE=(CYL,(2,2)),UNIT=SYSDA
//SYSPUNCH      DD   DSN=&&SUBOBJ,
//              DSIP=(NEW,PASS,DELETE),
//              SPACE=(80,(400,40)),
//              UNIT=SYSDA
//SYSIN         DD   DSN=&&SBSCHEM,DISP=(OLD,DELETE)
//LINKSUBS      EXEC PGM=IEWL,PARM='XREF,LIST,LET'
//SYSPRINT      DD   SYSOUT=A
//LIB           DD   DSN=DSFH7.IDMS,DISP=SHR
//SYSLIN        DD   DSN=&&SUBOBJ,DISP=(OLD,DELETE)
//SYSUT1        DD   SPACE=(TRK,(20,5)),UNIT=SYSDA
//SYSLIB        DD   DUMMY
//              PEND
//STEP1         EXEC SUBSCHMA
//SUBSCHMA.SYSIPT  DD  *
/*                   SUBSCHEMA DDL CARDS PRECEDE THIS CARD
```

DML COMPILE PROCEDURE

```
//DML           PROC
//DML5          EXEC PGM=IDMSDML5
//STEPLIB       DD   DSN=DSFH7.IDMS,DIPS=SHR
//SYSDDL        DD   DSN=DSOC7.DICT,DISP=OLD
//SYSJRNL       DD   DUMMY
//SYSLST        DD   SYSOUT=A,DCB=BLKSIZE=133
//SYSPCH        DD   DSN=&&DMLOT,
//              DISP=(NEW,PASS,DELETE),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//              SPACE=(TRK,(5,5),RLSE),
//              UNIT=SYSDA
//COB           EXEC PGM=IKFCBL00,
//              PARM='BUFF=32768,SIZE=200000,TRUNC,DMAP,SXREF'
//STEPLIB       DD   DSN=SYS1.CBV32LNK,DISP=SHR
//SYSPRINT      DD   SYSOUT=A
//SYSPUNCH      DD   DUMMY
//SYSUT1        DD   SPACE=(TRK,(100,10)),UNIT=SYSDA
```

```
//SYSUT2          DD    SPACE=(TRK,(100,10)),UNIT=SYSDA
//SYSUT3          DD    SPACE=(TRK,(100,10)),UNIT=SYSDA
//SYSUT4          DD    SPACE=(TRK,(100,10)),UNIT=SYSDA
//SYSLIN          DD    DSN=&&LOADSET,
//                DISP=(MOD,PASS,DELETE),
//                DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),
//                SPACE=(3200,(20,10)),
//                UNIT=SYSDA
//SYSIN           DD    DSN=&&DMLOT,
//                DISP=(OLD,DELETE,DELETE)
//LKED            EXEC PGM=IEWL,PARM='LET,LIST,MAP,XREF',COND=(5,LT,COB)
//SYSLIN          DD    DSN=&&LOADSET,
//                DISP=(OLD,DELETE)
//                DD    DDNAME=SYSIN
//SYSLMOD         DD    DSN=&&GOSET(GO),
//                DISP=(MOD,PASS,DELETE),
//                DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//                SPACE=(TRK,(8,5,1)),
//                UNIT=SYSDA
//SYSLIB          DD    DSN=SYS1.CBV32LIB,DISP=SHR
//                DD    DSN=SYS1.SUBR,DISP=SHR
//                DD    DSN=DSOC7.LIB,DISP=SHR
//                DD    DSN=DSFH7.IDMS,DISP=SHR
//SYSUT1          DD    SPACE=(1024,(50,20)),
//                UNIT=(SYSDA,SEP=(SYSLIN,SYSLMOD))
//SYSPRINT        DD    SYSOUT=A
//                PEND
//STEP1           EXEC DML
//DML5.SYSIPT     DD    *
/*                      COBOL AND DML SOURCE CARDS PRECEDE THIS CARD
//LKED.SYSLMOD    DD    DSN=DSOC7.LIB(PBU01AAA),DISP=SHR
//LKED.SYSIN      DD    *
  LIBRARY (IDMSTSKM,IDMSTSKT)
  INCLUDE SYSLIB(IDMS,IDMSCANC,BU01SS01)
/*                      LINKAGE EDITOR CONTROL CARDS PRECEDE THIS CARD
```

DATA BASE INITIALIZATION

```
//INITDB          EXEC PGM=IDMSINIT
//STEPLIB         DD    DSN=DSFH7.IDMS,DISP=SHR
//                DD    DSN=DSOC7.LIB,DISP=SHR
//SYSUDUMP        DD    SYSOUT=A
//SYSLST          DD    SYSOUT=A
//BU01BDGT        DD    DSN=DSOC7.BASE,
//                DISP=(NEW,CATLG,DELETE),
//                DCB=BLKSIZE=3156,
//                VOL=SER=KSCC03,
//                SPACE=(3156,1000),
//                UNIT=SYSDA
//SYSIPT          DD    *
PROCESS=TOTAL,DMCL=BU01DMCL
/*
```

IDMS SECURITY DUMP

```
/*TAPE9
//IDMSDUMP        EXEC PGM=IDMDUMP
//STEPLIB         DD   DSN=DSFH7.IDMS,DISP=SHR
//SYS001          DD   DSN=DSOC7.DICT.BACKUP,
//                DISP=(NEW,KEEP,DELETE),
//                DCB=BLKSIZE=3156,
//                VOL=SER=9A59BW,
//                UNIT=TAPE9
//SYSDDL          DD   DSN=DSOC7.DICT,
//                DISP=(OLD,DELETE,KEEP)
//SYSLST          DD   SYSOUT=A,DCB=BLKSIZE=133
//SYSIPT          DD   *
PROCESS=TOTAL,DMCL=IDMSBASE,REPORTS=YES
/*
```

IDMS SECURITY RESTORE

```
/*TAPE9
//IDMSRSTR        EXEC PGM=IDMSRSTR
//STEPLIB         DD   DSN=DSFH7.IDMS,DISP=SHR
//SYS001          DD   DSN=DSOC7.DICT.BACKUP,
//                DISP=(OLD,KEEP,KEEP),
//                DCB=BLKSIZE=3156,
//                VOL=SER=9A59BW,
//                UNIT=TAPE9
//SYSDDL          DD   DSN=DSOC7.DICT,DISP=OLD
//SYSLST          DD   SYSOUT=A,DCB=BLKSIZE=133
//SYSIPT          DD   *
PROCESS=TOTAL,DMCL=IDMSBASE
/*
```

PBU01AAA PROCEDURE

```
//STEP1           EXEC PGM=PBU01AAA
//STEPLIB         DD   DSN=DSOC7.LIB,DISP=SHR
//                DD   DSN=DSFH7.IDMS,DISP=SHR
//IBUAAAA         DD   *
/*                     COLLEGE UPDATE CARDS PRECEDE THIS CARD
//OBUAAAA         DD   SYSOUT=A
//SYSOUT          DD   SYSOUT=A
//SYSUDUMP        DD   SYSOUT=A
//BU01JRNL        DD   DUMMY
//BU01BDGT        DD   DSN=DSOC7.BASE,DISP=OLD
```

PBU01AAB PROCEDURE

```
//STEP1          EXEC PGM=PBU01AAB
//STEPLIB        DD   DSN=DSOC7.LIB,DISP=SHR
//               DD   DSN=DSFH7.IDMS,DISP=SHR
//IBUAABA        DD   *
/*                    DEPARTMENT UPDATE CARDS PRECEDE THIS CARD
//OBUAABB        DD   SYSOUT=A
//SYSOUT         DD   SYSOUT=A
//SYSUDUMP       DD   SYSOUT=A
//BU01JRNL       DD   DUMMY
//BU01BDGT       DD   DSN=DSOC7.BASE,DISP=OLD
```

PBU01AAC PROCEDURE

```
//STEP1          EXEC PGM=PBU01AAC
//STEPLIB        DD   DSN=DSOC7.LIB,DISP=SHR
//               DD   DSN=DSFH7.IDMS,DISP=SHR
//IBUAACA        DD   *
/*                    FUND UPDATE CARDS PRECEDE THIS CARD
//OBUAACB        DD   SYSOUT=A
//SYSOUT         DD   SYSOUT=A
//SYSUDUMP       DD   SYSOUT=A
//BU01JRNL       DD   DUMMY
//BU01BDGT       DD   DSN=DSOC7.BASE,DISP=OLD
```

PBU01AAD PROCEDURE

```
//STEP1          EXEC PGM=PBU01AAD
//STEPLIB        DD   DSN=DSOC7.LIB,DISP=SHR
//               DD   DSN=DSFH7.IDMS,DISP=SHR
//IBUAADA        DD   *
/*                    BUDGET UPDATE CARDS PRECEDE THIS CARD
//OBUAADB        DD   SYSOUT=A
//SYSOUT         DD   SYSOUT=A
//SYSUDUMP       DD   SYSOUT=A
//BU01JRNL       DD   DUMMY
//BU01BDGT       DD   DSN=DSOC7.BASE,DISP=OLD
//SORTLIB        DD   DSN=SYS1.SORTLIB,DISP=SHR
//SYSOUX         DD   SYSOUT=A
//SORTWK01       DD   UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
//SORTWK02       DD   UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
//SORTWK03       DD   UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
```

PBU01AAE PROCEDURE

```
//STEP1           EXEC PGM=PBU01AAE
//STEPLIB         DD    DSN=DSOC7.LIB,DISP=SHR
//                DD    DSN=DSFH7.IDMS,DISP=SHR
//IBUAAEA         DD    *
/*                      SELECTION CARDS PRECEDE THIS CARD
//OBUAAEB         DD    SYSOUT=A
//SYSOUT          DD    SYSOUT=A
//SYSUDUMP        DD    SYSOUT=A
//BU01JRNL        DD    DUMMY
//BU01BDGT        DD    DSN=DSOC7.BASE,DISP=SHR
```

PBU01AAF PROCEDURE

```
//STEP1           EXEC PGM=PBU01AAF
//STEPLIB         DD    DSN=DSOC7.LIB,DISP=SHR
//                DD    DSN=DSFH7.IDMS,DISP=SHR
//IBUAAFA         DD    *
/*                      SELECTION CARDS PRECEDE THIS CARD
//OBUAAFB         DD    SYSOUT=A
//SYSOUT          DD    SYSOUT=A
//SYSUDUMP        DD    SYSOUT=A
//BU01JRNL        DD    DUMMY
//BU01BDGT        DD    DSN=DSOC7.BASE,DISP=SHR
```

PBU01AAG PROCEDURE

```
/*TAPE9
//STEP1           EXEC PGM=PBU01AAG
//STEPLIB         DD    DSN=DSOC7.LIB,DISP=SHR
//                DD    DSN=DSFH7.IDMS,DISP=SHR
//OBUAAGA         DD    SYSOUT=A
//IBUAAGB         DD    DSN=COLLEGE.POSITION,
//                DISP=(OLD,KEEP,KEEP),
//                DCB=(RECFM=FB,LRECL=120,BLKSIZE=1200,DEN=2),
//                LABEL=(,LTM),
//                VOL=SER=9852BW,
//                UNIT=TAPE9
//IBUAAGC         DD    *
75-76                   DATE CONTROL CARD    FORMAT YY-YY
403                     COLLEGE CONTROL CARD
/*
//BU01JRNL        DD    DUMMY
//BU01BDGT        DD    DSN=DSOC7.BASE,DISP=OLD
//SYSOUT          DD    SYSOUT=A
//SYSUDUMP        DD    SYSOUT=A
//SORTLIB         DD    DSN=SYS1.SORTLIB,DISP=SHR
//SYSOUX          DD    SYSOUT=A
//SORTWK01        DD    UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
//SORTWK02        DD    UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
//SORTWK03        DD    UNIT=SYSDA,SPACE=(CYL,1,,CONTIG)
```

APPENDIX C

PROGRAM SOURCE LISTINGS

Due to their excessive volume, the source listings will not be appended to this report, but will be available for inspection in a special binder in the Computer Science Office.

APPENDIX E

SAMPLE PROGRAM REPORTS

```
PBUO1AAA                     K A N S A S   S T A T E   U N I V E R S I T Y                   DATE 04/19/76
ROC1                                   COLLEGE UPDATE REPORT                                  PAGE    1
                        TRANS    CCLLEGE                          TITLE
                          A        403        ARTS AND SCIENCES
    NO ERRCRS HAVE BEEN ISSUED
                                          END CF COLLEGE UPDATE REPORT
```

```
PBUO1AAB                    K A N S A S   S T A T E   U N I V E R S I T Y                        DATE 04/19/76
R002                                     COLLEGE OF ARTS AND SCIENCES                              PAGE   1
                                       DEPARTMENT UPDATE REPORT
                        TRANS   COLLEGE   DEPT                      TITLE
                          A       403      01    DEAN OF ARTS AND SCIENCES
                          A       403      09    ART
                          A       403      10    CIVISIGN OF BIOLOGY
                          A       403      13    BIOCHEMISTRY
                          A       403      14    CHEMISTRY
                          A       403      16    ECCNCMICS
                          A       403      17    CCMPUTER SCIENCE
                          A       4C3      20    ENGLISH
                          A       403      21    GEOGRAPHY
                          A       403      22    GECLCGY
                          A       403      24    HISTORY
                          A       403      25    MATHEMATICS
                          A       4C3      27    MCDERN LANGUAGE
                          A       403      28    MUSIC
                          A       403      29    HEALTH,PHYSICAL EDUCATION & RECREATION
                          A       403      30    PHYSICS
                          A       403      31    PSYCHOLOGY
                          A       4C3      32    SPEECH
                          A       403      33    STATISTICS
                          A       403      34    JCURNALISM AND MASS COMMUNICATIONS
                          A       403      38    POLITICAL SCIENCE
                          A       403      39    PHILOSOPHY
                          A       403      40    SOCIOLOGY AND ANTHROPOLOGY
                          A       403      57    ATHLETIC DEPARTMENT
     NO ERRORS HAVE BEEN ISSUED
```

```
PBUO1AAC                      K A N S A S   S T A T E   U N I V E R S I T Y                    DATE 04/15/76
ROC3                                 COLLEGE OF ARTS AND SCIENCES                               PAGE   1
                                         FUND UPDATE REPORT

                    TRANS   COLLEGE   FUND                   TITLE
                      A       403     -3315
                      A       403     -3715
                      A       403      23 2
                      A       403      27 2
                      A       4C3     43010
                      A       403     43041
                      A       403     43047
                      A       4C3     43110
                      A       403     43141
                      A       4C3     43142
                      A       403     43243
                      A       403     43443
                      A       4C3     50853
                      A       403     50933
                      A       403     50953
                      A       403     51843
                      A       403     51910
                      A       403     51943
                      A       403     52810
                      A       403     55810
                      A       403     55843
                      A       403     55849
                      A       403     55852
                      A       4C3     55911
```

```
PBL01AAD                                    K A N S A S   S T A T E   U N I V E R S I T Y                                   DATE 05/06/76
R004                                                  COLLEGE OF ARTS AND SCIENCES                                                PAGE   1
                                                          BUDGET UPDATE REPORT
TR  COLL  DPT OBJ RANK  LIN TYP  ST  MO PLAN          NAME           SOC SEC NO   TE  SALARY    BEG DATE  END DATE       OBJ  RANK LINE

                                 CODE  FUND  TE  OBJ  RANK  LINE  CODE  FUND

 A  403   57   2  9279  003  B   60  09   A GRADUATE STUDENTS                     60  50,000    05/01/75  09/01/75
 A  403   57   2  9279  003  S    2   43010  60
 A  403   57   2  9279  005  B   21  09   B JIMMY JAMES           999-99-9999  10  20,000    06/18/75  10/17/75
 A  403   57   2  9279  005  S    3   43041  05
 A  403   57   2  9279  005  S    4   43010  05
 A  403   57   2  9279  006  B   21  09   B BARBARA WHITE         888-88-8888  09  28,000    09/01/75  03/01/76
 A  403   57   2  9279  006  S    2   43010  05
 A  403   57   2  9279  007  B   21  09   A TOM JONES             777-77-7777  05   5,678    10/14/75  12/30/76
 A  403   57   2  9279  007  S    2   43010  05   2   9279   003
 A  403   57   2  9279  008  B   21  12   A JOHN SMITH            666-66-6666  06   6,000    12/01/75  09/30/76
 A  403   57   2  9279  008  S    5   43010  06
 A  403   57   2  9279  009  B   21  12   A MARY HARTMAN          555-55-5555  10  56,780    05/10/75  10/25/75
 A  403   57   2  9279  009  S    4   43041  04
 A  403   57   2  9279  009  S    5   55810  06
 A  403   57   2  9279  010  B   31  12   A GREGG DANIEL          444-44-4444  05  54,321    10/10/75  10/09/76
 A  403   57   2  9279  010  S    4   43010  05
 A  403   57   2  9279  011  B   58  12   A STEPHEN JAMES         333-33-3333  03  12,345    12/12/75  10/10/76
 A  403   57   2  9279  011  S    2   43010  03
 A  403   57   2  9279  012  B   28  12   A DANIEL BRUCE          222-22-2222  07     900    06/06/75  06/30/75
 A  403   57   2  9279  012  S    4   43010  04
 A  403   57   2  9279  012  S    5   43041  03
*************** BUDGET RECORD ALREADY EXISTS                ***************
 A  403   57   2  9279  013  B   21  12   A DAVID BOONE           111-11-1111  01              06/18/75  06/17/76
 A  403   57   2  9279  013  S    2   43010  01
```

```
PBLOIAAD                                 K A N S A S   S T A T E   U N I V E R S I T Y                                   DATE 05/06/76
ROC4                                            COLLEGE CF ARTS ANC SCIENCES                                              PAGE    2
                                                   BUDGET UPCATE REPORT
TR  COLL  DPT C3J RANK  LIN TYP  ST  MO PLAN          NAME           SOC SEC NO   TE  SALARY   BEG DATE  END DATE      OBJ  RANK LINE

                                 COCE  FUND  TE  CBJ  RANK  LINE  CCCE  FUND

 C  403   57   2  9279  CC5  B   31
 C  403   57   2  9279  006  B   28
 C  403   57   2  9279  CC7  B              ARCHY LITTLE                      07   6,000   11/01/75
 C  403   57   2  9279  CC7  S    2  43C1C  C7
 C  403   57   2  9279  008  B   22                          123-45-6789          7,000   08/01/76
 C  403   57   2  9279  C1C  B   21
```

```
PBUOIAAD                                K A N S A S   S T A T E   U N I V E R S I T Y                              DATE 05/06/76
RCC4                                              COLLEGE CF ARTS AND SCIENCES                                       PAGE    3
                                                      BUDGET UPCATE REPORT
TR  CCLL  CPT CBJ RANK  LIN TYP  ST  MO PLAN         NAME          SOC SEC NO   TE  SALARY    BEG CATE  END DATE       OBJ  RANK LINE
                                 CODE  FLND  TE  CBJ  RANK  LINE  CCCE  FUND

 D  4C3   57   2  9125  CC1
 C  403   57   2  9277  CC3
 D  403   57   2  9279  011
```

```
PBUOIAAD                         K A N S A S   S T A T E   U N I V E R S I T Y                                DATE 05/06/76
R004                                       CCLLEGE CF ARTS ANC SCIENCES                                       PAGE    4
                                             BUDGET UPDATE REPORT

     ------------ LOAN FRCM ------------       ------------ LOAN TO ------------   NINE MONTH    TWELVE MCNTH
TR    ACT     FUND   CBJ   RANK    LINE          ACT     FUND   CBJ   RANK   LINE     TENTHS        TENTHS        DCLLARS

 L   40357    43C10   2    9279    0C3         40357    43010    2    9279   013        5                          5,0C0
```

```
PBLO1AAD                                         K A N S A S   S T A T E   L N I V E R S I T Y                                       DATE 05/06/76
ROC4                                                    CCLLEGE CF ARTS ANC SCIENCES                                                      PAGE    5
                                                           BLCGET UPCATE REPCRT

                     NINE MCNTH TENTHS                   TWELVE MONTH TENTHS                                    DOLLARS
               ---- FREE ----  ---- USED ----     ---- FREE ----  ---- USED ----    --------- FREE ---------   --------- USED ---------


 M 40357 43010             50             60                 70              80                 123,456,789                        987,654,321
                     78   128      46    106        72      142      103    183    123,412,468  246,869,257  987,811,609  975,465,530

      01 EFFORS HAVE BEEN ISSUED
                                                        END CF BUCGET UPCATE REPORT
```

```
PBLO1AAE                              K A N S A S   S T A T E   L N I V E R S I T Y                                      DATE 05/10/76
RCC5                                          CCLLEGE CF ARTS ANC SCIENCES                                               PAGE  38
                                                     BLCGET LISTING

                                   XXXXX DEPT. NAME                                    4.

          LINE                                                  EFFECTIVE CATES   MC  MONTHLY ANNUAL  BUDCETED    USED         FREE
CBJ RANK  NO               NAME              SOC SEC NC STAT TE  BEG CATE END CATE SER SALARY  SALARY  DCLLARS TE DCLLARS TE DOLLARS CD FUND
                                                             5                                  8,C1C   8,C10  5  8,010                4 43041

 2  7230  002  NAME 1                                   21  10  C9/C1/75 05/31/76 09   1,510  13,590  13,590
                                                             5                                  6,795   6,795  5  6,795                C 43041
                                                             5                                  6,795   6,795  5  6,795                4 43C41

 2  7230  003                                           31  10  C9/C1/75 05/31/76 09   1,540  13,860  13,860
                                                             5                                  6,930   6,93C             5  6,930  0 43041
                                                             5                                  6,930   6,930             5  6,93C  4 43C41

 2  7230  013                                           31  10  C9/C1/75 05/31/76 09   1,350  12,150  12,15C
                                                             5                                  6,075   6,C75             5  6,C75  C 43C41
                                                             5                                  6,C75   6,075             5  6,C75  4 43041

 2  7230  015  NAME 2                                   21  10  09/C1/75 05/31/76 09   1,735  15,615  15,615
                                                             5                                  7,808   7,8C8  5  7,808                C 43C41
                                                             5                                  7,8C7   7,8C7  5  7,807                4 43041

 2  7230  019  NAME 3                                   21  10  09/01/75 05/31/76 09   1,630  14,670  14,67C
                                                             5                                  7,335   7,335  5  7,335                C 43C41
                                                             5                                  7,335   7,335  5  7,335                4 43041

 2  9305  C17                                           60  28  09/01/75 05/31/76 09   2,618  23,560  23,56C
                                                            28                                 23,560  23,560            28 23,56C  4 43041

 2  93C5  018                                           66  04  C9/01/75 05/31/76 09     376   3,386   3,366
                                                             4                                  3,386   3,3E6             4  3,386  4 43C41

                     ---- NINE MCNTH TENTHS ----    --- TWELVE MCNTH TENTHS ---    ---------------- DCLLARS ----------------
                      USED      FREE      ALLCC      USED      FREE      ALLOC              USED          FREE           ALLOC
***** CEPARTMENT TOTALS
```

```
PBLO1AAF                          K A N S A S   S T A T E   U N I V E R S I T Y                                 DATE C4/20/76
ROC6                                        CCLLEGE CF ARTS ANC SCIENCES                                       PAGE  17
                                         DEPARTMENT ANALYSIS BY FUND REPORT

                                      XXXXA DEPT TITLE

        ------ NINE MCNTH TENTHS ------   ----- TWELVE MONTH TENTHS -----   ---------------------- DOLLARS ----------------------
FUND     LCAN    BCRR  IN USE   FREE       LCAN    BCRR  IN USE   FREE        LCAN         BCPR         IN USE        FREE
43041                   158      24                        12                                           295,045       29,396
51843                                                       2                                             6,528
TCTAL                   158      24                        14                                           301,573       29,396
ALLCCATEC                       182                                14                                                330,969
```

```
PBUO1AAG                              K A N S A S   S T A T E   U N I V E R S I T Y                                    DATE 04/19/76
RCC7                                          CCLLEGE OF ARTS ANC SCIENCES                                              PAGE  43
                                                    INITIAL BUDGET LOAD
TR  CCLL  DPT CBJ RANK  LIN TYP  ST  MO PLAN            NAME            SOC SEC NO   TE   SALARY   BEG DATE  END DATE

                                 CCDE  FUNC  TE

 A  403   XX   2  6225  016  B   22  09   A   NAME 1                              10  16,020   09/01/75  05/31/76
 A  403   XX   2  6225  016  S    C   43C41  C5
 A  4C3   XX   2  6225  016  S    4   43041  C5
 A  403   XX   2  7230  002  B   21  09   A   NAME 2                              10  13,590   09/01/75  05/31/76
 A  403   XX   2  723C  CC2  S    C   43041  C5
 A  403   XX   2  7230  002  S    4   43041  05
 A  403   XX   2  723C  003  B   31  09   A                                       10  13,860   09/01/75  05/31/76
 A  403   XX   2  7230  003  S    0   43C41  C5
 A  4C3   XX   2  7230  003  S    4   43041  05
 A  403   XX   2  7230  013  B   31  C9   A                                       10  12,150   09/01/75  05/31/76
 A  403   XX   2  7230  013  S    0   43041  05
 A  403   XX   2  7230  013  S    4   43041  05
 A  403   XX   2  7230  015  B   21  C9   A   NAME 3                              10  15,615   09/01/75  05/31/76
 A  4C3   XX   2  723C  015  S    0   43041  05
 A  403   XX   2  7230  015  S    4   43C41  05
 A  403   XX   2  7230  019  B   21  09   A   NAME 4                              10  14,670   09/01/75  05/31/76
 A  4C3   XX   2  7230  C19  S    0   43C41  C5
 A  403   XX   2  7230  019  S    4   43C41  05
 A  403   XX   2  9305  C17  B   6C  09   A                                       28  23,560   09/01/75  05/31/76
 A  403   XX   2  93C5  017  S    4   43041  28
 A  403   XX   2  93C5  018  B   66  09   A                                       04   3,200   09/01/75  05/31/76
 A  4C3   XX   2  93C5  018  S    4   43C41  04
 A  4C3   XX   2  1210  001  B   21  12   A   NAME 5                              10  27,000   06/18/75  06/17/76
```

AN IMPLEMENTATION OF A BUDGETARY SYSTEM FOR THE COLLEGE OF ARTS AND SCIENCES

by

WILLIAM PETER WEBER

B. S., Kansas State University, 1970

A MASTER'S ABSTRACT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1976

ABSTRACT

For several years now the dean's office of the College of Arts and Sciences has been in need of an automated budgetary system. A previous study demonstrated that it was feasible to implement a system with both short and long term objectives.

This master's report describes the needs of the dean's office and outlines the goals for the project. A detailed analysis of the existing manual and computerized procedures, and the impact the proposed budgetary system will have on them is also included.

The problem specification describes the scope of the problem, performance considerations for the system and features of both file management and data base management systems.

Before any implementation is attempted, data structures and structural relations must be defined. These topics and secondary storage requirements are presented together.

A data base management system incorporates the concepts of a data description language and data manipulation language to achieve data independence. To this end the software vendor of the data base management system has established procedures to describe data structures and structural relationships and to provide the user with an interface between the application and data base management routines.

The software system is designed to provide the user with protected control over all data. To achieve this goal programs are described which allow limited access to specified data structures and set relations. Special locks and verification routines in each program insure data integrity and consistency.