HOUSING ASSISTANCE SUPPLY EXPERIMENT

A WORKING NOTE

This Note was prepared for the DEPARTMENT OF HOUSING AND URBAN DEVELOPMENT, under Contract No. H-1789. It is intended to facilitate communication of preliminary research results. Views or conclusions expressed herein may be tentative and do not represent the official opinion of the sponsoring agency.

Rand
SANTA MONICA, CA. 90406

WN-7953-HUD

DATA MANAGEMENT SYSTEM: PART II, THE MANAGEMENT OF DATA FOR ANALYSIS

G. Levitt

August 1972

This Note was prepared for the DEPARTMENT OF HOUSING AND URBAN DEVELOPMENT, under Contract No. H-1789. It is intended to facilitate communication of preliminary research results. Views or conclusions expressed herein may be tentative and do not represent the official opinion of the sponsoring agency.

Rand
SANTA MONICA, CA. 90406

PREFACE

This Working Note was prepared for the Office of Research and Technology of the U.S. Department of Housing and Urban Development. It is the second in a series describing the data management system to be used for manipulating data generated by and used in the Housing Assistance Supply Experiment. This Note discusses the means by which Supply Experiment data will be audited, stored, organized, and prepared for analysis. Material in this Note is based on the data requirements and data collection methodology described in Working Notes previously submitted to HUD: WN-7866-HUD, *Preliminary Design for the Housing Assistance Supply Experiment*; WN-7883-HUD, *Preliminary Description of Survey Instruments*; and WN-7885-HUD, *Data Management System: Part I, Fieldwork Data and Data Transfer Specifications*.

This Working Note fulfills the requirements set forth in Sec. II.B, Phase I, Task 1, subparagraph (6), and Task 2, subparagraphs (4) and (5), of HUD Contract H-1789.

The author would like to acknowledge the help of the Housing Assistance Supply Experiment staff members who reviewed this report and suggested useful modifications. The present draft was edited by Janet DeLand and Roberta Schneider.

CONTENTS

I. INTRODUCTION

This Working Note describes the data management techniques and processes that will be used to audit, store, organize, and process for analysis the data collected by the Housing Assistance Supply Experiment.

The main purpose of the Supply Experiment is to investigate the housing supply response to an increase in effective demand generated by providing low-income households with some form of housing allowance. The data that will be used to test the supply response (and to meet other, related study objectives) will be supplied primarily through formal surveys conducted at two experimental sites. Additional data will be collected when certain other scheduled and unscheduled events occur, such as participant enrollment, change of participant eligibility status, and participant recertification. Other less-structured data will be captured through informal observations by site monitors and others involved in fieldwork activities. (Appendix A provides additional information on data sources.)

Most field data will be edited by staff for content and logical errors, secured to protect respondent confidentiality, coded to standard forms and formats, and digitized for computer cleaning. These field data-processing* steps will be executed in roughly the above order and will be completed so that high-quality, timely data will be consistently available.

The manipulation of the data following field data processing will include the following steps:

1. Data will be audited as a quality-control check on field operations.
2. Data will be stored in a flexible format so that they may be easily reorganized to meet the demands of analysis as they arise.

---

*Note that as used here, "field data-processing" includes activities that do require use of the computer. A more complete description of field data-processing activities can be found in WN-7885-HUD, *Data Management System: Part I, Fieldwork Data and Data Transfer Specifications*.

3. Data will be reorganized for longitudinal and/or linked analysis.
4. Data will be formatted for use by the various data analysis computer programs.

These steps are described in Sections III, IV, and V. Section II presents the design goals and approaches used in building the computer system to carry out these steps. Schedules, operational procedures, and design specifications are mentioned briefly; they will be described more fully in a forthcoming Working Note.

II. STRATEGY AND GOALS

In the Supply Experiment, inputs of large amounts of different types of data will arrive in volume, requiring a variety of special and general-purpose analytical techniques. Under such conditions, data quality will inevitably vary over time, and changes in experimental design will occur throughout the analyses. As a rule, therefore, data-processing will have to be quickly adaptable to changing conditions as the project develops. Modifications to field survey instruments, varying data volumes, new hypotheses, and changing models will also cause changes in file definitions, data storage formats, computer programs for analysis, and file manipulation strategy. Furthermore, the typical researcher who will be using the data has little familiarity with data-processing techniques and the use of computer programs; indeed, there should be no reason for the researchers to acquire such skills.

The major goals of the Supply Experiment's Data Systems Group,* therefore, will be to provide a continuous stream of high-quality data for analysis; to prepare appropriate procedures, computer programs, and documentation to meet the varying project demands; and to provide these items in a form that could, if necessary, be executed and used directly by an operations staff.

It will be necessary to incorporate some degree of flexibility into our design and computer programs. The extent to which we can accomplish this, however, will depend greatly on schedule and demand constraints. Even in the best of situations, it is naive to assume that complete flexibility is possible. Flexibility is usually traded for simplified programming and program efficiency. Our system design approach includes this tradeoff as well as others.

We have made a crucial tradeoff, for example, in our approach to system implementation: Initially, we shall use established techniques

---

*The Data Systems Group is distinguished from the Field Data-Processing Group in that the function of the latter is to provide clean, computer readable data for the former to manipulate.

and standards in order to have functioning and reliable software ready to process baseline data as they arrive. Our total system design, however, includes more innovative approaches that we plan to implement in an evolutionary fashion. Thus, our *initial* system will not be very flexible and will require close attention by the Data Systems Group. But our *ultimate* system will emphasize ease of use by nonprogramming personnel (i.e., an operations staff) and will allow a great variety of tasks to be accomplished automatically and quickly.

The following sections present an overall description of the system. Each processing step (audit, storage, organization, and analysis) is described individually. It should be kept in mind that these steps are to be connected initially by well-defined, but manual operational techniques and that ultimately they will be integrated and connected by automatic processing methods.

III. DATA AUDIT

The Data Systems Group will audit data arriving from the field for quality, timeliness, and completeness. This audit will provide consistent feedback to the field survey and data processing staff on the quality of their output. We expect this process to uncover and pinpoint human, machine, and system malfunctions.

The audit will be performed by a computer program (called the Audit program) that accepts field data tapes and produces (1) an audit data base of statistics and quality-control information, (2) reports summarizing the results of the audit computer run, (3) printouts of questionnaires formatted for readability, and (4) a series of ancillary tape files. Figure 1 illustrates the overall audit flow.

Field data will periodically arrive at Rand on computer readable tapes (at a minimum of one shipment per week) containing several questionnaire batches, each of which will be homogeneous with respect to data type and collection site. Each shipment will include a complete description of the contents of the accompanying computer tape. This description will list all questionnaires by respondent identification in the order in which they are stored on the tape. Each questionnaire will occupy one logical tape record, and to each record the field data-processing group will add relevant data control information including the questionnaire type (tenant, etc.), the experimental site, the interview period (baseline, wave number, etc.), and the processing batch numbers.

The field data tape is input directly to the Audit program, which has access to file description, quota data, key variables to be audited, and sample sizes for questionnaire computer-validity checking. All of this information is stored and maintained on disk by the Audit Data Base program. The Audit program itself uses and updates this information as it reads the field questionnaire tape. Data quality is tested by running marginals* on key variables in each file and by

*The response distributions obtained by running marginals on the incoming questionnaires will indicate the type and frequency of invalid responses, should they occur.

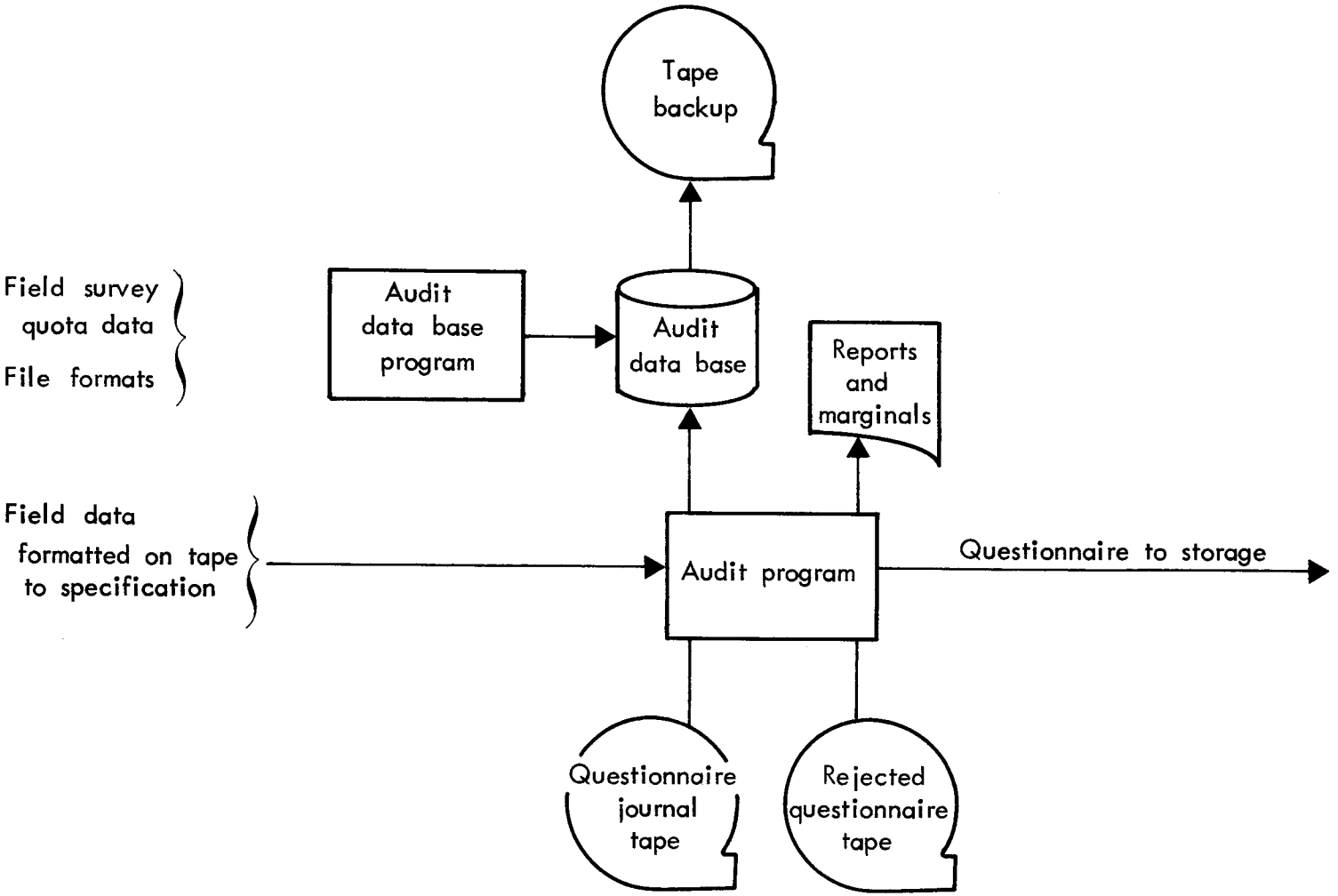


Fig. 1 — Data audit process

performing direct comparisons of a selected sample of records with their source documents; a thorough computer check is also made of the content and logic validity of the sample records. Completeness is checked by counting missing value codes; and scheduling and operations consistency is monitored by checking percentage quotas and accuracy of documentation.* Table 1 summarizes the audit processes.

Table 1

AUDIT PROCESSING SUMMARY

Type of Audit	Action
Data quality	Running marginals on key variables; comparison of audit hardcopy with questionnaire source documents for a small sample of records; analysis of content and logic for the sample records; comparison of audit statistics with field operations completion statistics.
Data completeness	Count of missing value codes.
Scheduling and operations	Comparison of incoming questionnaire counts (by type, size, and period) with quota estimates.

The Audit program produces the following ancillary tape files:

1. A file of all questionnaires received from the field, stored in the order in which they are processed.
2. A file of rejected questionnaires.
3. A backup audit data base.

The backup audit data base is formed at the end of each audit production run and stored on tape. Should the audit data base on disk be inadvertently destroyed, it can be restored immediately from this file.

*Schedule and operations monitoring will be augmented by periodic on-site field data collection and data-processing audits. Further schedule monitoring will be accomplished by comparing periodic field operations reports with audit data base statistics.

Questionnaire records that pass audit inspection are processed for storage immediately, and a report summarizing the audit is produced. This report will be sent directly to the Rand field operations manager.

Table 2 summarizes the reports generated by the audit procedure.

Table 2

AUDIT REPORT SUMMARY

Type of Report	Contents
Journal tape	Description of each questionnaire processed by the Audit program and stored on the questionnaire journal tape; output includes questionnaire identification, type, site, period, batch number, and errors, if any.
Questionnaire reject tape	Same as journal tape, but covers only those questionnaires in which some error is found.
Summary report[a]	Summary of questionnaires by tape, period, site, batch number, errors detected, and schedule fulfillment.

[a]The summary report is immediately sent to the Rand field operations manager; it is also filed as part of the data-control system.

IV. DATA STORAGE

In the data storage process, data are stored so that they may be easily reordered, retrieved, or reorganized to generate special data files for analysis. Our data storage design strategy is based on the following criteria:

1. The data must be formatted so that they can be manipulated by existing special or general-purpose commercially available data-base management systems.
2. The data must be stored so that they can be easily read and manipulated through higher-level programming languages.
3. The data must be processable by existing computer software analysis packages (e.g., SPSS and BMD).*
4. Data files must be self-defining and self-describing, i.e., a definition and description of the data must be stored with the data.
5. Data must be in a form that makes them immediately available for the most frequent and common types of analysis while also minimizing the difficulties and complexities of reorganizing them for less frequent and unanticipated analysis requirements.
6. The data-processing techniques used must be well understood and tested.

Our storage design, therefore, includes the following basic features:

1. Data fields within records are positionally stored as characters.
2. All file records are stored and retrieved sequentially.
3. A file for each of the major units of analysis is created and maintained separately.
4. Each file is divided into two parts: (a) the data definition and description header, and (b) the data records.

---

*N. H. Nie, D. H. Best, and C. H. Hull, *SPSS--Statistical Package for the Social Sciences*, Reference Manual, McGraw-Hill Book Company, Inc., New York, 1970; and W. J. Dixon (ed.), *BMD--Biomedical Computer Programs*, University of California Press, Berkeley, 1970.

5. Internal record data formats are acceptable for processing by available in-house data-base management software.

As shown in Fig. 2, the Data Storage program accepts questionnaire records from the Audit program and stores those records in one of three master files;* the basic unit of analysis of the data determines the master file to be used. The Storage program also does longitudinal data validity checks and computes and inserts values for variables that are not derived directly from the questionnaire. (Such variables include, for example, the sum of the separate incomes of a head of household.) The Storage program, of course, verifies that the correct master files have been mounted for use and that a duplicate of each such file is available in the event the primary file is damaged.

MASTER-FILE ORGANIZATION

The basic master-file organization consists of two parts: (1) the file definition and description header, and (2) the data records. The definition and description header (hereafter called the *file header*) has two important functions: First, it serves as documentation that can always be related to the file it is stored with; in this function, it minimizes the risk of inadvertent misuse by enabling precise identification of the file being manipulated. Second, it provides an extremely flexible base for adding and modifying file management software. It permits the use of certain programming features such as the COBOL copy command, and it allows programs to be written so that they require little or no change if the files on which they operate are modified.

The file header, located at the beginning of each file, consists of three parts:

1. The *file code book*, which defines each variable name, position, value type and mode, multiplicity, and valid-value encodings.
2. The *file commentary*, which describes the source, type, and uses of the data in the file.

---

*These files (for household, landlord, and housing-structure data) are the primary sources for all special data files produced for analysis.

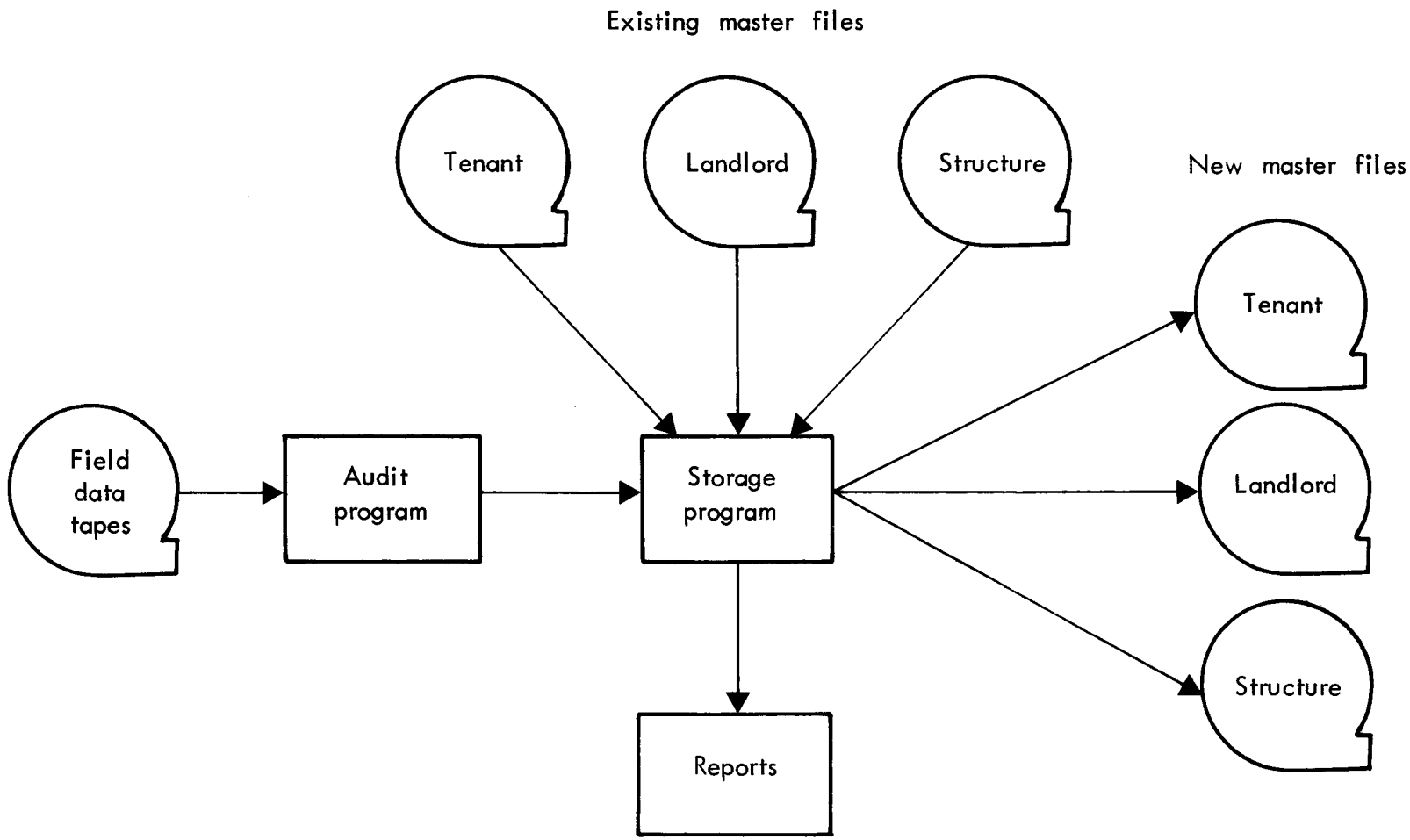


Fig.2 — Data storage process

3. The *file log*, which is a running description of the maintenance history of the file.

The data records that follow the file header are organized for sequential retrieval and have a fixed-size, positional internal storage format. The values of one variable, therefore, may be found only by knowing the position of the values that precede it in the record. If a variable has many values in a record, they are stored in neighboring fixed-size data fields together with a count of the number of values occurring.

Records will consequently vary in length and will be stored sequentially (i.e., one after the other) with respect to each other. In addition, during field processing each record will be assigned a unique identification for record maintenance and bookkeeping purposes. Files will be stored primarily on magnetic tape, although during maintenance, some files may be moved to disk in order to provide random record retrieval. Table 3 summarizes the master-file organization.

Table 3

ORGANIZATION OF MASTER FILE

File Header			
Codebook	Commentary	Log	Data Record
Description of variable name Value field length Value type (real integer)	File uses, source, and type	Maintenance history of file	Data ordered for sequential access by respondent identification; positional internal data structure; records of variable length

DATA FILE MAINTENANCE

Modifications to existing files are necessitated by corrections and updates to individual respondent records detected after field data-processing and changes in corresponding survey instruments (e.g., the

addition or deletion of a question). Because these two sources of change have essential differences, we use different techniques in our operational approach to each.

We expect individual record updates to occur at a low but constant rate. (We expect that tight field quality control will keep the error rate minimal.) These updates will occur on specific records that can be retrieved by unique identification; most often, they will be of the variety requiring values to be added, changed, or deleted.

File modifications occurring because of changes to questionnaires, on the other hand, are much more extensive and usually require changes to every record in a master file. These modifications result from the addition or deletion of questions and from changes to valid response ranges and interresponse dependencies. They are likely to occur at a high rate near the start of data collection and should asymptotically level off to zero toward the end of the collection period.

Record Maintenance

Individual records will be updated by a commercial data-base management system. We currently plan to employ the Informatics Mark IV system,* which is available at Rand, although we are also considering other systems. Updates will be accumulated and logged as they arrive from the field and will be processed at the rate of one batch every one to two weeks. Field operations will identify each update by unique record identification, the name of the variable to be updated, its current value(s), and its new value(s). After update processing is complete, a summary report will be sent to the field operations manager, in addition to being entered in the data control system.

These updates will affect only the master-file data base; they will not affect files that have been created from those basic master files. The problem of maintaining files created from the master files is discussed in Section VI.

---

*_Mark IV, Reference Manual_, Informatics, Inc., Canoga Park, California, 1970.

Maintenance Resulting from Questionnaire Modifications

Maintenance requirements generated by changes to questionnaires are handled by special software, as shown in Fig. 3.

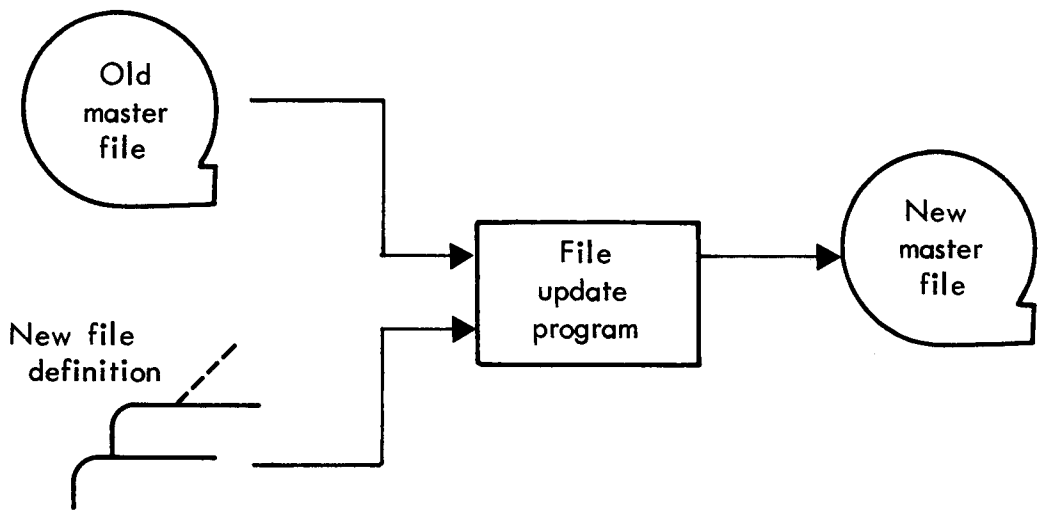


Fig.3 — Process for handling questionnaire changes

The Update program accepts as input an existing master file and a new file definition or modification to the existing definition. From these inputs the update program produces a new master file and in the process makes any or all of the following changes:

1. Deletion of data fields and compressing of the data in each record.
2. Addition of data fields and insertion of missing data value codes.
3. Transforming of response codings.
4. Updating of the file header codebook and log.

The File Update program allows us to react quickly and accurately to complicated requests. Alternatives to the creation of such a program would require the ad hoc generation of special software each time a modification was required.

V. FILE ORGANIZATION AND DATA ANALYSIS

Preliminary designs for analysis have been specified in an earlier Working Note.* Translated into data-processing requirements, these specifications fall into three different categories:

1. Direct analysis of data in the master files (e.g., for demographic characterization of experiment participants).
2. Longitudinal analysis of data in the master files (e.g., for studies of household movement patterns).
3. Analysis of linked data from different files (e.g., for measurement of supply response).

These categories are similar in that the processing paradigm, illustrated in Fig. 4, is identical for each. All processing starts with data in the master file(s).** Except for some direct analysis (type 1 above), all data go through a file manipulation step to produce reorganized files for analysis. These file-processing tasks include the creation of subsets and linked and longitudinal files. Analysis programs that range from existing data analysis packages such as SPSS and BMD to specially created programs such as those for producing household tracking patterns are then applied to these reorganized files. The analyses themselves may produce additional data files for further analysis.

The three processing categories differ greatly, however, in terms of the complexity and type of file manipulation and analysis techniques employed.

DIRECT MASTER-FILE ANALYSIS

Of the three categories, processing master-file data for analysis is the least difficult. Master-file record formats are designed to

---

*I. S. Lowry, *Preliminary Design for the Housing Assistance Supply Experiment*, The Rand Corporation, WN-7866-HUD, June 1972.

**It is possible that other files, such as enrollment and disbursement data and informal field observational data, may also be used.

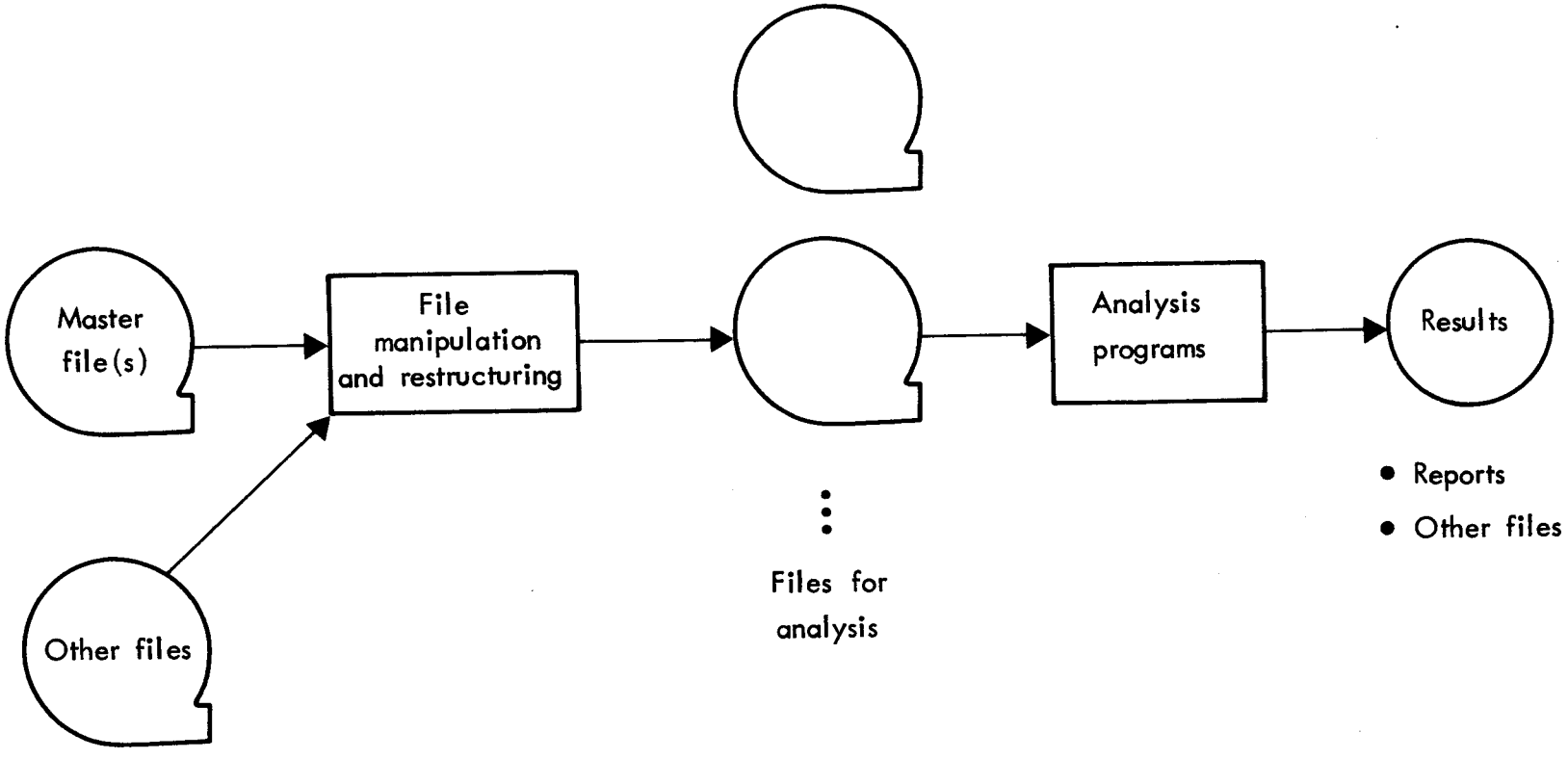


Fig.4 — Data-processing paradigm

facilitate the use of existing analysis programs. With the possible exception of manipulating multiple observation data (i.e., where a variable in one record has more than one value), no special file-processing programs need be created. When special programs are required, however, the difficulty of implementation will depend largely on the complexities of the analysis itself.

LONGITUDINAL FILE ANALYSIS

Longitudinal file analysis is more complex than direct master-file analysis but should present no extraordinary problems. Because each master file will be ordered by respondent identification, the periodic interviews of a single respondent will be stored in neighboring records, thus eliminating the need for complicated file sorts, matches, and merges. We will develop a special program to create longitudinal files, although we may also use the Mark IV system.

It is important in longitudinal file creation to employ a program that is designed to create files for varying amounts of source data arriving at different time intervals. This program will be designed to operate only on master files, so that as new data arrive, existing longitudinal files will be replaced rather than updated. By not maintaining the longitudinal files directly, both the programming and the operational logistics involved are simplified. Programs need be written and applied only to particular files whose structure remains static over time, even though new data are being added. And file maintenance is limited to master files, thus reducing the extra bookkeeping and the hazards of dealing with many similar files. The main disadvantage of this approach is its somewhat higher cost in terms of computer running time.

The longitudinal file creation program illustrated in Fig. 5 creates files for each group of variables specified to it. The data are accumulated and reformatted for each panel of respondent records and are passed to special processing programs (P_1, ..., P_n) that create the output data for analysis. These processing programs, which are developed as needed, produce the data as required for the intended

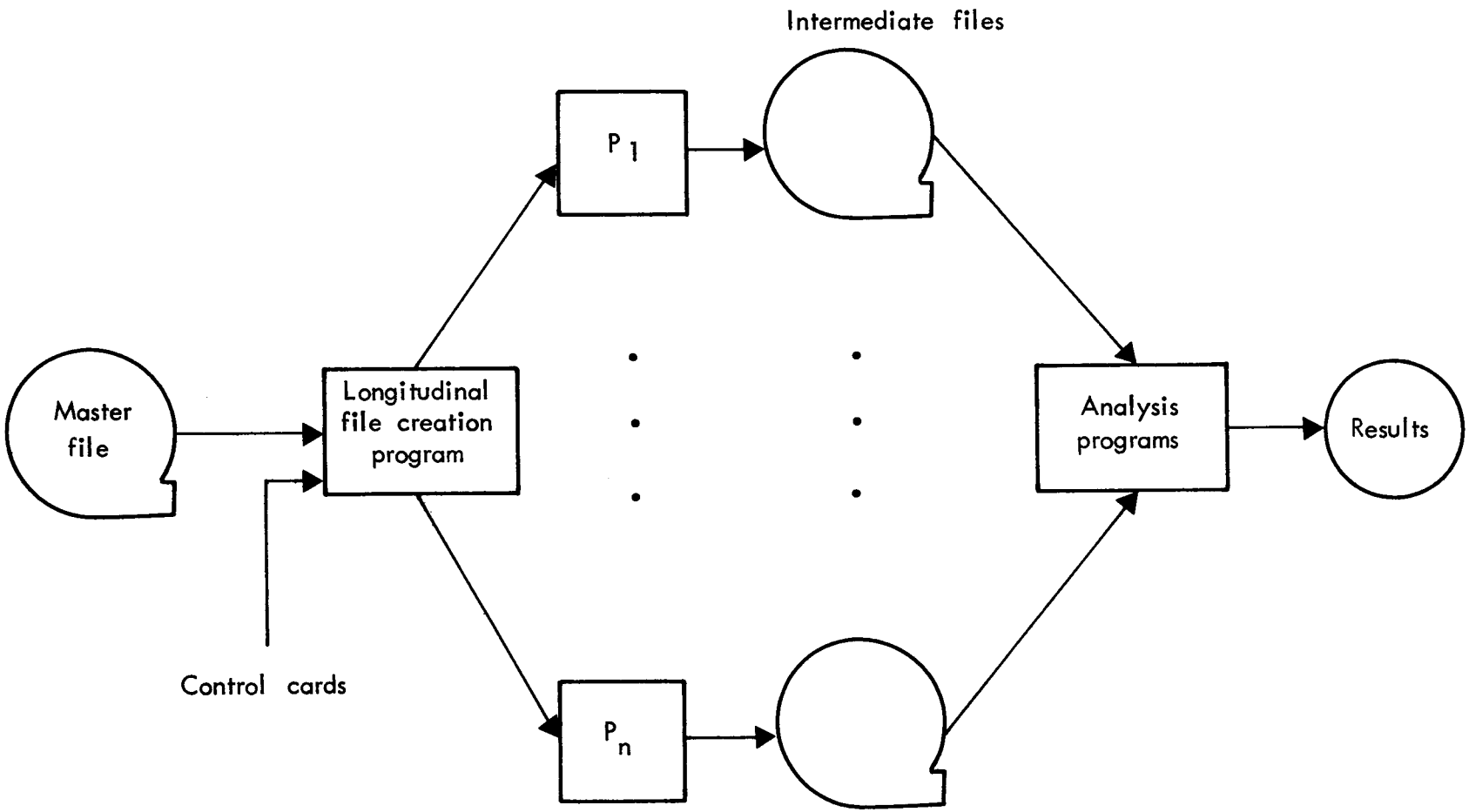


Fig.5 — Longitudinal file creation and anaysis process

analysis (e.g., a series of matrices or a file of master-file-type records). Special control cards input to the longitudinal file creation program specify input variables and output data formats.

LINKED DATA FILE ANALYSIS

Of the three categories, processing for linked data file analysis is by far the most difficult. It involves the most processing steps on the greatest number of files. Linking data in different files requires the following information:

1. The key or keys for relating the records of the different files.
2. The variables desired and the files from which they should be taken (if the same variable occurs in more than one).
3. The formats and relationships of data in the output files (e.g., longitudinal data with hierarchical relationships, matrices, master-file-type records, etc.).

The processing steps for linking data are complicated and difficult. Given the key for relating data, each individual file must first be sorted on that key; the file must then be matched on the appropriate data values, then it must be merged and reformatted for the eventual analysis. These steps require extensive tape handling and the creation of complicated computer programs for merging and reformatting. The extent of the programming difficulties depends in great part on the range of differences in the source files manipulated and the number of such files. (It is assumed that source files include longitudinal files, enrollment and disbursement data, and informal field data, as well as master-file data.)

Each processing step, as illustrated in Fig. 6, requires separate handling and programming techniques. The major step of matching, merging, and reformatting the data is usually itself a several-step process.

The processing for linked files will be described in more detail as the analysis design for this area progresses.

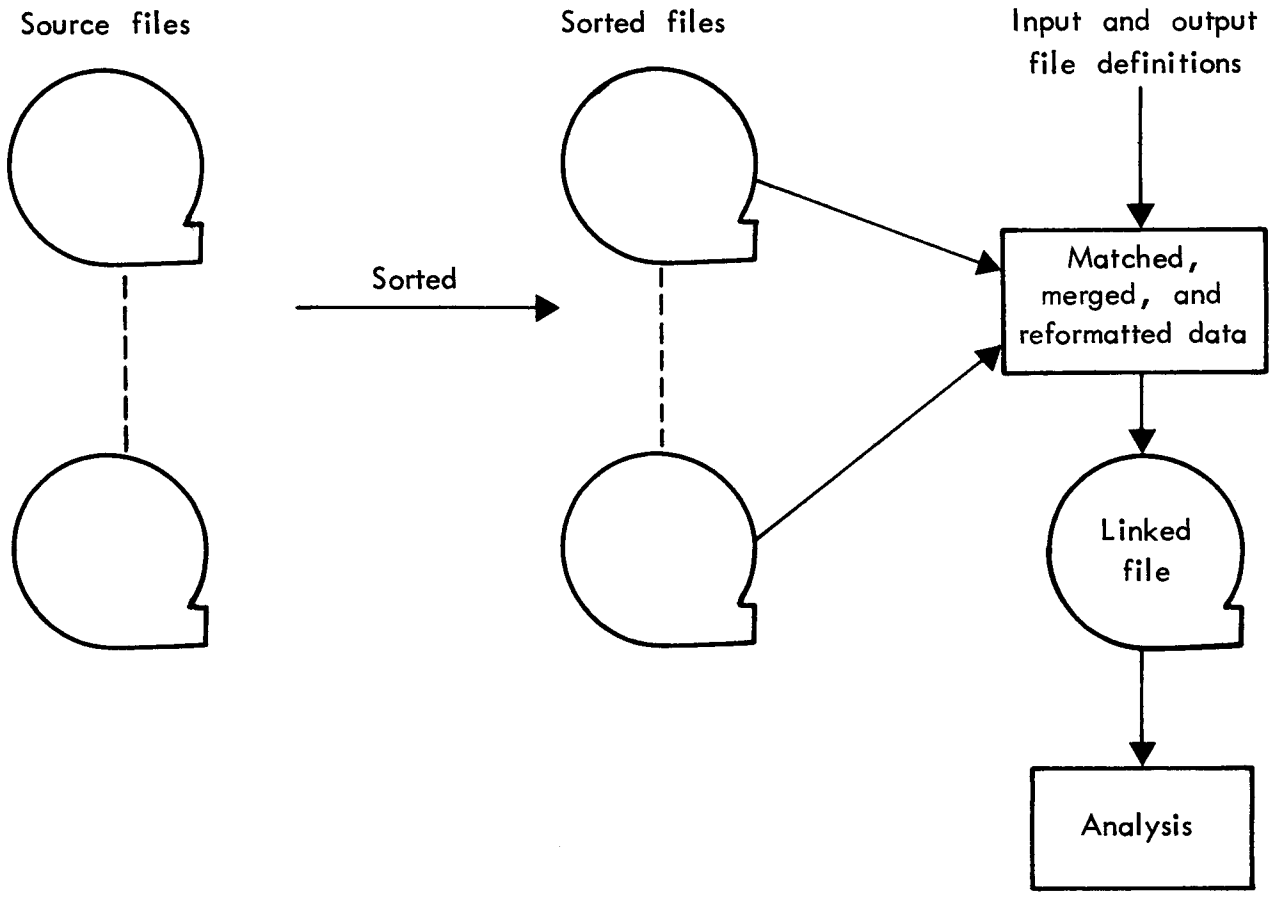


Fig.6 — Processing steps for linking data

VI. DATA MANAGEMENT OPERATIONS

EXECUTION

Initially, the data management steps described in this Working Note will be executed as individual, but related operations. An aide or clerk will use an operational manual describing the range of information needed to guide the data through audit to analysis. Programmers in the Data Systems Group will manage programming tools requiring complicated and sophisticated software design. All of these efforts, however, will be directed toward creating a production-oriented system that an operations staff can handle directly.

Ultimately, we hope to integrate each piece of the total system in such a way that human intervention is required only at key decision points (e.g., error detection and variable specification). Our approach to system integration is evolutionary and will proceed in roughly the following manner:

1. Integrate the data audit and storage programs.
2. Develop procedures and software for submitting audit and storage programs from a computer terminal (called remote job entry (RJE) procedures).
3. Integrate file management processes and software for creating longitudinal and linked files, where possible.
4. Develop RJE procedures and software for step 3.
5. Develop a user-oriented language (and respective software) for specifying and automatically executing the formation of data files and the desired analysis of these files.* (This facility would be used via computer terminal.)

SCHEDULING

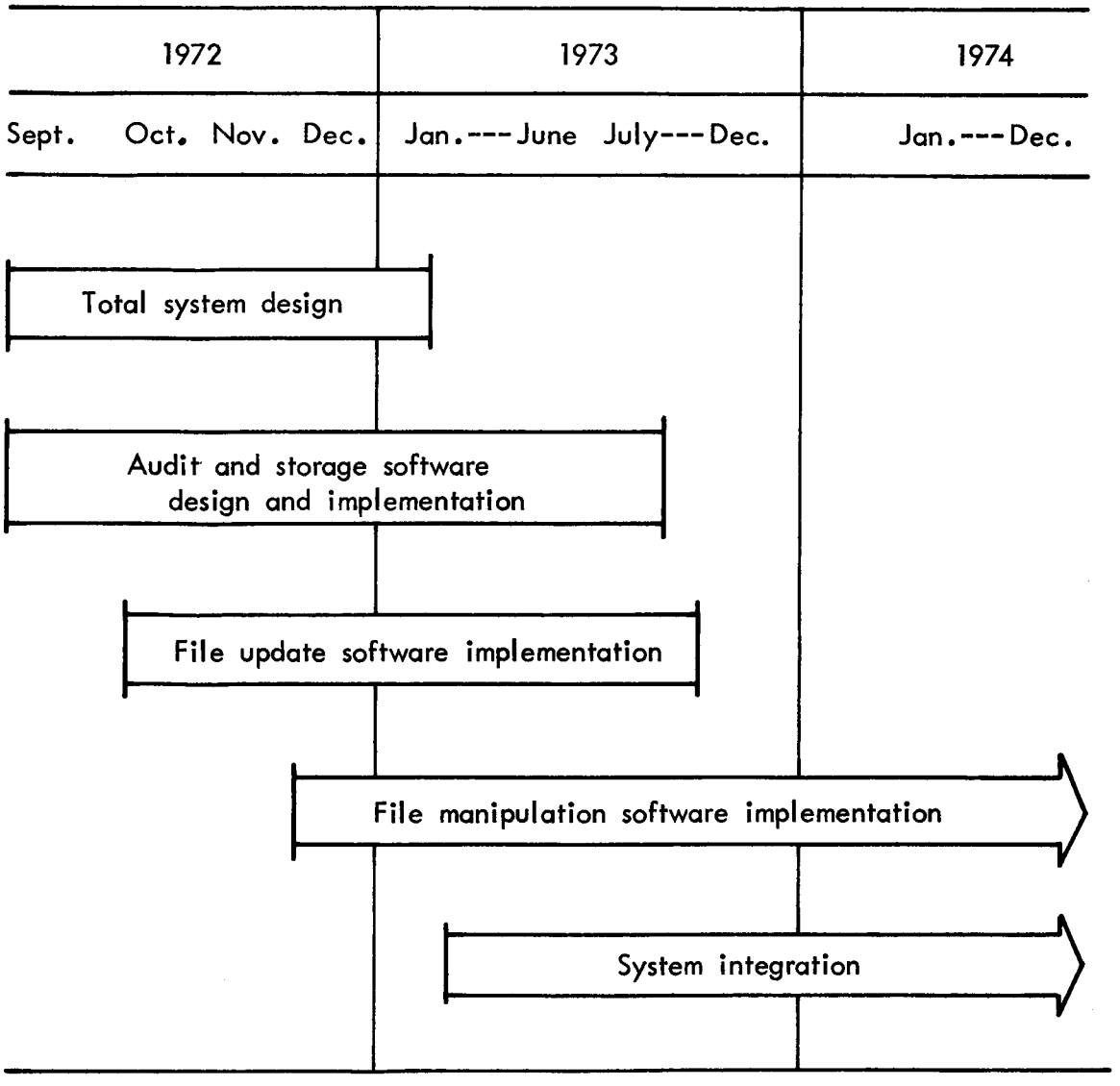
Total system design, including specifications for ultimate system integration, will begin immediately and will continue until the end of

---

*The extent to which this task will be completed is not yet clear. A complete statement of our intentions in this area of integration will require additional design analysis.

this year (1972). Design and implementation of audit and storage software will also begin immediately so that these programs will be ready to process baseline data as they arrive. Development of this software will, however, continue through the first year of the Supply Experiment.

Implementation of file manipulation software for the generation of longitudinal and linked files will begin toward the end of this year and will continue through the second year of the project, and system integration implementation will start at the beginning of the second year. The scheduling of these major events is summarized below. As system design proceeds, a more accurate and detailed dependency chart will be developed. (A summary of the data-processing tasks is presented in Appendix B.)



Appendix A

SUMMARY OF DATA COLLECTION TASKS

Field surveys will be administered both preceding and following participant enrollment in the experimental program.* Three pre-enrollment surveys are proposed.

1. Property appraisal and building survey. We propose systematic market-value appraisals of all residential property within selected modular neighborhoods, parcel by parcel; in conjunction with the appraisal, we propose to record systematic observations of the condition and use (including vacancies) of each building. In addition to residential appraisals, we expect to appraise all nonresidential parcels that might conceivably be converted to residential use during the course of the experiment; we do not anticipate that these will be numerous.
2. Landlord financial survey. We propose to interview each owner or manager of rental property to obtain a record of his rental revenues and his outlays for building maintenance and operations during the preceding year, documented insofar as feasible by ledger entries or receipts. At the same time, we will try to elicit information about his sources of, and problems in obtaining, mortgage financing and insurance, and about his difficulties with tenants, vandals, etc.
3. Tenant survey. We propose to interview all households living in the selected modular neighborhoods to obtain data on household composition, family characteristics, income, housing expenditures, and attitudes toward their housing, their landlords, and their neighborhoods. For homeowners, the survey would attempt to capture a year's record of operating and

---

*For a more detailed discussion of data collection, see WN-7866-HUD, op. cit., Sections II-VI; and Housing Assistance Supply Experiment Staff, *Preliminary Description of Survey Instruments*, The Rand Corporation, WN-7883-HUD, June 1972.

maintenance expenses and mortgage and insurance data similar to that requested of the owners of rental property.

We propose also to monitor the housing allowance experiment for a period of five years from enrollment. The following annual surveys are planned, to cover all housing and households within modular neighborhoods:

1. Building survey. The preenrollment survey of buildings will be repeated each year with emphasis on detecting changes since the preceding survey in the physical condition of residential structures, the incidence of vacancies, conversions to and from nonresidential uses, new residential construction, and residential demolitions or abandonment.
2. Landlord financial survey. We propose to reinterview owners or managers of all rental property each year, to obtain a record of rental revenues and outlays for building maintenance and operations, comparable to the data gathered in the preenrollment survey. We will also inquire about capital improvements made during the year, and their cost. Finally, we will repeat our inquiries about sources and terms of mortgage financing and insurance, and difficulties with tenants, vandals, etc.
3. Tenant survey. The preenrollment tenant survey will also be repeated, with an emphasis on detecting changes that have occurred in the intervening year, both in the population of tenants and in the housing expenditures, incomes, and attitudes of those who had been surveyed previously. (Note that these tenant surveys in the modular neighborhoods include both allowance recipients and nonrecipients.) For homeowners, the survey would again attempt to elicit a year's record of operating and maintenance costs and mortgage and insurance data similar to that requested of the owners of rental property.

Questionnaire design is currently under way and the survey sample design is still being developed; consequently the conditions under which these surveys will be applied is not yet fixed.

Sample sizes for the several surveys are also undetermined and will remain so until the two experimental sites are selected. Initial monitoring plans, however, imply roughly the following numbers for the baseline surveys at each site:

- o 5,000 household interviews (renters and homeowners)
- o 1,850 - 2,800 building evaluations (one per single or multiple dwelling
- o 1,000 - 1,500 landlord interviews (one per rental structure)

Additional records will be created by office interviews of each household enrolled in the allowance program, updated semiannually. We anticipate possibly 10,000 enrollees at each site. Enrollment will begin after baseline surveys are completed.

Appendix B

SUMMARY OF PROGRAMMING TASKS

The following table summarizes the programming tasks and activities suggested by our data management design approach and strategy.

Task	Program/Task Description
Data audit	
Audit data base program	Initializes and updates audit data base.
Audit data base interrogation program	Permits on-line computer terminal access to audit data base statistics.
Audit program	Audits incoming questionnaire tapes by producing marginals and checking for internal logic and content validity of questionnaires; produces reports and ancillary files.
Storage	
Storage program	Adds questionnaire records to the appropriate master file; creates variables and validates the integrity of its file; updates the file header log.
Integrity check program	Validates that all data stored in a master file belong there and are structurally well formed.
File header creation program	Builds a file header; maintains the header codebook and commentary.
File maintenance programs	Update master-file header and records for major changes to questionnaires.
Data-base management software	Updates individual records of master file.
File management	
Longitudinal file creation programs	Create longitudinal files designed to be employed over the entire data collection period.
Analysis	
Interface programs	Set up data so that they are processable by the various analysis programs.
Analysis programs	Modify or create special data analysis packages and programs for analysis.
Operations	
Manuals	Design and create operating manuals, codebooks, bookkeeping devices, special forms, etc.
Program documentation	Documentation of programming system for maintenance.
Integration	
Design	Completion of design of integrated system, including program design and implementation procedures.