ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

ORNL/TM-11215

MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES

3 4456 0300723 4

THE INTERNAL DESIGN PHASE OF THE BREEDING AND MULTIGENERATION SUPPORT SYSTEM: A TRACKING AND DECISION SUPPORT SYSTEM FOR NCTR

R. STRAND
T. L. COX
A. SJOREEN
D. ALVIC

OAK RIDGE NATIONAL LABORATORY
CENTRAL RESEARCH LIBRARY
CIRCULATION SECTION
4500N ROOM 175
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON
If you wish someone else to see this report, send in name with report and the library will arrange a loan.

Prepared for the
National Center for
Toxicological Research

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831; prices available from (615) 576-8401, FTS 626-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161.
NTIS price codes—Printed Copy: A18 Microfiche A01

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/TM-11215

Energy Division
Transportation and Systems Research Section

THE INTERNAL DESIGN PHASE OF THE BREEDING MULTIGENERATION SUPPORT SYSTEM: A TRACKING AND DECISION SUPPORT SYSTEM FOR NCTR

R. Strand*
T. L. Cox*
A. Sjoreen**
D. Alvic*

*University of Tennessee, Knoxville
**Computing and Telecommunications Division, ORNL

Date Published: June 1989

Research sponsored by
National Center for Toxicological Research
of the
U.S. Food and Drug Administration

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6285
operated by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U. S. DEPARTMENT OF ENERGY
under
Contract DE-AC05-84OR21400

MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES

3 4456 0300723 4

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1. REFERENCES 2

2.0. SYSTEM FLOW 5

2.1. DATA FLOW 5

2.1.1. Data Entry 7

2.1.2. RMS Data Files 8

2.1.3. Nightly Driver 9

2.1.4. Historical Data Error Correction 23

2.1.5. Access to the BMSS and EB 29

2.1.6. ADABAS files 33

2.1.7. Batch Reporting 34

2.1.8. Allocation Flow 36

2.2. USER VIEW 41

2.2.1. Interactive Update Functions. 43

2.2.2. On-line and Batch Reporting 49

2.2.3. Data Re-entry and Validation 51

2.3. DATA BASE CONSIDERATIONS 54

2.4. MODULES AND INTERNAL FUNCTIONS 57

2.4.1. BMSS Startup 57

2.4.2. On-line Update Summary 57

2.4.3. UIN Assignment 59

2.4.4. EB Single Caged Housing and Animal Assignment 60

2.4.5. BMSS/INLIFE Shared Functions and the BMSS Death Check 61

3.0. BMSS STRUCTURE 64

3.1. TERMINAL SCREEN LAYOUTS 64

3.1.1. Overview 64

3.1.2. Screen Standards 64

3.2. REPORT LAYOUTS 65
3.2.1. Report Revisions. 65
3.2.2. Report Characteristics 65
3.3. LOGS AND FORM LAYOUTS 65
3.4. DATA BASE STRUCTURE/DESIGN 69
3.5. TABLE DESIGN 71
3.6. BMSS MODULE SPECIFICATIONS 74
3.6.1. BREEDING Screen Module Specifications 74
3.6.2. BREEDING Report Module Specifications 76

4.0. SYSTEM TEST PLAN FOR THE BMSS 78
4.1. INTRODUCTION 78
4.1.1. Objectives 78
4.1.2. Scope 78
4.1.3. Applicable Documents 78
4.2. TEST ITEMS 79
4.3. FEATURES TO BE TESTED 79
4.4. FEATURES NOT TO BE TESTED 82
4.5. APPROACH 82
4.5.1. Method 82
4.5.2. Testing Categories 83
4.5.3. Functionality Testing 84
4.5.4. Incident Tracking 84
4.6. ITEM PASS/FAIL CRITERIA 86
4.7. SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS 87
4.7.1. Suspension Criteria 87
4.7.2. Resumption Requirements 87
4.8. TEST DELIVERABLES 88
4.9. TESTING TASKS 88

4.10. ENVIRONMENTAL NEEDS 91
4.10.1. Hardware 91
4.10.2. Software 91
4.10.3. Tools 92
4.10.4. Security 92
4.10.5. Test Environment 92
4.11. RESPONSIBILITIES 92
4.11.1. Test Specialists 92
4.11.2. Build Test Leader 93
4.11.3. Project Leader 93
4.11.4. ADPE Manager 93
4.11.5. Responsible Programmers (RPs) 93
4.11.6. Software Specialists 94
4.12. STAFFING NEEDS 94
4.13. SCHEDULE 94
4.14. RISKS AND CONTINGENCIES 95

5.0. BMSS DESIGN APPROVALS 96

LIST OF TABLES

Table 3.1. Frequency of generation and estimated volumes for the BREEDING Multigeneration Support System Reports 66
Table 3.2. BREEDING Multigeneration Support System reports and their equivalent numbers and names in the preceding Breeding Information System. 67
Table 3.3. Distribution characteristics of BMSS reports. 68
Table 4.1. Functions Required for Breeding and Experimental Breeding. 80
Table 4.2. Tasks Required to Test the NCTR BMSS 89

LIST OF FIGURES

Figure 2.1. Schematic of General BMSS Processing Flow 6
Figure 2.2. Overview of HDEC Data Flow 25
Figure 2.3. Allocation Flow 38
Figure 2.4. BMSS Function Access from NARSS 42
Figure 2.5. General BMSS Screen Flow 48
Figure 2.6. General BMSS Report Flow. 52

List of Acronyms

Term	Meaning
ADABAS	Adaptable Data Base System
ARUA	Animal Room Update Actions
BIS	Breeding Information System
BMSS	Breeding Multigeneration Support System
CAN	Common Account Number
CID	Carcass IDentification Number
DCL	Data Control Language
DEC	Digital Equipment Corporation
DRIMS	Division of Research Information Management Systems
EB	Experimental Breeding (also called Multigeneration)
FDA	Food and Drug Administration
FDU	Forms Definition Utility
HDEC	Historical Data Error Correction
MU	MUltiple Field Values
NARSS	NCTR Automated Research Support System
NCTR	National Center for Toxicological Research
PACS	Project Accounting System
PE	PEriodic Groups
RMS	Records Managment System
RDU	Request Definition Utility
TDMS	Terminal Data Management System
UIC	Unique Identification Code
UIN	Unique Identification Number

ABSTRACT

The National Center for Toxicological Research (NCTR) is the basic research arm of the U. S. Food and Drug Administration (FDA). The NCTR has upgraded and standardized its computer operations on Digital Equipment Corporation VAX minicomputers using Software AG's ADABAS data base management system for all research applications. The NCTR is currently performing a large study to improve the functionality of the animal husbandry systems and applications called Breeding/Multigeneration Support System (BMSS). When functional, it will operate on VAX equipment using the ADABAS data base management system, TDMS, and COBOL. Oak Ridge National Laboratory (ORNL) is supporting NCTR in the design, prototyping, and software engineering of the BMSS. This document summarizes the internal design elements that include data structures, file structures, and system attributes that were required to facilitate the decision support requirements defined in the external design work. Prototype pseudocode then was developed for the recommended system attributes and file and data structures. Finally, ORNL described the processing requirements including the initial access of the BMSS, integration of the existing INLIFE system and the STUDY DEFINITION system under development, data system initialization and maintenance, and BMSS testing and verification. This document describes ORNL's recommendations for the internal design of the BMSS. ORNL will provide research support to NCTR in the additional phases of systems life cycle development for BMSS. ORNL has prepared this document according to NCTR's Standard Operating Procedures for Systems Development.

1. INTRODUCTION

This document defines the internal design and development phases for the development of the Breeding Multigeneration Support System (BMSS). This system is being designed, prototyped, and engineered to aid the National Center for Toxicological Research (NCTR) in its animal husbandry and experimental testing activities. NCTR is a component of the Food and Drug Administration (FDA). Support for this effort is authorized through Interagency Agreement DOE No. 1860-A024-A1 between Department of Energy (DOE) and the FDA.

The BMSS will be engineered to serve four major groups of users:

1. the animal technicians who use the system to enter animal data and monitor their work,
2. principal investigators who will be monitoring an experiment and using the data for later analysis,
3. user services personnel who must maintain the system, and
4. quality assurance personnel who monitor data quality.

The BMSS system is needed because the current system's hardware is antiquated and difficult to maintain. In addition to this, the current system, Breeding Information System (BIS), is incompletely documented. NCTR is phasing out its non-Digital Equipment Corporation (DEC) equipment and consolidating its computer systems into DEC equipment. Because the current BIS operates interactively on a MODCOMP CPU with the master data base residing on an IBM 4341, BIS must be rewritten to DEC equipment. The new BMSS system will include the old BIS functionality (allowing for analysis of the data collection needs for the breeding colony) as well as functions for experimental breeding. The BMSS development will result in a system that more accurately meets the needs of the NCTR's Animal Husbandry and Experimental Breeding personnel.

This document contains the internal design phase of BMSS. The purpose of the internal design phase is to refine the external design into a computer design with the specification of all activity necessary for development of the computer system in subsequent phases. The next phase is to use specifications prepared in this internal design phase to create and perform initial testing of all computer code necessary to support BMSS development. Specifically, this document addresses:

- Data and Process Flow,
- Screen, Report, Form Layouts and Revisions from External Design,
- Data Base Structure/Design,
- Table Design, and
- Module Specifications.

As mentioned above, this completed internal design document will identify specific logical and physical file design as well as data base design and processing specifications. The next stage of NCTR life cycle management will initiate the System Implementation and Test Phase.

1.1. REFERENCES

A number of documents are available to ORNL for the design and engineering of BMSS and to NCTR for the review and approval of BMSS.

The available documents are:

1. Research Proposal for Development of BREEDING/MULTIGENERATION Support System, submitted to National Center for Toxicological Research, Jefferson, Arkansas.

2. Task Order No. 1 for the Requirements Phase of BREEDING/MULTIGENERATION Support System.

3. Interagency Agreement between the Food and Drug Administration (FDA) and the Department of Energy (DOE), Oak Ridge Operations (ORO). (DOE Number 1860-AO24-A1, FDA Number 224-87-0004).

4. INLIFE External Design, dated April 14, 1986.

5. INLIFE Data Collection System User's Guide NCTR Division of Resource Information Management Systems (DRIMS) Document. 4141-INL-1.0.

6. INLIFE Phase I Technical Documentation (DRAFT).

7. NCTR Automated Research Support System (NARSS) Data Definition Document dated 10/22/87.

8. NARSS Project Status Report dated June 19, 1984.

9. Toxicology Data Management Systems: Systems Development Guidelines, Standard Operating Procedures, TDMS Document 4505-SDG-2.0.

10. BREEDING Information System Terminal Operation Procedures, dated April 4, 1977.

11. BREEDING Information System Functional Specifications, prepared March 21, 1977.

12. Standard Operating Procedures for BREEDING Information System, dated November 1982.

13. Guide to Accessing New BREEDING Information System Data Base, prepared by Mike Holland, SISD on March 7, 1977.

14. NCTR Computer Systems Designed for Toxicologic Experimentation III. BREEDING Information System, prepared by Ron Barsh.

15. Requested Changes and Additions to BREEDING Information System, prepared by Charlie Schmieder in October 1987.

16. Code of Federal Regulations - Good Laboratory Standards Practice, Subpart A---General Provisions, Chapter 1, Part 160, July 7, 1987.

17. BREEDING Information System Data Base Record Segment Definitions, received from Ron Barsh dated October 22, 1987.

18. BREEDING Information System Flags Values, received from Ron Barsh dated October 22, 1987.

19. BIS Reports Number List.

20. Kathy Carroll's Requirements List (Personal communication).

21. March 7, 1977 Memorandum from Mike Holland. "Guide to Accessing New BREEDING Information System Data Base".

22. Copies of reports: BS-013, BS-015, BS-016, BS-017, and BS-018.

23. Copies of the two Daily Reports: BS-002 and BS-003.

24. Automated BREEDING Information System with copies of reports and descriptions.

25. Six Flowcharts: (1) BIS Edit (Low Level), (2) E2GENER (Low Level Flow), (3) UPDATE (Low Level Flow), (4) E2GENER (High Level Flow), (5) UPDATE (High Level Flow), (6) BIS Data Flow.

26. Standard Operating Procedures for the BREEDING Information System.

27. INLIFE Data Collection System Report Formats and Descriptions, #4114-INL/REP-1.0, November 1986.

28. INLIFE Data Collection System Report Formats and Descriptions, #4114-INL-REP-1.1 DRAFT.

29. INLIFE Group II Reports.

30. BREEDING MULTIGENERATION SUPPORT SYSTEM: REQUIREMENTS ANALYSIS, 04-FEBRUARY-1988, ORNL (DRAFT). 93 pp.

31. Requirements Analysis Report For The NCTR Automated Research Support System. Prepared by System Development Corporation, June 1984. 30 pp.

32. INLIFE Error Correction System Terminal Operator Guide, DRIMS 3311-INEC-1.1. Prepared by BIOTECHNICAL SERVICES, INC. for UNISYS under contract with National Center for Toxicological Research. June 15, 1988.

33. INLIFE Table Maintenance System Terminal Operator's Guide, DRIMS 4311-INL/TBM-1.0. December 4, 1987.

34. INLIFE Liaison Experiment Startup Standard Operating Procedures, DRIMS Doct 4545-INL/ES-1.0.

35. NARSS Physical ADABAS File Diagram--Proposed, June 8, 1988.

36. NARSS Data Definition--Proposed, June 9, 1988.

37. Code of Federal Regulations Title 21-Food and Drugs-Chapter 1--Food and Drug Administration, Department of Health, Education, and Welfare, Nonclinical Laboratory Studies Good Laboratory Practice Regulations (Part II). Federal Register, Vol 43, No. 247. December 22, 1978.

38. Bengtson, P.Y., R. Strand, and A. Sjorcen. BREEDING MULTIGENERATION SUPPORT SYSTEM: REQUIREMENTS ANALYSIS FOR NCTR, 04-FEBRUARY-1988, ORNL-TM-10729. 95 pp.

39. BREEDING MULTIGENERATION SUPPORT SYSTEM: EXTERNAL DESIGN, 14-JULY-1988, ORNL (FINAL DRAFT), 500pp.

40. Strand, R. H., et.al., THE EXTERNAL DESIGN PHASE OF THE BREEDING MULTIGENERATION SUPPORT SYSTEM: A TRACKING AND DECISION SUPPORT SYSTEM FOR NCTR 14-MAY-1989, ORNL-TM-11412, 500 pp.

41. Speas, Rita, et.al., STUDY DEFINITION SYSTEM: INTERNAL DESIGN PHASE, FEBRUARY, 1989, 850 pp.

42. BREEDING MULTIGENERATION SUPPORT SYSTEM: INTERNAL DESIGN, 13-MARCH-89, ORNL (FINAL DRAFT), 500pp.

2.0. SYSTEM FLOW

This section provides an overview of the BMSS system flow, including data and process flow. Details pertaining to individual screen, report, and driver programs are provided in Sections 2.3-2.4, and Appendices A and B. Pseudocode for the screens and reports is found in Appendices D and E, respectively.

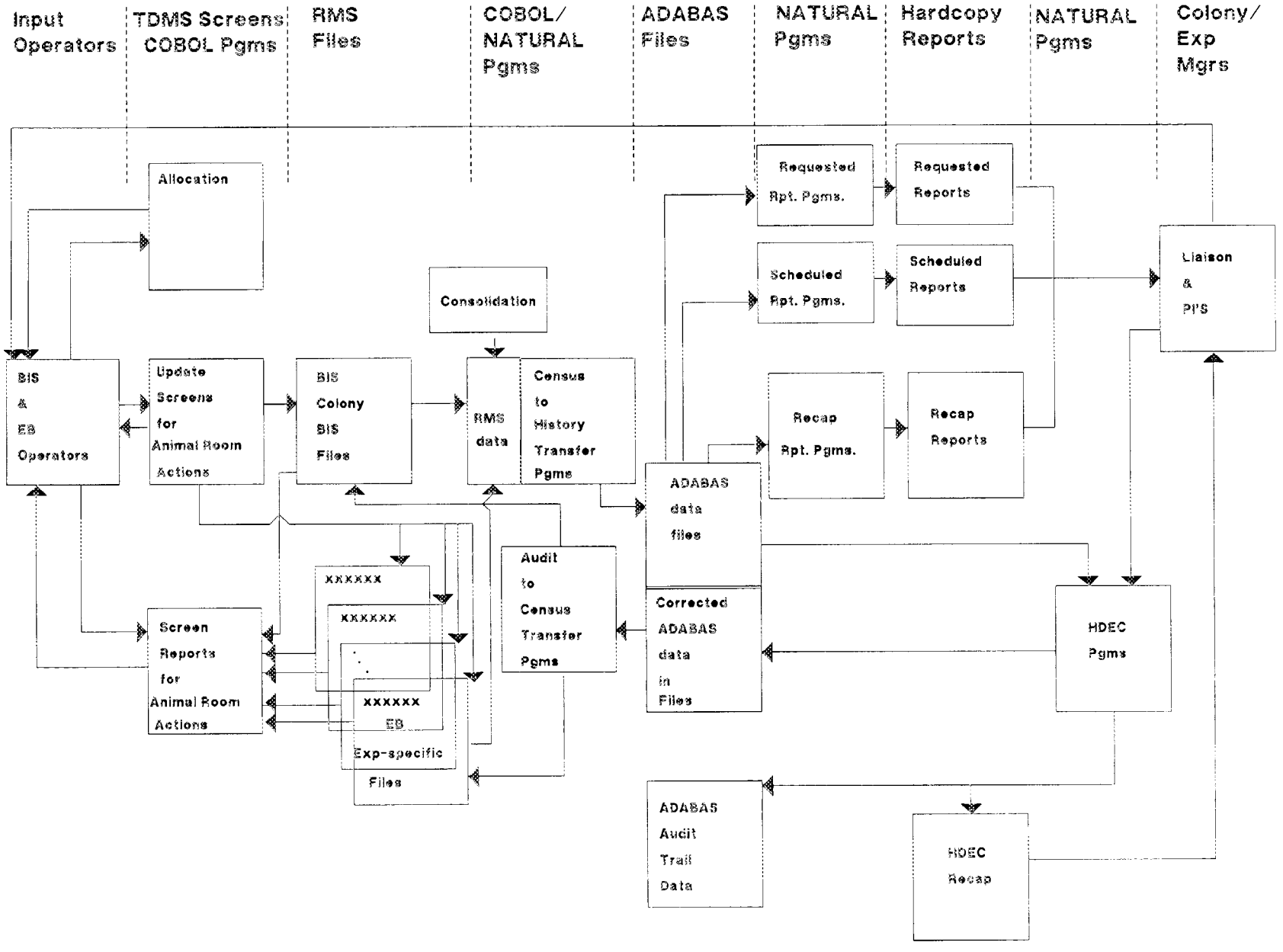
All BMSS processing is initiated from the NCTR Automated Research Support System (NARSS) main menu. The NARSS is composed of a number of automated systems supporting research performed at NCTR. BMSS and INLIFE are two of these systems. Current options available from the NARSS main menu include INLIFE screens and INLIFE reports. The INLIFE component is used to maintain information related to an animal that has been assigned to an NCTR study. INLIFE functions are described in separate INLIFE documentation. The new BMSS options are discussed in Section 2.2 below.

The BMSS flow is described in four sections. Section 2.1 discusses the data-related functions making up the BMSS, including those functions with which the user normally does not interact (automated batch job submissions and the like). Section 2.2 discusses the operational or processing flow of the BMSS from the user's view. Section 2.3 discusses data base concerns, and Section 2.4 discusses BMSS modules and internal functions.

2.1. DATA FLOW

A general schematic of the BMSS processing flow is shown in Figure 2.1. The BMSS system is composed of a number of parts that make specific contributions to the entry, reporting, and maintenance of breeding colony and experimental breeding data. Some of these parts are specifically accessible to the operator through interactive functions using DEC's Terminal Data Management System (TDMS) and NATURAL. However, other portions operate in the background and provide for automated transfer of data from one part of the BMSS data storage to another, or provide for the generation of reports without the need for specific operator intervention. This section will list and discuss in detail the processing of each of the functions shown in the diagram, emphasizing the data flow and storage of the BMSS data elements. Specific mention is made of functions that perform "background" processing of BMSS data.

Figure 2.1. Schematic of General BMSS Processing Flow



2.1.1. Data Entry

NCTR maintains a population of breeding animals to be used in experiments. Animals are housed in various rooms, racks, and cages within the NCTR facilities. A number of actions are required to set up animals as breeders, maintain each animal and its offspring, and distribute the animals to specific experiments. The BMSS on-line update functions, available through TDMS screens, allow the operator to record the various actions associated with an animal during its time in the breeding colony or as a breeder while in an Experimental Breeding study.

The data generated by this process are maintained in DEC's Records Management System (RMS) files on the NCTR VAX 785, and fall into two categories---the overall breeding colony data and the experiment-specific data (composed of INLIFE dose and observation data and experimental breeding (EB) data). See Section 2.1.2 for further information regarding storage of the RMS data.

The Animal Room Update Action (ARUA) functions use and add to the data stored in the RMS files but do not modify existing field values. The data involved in the ARUA are known as the Census data. These are the data required for day-to-day activities associated with the active members of the NCTR breeding colony. Access to these data are provided via TDMS screen calls written in the COBOL environment in a manner that parallels the existing INLIFE system. Each operator uses an individual copy of these COBOL programs to access the data relating to the selected BMSS functions. See Section 3.1 and Appendices A and D for additional details pertaining to the screen modules. See Section 2.1.4 for information regarding the modification of existing field values.

On-line updates of Census data are conducted on a daily basis. Each record added or changed is marked with a reference to the session in which the data were created or modified. The operator selects whatever functions are required to record the activity in the animal rooms on that day and the data referenced are changed in the RMS files immediately upon operator confirmation. Most update actions operate on the contents of a single cage. While operating on a single cage, the operator is allowed to edit data pertaining to functions selected for that cage. After confirmation of the requested data manipulation, updates are immediately posted to the RMS files. The selection of a new cage is considered to be implicit confirmation of the data entered for the previous cage. Sequential modes exist to allow for automatic generation of data for selected functions. Editing of the generated data is not allowed during sequential processing. However, these data are displayed to the screen. Sequential modes will allow for automated processing of all cages, littered cages, and cages containing litters at the wean age.

2.1.2. RMS Data Files

Data pertaining to the animals currently active at NCTR are stored in RMS files on the VAX cluster. These files are indexed based on selected field contents in order to provide for rapid access to operator requested records. See Appendix C for the detailed file structures and indexed fields.

The data stored using RMS actually reside in a number of files with identical structure. In an experimental breeding study, the data relate only to those animals identified with that particular experiment. For normal breeding colony activities, the files involved are those containing animals in the general breeding colony. In order to keep these data separate and to allow the data to be assigned to a particular study, there are a number of separate files making up BMSS breeding data, each owned by either an individual experiment or by the BREEDING portion of BMSS.

Only the BMSS files are required for the breeding colony. No INLIFE observations are ever made on the animals in the breeding colony, so the INLIFE files are not needed. However, experimental studies require the collection of observations on the experimental animals, and therefore a repository is required for these data. Studies with no experimental breeding require only the INLIFE files, and correspond to the previously existing INLIFE system. Studies in which breeding data are to be collected must possess both INLIFE and BMSS files in order to allow for all required functionality. Each study has a separate copy of the files used to contain INLIFE data and, if the study employs Experimental Breeding, it must contain the BMSS files. The INLIFE files are stored in a separate subdirectory affiliated with the study, and if the breeding files are required, they will be stored in the same subdirectory. This allows access to the data in either set of files while logged in to a single (Unique Identification Code) UIC. Such access will be a routine requirement for EB studies. The decision to create INLIFE files or INLIFE and BMSS files for a study is made based on Study Definition data. If Study Definition indicates that a particular study is an Experimental Breeding study, then both INLIFE and BMSS files are created.

The RMS Census data will be modified to reflect the activities carried out in the breeding colony or, in Experimental Breeding, a set of experimental breeders specifically associated with an individual NCTR study. Resulting data are transferred to the historical repository each day for all breeding data for later use or retrieval. Only the data needed to provide the functions maintained in the Animal Room Update screens are retained in the Census file. Data no longer needed in the Census data base are removed after having been successfully copied to the Historical data base. See Section 2.1.3 for additional details regarding the data transfer and cleanup processes.

2.1.3. Nightly Driver

The BMSS stores data in two separate data bases, each handling specific needs and functions of the overall BMSS process. Because there are two separate data bases (Census and Historical), there is a need to transfer data between them. The process of data transfer involves more than simply copying records from the Census data base to the Historical data base because various functions exist to manipulate and report data in both data bases.

The transfer of data between Census and History is scheduled to be performed after each day of operator processing. The entire process is frequently referred to as the "nightly run". It consists of a number of smaller, specific processes grouped together in a control structure with restart capability. The entire sequence of processes used to perform the regularly scheduled data transfer and reporting is bound together by a single driver program, though individual parts of the sequence can stand alone. Under normal circumstances, the process functions sequentially and, if interrupted, is restarted from scratch. However, if required, certain parts of the process may be submitted as independent processes. For example, the report processing can be started under the control of the processing control group operators. In summary, the nightly driver initiates the following processes that are described in Sections 2.1.3.A through F:

A.) Reflection of Historical Data Error Corrections (HDEC) into the Census files

B.) Extraction/Consolidation of new/modified RMS data

C.) Census-Historical Data transfer/validation

D.) Submission of scheduled and operator-selected reports

E.) Removal of Census data no longer required

F.) Summary of actions performed

The nightly run is intended to be started each evening after all interactive processing for the day is complete. Since this process may be delayed by system downtime, the day for which the process was to be run is entered as a parameter, rather than using today's date unconditionally. Daily reporting of the update activities is required; therefore, the nightly driver must be initiated and proceed to completion after the entry of each day's data. This fact becomes important in the case of delayed data entry (See Section 2.1.5.4).

After the target date is specified and stored for future reference, each of the subprocesses listed above is performed in turn. Each process can be restarted from the beginning, if required. The nightly driver is submitted by personnel in the NCTR process control group, and each step is available for manual startup so that only those portions that fail to complete need be resubmitted. The various portions of the nightly driver are to be written in DEC's Data Control Language (DCL) code and will cause the submission of batch jobs for each of the subfunctions involved.

2.1.3.A. Reflection Driver

Before daily additions to the Census data can be transferred to the Historical data base, it is necessary to reflect any changes that have been made to existing Census data using the Historical Data Error Correction (HDEC) component. The reflection of HDEC changes is only required for data that reside in both the Census and the Historical data bases. This process must be performed before data transfer from Census to History to insure that 1) the data in Census are consistent with that stored in the Historical data base, and 2) when the data are transferred, old (incorrect) data will not overlay the data corrected using HDEC. The reflection process is really a part of HDEC, in that the existence of HDEC forces the need for data reflection to Census. However, reflection must be coordinated with the normal data transfer process, rather than performed on a record-by-record basis in HDEC.

During the HDEC process, Historical data are changed. Changes may be made to the low level fields in a record or to the elements of an MU (Multiple) or PE (Periodic) group. HDEC changes should be made only when the on-line update system cannot make the data change required. For example, if an animal was incorrectly changed from feeder to breeder status, the HDEC should not be used. This error can be corrected by using the on-line data entry screens to change the status back to feeder. However, if an animal was incorrectly sexed, then the HDEC must be used, as there is no provision to change the animal sex in the on-line screens.

As was mentioned in Section 2.1.1, the on-line screens are used only to add new data. Section 2.1.4 points out that the HDEC screens are used only to change old data. Therefore, the two processes are partially independent. But, because each process can eventually cause changes to the same ADABAS record, care must be taken to insure that the data are changed in a manner that does not result in lost or inaccurate information.

For example, if the sex of an animal had been entered incorrectly it might be noticed as being incorrect several days after weaning. At this time a request would be made to make an HDEC correction to this value and the change would occur. However, at the same time an operator could be making other changes that alter some other value in the ANIMAL record (the status, perhaps). If the sex error in the Historical data were corrected, and then the data transfer overlaid the record, the HDEC correction would be lost. The following considerations provide for a resolution of this problem:

A. As will be seen in section 2.1.4, HDEC is not allowed to add MUs and PEs, but is allowed to delete them. If HDEC were only allowed to change existing values, then the application of changes could proceed as follows:

> Before the normal data transfer, apply the HDECed data to the RMS record. Only old MU or PE values would change (along with any flat field entries). These could not have been touched by on-line access. The process would not be a complete overlay, but would use the MU and PE counts from the Historical record to determine which elements to update. After this "reflection" of HDEC changes has occurred, the normal data transfer process overlays the HDECed record in ADABAS with the new data. The HDECed fields will now be overlaid with the same data that were just reflected, and any new data will be "added" to the Historical record. This process would provide a complete solution except for the fact that HDEC is to be allowed to delete MU and PE elements.

B. The HDEC will be allowed to delete MU and PE elements. However, all data that have just been altered by HDEC are excluded from the reports. The internal format of the data in the record is therefore not critical. The process can therefore proceed as follows:

> The HDEC is allowed to delete elements by blanking them out rather than by removing them with automatic compression. The considerations in Step A above indicate that blanking data elements will allow for proper transfer. When data are transferred back to the Census data base, they are marked as having been changed. The normal data transfer process must be modified to incorporate MU and PE element compression, providing for the proper handling of HDECed data deletions. Data that were HDECed, but not modified on-line, are still marked as modified and

therefore are included in the normal data transfer process. After both of these processes have been completed, the data are consistent in both data bases. As in consideration A above, the reflection of HDECed data is assumed to occur before the data transfer process.

The processes described here work only because the HDEC and the ARUA do not alter the same fields (field occurrences) and the HDEC cannot add data. The HDEC reflection routines will be somewhat more complicated than the data transfer routines. This complexity requires the use of the MU and PE counts to determine which Census data to overlay. Using MU and PE counts is not anticipated to pose any technical problems. The data retrieved will have Format and Record Buffers that are dynamically created and only those fields will be overlaid onto the Census record before it is updated.

In summary, HDECed records must contain marks to indicate that they need to be reflected and the reflection must occur. The reflection will require that the appropriate file in the appropriate area be opened for write access and the data modified as indicated above. Considerations will have to be made regarding the ordering of this processing so that each file affected only has to be opened once. While there are expected to be very few records, this step will be early in the nightly run and nothing can proceed until it completes. The process described in the methodology listed above is restartable by resubmission. The reflection process will change the HDEC marks on corrected data from "C" (Corrected) to "V" (Validated), so that successful record updates will remove their own references in the Historical data base as the process proceeds. A restart by resubmission will therefore process only those records that were not successfully processed in any earlier attempt to reflect the data to the Census.

2.1.3.B. Consolidation of RMS Data

The BMSS data to be transferred to the Historical data base consist of all new and modified data across all copies of the BMSS RMS files. Data that have not been modified need not be transferred. While it is not absolutely required to consolidate all such data before transfer to the Historical data base, it is more efficient to do so. Therefore, there is a function which consolidates the BMSS data. This function is patterned after the analogous INLIFE function.

During daily processing of the Census data, RMS records are created and modified. Each study contains a separate copy of each of the various BMSS and INLIFE RMS files (containing the experiment

specific animal and observation information). The breeding colony also contains its own copies of these files.

As daily processing proceeds, data in the RMS files may be modified, or new records may be added, depending on the events observed or conducted in the animal room. Not all records are modified during the course of a day, but those that are modified contain new data that must be transferred (copied) to the ADABAS Historical files. The actual transfer process is detailed in the section on Census-to-Historical data transfer (Section 2.1.3.C). Prior to the execution of that transfer, the new and modified data from each of the studies are assembled into a single area. The consolidated data are processed by the transfer programs.

The consolidation process requires read access to the data in all studies and the breeding colony. A list of the appropriate subdirectories is maintained and is used to drive the consolidation process. The consolidation program reads through the colony and study BMSS files, extracts new or modified records (based on the record creation date or latest modification date), and appends the selected records to an RMS file in a single area.

The consolidation area contains the same BMSS files as are maintained for any study, and while the extract program is processing, each study specific file is opened, processed, and closed in turn. Each file in the study specific subdirectory is processed in turn. After all files have been processed for a single study (or the colony), processing begins for the next study. This process continues until all files in all study subdirectories have had data extracted and consolidated. It would be difficult to track the progress of the consolidation process, so consolidation will be designed to be restarted from scratch. In order to allow for this kind of restart processing, the extraction/consolidation program must initialize the output files in the consolidation subdirectory as the first step in its processing.

2.1.3.C. Data Transfer from RMS to ADABAS

After the data from multiple user areas have been placed in a single area, they are ready for transmittal to the Historical data base. Separate programs are to be created for each data file. Each program will read the Census data in a particular file and determine if the data referenced exist in the Historical data base. If a copy of the record does not exist in History, then a new record is created and added to the Historical data base. If the data exist in the Historical data base, then the record in question was modified that day and will be used to replace the old version in the Historical data. The data in the

BMSS are structured in such a way that the history of each animal is kept as changes are made, so that the records that are "modified" are, in fact, augmented only; that is, the previously existing data are not changed, but maintained intact as a Historical record of the animal's movement or change in status. Augmentation allows for the existing Historical record to be replaced with the new copy without losing the trail of how that record (and therefore the corresponding animal or other BMSS entity) has arrived at its current status.

The Census data are stored in RMS files. There are a number of files for each study and the same files exist for the breeding colony. The data stored in each set of files reside in a subdirectory owned by the study or the colony. The Historical data are stored in ADABAS files having structure similar to that of the RMS files. All Historical data are stored in a single set of ADABAS files, representing the consolidated information from all studies and all past colony history. The RMS data are modified via full screen access and must be transferred to the Historical ADABAS files periodically. The Census-to-History Driver performs this function.

The Census-to-History data transfer is performed as part of the nightly run. The data that have been added to Census or changed in Census are moved from the RMS files to the ADABAS files. Prior to this transfer, the data from the individual RMS files are consolidated into a single area. The consolidation process is described in detail in Section 2.1.3.B.

The process of updating the ADABAS data with the new or modified RMS data records is straightforward. The consolidated RMS data are composed only of records that are new or changed; therefore, every record in the consolidated file must be transferred to the Historical ADABAS data base. Modified records are copied over their corresponding Historical record in ADABAS. The design of the individual record structures was made in such a way as to allow wholesale overlay of the old ADABAS record with the modified RMS data. New RMS records are simply added to the ADABAS file by copying their contents to a new ADABAS record in the appropriate file.

NATURAL 1.3 does not support data structures and, therefore, the transfer of data from the RMS structures to the ADABAS files with NATURAL requires specification of individual field names and occurrence numbers. While specification is not impossible, ADABAS Direct Calls will be used for the data transfer program, which is to be written in COBOL. Use of NATURAL would require a hybrid system, part in NATURAL and part in COBOL. Because COBOL supports the use of the Common Data Dictionary, the code used to perform the data transfer can be made to reference both the RMS data structures and the

ADABAS file structures in a straightforward way. Mapping of all data values in the structures can be performed by simple assignment statements.

Separate programs are to be created for each file making up the breeding portion of the BMSS. Each such program will read the corresponding RMS file, map the data into the ADABAS structure, and update or create the record on the ADABAS side. Historical records that are modified or created during the data transfer process will be marked on the ADABAS side to indicate their status. During the data transfer process, an attempt is made to read a Historical record having the same key fields as that found in the Census record. If such a record is found, then the data are overlaid. If not, then the data are added as a new ADABAS record. The data transfer process and the process that validates the data transferred are separate programs. Data loaded to the ADABAS files are flagged as requiring verification and verified after all the data have been loaded. See Section 2.1.4 for further details pertaining to the marking of data during load, validation, and error correction.

Each ADABAS file is composed of a number of fields, and some may be ADABAS MU or PE elements. Each record has fields that indicate the creation date and time and other fields that identify the record. All such data are intended to be entered via the TDMS update screens, but are not allowed to be altered by the operators after initial entry and confirmation. Other fields are maintained as MU or PE elements, and each time data are modified in these fields, the result is the addition of new data to the MU or PE. The existing elements of these fields also cannot be modified by the operator. All data are stored in both the RMS and ADABAS files. Because of this structure and update methodology on the RMS side, the overlay of all fields would result in changes only to the fields that had been added to the record by the operator. The total daily transaction volume is estimated to be small (less than one thousand records per day) so that the total updating would require a relatively small amount of processing time. Other techniques could be used to determine exactly which fields had been modified and used to update only those portions of the record that changed. However, the additional programming complexity is not estimated to be worthwhile. The tradeoff is basically simplicity of programming (and therefore, maintenance) versus processing time. The processing time considerations are not felt to be a driving force because of the low daily data volume.

After all data have been transferred, all Historical data not marked with a "V" (for validated) will be processed by the data validation routines. It is assumed that all data content checks are performed during the entry of the data via the TDMS screens so that they do not need to be revalidated here. The data

validation routines are used to verify that the data that have been loaded or modified possess the proper links to other, related items.

For example, the entry of a DATE ESTABLISHED record requires the use of the Dam and Sire UIN. The values found in the DATE ESTABLISHED record must exist in the BMSS Historical data or there will be no link from an animal to this record. In some cases the data transferred from Census to History may reference parent-child relationships in this manner, and the order in which the data are loaded is important. The parent records must be loaded first to insure that future data checks will succeed. In order to allow for checking of new data as they are loaded, the data will be loaded in the following order: session file, animal file, date established file, then all other files.

New records with linkage errors will be loaded to the Historical file, but will be flagged as requiring HDEC. If an existing record is found to contain a linkage error, it will also be loaded to the data base and marked as requiring attention in HDEC. The loading of known erroneous data to the Historical data base should not present any problem. Erroneous data are entered into the RMS files and transfered to the Historical file as raw data. Provisions exist for correcting such errors with an audit trail (See HDEC, Section 2.1.4). A report of load errors will be produced by each of the data transfer programs. This report will then be considered by the PIs or Liaisons to determine the appropriate HDEC measures that must be exercised to correct the problem.

The data transfer process must possess restart capabilities. The data transfer process can be restarted from scratch, because the re-application of data from the current consolidated RMS data files will result in no problems. Records that had just been added would be overlaid with duplicate field data and records that had been previously overlaid with new data would have the same data overlaid again. If the data volume involved in the transfer ever became quite large, the reapplication of each record could result in significant wasted resources, and individual tracking of the records updated in History could be made by examination of the value in the flag field. Because of the anticipated data volume and the low probability that restart will be required for any particular day, it is felt that restarting from scratch is acceptable.

After all data have been processed, the consolidated file will be renamed so that a restart required in a subfunction after the data transfer process will find no files requiring transfer of data. The renamed file can then be archived to tape if desired.

The process of data transfer is closely linked to the re-application of HDEC corrections to the Census data. Because of the need to insure that the data transferred from Census to History contain all new and changed data, while simultaneously allowing the change of Historical data using HDEC, a function exists to "reflect" changes made by HDEC in the corresponding Census data. That process is described in Section 2.1.3.A.

2.1.3.D. Report Driver

As part of the nightly BMSS batch run, all regularly scheduled and operator requested reports are submitted for processing. The submission of these reports is carried out automatically by the BMSS report driver. The BMSS report driver reads the BMSS report parameters file and determines which reports should be submitted. The reports to be submitted include regularly scheduled reports for which the proper time interval has passed since the last submission and all operator requested reports. The reports requested are submitted to a single batch queue in an order which prioritizes the reports based on need. The daily recap reports receive highest priority, weekly reports come next, then monthly reports, then operator requested reports. The regularly scheduled reports provide for output of breeding colony and study-specific data in the same report, so that individual studies need not have independent job submissions. The resulting report is divided by study after printing and distributed to the appropriate individuals. The techniques used to perform this division must include the generation of a separate banner page for each study in order that standard NCTR distribution techniques can be used for report distribution. Operator requested reports are submitted as separate jobs and are directed to operate on a specific study or the breeding colony alone.

All data used to drive the report submission reside in the ADABAS Report Parameters file. The data pertaining to regularly scheduled reports reside there permanently, while the data pertaining to operator-selected reports are generated on the RMS side by operator actions and are transferred to the ADABAS side each night. The report driver operates on the ADABAS data and therefore must perform processing after the data transfer process has been successfully completed.

The report driver is a NATURAL program that reads the ADABAS records from the Report Parameters file. Each record corresponds to a regularly scheduled or operator-selected report. The driver causes the various reports to be submitted for execution. Each such report is described in much greater detail in Appendix E. After each regularly scheduled report completes normal processing, the driver reads and modifies the corresponding record in the Report Parameters file and changes the last run date to

indicate successful completion. Similar processing takes place in the operator-selected reports---after normal processing is complete, the report causes the corresponding record to be eliminated from the Report Parameters file. This technique eliminates any need to purge the Report Parameters file on a periodic basis.

The process by which the information in the Report Parameters file is updated by the successful completion of a report allows for simple restarting of the report driver processing. Each report that successfully completes will remove the data from the Report Parameters file that would cause report submission if the driver were restarted. Therefore, if the report driver is restarted, it will resume by submitting those reports that had not successfully completed. At the worst case, the processing would have been interrupted after some report successfully completed but before that report had modified or deleted the corresponding report parameter. In that case, a single report would be re-run.

Because restart capability exists and functions in the manner described above, the report driver may be restarted automatically or through operator intervention.

NCTR will be examining the methods by which INLIFE report costs are allocated to Project Accounting System (PACS) accounts and will supply these methods for future inclusion in the report driver. No specifics regarding chargebacks will be included in this document.

The existing INLIFE reporting function possesses report queue areas named USER4$DISK:[INRP...DAILY] and [...WEEKLY] for the storage of regularly scheduled reports. The report output files in these areas are set up for a seven day retention period. BMSS will use similar names and the same retention period.

During submission of the reports, the operator will be allowed to indicate that the report should not be printed. In this case, the report will be run and queued to the viewing area but not printed. Later, while viewing the report output, the operator may request that the report be queued for hardcopy. The report viewing function will be patterned after the existing INLIFE function.

Reports will be submitted to a generic BMSS report queue, from which they will enter other NCTR batch queues for actual execution. Entry into these execution queues will be made automatically or under the direction of the NCTR process control operator.

Report launches are scheduled for each night after daily processing is complete and the Census data have been successfully transferred to the ADABAS data base files. A separate job is submitted for each report requested, and the final action of each operator-selected report is the deletion of the report record that caused submission. Regularly scheduled reports mark the report submission record with the date of the successful completion.

The report driver reads ADABAS data records and will be programmed in NATURAL. The report driver will be called from the nightly driver (which is described elsewhere). The process initiated by the report driver may be interrupted by unanticipated system downtime; so, the process must be restartable. Because each successful report eliminates the record responsible for the creation of that report, record deletion after successful completion insures that the process may be resubmitted to process all active report records. Operator-selected reports that had been processed will no longer have report submission records, while those that had not been processed will have such records. Regularly scheduled reports will show that they were successfully completed if they actually ran to completion; otherwise, they will be resubmitted. In the event of a system failure, the process control operator will determine when the report process should be resubmitted based on the contents of the batch queues. The resubmission will be performed when no pending BMSS reports remain.

2.1.3.E. Cleanup of the RMS data files

Data are transferred to the Historical data base from the consolidated RMS files and any records that are not needed in the Census data base are removed. The Census data will contain information pertaining only to those animals that are active in the colony. A program will be run to determine which animals are no longer active based on several criteria. For example, any animals with a status of dead, culled, or surplused or with a disposition of Pathology, Microbiology, or Chemistry would be removed from the Census data. These data will be maintained in the Historical data base, but will no longer be available to operators via the user input screens.

After all data moved from HDEC to Census and History have been updated with all modified records, the Census file is ready for cleanup. Records referring only to animals that have been removed from the colony or study should be removed from the RMS files periodically. Normally, the "out-dated" data will not be removed from the RMS files on the same day in which it was discovered to be "out-dated" (or rather, during the nightly processing that corresponds to the day on which the records became superfluous

in Census), but one week later. The reason for this delay relates to the possibility of unforeseen circumstances arising in which the data are found to have been updated incorrectly. Under these circumstances, the data should not have been removed, and it is almost certain that related records should also not have been removed. For example, if a block of animals is inadvertently retired, certain records have to be reconstituted using HDEC and data reflection while others would have to be reinstalled. While parent records (like ANIMAL, for instance) can be modified directly by the HDEC System and would then reappear in Census because of data reflection, related records (like DATE ESTABLISHED) possess no actual error and should not be handled with the HDEC. Such records would have to be manually retrieved based on the changes made to the parent record and reinstalled in the RMS file. Rather than adding the complexity of the related record processing to the HDEC, it appears that a more direct course is to cause the removal of Census records to be delayed until after a suitable interval for HDEC corrections has passed. The period of a calendar week should be sufficient to allow for the correction of any such errors. The actual criteria for Census record elimination are as follows:

Data should be retained in Census as long as they are needed for processing. As soon as they are no longer needed, they can be removed from Census, and will be preserved in the Historical data base. Screen processing was examined to determine how long data should remain in Census. For example, in some screens, the litters of an animal need to be found. However, the parents of an animal never need to be found for screen processing. This information can then be used to determine if a corresponding litter record will be removed when an animal is removed. In similar manner, other screen processing considerations were made to determine data retention needs.

The time interval for retention of old Census data will be parameterized, and is referred to here as the input parameter PURGE_INTERVAL (to be specified in days).

ANIMAL RECORDS

Two alternatives exist for removing an animal from the Census data. In either method, step (1) should be executed before steps (2) or (3). Steps (2) and (3) can be executed in either order. The first method uses the results of ANIMAL file deletions to determine candidate records in subsequent steps. The first method would require more programming time and less processing time. The second method purges each file independently and would require less programming time, but more processing time. Method 1 has been selected because it

is believed to be more efficient based on the number of records to be processed, but method 2 is included for possible future reference.

Method 1.

(1) All ANIMAL records should be searched to see if the ANIM_REMOVAL_DATE is further from the present date than the given parameter PURGE_INTERVAL. If so, the ANIMAL record should be removed from Census. The values of ANIM_BIS_UIN and ANIM_LITTER_UIN should be retained by the cleanup program for further processing.

(2) For each ANIMAL record removed, search the DATE_ESTABLISHED file for occurrences of the corresponding ANIM_BIS_UIN. Do the following for each DATE_ESTABLISHED record found:

Search the ANIMAL file for occurrences of the partner(s) of the removed animal described by ANIM_BIS_UIN. If no corresponding ANIMAL records are found, this means that all partners have already been removed, and the DATE_ESTABLISHED record can be removed from Census. If any ANIMAL records are found for the partner(s), then the DATE_ESTABLISHED record should be left intact in Census.

(3) For each ANIM_BIS_UIN isolated in step (1) above, search the LITTERS file for the record that corresponds to ANIM_LITTER_UIN. Using the LIT_PARENTS structure, search for the ANIMAL records for the parents of the animal identified by ANIM_BIS_UIN. If ANIMAL records for either parent exist, exit this step, because the LITTER record should be left intact.

Using the same LITTER record already found, search the ANIMAL file for siblings of the animal identified by ANIM_BIS_UIN. If ANIMAL records for any sibling exist, exit this step.

If no parents or siblings of the animal identified by ANIM_BIS_UIN exist, remove the LITTER record corresponding to this animal from Census.

Method 2.

(1) All ANIMAL records should be searched to see if the ANIM_REMOVAL_DATE is further from the present date than the given parameter PURGE_INTERVAL. If so, the ANIMAL record should be removed from Census.

(2) All DATE_ESTABLISHED records should be processed in the following way:

For each DATE_ESTABLISHED record, find the corresponding ANIMAL record for the Sire and the Dam. If no ANIMAL record exists for either animal, then both animals have been removed. In this case, remove the corresponding DATE_ESTABLISHED record from Census. If either animal has an ANIMAL record remaining, the DATE_ESTABLISHED record should be preserved.

(3) All LITTER records should be processed in the following way:

For each LITTER record, do the following:

(a) Find corresponding ANIMAL records with the same ANIM_LITTER_UIN. If any ANIMAL record is found, escape this step, because the LITTER record should be left intact.

(b) Find the corresponding ANIMAL records for the Dam and Sire of the litter. If any ANIMAL record is found, escape this step.

(c) If no ANIMAL records were found for siblings or parents, remove the LITTER record from Census.

ALLOCATION RECORDS

An ALLOCATION record should be removed when the difference between the current date and that record's ALC_ALLOCATE_DATE is greater than PURGE_INTERVAL.

ORDER RECORDS

An ORDER record should be removed when the difference between the current date and that date indicated as the ORD_DATE_FILLED is greater than PURGE_INTERVAL.

REPORT RECORDS

BREEDING_RPT_REC records will be removed as the requested reports are generated, so there will be no need to handle report records through Census cleanup.

CAGE RECORDS

CAGE records should never need to be removed.

2.1.3.F. Summary of Nightly Actions

The reflection, consolidation, data transfer, and Census data cleanup programs will each run in turn. After each step completes successfully, the next step in the process may proceed. Each step will produce a summary of the actions that were performed in that step. The resulting summary report will be appended to the contents of a single file, created each day as the first step in the reflection process.

The summary file will be printed as the last step of the data cleanup process. Completion of the Census data cleanup indicates that all processes related to the transfer and maintenance of data have been completed and that update access to the Census data may be granted. The daily processing cycle then restarts.

2.1.4. Historical Data Error Correction

HDEC for BMSS will allow for the correction of data in the Historical ADABAS files that cannot be accessed for modification by the operators during animal room activities, including data pertaining to animals still in the Census, or purely Historical data. The Historical ADABAS files include all data that

have been accumulated from the Census data, processed by the validation program, and stored in the ADABAS files. These data may have been successfully validated or rejected during the validation process (part of the Census-to-History data transfer). ORNL proposes that rejected records be stored in the same data base files, but marked with a flag to indicate the status of the record. Rejected records may then be modified with HDEC or reprocessed after modification to other (related) rejected records. The list of rejected records will be made available for access to the principal investigators (PIs) or liaisons so that the required action can be determined.

A user may access the Historical data through ADABAS Natural input screens by entering the function indicating the modification required and the appropriate key field and value for the file that contains the desired data. The rejection report will include the keys for each record rejected. Other records found to be in error will be identified based on field content information from the report in which the error was found. The selected record can be reviewed, modified, or deleted. For all modifications and deletions, audit trail records are generated. Deletions are immediately applied to the data upon confirmation from the user. Note that entire records are never actually deleted, but that PE and MU elements may be deleted. The records that are modified are marked as such, and are automatically included for revalidation the next time the validation routines are run. These routines may be executed the next day during the normal Census-to-History load process, or may be executed as a separate process. Note that the validation process does not perform the reflection of modified data back into Census. Validation merely determines which records are valid for use in the Historical data and which are not. See Section 2.1.3.A for a description of the reflection process.

A simplified diagram of the process involved in HDEC is included in Figure 2.2. Each element of the diagram is discussed below.

2.1.4.1. Load/Validation Process

After collection from the operators' RMS files by the consolidation process and movement to single files, data are processed by the Load/Validation processor and loaded into the Historical ADABAS files. This process actually consists of two steps, (1) loading the data into the ADABAS files (Load) and (2)

Figure 2.2. Overview of HDEC Data Flow

```
 ------------------------------
|  Consolidated Census Data    |
 ------------------------------
           |
           v
 ----------------------------     -------------------
|  BMSS                      | <-----General Tables
|  Load     | Validation     |    -------------------
| Process   | Routines       | <-----Study Definition
 ----------------------------     -------------------
           ^                 |    -------------------
           |                 |->| Update Summary    |
           |                 |    -------------------
           |                 |    -------------------
           |                 |->| Rejection Rep.    |
           v                     -------------------
 ----------------------------     -------------------
|        BMSS                |   | History to        |
| Historical-ADABAS *        |->|Census Process     |
 ----------------------------     -------------------
           ^
           |
           v
  --------------------------      -------------------
 |     BMSS HDEC            |<-->| INLIFE Error      |
 |                          |    | Correction        |
 | (Also Calls Validation)  |    | System            |
  --------------------------      -------------------
           |
           v
  --------------------------
        BMSS Audit Files
  --------------------------
```

* - Note that records modified by HDEC remain in the Historical data base. The batch version of the validation process occurs during the batch load process, but contains as a component a portion that revalidates records in History that are marked as HDEC corrected or still invalid, even if the load did not "create" those particular records, allowing for automatic revalidation of data linked to a HDEC-corrected parent record. Note that the batch validation process may be issued as the final action of the HDEC, to cause "immediate" automatic revalidation of rejected records that were not specifically altered with HDEC changes.

validating the data (Validation). The Load (Census to Historical) programs are written in COBOL using ADABAS Direct Calls and are described in Section 2.1.3.C. The Validation routines are to be written in Natural and are run immediately after the Load program or optionally called from HDEC. The data loaded to the file are initially marked with the letter "L" (for "loaded). The validation routines operate on all data having a flag other than "V" (for validated), and the reports include only data marked as validated.

The input to the Validation routines is therefore new or user modified Census data, or records previously rejected by HDEC (which may or may not have been explicitly corrected). Validation consists of checking the data against the General Tables data, Study Definition data, and existing Historical data. This validation duplicates checks performed on the Census data during initial entry. In addition, related data base fields are checked to insure that all links between related data exist. For example, the LITTER record must reference a Dam and a Sire that exist. Both new and modified records are validated for linkages. Changes to PARENT records in the HDEC are performed in such a manner as to cause automatic change to the CHILDREN records if that is required to maintain consistency.

After passing validation, modified records will be posted to the data base and a copy of the original record (with audit information added) is created in the appropriate audit file. These two data base modifications are associated with the same END TRANSACTION to insure that the entire update process succeeds or fails. In the case of CHILDREN records modified based on changes to the PARENT record, the END TRANSACTION encompasses all records that are logically related.

2.1.4.2. Historical ADABAS Files

The Historical ADABAS files contain all data that have not been archived, i.e., current animal data that are also in the Census, and all unarchived data for previous BMSS activities. When loaded into the Historical files, data may have been successfully validated or rejected by the Validation/Load process. After the validation process completes, a record status flag exists for each record in the Historical files that indicates whether the record has been rejected (R), is valid (V), or has been modified and validated and is ready to be copied back to the Census data base (C). All "L" records are processed during validation and will be marked with "R" or "V" after validation completes. Previously existing "R" records are reprocessed during any batch validation pass and may then be accepted or remain rejected, depending on the status of the appropriate data checks. As was mentioned above, ORNL assumes that the reports will not include data marked with a rejected flag. This flag is to be added to the various file definitions.

2.1.4.3. Audit Files

Audit files are ADABAS files with essentially the same structure as the Historical ADABAS files. Audit information has been added to track all changes that were made to the corresponding validated records. As validated records are modified or deleted by the user, audit records are created by copying the original view of the records to be modified to the corresponding audit files. The HDEC operator is required to enter a reason for the change. Any changes made to any record stored in the Historical data file will result in the generation of audit trail information. Thus, if a failed attempt was made to add a record to the data base on Monday, and subsequent Tuesday corrections were also incorrect, but the Wednesday corrections worked, then the audit file would consequently contain the Monday and Tuesday versions, while the Wednesday version would be the only one that would finally reside in the Historical data files.

2.1.4.4. HDEC User Interaction

HDEC will be a menu-driven correction facility that consists of a sequence of ADABAS Natural user input screens. The screens will function much the same as the Inlife Error Correction (EC) System screens. The user will select the source of the record he wishes to change, either rejected data (R) or valid data (V). Even though the rejected and validated records are stored in the same files and it is possible to search the entire data base for a record without selecting R or D, the user will know the status of the record that requires modification. This value is used to provide a user view consistent with that used in INLIFE error correction even though the data storage methodology differs somewhat from the INLIFE method.

The user will then be prompted to enter the BMSS function under which the desired record can be found. The selection of a function determines which file(s) should be queried. Some functions (e.g., correcting session data) already exist in the Inlife Error Correction System and will be accessed by BMSS HDEC. When the appropriate record is found, the user may modify selected fields depending on the function selected. The user may modify, delete, or archive data through HDEC. Reprocessing of rejected records is performed automatically.

Only those data that cannot be modified via animal room actions and are not keys to files can be modified with HDEC. There are classes of errors that can, and presumably will, be corrected through

Animal Room Update Actions, as long as the data to be corrected remain in the Census data. Documentation of errors found may be required by the principal investigators so that the appropriate error correction procedure can be determined. All data that can be changed using the ARUA screens should be changed with the ARUA screens. HDEC should only be used to make corrections that are beyond the scope of ARUA or that apply to data that were discovered after an animal was removed from the colony.

When a record is modified, the modified fields are checked for consistency (i.e., numeric in numeric field) and the Validation routines are called to revalidate the record. If the record passes validation, the record status flag is set to C to indicate the record may need to be copied to the appropriate Census file. If the record does not pass the validation criteria, the user is prompted to correct the data in error. The user may elect to abort the correction process if the proper correction cannot be made at this time. No update occurs in this case.

If the user elects to delete a record, a confirmation message is given. The user must explicitly indicate that the deletion should occur. ORNL assumes that confirmed deletions will occur immediately and will merely result in the change of a flag value. GLP prohibits the deletion of data that reach the Historical files, therefore, no ADABAS records will ever be physically deleted. Record "deletions" will be handled by marking the record as deleted, and ignoring such records in the reports.

In the INLIFE Error Correction System there is an option to reprocess rejected records without making any changes. In the BMSS HDEC, reprocessing is unnecessary. Rejected records will be retrieved by the nightly validation process and revalidated automatically. If the validation succeeds, the record status flag is changed from R to V (valid). If the validation fails, nothing is changed, the record remains in the rejected status, and it appears on the next rejected record report.

Just as in the INLIFE system, records can be flagged for archiving. The use of this flag to perform the archiving of the marked records is performed by processes external to the BMSS, therefore, these processes are not described here.

2.1.5. Access to the BMSS and EB

The BMSS logon driver programs are used to initiate a BMSS session, allow the operator to select the desired functions, and eventually to terminate the BMSS session. The BMSS driver is a subsidiary of the NARSS menu and must be constructed in a manner that will allow integration of the BMSS functionality with the existing INLIFE and Study Definition components of NARSS.

The logon driver must establish the parameters required to drive the TDMS/COBOL session through which the Animal Room Update Actions are performed. Because of the nature of the BMSS, the logon driver must actually be capable of controlling four different kinds of BMSS sessions. These are:

- Normal Update---the typical day-to-day operations recording colony or experiment specific BMSS actions
- Training Room Sessions---the use of special data for the training of animal room operators, in which the colony manager accesses data for a fictitious room
- Supervisory Mode---the use of a special function to allow the screen I/O from another operator's session to be observed from a second terminal...this function is used to monitor the activity conducted in another room for screening of technique or to provide actual examples of animal room actions that occur in daily colony maintenance
- Delayed Entry of Normal Data---on some occasions, the entry of data will be delayed from the collection time because of system availability; data so entered must be allowed to have the session date and time specified, rather than using the system date and time

Just as a separate driver exists for the INLIFE function, separate drivers must be constructed for the BMSS and EB applications. Each of these four drivers will provide for internal identification through the use of a separate application ID prefix.

2.1.5.1. Normal Update

Animal Room Update Actions (ARUA) can be conducted on a daily basis. NCTR will enter into BMSS the actions performed on a particular day before the initiation of the nightly data transfer to the Historical data base. Under these circumstances, the logon driver allows for interactive update of the current Census data, so that as each action occurs, the data are changed. This technique assumes that the current system day and time can be entered into those fields that keep track of when the various animal-related

records were added or updated. Therefore, the time stamp information is the time of data entry, not the time of actual occurrence of the event recorded. The delays between the two actions are normally small enough that this difference is not considered to be a problem. While system downtime may cause additional delays, downtime is expected to occur with low frequency and the anticipated delays are judged to be acceptable by NCTR.

The data changed or added during such sessions are eventually transferred to the Historical data base when the nightly transfer process occurs. The cycle resumes the next day when the operator enters new data corresponding to new ARUAs.

2.1.5.2. Training Room Sessions

The driver can be directed to point to data that have been created exclusively for use in training sessions. In that case, the driver would be directed to open and use a special set of files that contain data that have the same format as normal BMSS data. However, these data are normally not transferred to History or used in reports and may, in fact, refer only to fictitious rooms, strains, and animals. All normal BMSS functions are allowed from the training room version of the driver, but the training room version is to be accessible only from certain accounts (the BMSS colony manager's, for example). The data involved, while checked for internal consistency during entry, would not be checked for consistency with the rest of the colony. Thus, the colony manager can manipulate the data at will, even performing operations that would never be normally performed in the real colony because of the operation's destructive effect.

In order to accomplish this segregation of data, the demonstration data will be established as a study, and access to demo data will be made by entering a special project number at the NARSS project prompt. The same BMSS logon procedure can be used for access to the demonstration data, actual study data, and the colony data. The only special provision that must be made is the creation of a separate UIN file for use in the demonstration data. This file will be used to assign UIN blocks to litters "born" in the demonstration room and will prevent the use of real UINs in what is otherwise entirely fictitious data. The use of the demonstration data is thus almost completely analogous to accessing any other BMSS data.

Two special functions will be made available from within this driver. The first function will be the ability to save the files in their current state by copying to another set of data sets. The second function would be the ability to restore to a previously stored state. These functions will allow the files to be

returned to a known state, so that unusual animal configurations will not require manual clean-up, and so that known baselines can be set and reused during training or testing.

NCTR has indicated that although reports on the demonstration data are not normally required, occasions do arise in which reports are of use. Use of a separate study to store the demonstration data will provide for the automated loading of the data from the RMS files to the ADABAS files. However, the transfer of data from the demonstration Census to ADABAS need not be performed every day, but only when a series of sessions occur that require reporting. Functions analogous to the RMS data save and restore should exist to insure that the demonstration ADABAS files can also be baselined for testing purposes.

2.1.5.3. Supervisory Mode

In addition to the training room session functions, it is also desired that the colony manager have the capability to view the activity on a BMSS terminal at another location. This capability involves directing duplicate screen I/O from one terminal to another. The second terminal session, from which the colony manager watches operator actions or provides demonstrations of actual breeding colony activity, must have sufficient privilege to allow this activity.

NCTR has agreed to provide the low level functionality required for this function and the BMSS integration team will make this function available within the BMSS logon sequence. Only the colony manager will be allowed to use this function.

2.1.5.4. Delayed Entry of Normal Data

Because normal data entry may occasionally be delayed due to the system not being available, provisions must exist for data entry sessions in which the actual date and time of the recorded activity are specified. Normally, the system date and time of data entry are used to mark the records added or modified. This function is analogous to INLIFE Manual Forms Entry.

In order to emphasize the difference between normal and delayed entry of data, access to this function will be by a separate NARSS menu selection. The same driver program can be used to perform

either normal or delayed entry. All that is required to differentiate between the two sessions is a parameter indicating whether to use the system date and time or to offset that date and time to an operator-specified starting point. Accessing the BMSS via this path will therefore entail the entry of an additional parameter, namely the date and time the input session should have begun. All session times will be offset by the difference between the current session date and time and the time entered. Also, data entry via this method will result in setting the Ses_Trn_Manual_Entry_Flag to 'Y' in the session record. Access to this data entry method will be limited to authorized individuals.

When an occasion arises in which delayed entry of data is required, the sequence of data entry and application of the nightly routines must be maintained, so that all changes to data are made in the actual sequence in which they occurred. This must be procedurally enforced or data integrity may be lost.

2.1.5.5. Logon Driver Description

The process level specifications for the logon driver appear in Appendix F as detailed pseudocode, but an overview of these functions is appropriate here.

User Access

The logon driver is accessed from the NARSS menu at the same level as INLIFE. Depending on operator authorization, the logon driver allows access to either the new BMSS system, EB, or both. The choice of system may be made automatically when the user enters the project number at the NARSS project number prompt if only a single function is allowed. Otherwise, the available functions are displayed in a menu and the user selects the desired subsystem.

Once the subsystem has been selected, the appropriate logon module (INLIFE, BMSS, or EB) is selected and the application prefix is set for the duration of the logical session. Note that some operators will have the authority to Change Process (CP), but that a change in process to another of the three components mentioned here will force the termination of the session and the creation of a new session with a new application prefix.

Entry of a study number at the NARSS project number prompt will automatically cause the selection of INLIFE or EB or the demonstration room, depending on the study number entered and the

nature of the study. Note that the colony manager must access the demonstration room by entering the study number, either directly or via a CP from the BMSS.

Process Setup

After the appropriate logon module has been selected and basic session information has been established (application prefix, operator code, date and time), the BMSS or EB logon module reads and sets up in local storage certain information that is required frequently during the session. The information referred to here consists of those data that require frequent access but that were not judged to be appropriate for inclusion in the TDMS screen definitions because new elements may be periodically added. For example, the basic animal status checks will be made using lists of acceptable values that have been incorporated into each screen that requires them, because these values are not expected to change over time. The list of NCTR strains, and their associated data (such as recommended retire criteria, etc.), will be included in the data that are loaded to local storage one time, at session setup, for use in subsequent functions.

After the required tables are loaded for future reference, the logon program will pass control to the appropriate driver program for the function selected. In the case of the BMSS, that program is the BIS driver routine, from which the various Animal Room Update Actions (ARUAs), On-line Report Screens (OLRSs), and certain other functions (namely the Animal Orders and Allocation functions) are available. In the case of the EB system, a single application prefix and session are used to track access to a modified (actually, truncated) version of INLIFE and BMSS, merged into a single unit. The functions of the parts of this system are accessed using drivers very similar to those used in the stand-alone INLIFE and BMSS.

2.1.6. ADABAS files

The Historical data base is maintained on a separate machine using ADABAS. The ADABAS and TDMS files contain nearly identical structure so that the update process is as straightforward as possible. The RMS CAGE file is not needed in the Historical data, therefore, no BMSS CAGE file is maintained using ADABAS. There are some fields that are kept in the Census data that are not strictly needed in the Historical data, for example, the "current" fields containing duplicates of the latest status group fields. These fields exist on the RMS side for ease of use. The file and field level detail for the ADABAS files can be found in Appendix C.

Each of the files containing data that must be administered under the Good Lab Practices procedures possesses a duplicate file for the storage of audit trail data. The audit files have structures identical to the base files except that several fields have been added to allow for the storage of date, time, operator, and comment information pertaining to the correction of the raw data. The records stored in the audit file are the data as originally entered. The comment field tells how the corresponding record in the base files differs from the original.

Audit files exist for the ANIMAL, DATE-ESTABLISHED, LITTERS, and ALLOCATION files. The file and field level detail for the audit files can also be found in Appendix C.

2.1.7. Batch Reporting

Three different categories of batch reports exist in the BMSS. These are:

- Recap Reports - these are run daily and are the primary source for confirmation of animal room activity
- Regularly Scheduled Reports - these are run on a regular basis (weekly or monthly) and serve to summarize activities in the colony or study
- Operator Requested Reports - these are reports with predetermined format that are selected

Reports are requested automatically based on a submission interval (daily, weekly, monthly) or by operator request using Screen A.2 (Operator Requested Report Submission).

The data used to produce reports are stored in a separate file. This file contains the information needed to submit the various reports including the report name, input parameters, frequency of run and other accounting data. This file is read by a portion of the nightly batch job after the update portions of the process are finished. Report accounting data are available so that the number of times a report has been run can be obtained. Each unscheduled report that is submitted is entered into the file as a separate record and these records are maintained to indicate the history of each such report, so that the report usage can be determined if NCTR desires.

The automatic reports have a single record stored in the BRP-REPORT-PARAMETERS file indicating the submission interval. Each time one of these reports is run the file is modified to reflect the actual run time. Subsequent submissions then determine if the required interval has passed, and if so, the report is submitted again.

User-selected reports cause a new record to be inserted into the RMS file indicating the request and storing the input supplied by the operator. These new records are then transferred to the Historical data base during the nightly run. After all data transfer is complete, the report records are scanned to determine which reports are to be run and the newly selected reports are submitted.

For each report requested by an operator, a sequence number is assigned. This number is displayed on the report for identification purposes.

A separate program is used to read the new unscheduled records and the regularly scheduled report records to determine which reports to run. These reports are then run in turn with the regularly scheduled reports taking priority, depending on the scheduled frequency. The recap reports can be considered regularly scheduled reports in which the schedule is to run every day.

The running of the recap reports is triggered automatically by the successful completion of the update process. In fact, the successful completion of the update process is the end of a larger process that interfaces with HDEC, the nightly driver (see Section 2.1.3).

In the BMSS, the daily recap reports will be run against the Historical data base after the Historical data base has been updated by the data transfer process. The recaps could be run directly against the new or changed data extracted from the Census data base, but if this were the case, it would still be possible for an error to occur in the Census-Historical transfer. Such an error might not be noticed until the data on the Historical file were summarized in a weekly or operator-requested report. After such a long time interval, it is possible that the data could not be corrected. Running the recaps directly against the updated Historical data would improve the chances that undetected errors in the data transfer process would have small impact on the Historical data. Running recaps directly against the updated Historical data would also prevent the need for running two sets of recap reports, one to verify that the data were extracted correctly from the Census data base and one to verify that the data were updated correctly to History.

While it might be expected that logically combining these operations could lead to problems in determining just where an error may be occurring, the similarity in data base structure between the RMS and ADABAS files should reduce the occurrences and simplify the detection of such errors during program development and testing. The end result of having only a single set of recap reports to review should help to insure that the data are checked adequately. The HDEC process provides a means to correct any error in the Historical data and the combination of the extract and update procedures into a single logical process simplifies the application of error correction in the BMSS. Errors detected in the Historical data base and corrected through HDEC will be automatically corrected in the Census data by a step in the HDEC process (see Section 2.1.3.A).

The efficiency of running the recaps against the Historical data is a concern. While running recaps against the transaction data would insure that the target report population would always be small, the following discussion explains why running the recaps against the Historical data should not present a problem of efficiency. It is likely that a typical Daily Recap Report will lead to the printing of only a hundred or so records. The total population of Historical data is expected to start with hundreds of thousands of records. Therefore, the extracted recap data are expected to be only a very small percentage of the total amount of data. The efficiency of such a retrieval depends on the method used to retrieve the data. Sorting the data before retrieval would be too slow. Previously, it had been planned to extract the data from the ADABAS files with a simple FIND statement, then to perform the reporting function in a separate program after externally sorting the target records. Use of even a simple FIND construction could be inefficient with such a large number of records. However, by changing such a FIND to a READ LOGICAL, the data can be simply and efficiently extracted even from a very large population of data. READ LOGICAL can easily be used in all of the recap reports. The READ LOGICAL command is superior to the FIND command in these reports because it makes use of efficient means for sorting data internal to ADABAS.

2.1.8. Allocation Flow

The Allocation subfunction of the BMSS is an independent function using inputs from other NARSS subsystems. The Allocation function is most frequently used to provide "transition" of an animal from the normal BMSS breeding colony to INLIFE or EB. Allocation may also be used to transfer an animal from one study to another. Allocation is the process of applying study definition rules in order to determine

how the issue animals will be assigned to the experiment cages and then performing this assignment. The Allocation process is diagrammed in Figure 2.3.

In order to determine animal placement, the Allocation function must have access to certain data. The data that Allocation requires consist of:

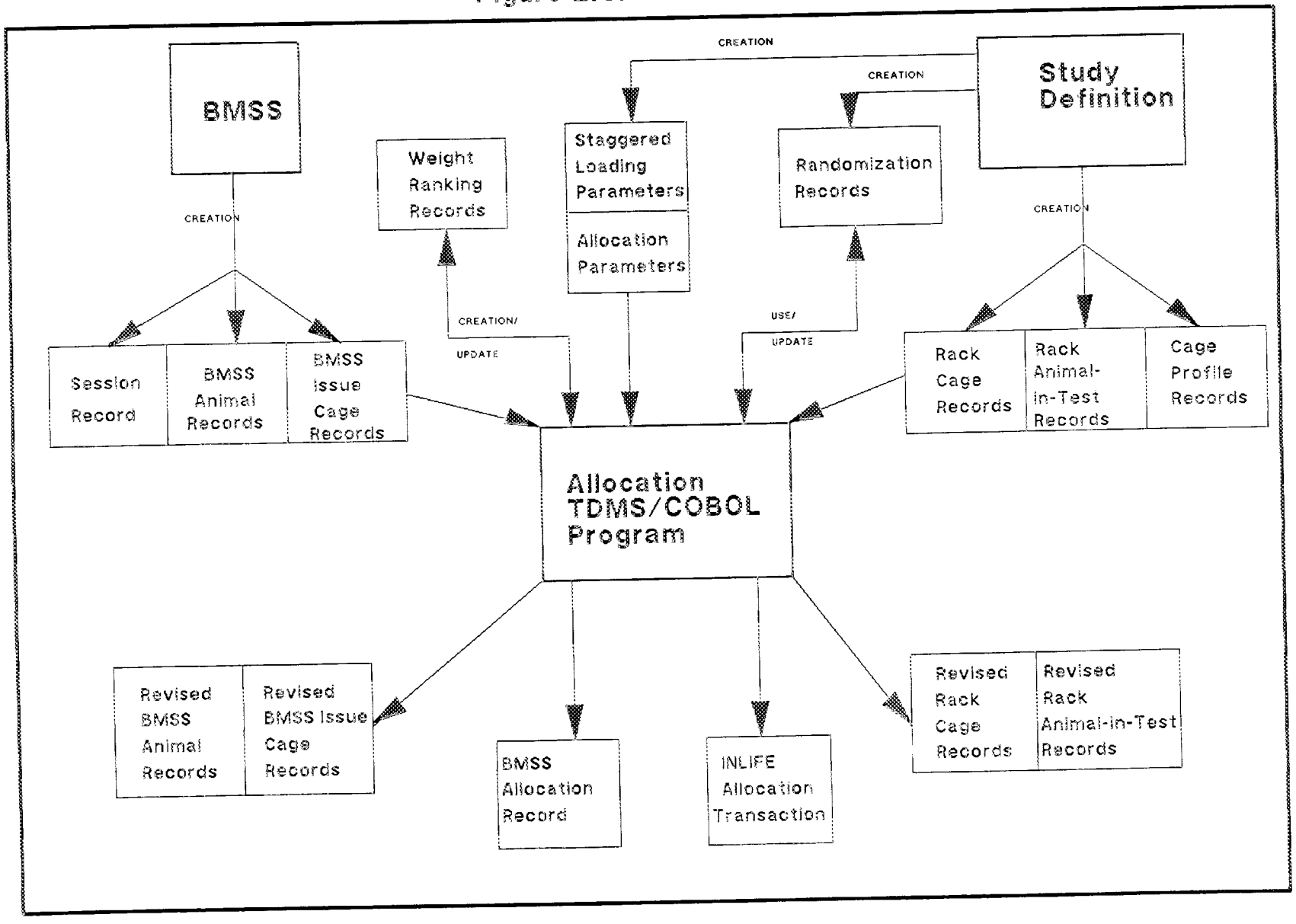
- o the BMSS issue cage data,
- o the INLIFE target cage data, and
- o destination cage filling rules, that include:
  - allocation parameters, and
  - randomization parameters.

Allocation therefore requires access to the source data files (general colony or prior EB BMSS data files), experiment-specific INLIFE and BMSS files associated with the target study, and the allocation and randomization parameters established in Study Definition. The allocation process is greatly simplified by storing the BMSS and INLIFE files associated with a particular experimental breeding study in the same subdirectory.

The issue cage contents are established within the colony or an EB study by using Animal Room Update Actions to assign animals from a breeding cage to an issue cage. These issue cages are then transported to the room into which the animals are to be allocated.

The rules by which issue animals are relocated into experiment cages are developed by the Principal Investigator and are then entered into the NARSS system through the Study Definition subsystem. These rules include the list of cages to which animals are to be assigned and the specific limitations on the animals to be placed in any one cage. In EB, each cage may contain only a single profile (because of the procedurally established single caged housing rule, see Section 2.4.4). In INLIFE, each cage may be assigned several CAGE PROFILEs and each such profile may be allowed to have several animals associated with it. However, no more than five animals may be allocated to a single INLIFE cage. CAGE PROFILES therefore represent a significant part of the application of cage-filling rules. The creation of the cage profiles is detailed in separate Study Definition documentation. Profiles must be created before animal allocation, and, in fact, before the specification of animal randomization (also defined in Study Definition).

Figure 2.3. Allocation Flow



At about the same time that the cage profiles are established in Study Definition, the allocation parameters are also specified. The allocation parameters indicate restrictions that must be placed on animals assigned to experiment cages. These criteria include such information as the rules detailing the placement of litter mates, whether animals are to be randomized by weight, and whether various parameters are to be evenly distributed across the experiment cages or not. The complete list of allocation parameters is described in the Study Definition Internal Design document. Some allocations are defined to occur as separate parts of a related whole. Each of these parts is referred to as a "staggered loading." Study Definition is also used to add the additional information pertaining to each of these events. In case of an allocation that occurs in a single event, there will be just one associated staggered loading record. Otherwise, there are as many staggered loading records as there are staggered loads associated with the test.

After all study information has been verified and entered into the Study Definition file and the necessary issue cages have been filled, then Allocation can proceed.

The operator initiates or continues an allocation by bar coding any experiment cage associated with the current allocation. The INLIFE cage file is then read to determine the test number associated with this particular cage, and the result is used as a key to find the appropriate Allocation record and Staggered Load record. By the time of allocation, the Staggered Load record has been used to create the Animal Randomization records that correspond to the cages in this staggered load. The randomization records are also obtained using the test number.

One pass is made through the Animal Randomization records to determine the exact cages present in the staggered load. This cage list together with the Staggered Load location fields are used to fill a buffer with a "map" of the cages belonging to this staggered allocation. If more than one rack is involved in this allocation, the map is formatted so that each rack is displayed as a separate screen with scrolling capability. The contents of the cages are also read, so that if this is a continuation of a previously interrupted allocation, the session can be resumed using the data as last entered. If this is a weight-ranked allocation, a check is made to determine if any previously established weight rankings have been retained and, if so, the data are read for use in the continuation of the session. Otherwise, the operator is prompted to weigh the issue animals. In this case, all animals in the issue cages must be weighed, and the results are stored in an RMS file identified by the test and staggered load numbers to allow for possible interruption of the

allocation session. ORNL assumes that only one animal may be present in each issue cage if weight-ranking is to be performed.

Sufficient information then exists to allow the allocation of animals, whether or not the allocation to this set of cages had previously been suspended or was just being started. The operator will be prompted to bar code one of the issue cages, and the contents of the issue cage will be determined by examination of the BMSS cage record corresponding to the issue cage selected. Based on the specific contents of the issue cage, a pass will be made through the Animal Randomization file to determine the first target cage that will accept the indicated animal. In the case in which an issue cage contains more than one animal, each animal should have the same sex, strain, and phenotype, and should be litter mates, so that any of the animals may be selected from the cage at the operator's discretion. If the animals can be uniquely identified by strain, sex, or phenotype, the first animal in the cage will be selected for processing, and the operator will be prompted to select that animal based on its unique attributes. In either event, the first target cage (based on the Allocation Number) in the animal randomization list that contains an empty, matching profile will be marked on the screen as the target cage and the operator will be prompted to clip the animal based on the clip sequence. The clip sequence is entered by the operator prior to bar-coding the first issue cage. If no animal in the issue cage fits any of the remaining cage profiles, then the operator will be prompted to bar code the next issue cage. The operator will also have the ability to override the automatic use of issue cage/target profile matching, but for each animal that is "manually" placed, a note will be required in the General Notes file to indicate the reason why the override was required.

Note that both sequential and manual allocations are also allowed. In the case of a sequential allocation, the procedure is the same as for a randomized allocation, except that the cage profiles in the animal randomization file are not assigned at random, but in ascending order. In the case of a manual allocation, the animal randomization file is set up the same as for a sequential allocation, but the operator is allowed to assign animals in any order, without requiring that notes be added to the General Notes file for each animal entered.

During the allocation process, a relatively large number of records are created or modified for each animal involved. For each animal moved from an issue cage:

- the BREEDING Issue Cage record is modified to reflect the movement of the animal from the cage

- the BREEDING Animal record is modified to indicate the animal's new (and possibly final) status; if the animal is found to be dead during allocation, then destination and CID information may be required

- if this is a weight ranked allocation, the corresponding Weight Ranking RMS file record is marked to indicate that the animal has been used.

When an animal is successfully allocated:

- a BREEDING allocation record is created,

- if the operator was forced to override the Study Definition parameters, then allocation notes are added to the General Notes record,

- the INLIFE Cage record is modified to indicate that the animal now resides in INLIFE (or EB), including the incrementing of the appropriate profile actual quantity field,

- the INLIFE Animal-In-Test record is modified to indicate the pertinent animal and test information,

- an INLIFE allocation transaction is created,

- the ANIMAL RANDOMIZATION file record is marked to indicate that the animal has been used.

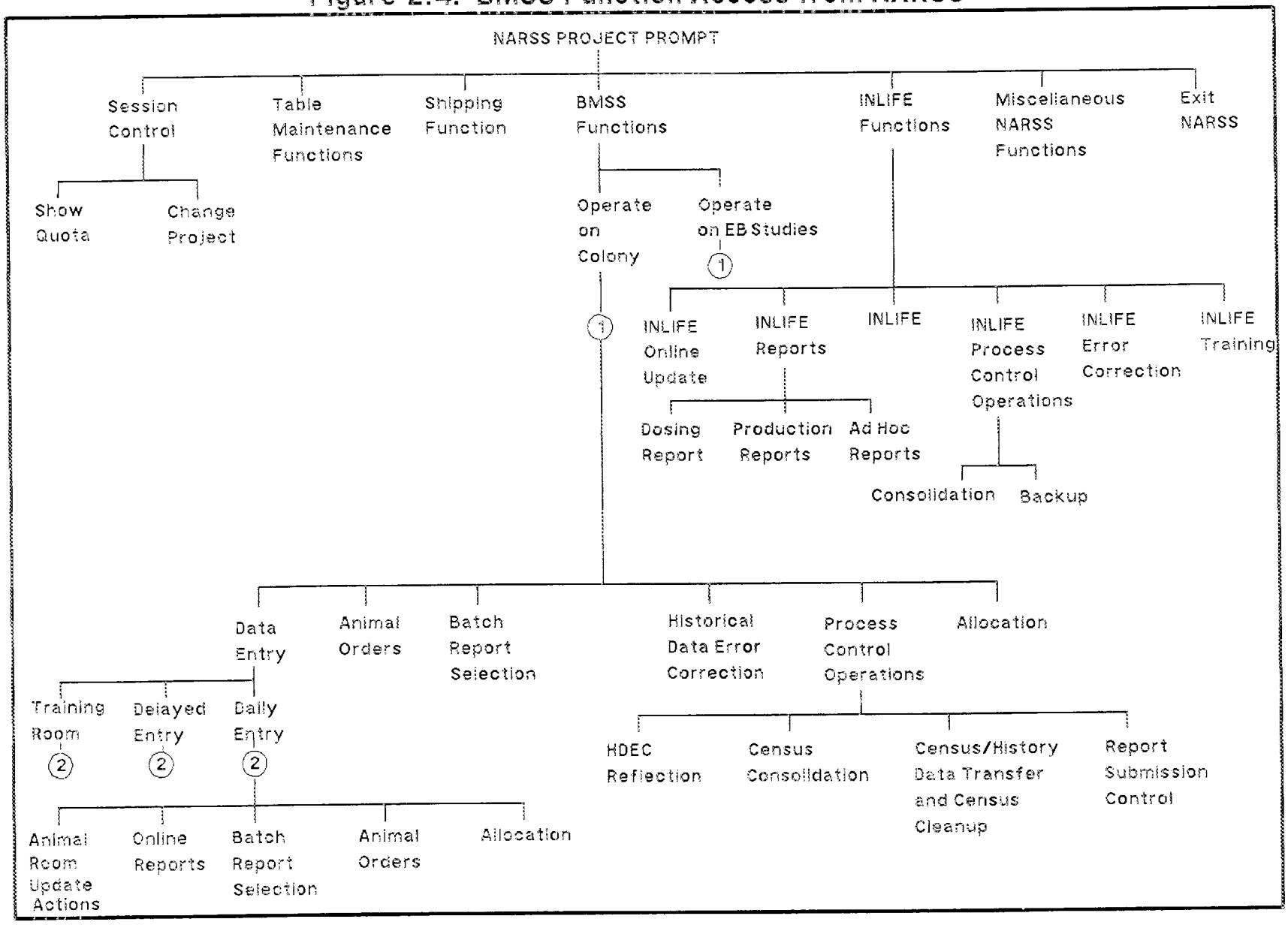
The General Notes function may be employed to record the operator IDs of additional operators assisting in this allocation.

2.2. USER VIEW

The operator sees the BMSS as a subsystem of NARSS and the BMSS is accessed directly from the NARSS menu on the NCTR VAX cluster. The BMSS can be thought of as a number of screens and reports that are used to input data relating to the breeding colony and experimental breeding and to review or assess the current or past status of the colony and experiments conducted. The BMSS functions accessible from NARSS are shown in Figure 2.4.

From the operator's standpoint, there are three high level functions available to perform the daily task of maintaining the breeding colony or, in the case of experimental breeding, the experimental-specific

Figure 2.4. BMSS Function Access from NARSS



breeding population. The operator can perform interactive update functions, on-line and batch reports, and data re-entry and validation.

2.2.1. Interactive Update Functions.

The interactive update functions are: Animal Room Update Actions, Animal Orders, Allocation of Animals to Experiment, and (in the case of experimental breeding) INLIFE Actions. These are functional groupings of the specific BMSS update functions, some of which are composed of a number of individually selected operator functions. Following the summary of these functions, screen access is discussed from the operator's point of view.

2.2.1.1. Animal Room Update Actions

Animal room update actions are used to record an animal's history in the colony from birth to eventual death or removal from the colony. The functions included here are:

New Animal Entry (Screen A.10), allowing the entry of new adults or litters into the colony from an outside source.

Change Cage Status (Screen A.11), allowing the changing of the status of all animals in a cage (between feeder and breeder status).

Move Cage (Screen A.12), allowing a cage to be physically relocated on a rack. The cage may contain animals or be empty.

Animal Assignment (Screen A.14), allowing an individual animal to be moved and/or assigned a new status. Note that a Move Animal function does not exist for INLIFE. This is discussed below.

Removals (Screen A.15), allowing for animals to be removed from the colony as a final action.

Retirement (Screen A.16), a special version of removal allowing for the single or simultaneous removal of members of a breeding pair; this is a final action in BMSS, but not in EB.

Pregnant or Plugged Observations (Screen A.17), allowing for the observation of either condition in the female of a breeding pair.

Litter Observations (Screen A.18), allowing for the observation of a litter and the assignment of a block of UIN's to that litter.

Issue Animal (Screen A.19), allowing for the placement of an animal in an issue cage for later transfer to an experiment or out of the colony.

Pup Weighing (Screen A.21), allowing pups to be weighed, as a litter, group, or individually.

2.2.1.2. Animal Orders Function

The animal orders function is used to allow for the entry, modification, and display of orders for animals from the breeding colony and consists of three separate screens (separate versions of Screen A.4).

Orders are entered based on information received from the PIs and it is likely that initial entry of the order will be by a liaison or the colony manager. An order for a single experiment and day may be composed of several parts, each requesting a different strain or sex, for example.

Orders can also be displayed and modified. In the case of display, the data are shown on the screen sorted by date required. In the case of modify, the data are displayed in similar fashion and the operator can select GOLD functions to insert additional lines for a given experiment and day or to delete lines entirely.

When an order is deleted, the data are not removed from the data base, but are merely marked as deleted. This allows a complete history of the changes to an order to be obtained, if desirable.

2.2.1.3. Allocation Function

The Allocation function (Screen A.3) is used to direct the transfer of animals from issue cages to experimental cages, thereby setting up the animals needed for an experiment. This function is considered to remove an animal from the breeding colony, though the animal may be transferred to an experiment within which breeding is required.

Use of the Allocation function requires that an INLIFE animal rack be prepared for receiving experiment animals. Animal rack definition is accomplished through the Study Definition function and is not detailed here. In summary, the set of cages that will contain the animals is defined and described by Study Definition so that the permissible contents of each cage are known.

Orders can be placed on the system using the Animal Orders function (Screen A.4) and filled by issuing animals (Screen A.19) or by using the Fill Animal Orders function. The orders required for a particular allocation may impose restrictions on the animals issued so that only animals of a particular sex or strain are made available. However, animals may be issued even if orders have not been entered.

If an order has been filled or the required animals have been issued, the issue cages are transferred to the room containing the destination rack, and the allocation itself may take place.

At this point, the Study Definition parameters are required for further processing. These parameters are identified uniquely in Study Definition using a technique that allows any cage to be linked to its corresponding Study Definition record. The operator may therefore bar-code any cage in the group, and the Study Definition data are obtained.

Study Definition defines parameters for the pairing of mates in EB; for example, whether pairs are to be of the same treatment, and, if not, what treatment combinations. Allocation within an experiment will be Study Definition-driven. By forcing the definition of parameters to be assigned to newly allocated animals, e.g., treatment/test of the animals to be mated and the corresponding litter, GLP will be followed and the data modification should be minimized.

Based on the contents of the Study Definition record, it may be necessary to weight-rank the issue cages. In weight ranking ORNL assumes that the issue cages contain only a single animal, and each cage is processed by weighing the occupant.

After any necessary weighing, the rack configuration that corresponds to the cages defined in the Study Definition record is displayed. A rack segment of up to 6 rows x 7 columns (6x7) of cages may be displayed on one screen. All rack configurations will display either a segment 6x7 cages or smaller (if fewer cages are involved), except 10x10 racks will be displayed as four individual 5x5 segments. The current cage contents will be displayed on the screen for each cage that is referenced by the Study Definition record. The operator initiates the allocation of animals by bar-coding an issue cage.

Once the cage contents are known, the Allocate function determines which cage will accept an animal of this description using the Study Definition information and current rack contents, and the function indicates the destination cage into which the animal should be placed.

This process continues until all the required cage positions are filled or all the issue cages are empty. At all times the current rack contents are taken into account when considering where to place the next issue animal; this allows interruption of processing so that breaks or partial allocations can be handled. In the case of a weight ranked allocation, the weight rankings are stored in a file containing sufficient information to link the file to the Study Definition record. This information may then be used if the weight ranked allocation is interrupted.

A partial allocation should not be confused with a staggered loading. In a staggered loading, a number of small groups of cages are requested to be filled in series. An example would be a set of five individual allocations in which two columns are filled each time. Each of these allocations may be requested to be performed on a separate day. A partial allocation is an allocation in which insufficient animals are available to complete the Study Definition request.

In the case of a partial allocation, the Principal Investigator may later request that empty slots be filled. If this is the case, the start date for the animals that were added later may need to be adjusted. The start date is assigned automatically, and it may be necessary, in the case of a partial allocation, to change the start date of the later animals. Such changes must be requested by the PI and noted. Changes are then accomplished by using Historical Data Error Correction.

During the allocation process, it is possible for animals to expire or escape. After the allocation is initially thought to be complete, empty cages or dead animals may be noted. An option exists to allow the operator to place animals to fill such gaps.

While the assignment of destination cage is normally determined by the allocation program based on the Study Definition rules, it may be necessary to override these rules on occasion. Only authorized operators are allowed to perform such overrides, and when an override is performed, the operator is requested to enter a description of the reasons for the override. These data are then stored in the Study Definition record for later use by the PI.

For some allocations, it is appropriate or acceptable for the operator to manually decide the destination cage. In this case, the Study Definition rules will note the study as requiring manual allocation, and the assignment of animals to destination cages will be made at operator discretion.

The full list of Study Definition parameters for use by the Allocate function is available in the Study Definition Internal Design Document.

2.2.1.4. INLIFE Actions

INLIFE actions are performed on animals that have been set up on experiment through ALLOCATE. INLIFE existed as a stand alone and fully implemented system before BMSS and INLIFE were integrated. There are many INLIFE functions, and descriptions of the individually selected operator functions are described in detail in the INLIFE documentation (See Section 1.1, References).

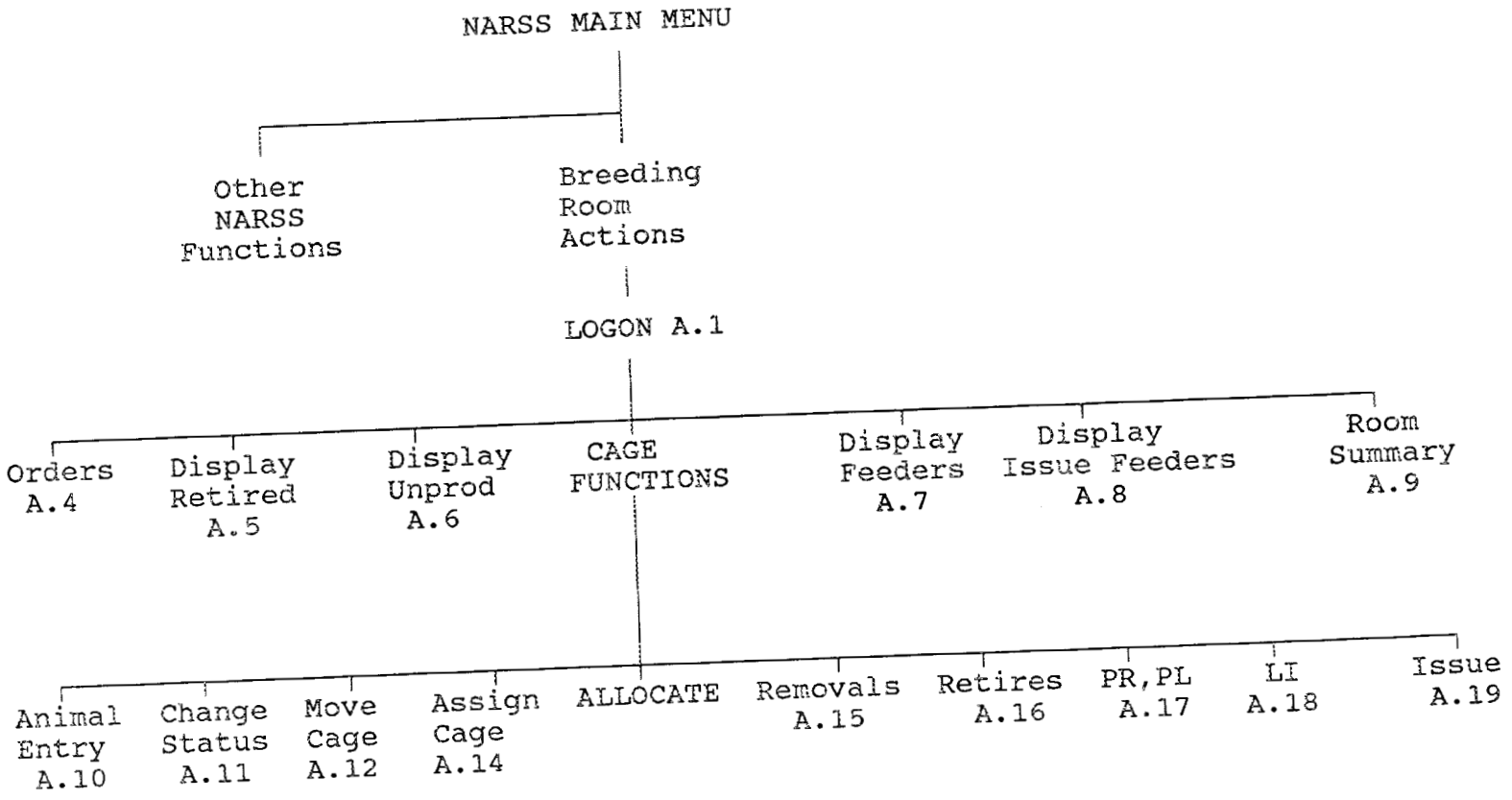
2.2.1.5. Screen Access

The various screens are accessed as individual entities that are arranged in a hierarchy under the NARSS menu in the manner shown in Figure 2.5. The options available from the NARSS menu vary depending on operator access privileges, and the menu displayed depends on those functions accessible to the operator. If only a single option is available, then the option is automatically selected, and the next menu or first function screen is displayed.

An operator will view only those functions indicated as authorized according to the authorized operator and valid functions files. At points deeper in the BMSS hierarchy, it is necessary for some functions to validate the Unique Identification Code (UIC) of the operator before allowing access. At this level, RMS file protection rules are also used to protect data. The BMSS operator authorization process will perform in the same manner currently employed by NARSS, BREEDING, and INLIFE.

In the case of the BMSS, the functions available constitute two general areas, each of which uses full-screen TDMS processing to allow for data entry. These areas are the BREEDING and INLIFE subcomponents. The INLIFE screens are described in other NCTR documentation. The BREEDING screens fall into several categories as described above, but all screens follow a generally similar format. Header information is displayed at the top of the screen. This information consists of column headers in the case of the on-screen reports and cage content information in the case of Animal Room Update Actions. The actual contents of the header information for each screen are shown in the screen layouts in Appendix A.

Figure 2.5. General BMSS Screen Flow.



The body of the screen is devoted to the on-screen report output, or the summary of Animal Room Update Actions. In the case of on-screen reports, the data displayed are associated with the current request; in the case of the update actions, the data are associated with the current cage (or range of cages if a sequential function has been selected).

The bottom of the screen is used for primary screen input and function selection. The operator enters the function or cage to select the Animal Room Functions or to describe the cage a previously selected function is to manipulate.

Each such function usually has a number of additional parameters required to fully qualify the report or update action. These prompts are displayed at the bottom of the screen in turn, and the operator enters the appropriate data. The data entered are validated against appropriate checks where the checks can be defined (See Appendix A). After the data have been entered at the bottom of the screen, they are transferred to the corresponding location in the body of the screen. Some data entered in the screens are more critical than others. For example, the number of the animals in a litter is used to create separate records in the animal file, and to reserve blocks of UIN's. Litter size will require double entry to insure that the value is entered correctly.

After all data for a cage have been entered and moved to the body of the screen, the operator is given a chance to review and make changes to any value entered. The body of the screen is opened for edit in full screen mode. The operator may tab to any field and modify its value. If more data exist than will fit in the display area, the operator will be allowed to scroll the data up or down to position the cursor to the desired field. After entering a change, the data are rechecked for fields possessing edit checks.

2.2.2. On-line and Batch Reporting

The reporting capabilities of the BMSS have been placed into two separate areas: On-line reports and Batch Reports. Animal Room On-line Reports are used to display selected information about groups of animals in the colony and may be used to determine which Animal Room Update Actions are to be performed. These reports are available from the same level at which the BMSS Animal Room Update Actions are accessed.

Display Retirement Candidates (Screen A.5), is used to indicate the animals that have reached the end of their useful breeding span.

Display Unproductive Breeders (Screen A.6) is used to display breeders that are no longer producing litters.

Display Feeders (Screen A.7) is used to display all animals in feeder status that have an age within an operator-specified interval.

Display Issue-Feeders (Screen A.8) is used to display all animals in issue-feeder status that have an age within an operator-specified interval.

The Room Work Summary (Screen A.9) is used to provide a summary of the animal update actions that have occurred in a given room during the current day.

The data displayed in the on-screen reports may occupy more lines than are available on one screen. The operator will be allowed to scroll the data by including non-display input fields on each line.

BMSS batch reports are divided into three groups: automatically submitted/scheduled reports, recap reports, and operator-selected reports. The automatically submitted/scheduled reports are general purpose reports relating to breeding colony activity on a weekly or monthly basis. The recap reports are also automatically submitted, but differ somewhat from the other automatically submitted reports. The recap reports serve as a record of what occurred in the breeding colony and are used by the PI or breeding colony manager to insure that the scheduled or requested activities took place correctly. Operator-selected reports (Screen A.2) are selected from within the same level at which the Animal Room Update Actions are performed, but the reports themselves are run against Historical data

and are performed in batch mode after the operator selects the desired reports. Figure 2.6 shows the general flow of report selection and automatic submission.

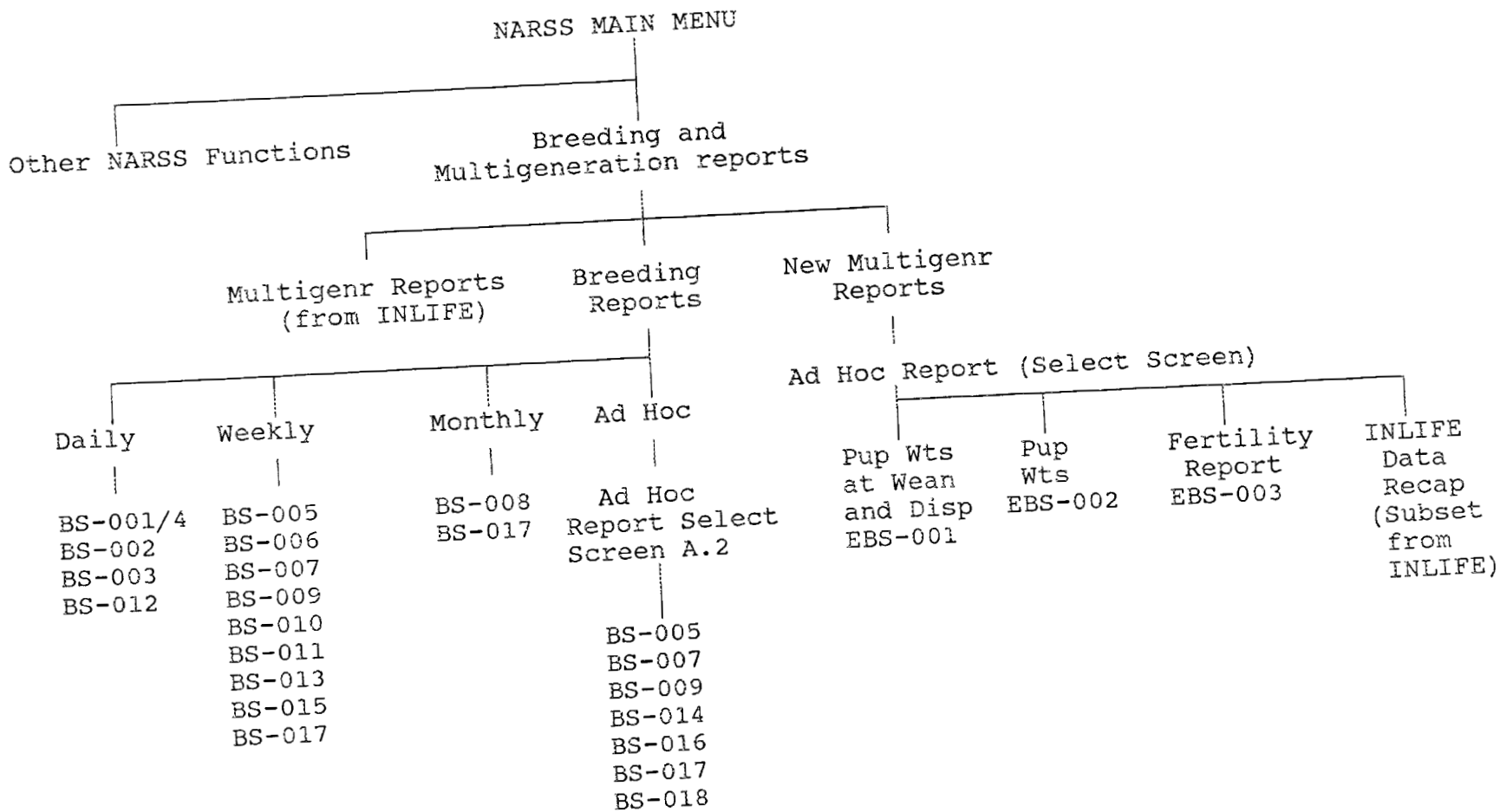
The 24 BMSS batch reports are listed along with a designation as to which category they belong in Table 1.1. of the Report Revisions section (Section 3.2.1). INLIFE reports are also available, but they are not referenced specifically here (See the INLIFE documentation references in Section 1.1).

2.2.3. Data Re-entry and Validation

While the screen processing described above contains data checks, requires double entry of critical data values, and allows operator edit of the newly entered data before posting the changes or additions to the data base, it is possible for erroneous data to be entered into the system. These data may be found to contain errors soon after entry, during checking of a recap report, or possibly even later. Errors made during daily maintenance that are not or cannot be corrected will eventually propagate to the Historical data base. Several methods exist for insuring that the BMSS data can be corrected if a data entry error occurs: performing additional Animal Room Update Actions to bring the data into correct form, HDEC, and ad hoc error correction.

In some circumstances the error that exists will be amenable to correction by issuing further Animal Room Update Actions. For example, if an animal was actually assigned to cage 35, but the digits of the cage number were transposed during data entry, then the system would show that the animal was in cage 53. To correct these data, the operator would enter data that indicate that the animal was assigned from cage 53 to cage 35. The on-line data would then match the actual cage contents. The animal would not need to be moved, because the animal is already present in the proper cage. In this, and similar cases, the incorrect data can be modified under operator control using normal Census data screen access.

Figure 2.6. General BMSS Report Flow



If, however, the data that are in error cannot be changed with the screens, then HDEC must be employed. The operator should make a note that a correction is required and wait until the Historical data are updated with the changes made on the day the error occurred. After this time, the data will be on the Historical data base. HDEC allows access to the ADABAS data stored in the Historical files and allows any field to be changed. For example, an animal may be incorrectly sexed, but no facility exists in the data entry screens to change the value entered for sex after the animal has been weaned. This data value must be changed by using the HDEC screens. All HDEC-changed data values are written in ADABAS NATURAL and are accessed as a separate NARSS function. When data are corrected using HDEC, the same change may be required in the corresponding on-line data, if the data refer to animals still in the Census. Application of the HDEC changes to the Census data are accomplished as a subsidiary process to the on-line changes carried out as part of HDEC, and is accomplished in the background, requiring no operator intervention (see Section 2.1.3.A.). If neither of the above methods can be employed to fix the error, then one of two possibilities exist:

First, the data that were entered in error may have already been used to assign other data values, and in such a way that the error must be accepted. An example of this kind of error is in the entry of an incorrect litter count. If the operator enters a litter count of 8 and there are actually 10 animals, then an unrecoverable error exists. The litter count is used to assign a block of unique identifiers to the litter, and because of the way in which these blocks are assigned, this block must be presumed to be assigned correctly from the outset. Later, at wean time, this block information is used to assign individual identifiers to the litter animals. In this case, when all such identifiers have been used, there will be two animals left in the cage. The operator has no choice but to surplus the animals. In this case, there is no corresponding data entry required---for all practical purposes, these animals never existed. In the case of entering a litter count that exceeds the actual number in the litter, then more individuals will appear to be in the cage than actually were born. After all the real animals are weaned (or otherwise removed from the cage), the operator will then have to go through the motions of surplusing these excess "animals", so that the system can be brought into synchronization with the real situation.

The second alternative is that the data could be corrected in principal, but no mechanism exists to change the data by using either Animal Room Update Actions or by using HDEC. This second alternative should occur infrequently, because error corrections of this type most likely represent an omission in either the update or data re-entry and validation system. In this case, however, the error would have to be corrected on an ad hoc basis, using routines written for that specific problem. If ad

hoc correction can be done, then such an error correction is a likely candidate for a new update or data re-entry and validation function.

2.3. DATA BASE CONSIDERATIONS

During the internal design of the BMSS, a number of design questions arose that were eventually resolved by discussion with NCTR and analysis of the required data structures and processing limitations specified in the Requirements phase.

The Historical data will be stored in ADABAS files, providing consistency with other NCTR data bases and to allow the use of NATURAL for reporting purposes.

The Terminal Data Management System (TDMS) was selected by NCTR to perform the full screen applications of the BMSS in preference over ADABAS NATURAL. This decision was made based on a number of considerations:

- o ADABAS is licensed only on the VAX 8650
- o Running on a separate machine reduces normal resource competition
- o Session priority cannot be meaningfully assigned if all data base access must be threaded through a single process (the ADABAS nucleus)
- o Integration of BREEDING and INLIFE functions should be easier if the screen processing methods used are based on the same software tool

ORNL also concluded that the level of integration of the existing INLIFE and the new BMSS functions should be relatively high. High level integration eventually led to the physical integration of the INLIFE and BREEDING ANIMAL files along with some important ramifications:

- o The INLIFE system has already been designed and is in use. The BMSS will replace and augment the existing BIS (on MODCOMP and IBM) and will be integrated with INLIFE as one of the NARSS subsystems.
- o INLIFE and BREEDING (whether BREEDING or EB) have in common that they deal with individual animals. INLIFE identifies an animal by its UAI (Unique Animal Identifier), while BREEDING will continue to use the UIN (Unique Identification Number). In order to provide tight integration of the two systems, which is required for the most part by the need for EB, the INLIFE ANIMAL file will be used to store both INLIFE-specific and BREEDING-specific

information. This file will contain both the UAI and the UIN, thus permitting both integration of the two systems and continuity with the previous BIS data. Another project, BIS Data Conversion, is being worked on simultaneously to provide access to the old BIS data from the new system.

- In INLIFE, the Animal file had previously contained those data elements specific to an animal that never changed status. The integrated file will permit tracking an animal's history as a breeder or feeder prior to an allocation or while on an EB study. The ADABAS data structure shown in Appendix C notes the fields previously found in the INLIFE animal file.

- BIS and INLIFE also use the strain code, but each provides a different format. The strain code has therefore been included in both formats in the Animal file.

- INLIFE Census data are stored in separate subdirectories on the VAX, with each subdirectory corresponding to an individual study. Because of EB requirements, BMSS will do the same; individual copies of the various BREEDING files will be stored in the study's subdirectory.

- Collection of individual study data to a single area for posting to the Historical data base will employ methods similar to those used by INLIFE.

- In the case of ordinary breeding colony data that are not associated with a study, a single subdirectory will be used to store the data.

Historical information related to an animal's previous status or location, the litters born to a breeding pair, and related information will be stored in the Historical data base as ADABAS periodic group data. Processing considerations indicate that these data should be stored in the group in order from oldest to newest. Because some functions require access to the most recent data from certain periodic group data, duplicate fields for the most recent occurrence have been added to the file definitions.

The physical file structure for the BMSS was determined by considering the logical structure presented in the External Design Document and the fact that ORNL and NCTR had decided to integrate the INLIFE and BMSS ANIMAL files into a single structure.

INLIFE uses only session identifiers to provide a link to a session record. In the case of the BMSS, ORNL and NCTR decided that there are two major input data sources, and each must be handled in a different manner:

- new data entry will allow for full use and storage of session information using a session record, in a manner patterned after INLIFE;

- converted data from the previous BIS system will require individual fields, but only for operator and date, because much of the new session information was never kept or is very difficult to derive from the old data.

Use of this technique should not provide any significant problems. Each portion of the system (data conversion or new BMSS functions) will be directed to use the appropriate fields in the Historical data base, and the fields that are unused will automatically be compressed by ADABAS. The individual operator, time, and date fields are not needed in the TDMS data structures.

Because of GLP procedures, the use of HDEC leads to a need for an audit trail, in a manner analogous to that used in INLIFE. Separate files have been defined to hold these audit trail data. They were created by adding the standard INLIFE audit trail structure to each BMSS file.

None of the various files are thought to be likely to overflow the ADABAS physical record size limit (around 3000 bytes) after compression. The litter and animal records each contain periodic group data which could become quite large in the case of a very large litter in which individual pup weighings were performed or in the case of an animal that changed status an abnormally large number of times. However, the chances of large periodic group data occurring is small. No specific means for dealing with many weighings of large litters is outlined here, because the practical limits on litter size and weighing schedules indicate that the probability of such an overflow occurring is extremely small. In the case of an animal with a large number of status occurrences, the BMSSLOOK report will indicate if an animal has a large number of status elements. If the number becomes too large, the animal could be removed from the colony and re-entered through New Animal Entry, or the oldest status elements could be removed using a program written specifically for that purpose.

While a specific NCTR project is being carried out in parallel to populate the Historical data base with the converted BIS data, no such project is expected for the Census data. Initial population of the Census files with data is expected to be performed by the routines that are normally used to reflect changes carried out by the HDEC process. A special run will be made in which the data in the ADABAS files are all considered as having been changed and all pertinent Census data should be derived. ORNL realizes that this method will convert some bad data from History and plan to take this into account in the BMSS engineering.

The experiment number has been added to each BMSS file in order to logically separate BREEDING and EB data.

In EB, there may also be a requirement for Study Definition to define experiment specific issue cages complete with cage profiles, so that pups from a given treatment may be issued to the proper type of cage from which to be allocated.

2.4. MODULES AND INTERNAL FUNCTIONS

This section presents additional description of certain modules and internal functions of the BMSS.

2.4.1. BMSS Startup

The entire logon, access, and menu display portions of the BMSS are to be modeled closely after existing INLIFE code. The same authorization and function definition files are expected to be used to determine authorized access and to then determine individual functions available to the operator. The actual incorporation of the new operator and function data into the production files will be accomplished by the NCTR Software Control Group. During implementation and testing, copies of these files will be modified for use by the implementation team.

Session information will be stored and maintained using the existing INLIFE files, and code modeled after the INLIFE code will be used to maintain the session information. BMSS session data will, however, be transferred to the ADABAS session files using separate load programs and will be corrected with separate HDEC routines. For the purposes of BREEDING colony sessions, a dummy test number will be required to link sessions back to the colony specific area (because BREEDING is not associated with a specific experiment).

2.4.2. On-line Update Summary

New ANIMAL records are created at wean time (accomplished in the Animal Assignment and Issue Animal screens) and when newly purchased animals are entered into the system (accomplished in the New

Animal Entry screen). At the time the ANIMAL record is created, all fields can be entered (by option or requirement) except the REMOVAL-DESTINATION-GROUP. The REMOVAL-DESTINATION-GROUP is updated at removal time by the Animal Removal screen, and at retirement time by the Animal Retirement screen. Additionally, the Status Group and the Current Status are updated by the following screens: Change Cage Status, Move Cage, Animal Removal, and Animal Retirement.

New DATE-ESTABLISHED records are created by the New Animal entry and Animal Assignment screens. At this time, the following fields are entered: Dam UIN, Sire UIN, Date Established, Create session information, and Progeny Strain. Each time a litter observation is noted by an operator, a new entry is made into the Pair Observation group. These observations are entered through the screens Note Pregnant or Plugged, and Note Litter. When the breeder pair is dissolved (Removed, Assigned, or Retired (in BMSS, but not EB), the Date Pair Dissolved field and session information pertaining to the dissolution event are entered.

New LITTER records are created by the New Animal Entry and Note Litter screens. At the time of creation, all fields except those pertaining to Pup Weighing may be entered by option or requirement. Pup Weighing information is supplied by the Pup Weighing screen and the Issue Animal screens.

ALLOCATION records are created by the Allocation screen.

ORDERS records are created by the Enter Animal Orders screen, and are updated by the Fill Animal Orders screen. The ISSUE screen will increment the number delivered to the next available order and will cause records that have the total number of animals requested to be marked as filled.

The CAGE file is used to allow rapid display of the information related to a single cage and to identify the individual animals or the litter in a cage. CAGE records are updated by the following screens: New Animal Entry, Change Cage Status, Move Cage, Animal Assignment, Animal Removal, Animal Retirement, Note Litter, Issue Animal, and Allocate. The CAGE data are stored only in the RMS files are not maintained in the ADABAS files.

The BRP-REPORT-PARAMS file is used to store information necessary for generation of breeding reports. A new record is created each time an operator requested report is selected. One record exists for each regularly scheduled report.

The BMSS files are stored as RMS data, ADABAS data, or both. The RMS files will be updated directly by the screen processing modules, as indicated above. The ADABAS files will be updated when the information is transferred from Census to Historical. Additionally, the following ADABAS files store audit information: AUD-ANIMAL, AUD-DATE-ESTABLISHED, AUD-LITTERS, and AUD-ALLOCATION. These files will be used to store HDEC corrections to the Historical data base. They are identical to the ADABAS files listed above, except that they contain additional audit information.

2.4.3. UIN Assignment

During update operations, NCTR occasionally is required to allocate a block of UINs to a litter. Each set of UINs is specific to the strain, so that the numeric portion of the UIN starts with 0000001 for the first animal in each strain. Subsequent UINs are assigned in blocks corresponding to the individual pups counted in a litter at the time the litter is first observed.

The assignment of block UINs is to be made across all experiments, however, so that if a strain appears in several experiments, the UIN assignment will reflect the true order of birth. For example, if a litter is observed in the breeding colony in the morning, the litter may be assigned numbers 1000001 through 1000005. Later in the day, a litter in the same strain may be observed in one of the EB studies, and the litter will be assigned numbers beginning with 1000006. There is, therefore, only one file that will contain UINs and this file is shared across all BREEDING functions and EB experiments.

The assignment of UIN blocks can be made only one time, so it is imperative that the number in the litter accurately reflect the pup count. This number will therefore require double entry. The actual assignment of the litter block will not occur until the operator confirms that the other data on the screen are correct by requesting another cage or a non-cage-related function. During actual UIN block assignment, the UIN file will be locked so that only one operator can request blocks at a time.

Because of file locking, it may happen that one operator is forced to wait until another operator already in this function frees the UIN file. Operator contention for files is thought to be a low probability event by ORNL and NCTR, and NCTR agrees that imposing such a wait is acceptable in order to insure that block assignments occur as required.

2.4.4. EB Single Caged Housing and Animal Assignment

Animals assigned to an experimental breeding study will have data stored in two sets of RMS files, INLIFE and BMSS, allowing both INLIFE observations and BMSS breeding activities to be recorded. However, two sets of files also impose certain restrictions on the functions that can be performed and the manner in which they are performed. Care must be taken to insure that both systems are updated at the same time, and that the physical configuration of cages and animals is accurately reflected in the on-line data.

With one important exception, the data stored in the INLIFE and BMSS files are independent. The cage number is shared between INLIFE and BMSS. The cage data are maintained separately in the two systems, and this fact is used to allow certain processing of animals used in an experimental breeding study.

INLIFE does not make provisions for moving animals from cage to cage. Moving animals from cage to cage is important because animals must be moved from one cage to another on a temporary basis to allow for mating of animals during experimental breeding. BMSS does allow such movement. In order to handle the disparity between INLIFE and BMSS, NCTR has established a procedure that will allow for the animal to be moved and still preserve existing INLIFE functionality.

During breeding, the Sire will be physically moved to the Dam's cage, but the INLIFE cage card will be moved at the same time. The animal will be moved using the BMSS assignment function. No terminal entry will take place to record the movement using INLIFE. BMSS will therefore indicate that the animal is present in the BMSS cage corresponding to the Dam's cage, while INLIFE will show that the Sire is present in the original cage. In order to insure that this methodology will work, NCTR has elected to procedurally insure that experimental breeding cages possess only one occupant, except while breeding is taking place. This condition is termed "single caged housing" and will be enforced for all EB studies. Limiting the normal number of animals occupying a cage to one insures that EB breeding cages can be identified by the BMSS cage contents and that the Sire's original cage can be tracked using INLIFE data. The use of this procedure also has ramifications in regard to animal removals. These are discussed in the next section.

2.4.5. BMSS/INLIFE Shared Functions and the BMSS Death Check

INLIFE and BMSS share a number of functions. These functions are divided into two categories:

Class I: Session ID Maintenance functions - both BMSS and INLIFE use very similar session information; this information should be allowed to be entered from each subsystem independently; each "version" will reference the application prefix of the application performing the manipulation, and the data will eventually reside in the same ADABAS session file. The functions referred to here include balance calibration.

Class II: Animal Maintenance Functions - these are the functions that must be restricted so that the function can occur in only one location to guarantee consistency. ORNL and NCTR previously agreed that the functions that fell into this category would be eliminated from the INLIFE routines called for EB studies, and that the functions would be accomplished by setting Study Definition parameters if Study Definition already made provisions for doing so.

The Class II shared functions are INLF_RELOCATE_CAGE and BMSS Screen A.12 (Move Cage), INLF_CHECK_DEAD and the new BMSS Death Check function, and INLF_REMOVE_ANIMAL and BMSS Screens A.15, A.16 (Removals and Retirements). The INLIFE relocate cage function should be disabled, and the BMSS Move Cage function should generate INLIFE transactions in EB studies. Animal removals should be disallowed in INLIFE and all removals should be accomplished using BMSS functions. In order to provide compatibility with the INLIFE Death Check Function, a new BMSS Death Check Function will be created. Animals in a study are normally not re-issued, but ISSUE should be explicitly disallowed in EB. If required, this function should only be performed using the REALLOCATION function.

The BMSS Death Check function will be modeled closely after the corresponding INLIFE function, but will be based on the procedural rules for animal housing listed in the previous section. In the Inlife Death Check function, the technician barcodes any cage card from a rack. Death Check verifies the rack upon which the cage is located and writes a death check transaction. If the technician finds a dead animal on that rack, the corresponding cage card is barcoded; if no dead animals are found, the technician moves

to the next rack. If the current cage is located on the same rack as the previous cage, death check calls INLF_REMOVE_ANIMAL. When control is returned to death check, the technician may barcode another cage or exit. This translates to pseudocode as:

```
Process DEATH_CHECK::
SET rack number to non-existent
LOOP    ! while BARCODING, until SELECT-EXIT
    BARCODE   anycage  or   SELECT-EXIT
    DETERMINE rack number
    if CHANGE-IN     rack number
        WRITE  death check transaction
    if NO-CHANGE   in rack number
        CALL          INLF_REMOVE_ANIMAL
```

An example is supplied to illustrate the potential difficulties with using the existing INLIFE death check function in the case where breeding must occur.

Example:

Female 1A-1 is in cage A of rack AA.
Male 1A-2 is in cage B of rack BB.

It is desired to set up the pair as breeders. To do this, 1A-2 is moved from B to A, using BMSS functions and the status of both animals is changed to BREEDER. The actual movement of the animal is therefore tracked in the BMSS data base.

Meanwhile, no entries are made in the corresponding INLIFE files, because INLIFE does not support animal movement. As far as INLIFE is concerned, the animals are still housed in separate cages (A and B). At the same time that the Sire is moved into the Dam's cage, the Sire's cage card is moved to the Dam's cage.

This introduces possible difficulties in the INLIFE sequential functions, specifically, the death check function. Based on the way that the death check function works (above), it is possible that the death check of rack BB could be conducted, and it is also possible that the Sire has, indeed, died. This might not be noted during the death check of rack BB because the animal now resides physically on rack AA.

Similarly, the death check on rack AA could have problems, because the barcoding of the Sire's cage card while processing rack AA would be interpreted as the barcoding of the first cage on a separate rack (namely, BB), and would therefore not have the desired result of calling INLF_REMOVE_ANIMAL.

The case of missing the dead animal during the check of rack BB does not appear to be a particularly large problem, because no other rack BB operations or data are changed. However, the situation during the death check of rack AA could lead to incorrect data entry and would almost certainly lead to confusion.

NCTR has agreed that the INLIFE death check function will be disabled in EB studies and that a new BMSS death check function will be created. This function will take into account the fact that the location of breeder pair animals is accurately depicted in BMSS and should therefore "automatically" circumvent the problem as just described. It must, however, be kept in mind that the use of the BMSS death check function must still generate INLIFE death check data and remove the dead animal from both systems.

In terms of pseudocode the BMSS death check function will perform logically as follows:

```
Process BMSS_DEATH_CHECK::
SET rack number to non-existent
LOOP    ! while BARCODING, until SELECT_EXIT
    BARCODE    any cage or    SELECT_EXIT
    if EMPTY    according to BMSS, ignore: cage is SIRE cage but he may be on the wrong
                rack...because of EB procedures: loop back to BARCODE with a message telling
                to barcode another cage
    DETERMINE rack number
    if CHANGE_IN    rack number, then
        WRITE  death check transaction
        GOTO        BARCODE
    if PAIR_HERE    (according to BMSS) then
        DETERMINE_SIRES_INLF_CAGE (you don't have to determine the DAM's, it is the cage
        just barcoded)
        PROMPT      to determine if SIRE, DAM, or BOTH are to be removed (checked dead)
        if BOTH then set SIRE and DAM
        if DAM  CALL    INLF_BMSS_REMOVE_ANIMAL with DAM info
        if SIRE  CALL INLF_BMSS_REMOVE_ANIMAL w.  SIRE info


Process INLF_BMSS_REMOVE_ANIMAL
    ! note that all EB removals will probably have to be funneled through this routine
    CALL    INLF_REMOVE_ANIMAL (which functions as before, "removing" the animal from the
            appropriate INLIFE files)
    CALL    BMSS_REMOVE_ANIMAL (which "removes" the animal from the appropriate BMSS files
            and is called from various BMSS screens directly)
```

Note that the identification of the animal to be removed is actually done in INLF_REMOVE_ANIMAL, but that the cage number has to be passed to INLF_REMOVE_ANIMAL. In INLIFE, there will only be one animal with a UAI in any cage (litter animals do not have UAI's). The BMSS will keep the animal identification identical to that used in INLIFE.

3.0. BMSS STRUCTURE

3.1. TERMINAL SCREEN LAYOUTS

3.1.1. Overview

BMSS terminal screens for both data entry and display are described in the following sections. The screens and accompanying descriptions presented in the Final Draft of the External Design (19 August 1988) have been revised considerably to increase the detail and reflect the changes requested in the early stages of the internal design. The revised screens are in Appendix A.1. The INLIFE screens will not be modified in this task and are not listed. The EB screens are a combination of the BREEDING and INLIFE screens. Screen A.21 has been created specifically for EB to allow for weighing of pups, individually or all together, during an experiment. The discussion of the EB screens will refer to material in the other sections, rather than duplicating it.

Help messages for each screen in the breeding portion of the BMSS have been prepared. These messages are presented in Appendix A.2.

3.1.2. Screen Standards

The data input screens and display screens for BMSS are shown in Figs. A.1 through A.21 (Appendix A). These screens conform to the NCTR screen standards which were used during the development of the INLIFE screens. Because the MULTIGENERATION component of BMSS will use both BREEDING and INLIFE screens, the two sets of screens must have compatible formats. A sample of the standard as it applies to the BREEDING screens is shown in the Animal Assignment screen in Fig. A.14, Appendix A. The screen has three sections, divided by horizontal lines. The top section contains header information, including the screen name on the upper left and the date and time on the upper right. The bottom section contains information about active function keys. Data entry occurs in the center section. Note that there will be a help message for each data entry field and a help screen for each data entry field.

3.2. REPORT LAYOUTS

3.2.1. Report Revisions.

Twenty-five BMSS reports have now been specified. Twenty-one reports were submitted with the Breeding Multigeneration Support System (BMSS) External Design (19 August 1988). During the initial stages of the internal design, selected reports were revised and additional reports requested. These reports with editorial corrections are provided in Appendix B. Four new reports are also included. BS-002 (Daily Recap Report-Observations) has been separated into two reports. One (BS-002o) provides separate daily recaps of the observations of pregnant, plugged, litter, and newborn animals. The second (BS-002p) provides a separate report for phenotype observations. Another of the new reports provides the ability of users to request reports (BS-019). The fourth report is a data dump report referred to as "BISLOOK" in the past. This is now report BS-020 and will be called "BMSSLOOK". Part B.1. provides the written description of the reports while the physical layouts are in Part B.2. The BMSS reports are cross-referenced to the Breeding Information System Reports in Table 1.1 in Section 3.2.1.

3.2.2. Report Characteristics

The frequency of reporting and estimated volume of each report are provided in Table 1.2 of Section 3.2.1. Only three of the reports are run daily. Most reports are run weekly and/or by operator request. The estimated report volumes are moderate in size. Table 1.3 of Section 3.2.1 shows the report distribution characteristics.

3.3. LOGS AND FORM LAYOUTS

The development and maintenance of activity logs are very important functions within the present BREEDING system and are manual procedures used for recording new litters. The procedure consists of two log sheets that are filled in according to well-defined standard operating procedures (SOP). Log sheet 1 is used for recording the new litter (e.g., to list assigned UIN's, etc.). Log sheet 2 is a carbon copy of 1 (i.e., duplicate information on a copy) and is used at the time the litters are weaned to record sex, animals weaned or dead, the disposition, and other information. The SOP for each log sheet and an example of

Table 3.1. Frequency of generation and estimated volumes for the Breeding Multigeneration Support System reports. (Abbreviations used in volume estimates are pp=pages and kb=kilobytes).

BMSS REPORTS	REPORT FREQUENCY				VOLUME
	daily	weekly	monthly	ad hoc	VOLUME
1. BS-001. Daily Recap Report-Assignments	X				2pp/10kb
2. BS-002o. Daily Recap Report-Pregnant, Plugged, Litter, and Newborn Observations	X				2pp/10kb
3. BS-002p. Daily Recap Report-Phenotype Observations	X				2pp/10kb
4. BS-003. Daily Recap Report-Removals	X				1pp/5kb
5. BS-004. Female and Male Breeder Card	X			X	10pp/50kb
6. BS-005. Animal Birth Report		X			8pp/40kb
7. BS-006. Weaned Animal Report		X		X	9pp/45kb
8. BS-007. Breeder Assignment		X			12pp/60kb
9. BS-008. Active Breeder Report			X	X	4pp/20kb
10.BS-009. Average Litter Size		X			6pp/30kb
11.BS-010. Unproductive Breeder Report		X			8pp/40kb
12.BS-011. Retirement Candidates		X			6pp/30kb
13.BS-012. Daily Allocation	X				10pp/50kb
14.BS-013. Female Breeder Status		X		X	10pp/50kb
15.BS-014. Productivity Index Report					10pp/50kb
16.BS-015. Feeder Status Report		X		X	10pp/50kb
17.BS-016. Issue Cage Census Report				X	10pp/50kb
18.BS-017. Animal Orders		X	X	X	10pp/50kb
19.BS-018. Target Trace				X	10pp/50kb
20.BS-019. Ad Hoc Reports				X	10pp/50kb
21.BS-020. BMSSLOOK				X	10pp/50kb
22.EBS-001. Pup Weights at Weaning and Disposition				X	10pp/50kb
23.EBS-002. Pup Weights				X	10pp/50kb
24.EBS-003. Fertility				X	10pp/50kb

Table 3.2. Breeding Multigeneration Support System (BMSS) reports and their equivalent numbers and names in the preceding Breeding Information System (BIS).

BMSS REPORTS	BIS REPORT
1. BS-001. Daily Recap Report-Assignments	BS-001. Daily Recap Report-Assignments
2. BS-002o. Daily Recap Report-Pregnant, Plugged, Litter, and Newborn Observations	BS-002. Daily Recap Report-Observations
3. BS-002p. Daily Recap Report-Phenotype Observations	BS-002. Daily Recap Report-Observations
4. BS-003. Daily Recap Report-Removals	BS-003. Daily Recap Report-Removals
5. BS-004. Female Breeder and Male Breeder Cards	BS-004. Female Breeder and Male Breeder Cards
6. BS-005. Animal Birth Report	BS-005. Animal Birth Report
7. BS-006. Weaned Animal Report	BS-006. Weaned Animal Report
8. BS-007. Breeder Assignment Report	BS-007. Breeder Assignment Report
9. BS-008. Active Breeder Report	BS-008. Active Breeder Report
10. BS-009. Average Litter Size	BS-009. Average Litter Size
11. BS-010. Unproductive Breeder Report	BS-010. Unproductive Breeder Report
12. BS-011. Retirement Candidates	BS-011. Retirement Candidates
13. BS-012. Daily Allocation	BS-012. Daily Allocation
14. BS-013. Female Breeder Status	Combination of BS-013 (Daily Recap Report-Allocations), BS-018 (Female Breeder History), and BS-021 (Allocation Weights)
15. BS-014. Productivity Index Report	BS-023. Productivity Index Report
16. BS-015. Feeder Status Report	BS-015. Feeder Status Report
17. BS-016. Issue Cage Census Report	NEW REPORT
18. BS-017. Animal Orders	NEW REPORT
19. BS-018. Target Trace	Target Trace (Special Ad hoc report, no BS designation)
20. BS-019. Ad Hoc Reports	NEW REPORT
21. BS-020. BMSSLOOK	BISLOOK
22. EBS-001. Pup Weights at Weaning and Disposition	NEW REPORT FOR EXPERIMENTAL BREEDING
23. EBS-002. Pup Weights	NEW REPORT FOR EXPERIMENTAL BREEDING
24. EBS-003. Fertility	NEW REPORT FOR EXPERIMENTAL BREEDING

Table 3.3. Distribution characteristics of BMSS reports.

BMSS REPORTS	DISTRIBUTED TO:	# COPIES	SCREEN	HARDCOPY	DISKFILE
1. BS-001. Daily Recap Report-Assignments	Printer XX, Bldg-Room XXX-XXXX	10		X	X
2. BS-002o. Daily Recap Report-Pregnant, Plugged, Litter, and Newborn Observations	Printer XX, Bldg-Room XXX-XXXX	5		X	X
3. BS-002p. Daily Recap Report- Phenotype Observations	Printer XX, Bldg-Room XXX-XXXX	10		X	X
4. BS-003. Daily Recap Report-Removals	Printer XX, Bldg- Room XXX-XXXX	10			
5. BS-004. Female Breeder Card and Male Breeder Card	All Cages	one/cage		X	X
6. BS-005. Animal Birth Report	Printer XX, Bldg-Room XXX-XXXX	10		X	X
7. BS-006. Weaned Animal Report	Printer XX, Bldg-Room XXX-XXXX	15		X	X
8. BS-007. Breeder Assignment Report	Printer XX, Bldg-Room XXX-XXXX	15		X	X
9. BS-008. Active Breeder Report	Printer XX, Bldg-Room XXX-XXXX	10		X	X
10.BS-009. Average Litter Size	Printer XX, Bldg-Room XXX-XXXX	15		X	X
11. BS-010. Unproductive Breeder Report	Printer XX, Bldg-Room XXX-XXXX	15		X	X
12. BS-011. Retirement Candidates	Printer XX, Bldg-Room XXX-XXXX	10		X	X
13. BS-012. Daily Allocation	Printer XX, Bldg-Room XXX-XXXX	10		X	X
14. BS-013. Female Breeder Status	Printer XX, Bldg-Room XX-XXX	10		X	X
15. BS-014. Productivity Index Report	Printer XX, Bldg-Room XX-XXX	15		X	X
16. BS-015. Feeder Status Report	Printer XX, Bldg-Room XXX-XXXX	10		X	X
17. BS-016. Issue Cage Census Report	Printer XX, Bldg-Room XXX-XXXX	15	X	X	X
18. BS-017. Animal Orders	Printer XX, Bldg-Room XXX-XXXX	10		X	X
19. BS-018. Target Trace	Printer XX, Bldg-Room XXX-XXXX	10		X	X
20. BS-019. Ad Hoc Reports	Printer XX, Bldg-Room XXX-XXXX	10		X	X
21. BS-020. BMSSLOOK	Printer XX, Bldg-Room XXX-XXXX	10		X	X
22. EBS-021. Pup Weights at Weaning and disposition	Printer XX, Bldg-Room XXX-XXXX	10		X	X
23. EBS-002. Pup Weights	Printer XX, Bldg-Room XXX-XXXX	10			
24. EBS-003. Fertility					

each sheet and the layout is shown in Part B.4 of Appendix B of the External Design. A requirement of the new system will be the capability to enter, store, retrieve, and print these data as a daily recap report. No revisions were required for these logs.

Forms for BREEDING that exist in the present BIS system are the Male Breeder card and the Female Breeder card. These are preprinted cards that are prepared for each breeder pair listed on the Breeder Assignment Report at the time the pair is established as breeders. Each set of cards is attached to a breeder cage and is used to log the birth and weaning of litters produced by that breeder pair and serves as quick visual check for that cage. New litters and their subsequent weaning are recorded on these preprinted cards and the data terminal. These cards are described in detail along with their layout in Appendix B (Parts B.1.1, B.2.1, B.3.1) of the External Design. No revisions were requested for these forms.

3.4. DATA BASE STRUCTURE/DESIGN

The data elements used in the formulation of the data base structure are being maintained in an on-line data element dictionary (DED). The DED was included in the External Design Document. Because of its size, the DED is not included in this document. Instead, the DEC will be actively upgraded as data element information changes and published at a later date under separate cover. The data base required for storage of the BMSS data requires four separate components. Both BREEDING and MULTI-GENERATION STUDIES require on-line and Historical data. The structure of the data for MULTIGENERATION STUDIES will be obtained as the union of INLIFE and BREEDING files. In addition, BMSS will require access to certain data maintained by Study Definition. These data are required to implement various experiment-specific allocation schemes and will be provided by Study Definition.

The BREEDING system will require seven new primary files, although one of these files is needed only for on-line processing. The files required are: ANIMAL, DATE-ESTABLISHED, LITTERS, ALLOCATION, ORDERS, BMSS-REPORT-PARAMETERS, and CAGE (required only for on-line processing). The seven primary files are used by TDMS during daily operator usage and are maintained as RMS files. The record definitions for these files are maintained as Common Data Dictionary objects containing Request Definition Utility (RDU) record specifications.

Separate copies of these files exist for each EB experiment as well as for the breeding colony. An intermediate file (used to collect all daily transactions, regardless of experiment) also exists; its structures are identical to that of the experiment and colony-specific files.

The six files containing Historical data are also maintained as ADABAS files. The structure of the ADABAS files has been designed to be as similar as possible to the TDMS record definitions.

The ADABAS and TDMS record structure layouts are contained in Appendix C.

The ANIMAL file contains information related to an individual animal as identified by its UIN. Information regarding one-time events (birth date, for example) or permanent characteristics of the animal is stored in this file. In addition, information pertaining to the animal's history and movement within the breeding colony is stored here. Note that INLIFE has no ability to move an animal once the animal is established in an experiment. This capability is being discussed by NCTR staff and will be jointly developed between NCTR and ORNL.

The DATE ESTABLISHED file contains information pertaining to breeder pairs. In addition to the date the pair was established, the dissolve date and any pair observations are maintained in this file.

The LITTER file contains information identifying the litter and its parents, along with selected wean and weight information that pertains to the entire litter or individuals of that litter.

The ALLOCATION file is used to store information recording the experiment and location to which an animal was allocated.

The ORDERS file is used to store animal orders (requests for animal delivery to an experiment).

The CAGE file is used to allow rapid display of the information related to a single cage and to identify the individual animals or the litter in a cage.

The BMSS-REPORT-PARAMETERS file contains information pertaining to scheduled and operator-selected reports.

For each file, the name, prime key, projected growth rate, and access method are indicated in Appendix C. Alternate keys are indicated as descriptors in the ADABAS file description.

The sizes of the files required for storage of the Census data should remain relatively constant and can be based on the maximum number of animals to be maintained in the BREEDING colony at one time.

In addition to the new files listed, the BMSS will also use the INLIFE session file to store session information.

3.5. TABLE DESIGN

The BMSS contains a number of fields for which there is a list of allowed values. Allowed value lists suggests the use of tables to store valid entries, and to store expanded data values for those fields in which coded values are usually entered. However, a number of the tables required are already stored in other NCTR subsystems.

Some of the proposed tabular values exist in the NCTR General Tables, which are maintained in both ADABAS and RMS form. These tables should be accessible to BMSS with a minimum of effort. Other tables may require the creation of new entries in the General Tables, or, if the format differs too much from the General Table format, may require the creation of a separate BREEDING-Validation Table.

With the exception of help and error messages, the tables tend to be rather small. ORNL recommends that the various tabular information required by BMSS should be read into memory at session initialization or used directly in the TDMS field level checks where the values are not expected to change frequently. Memory initialization of the TDMS field level checks will allow direct references to be made to tabular information, rather than requiring individual accesses to the tables (on disk) during screen operations. In TDMS, field level checks can be accomplished by creating a TDMS record structure to contain the various items in a high level routine designated to read the tabular information. In NATURAL, global variables would be required.

The list of table oriented items follows:

BLDG-ROOM	-	Building room values, available from TST-RACK-DATA for individual tests
TERMINAL, BARCODE, KEYBOARD, BALANCE	-	Based on available NCTR feedback, it is suggested that these values not actually be tabled, merely verified as containing only numeric digits
OPERATOR	-	Operator barcode ID's, already in GENERAL TABLES
CAGE	-	Can be checked with ranges appropriate to BREEDING or INLIFE; in BREEDING, 1-1200 are breeder cage numbers, while 3000 and greater are issue cages; in INLIFE, cage numbers can be any value from 1 to 9999
EXPERIMENT	-	Experiment numbers; these are expected to be supplied via an INLIFE record created by Study Definition, and must include the PI-Experiment link.
SEX	-	Several different tables exist, each depending on a specific screen operation. The comprehensive list includes B(oth), F(emale), L(itter), M(ale), and U(nsexed). It may be appropriate to have one table to convert the 1-character coded values to the longer value, but to make validation specific to the function called. The formatted values of these codes are unlikely to change with time.
STRAIN	-	The Strain-Sex-Group from Study Definition should be able to be modified to contain all required strain information. The values required by BMSS are the strain code, the retirement age criteria, the breeding inactivity criteria.
PHENOTYPE	-	Phenotype values apply only to certain strains and then indicate a number of different coded values corresponding to an animal phenotype (like "haired" or "agouti"). These data should probably become a component of the STRAIN information.
STATUS	-	Issue-Feeder, Issue-Experiment, Issue-Other, Feeder, Breeder, Surplused, Culled, Dead, Retired, Removed
CAGE STATUS	-	Breeder, Feeder, Issue
DISPOSITION, DESTINATION	-	Pathology, Chemistry, Microbiology
OBSERVATION	-	PR(egnant), PL(ugged), LI(tter)
ROW/COLUMN	-	Already exists in TST-RACK-DATA.
CLIP	-	N(one), L(eft), R(ight), B(oth)
ROOM/COLONY	-	R(oom) or C(olony), can be hardcoded in the screen in which room/colony is used, because these are the only two values ever expected.

MALE PRESENT - P(resent) or A(bsent), are derived values based on cage Census data; these could be hardcoded because the formatted values are not expected to change.

FUNCTION - This is a list of the character sequences used by the operator to request an Animal Room Function. These values are used for validation and selection, but are not used for translation.

O(ff) - Log Off BREEDING
A(llocate)- Perform an allocation sequence
N(ew) - New Animal Entry
S(tatus) - Assign Status
F(inal) - Final Actions (Removals)
R(etire) - Retire Breeders
P(regnant) - Note Pregnant or Plugged
L(itter) - Note Littered
I(ssue) - Issue Animal
C(hange) - Change Cage Status
M(ove) - Move Cage(s)
D(isplay) - Display, causes an additional prompt for subfunction:

C(age)
R(etirement Candidates)
U(nproductive Breeders)
F(eeders by Minimum Age)
I(ssue Feeders by Minimum Age)
W(ork Room Summary)

CONTROL FUNCTION - the standard NCTR set of Control and Gold sequences will be augmented with the BREEDING-specific control sequences:

Cntl-W - Sequential Wean
Cntl-A - All Cages
Cntl-L - Littered Cages
Gold-T - Display Table Values
Gold-N - Erase screen values and restart
PF2 - Help

HELP and ERROR - Coded help and error messages will be used throughout BREEDING. The text values associated with the coded values will be stored in a separate ADABAS file, if NATURAL is used, or in a TDMS record structure, if TDMS is used.

3.6. BMSS MODULE SPECIFICATIONS

Each module specification indicates the module name, descriptive title, method of invocation, inputs, outputs, called modules and description. Processing is indicated in some detail using structured English. Together with the data base file specifications and knowledge of the language used for implementation, these descriptions should provide a significant start toward actual implementation of the desired functions.

Descriptions of the BMSS screen and report modules are in Appendix D and E, respectively. Items discussed for each of the modules are listed below.

3.6.1. BREEDING Screen Module Specifications

An explanation of the processing performed in each report and screen shown in Appendices A and B follows. The physical structure shown or referenced in Section 3.4 and in Appendix C is used for reports and screens. The batch reports utilize the Historical data stored in the ADABAS files, while the screens and on-line reports access the Census data. To increase efficiency and reduce system response time, the Cage file contains the current status and contents of each cage. This file is used to quickly determine the UINs of the individuals in each cage and the cage status. There are no Historical data stored for this file.

The module descriptions discussed in this section show the data base queries and data validation required to produce the output. Certain processing information that is common to all modules is not shown in the Appendix, but is discussed here. Any assumptions that pertain to specific modules precede the processing outline of each module.

The following assumptions were made concerning all modules:

1. There will be a table of numbered error messages maintained on a TDMS record structure.
2. Study Definition data will be accessed where required to validate data entered.
3. HDEC corrections to existing Census data are reflected into the Census RMS files no earlier than the evening that they were corrected with HDEC. This process is described in Section 2.1.3.A.
4. The exact key sequences used to invoke functions are not shown in the module descriptions.

5. The formatting of output to the screen has been omitted but will follow the report and screen formats shown in Appendices A and B.

6. The precise error messages displayed for invalid data entered are shown in the screen formats Appendix A.

7. Data entered by the user are immediately displayed in the proper location on the screen.

8. The entry of the Operator identification is performed before the logon process using established NARSS procedures.

9. All dates entered by the user will be checked for accuracy.

10. In each function, data are retained on the screen to allow the user to scroll through the data to view and correct errors.

11. Because the module names are unknown at this time, the descriptive title is used.

12. The CIDs entered by the user will be checked for uniqueness. NCTR has agreed that daily checking of CIDs is sufficient, so CIDs will be checked against all CIDs existing in Census.

13. The sex of littered animals may be entered in any screen in which UINs are assigned. The animal records are stored when UINs are first assigned to litter animals.

14. During assignments from a block, UINs are assigned for males from the lower end of the block. UINs for females are assigned from the high end of the block. In order to make this operation function correctly, the COUNT field must be decremented whenever a litter animal is weaned, but the BLOCK UIN is changed only when a male is assigned, because the assigned male receives what was formerly the BLOCK UIN. The BLOCK UIN is therefore incremented by one when a male is assigned.

The assignment of UINs is accomplished by allocating blocks of contiguous UINs at the time that a litter is observed based on the highest UIN previously assigned in the strain under consideration. For example, if the last UIN previously assigned was number 1A-0000100 and a litter is observed in strain 1A having 10 pups, then the block of UINs from 1A-0000101 to 1A-0000110 would be assigned. In the CAGE record this would be reflected by the assignment of a Litter UIN of 1A-0000101 and a count of 10.

The Last UIN Assigned will be maintained in a separate, shared RMS file and incremented each time a block of UINs is assigned. Separate entries will exist for each strain. In the previous example, the Last UIN Assigned would have a value of 1A-0000100 before the litter is observed and 1A-0000110 afterwards. Because multiple operators may attempt to operate on a single strain at the same time, the last UIN record corresponding to the litter strain must be locked while UIN blocks are assigned. The assignment of UINs is not made until the other information on the screen has been verified.

3.6.2. BREEDING Report Module Specifications

Detailed module specifications for BREEDING reports can be found in Appendix E.

Any specific assumptions made in regard to report processing are listed within the detailed processing section. In some cases, these assumptions were made to insure that the reports would function properly, even if they could not be submitted according to the normal report submission schedule.

Access to the elements in ADABAS Periodic Groups was made under the assumption that the oldest element would be stored in occurrence number one. This storage order reduces the overhead during update operations.

Occasional notes are listed within the report specifications. Notes generally refer to a specific limitation of the code that was noted during review of the specification after the specification was thought to be complete, or suggest that an alternate solution deserves further consideration before implementation.

All reports listed here are assumed to operate on the BREEDING Historical data after they have been updated with the daily transactions from the modifications made to Census data during daily on-line processing.

Many of the reports require reporting the current cage, and some reports require sorting by this value. However, the definition of current cage may vary, depending on the exact nature of the report and the run date, which may be different from the desired run date, for example, when there is extended system downtime before the daily report runs. For the purposes of these reports, the current cage is assumed to be the cage in which the animal resided at the end of a particular date. This date is either the report run date or the end date of the selected date range. The code described will thus allow a report to be run listing the data as of some period in the past.

The fact that the current cage does not correspond to a fixed PE location precludes its direct use in NATURAL FIND and SORTED BY constructions. Many of the reports are therefore produced by a sequence of related programs, usually one program to perform data extraction, one to sort the data in the desired order, and a third to provide the report output and summary. The extracted data are written to

ADABAS work files, and data sorting is currently assumed to be performed externally to ADABAS and NATURAL using the VMS SORT command.

Report output may be stored for on-line viewing using the specifications indicated in Section 2.1.3.D.

4.0. SYSTEM TEST PLAN FOR THE BMSS

4.1. INTRODUCTION

4.1.1. Objectives

The objective of this test plan is to define the general system testing procedures for the NCTR BMSS computer processing system. This includes establishing the testing approach, the testing environment, the staff responsibilities and tasks, and the items to be tested.

4.1.2. Scope

Three major types of testing will be performed on the BMSS computer processing system - unit testing, integration testing, and system testing. Unit testing is the explicit testing of functions executed at the module level, and integration testing is the explicit testing of interfaces between modules. System testing is the verification of functions viewed from a system level. Interfaces between modules and module level functions are implicitly, not explicitly, tested in system testing. Of the three types of testing, only system testing will have a formalized structure with externally required documentation. Integration testing and unit testing will both be semi-formal tests with only internally required documentation. This document is the test plan for system testing of the BMSS computer processing system.

4.1.3. Applicable Documents

The following documents were used as sources of information for this test plan and are referenced within this test plan:

1. Research Proposal for Development of BREEDING/MULTIGENERATION Support System, submitted to National Center for Toxicological Research, Jefferson, Arkansas.

2. Toxicology Data Management Systems: Systems Development Guidelines, Standard Operating Procedures, TDMS Document 4505-SDG-2.0

3. Code of Federal Regulations - Good Laboratory Standards Practice, Subpart A--General Provisions, Chapter 1, Part 160, July 7, 1987

4. Standard Operating Procedures for the BREEDING Information System.

5. BREEDING MULTIGENERATION SUPPORT SYSTEM: REQUIREMENTS ANALYSIS, 04-FEBRUARY-1988, ORNL (DRAFT). 93 pp.

6. Requirements Analysis Report For The NCTR Automated Research Support System. Prepared by System Development Corporation, June 1984. 30 pp.

7. BREEDING MULTIGENERATION SUPPORT SYSTEM: EXTERNAL DESIGN 19-AUGUST-1988, ORNL (DRAFT). 481 pp (approx.).

8. BREEDING MULTIGENERATION SUPPORT SYSTEM: INTERNAL DESIGN 15-FEBRUARY-1989, ORNL (DRAFT). 500 pp (approx.).

4.2. TEST ITEMS

The BMSS computer processing system, as defined in user requirements, external design, and internal design documents, listed above (Section 1.3), will be tested.

4.3. FEATURES TO BE TESTED

The features to be tested are the system functions initially specified in the User Requirements document and refined in the External and Internal Design documents (See items 5, 7, and 8, respectively, in Section 1.3.)

Table 4.1. contains all the functions in the current BIS and INLIFE as defined in the Requirements Analysis and External Design Documents. Also included in this table are functions developed for use in Experimental Breeding that were not included in BIS or INLIFE. New functions identified for BMSS after the publication of the Requirements Analysis include:

- "Tracking Animal Orders" (Report BS017, Appendix B of this document),
- "Enter Allocation Data" (Screen A.3),
- "Weigh Animals--Group", and
- "BMSS Identification Function" (Screen A.1).

Table 4.1. Functions Required for Breeding and Experimental Breeding.

	SCREEN(S)	BREEDING	EXPER. BREEDING
CURRENT BIS FUNCTIONS			
Assign breeder as feeder	A.14	X	X
Assign feeder as breeder	A.14	X	X
Breed animals		X	X
Change animal status (Assign litter animal status)	A.14	X	X
Change cage status without moving cage	A.11	X	X
Determine reason for transfer out	A.15	X	X
Discard animal (Removal)	A.15,16	X	
Display data for animal technicians	A.6,7,0,9,13	X	X
Enter Demonstration Mode		X	
Enter Monitor Mode		X	X
Establish breeders	A.14	X	X
Identify animal destination		X	
Introduce a new strain	A.10	X	
Remove animal for analysis	A.15	X	X
Maintain feeders		X	X
Maintain the Breeding Colony		X	
Move a cage within a room or the colony	A.12	X	X
Female littered	A.18	X	X
Note pregnancy	A.17	X	X
Observe animals	A.17, 18	X	X
On-line correction		X	X
Process cages		X	X
Put animal in issue cage	A.19	X	
Transfer animals out of breeding colony	A.4	X	
Remove dead animal	A.15	X	X
Sign off the system		X	X
Sign on to the system	A.1	X	X
Transfer to a new breeder cage		X	X
Verify mark	A.4	X	X
Wean animals	A.14	X	X
Weigh animals--individual	A.4	X	X

Table 4.1. (Continued)

	SCREENS	BREEDING	EXPER. BREEDING
PROPOSED BMSS FUNCTIONS			
Allocate animal to an experiment	A.4	X	
Manual entry of data		X	X
Print log book		X	X
Error Correction		X	X
Weigh animals--groups			X
BMSS identification	A.1		X
Enter allocation data	A.3		X
Tracking animal orders	A.5	X	
CURRENT INLIFE FUNCTIONS			
Logon		X	X
Main Menu, Identification Information		X	X
Interactive Data Collection		X	X
Procedure Definition Screen			
Schedule Definition Screen			X
Balance Calibration Screen		X	X
Identification Correction Screen		X	X
Cage Menu Screen			
Process Container Weights		X	X
Cage Observations and Notes			X
Select Animal Process			X
Selection of unscheduled actions			X
Re-Identify Animals on Cage Screen			X
Animal Weight Process			X
Clinical Observation Process			X
Clinical Observation Table Screen			X
Site Table Screen			X
Size/Color Table Screen			X
Observation Summary Screen			X
Feed Check Screen			X
Death Check Screen			X
Cage Relocation Screen			X
Remove Animal Screen			X
Reason for Removal Screen			X
Selection Dispositions Screen			X

4.4. FEATURES NOT TO BE TESTED

No features will be initially tested using live data. The data to be used will be generated or simulated data. Informal testing of features requiring live data or real time communications support may be done as unit testing proceeds to the integration level. Informal testing implies that the test cases, test procedures, and test results will not be documented and will not be considered part of the formal system testing.

4.5. APPROACH

4.5.1. Method

The method that will be used to do system level testing of the BMSS is Build Testing and System Verification Testing. Build Testing is a modular approach to testing in which a system is tested in stages called "Builds." A subset of the BMSS is tested during each Build, and the functionality of the system increases with each successive Build. Features of BMSS tested in one Build may be used as testing tools in subsequent Builds. The major emphasis during Build Testing is to test the basic functioning of the BMSS as the system is being developed.

After all Build Testing is completed, two System Verification Tests (SVTs) will be run. The first SVT will be run at NCTR on the Test Bed. This will be SVT 1. The second SVT will be run at the operational site on the Target System. This will be SVT II. The SVTs will test the complete BMSS computer processing system. Because of possible differences between the Test Bed system and the Target System, SVT I will be a subset of SVT II.

The BMSS will be tested in two builds. System functions that will be most useful as testing tools, foundational system functions, and functions that have simple modular interfaces will be tested in the first Build. The more detailed functions such as displays and functions with more complex interfaces will be tested in the later Build. Each Build Test Specification will specify what system functions have been tested in that Build.

The Test Team will use any applicable system documentation to plan the test cases and test procedures for each Build and for both SVTs. Using the External/Internal documentation will help insure that all the documented features of the BMSS that should be tested are tested.

4.5.2. Testing Categories

The following eight categories of BMSS level testing will be used for system testing:

1. Functional Testing - testing the basic specified functions of the BMSS.

2. Volume Testing - testing the operation of the BMSS when the BMSS is subjected to heavy volumes of data.

3. Stress Testing - testing the operation of the BMSS when the BMSS is subjected to heavy volumes of data in a short span of time.

4. Security Testing - testing the security functions of the BMSS.

5. Performance Testing - testing the performance attributes of the BMSS.

6. Configuration Testing - testing the operation of the BMSS using the defined system hardware configurations.

7. Installation Testing - testing the installation procedures of the BMSS.

8. Recovery Testing - testing the recovery functions of the BMSS.

If one of the preceding categories does not have any BMSS requirements defined for it, then that category will not be used in System Testing.

Interface Testing is not listed as one of the categories because Interface Testing between modules will be explicitly tested during Integration Testing. The interface between modules will be implicitly tested

during System Testing when a test case is run to test a function that involves the interaction of two or more modules.

Since the emphasis of Build Testing is to test the basic functioning of the BMSS as the system is being developed and to accumulate a tool base for later Builds, Functionality Testing will be the predominant type of testing run during the Build Tests. The other types of testing, which usually require the whole system to be completed for accurate results, will be run during the System Verification tests.

4.5.3. Functionality Testing

Screen functionality and report generation tests will be performed at the beginning of each Build after Build 1. These tests will compare data entry screens and reports to those designated in the External Design and refined in the Internal Design. The test will be done by running the test cases from the previous Builds on the software for the current Build. Because of time constraints, the Build Test Leader may specify certain test cases from the previous Builds that do not have to be run during subsequent testing. The Build Test Leader determines omitted test cases which will cause minimal impact on the integrity of the BMSS. Furthermore, any test cases from the previous Builds that are superceded by software for the new Build will not be rerun.

All of the screen and report functions tested in the Builds will be tested in both SVT I and SVT II. These functions may be tested by rerunning the test cases used in the Builds or new test cases may be written.

Similar testing will also be done within a Build or within an SVT after a software fix has been integrated back into the Build Test software. The test case that uncovered the incident and any test cases that depend on the section(s) of software that were modified will be rerun.

4.5.4. Incident Tracking

An incident is some event that happens during testing that is inconsistent with what is expected. There are many reasons why an incident occurs such as incorrect documentation, software failures (bugs), or hardware problems. When an incident is noted by a tester, the tester will

- isolate the incident,

- determine the Responsible Programmer (RP) that needs to be notified of the incident,

- enter the incident information into a Build Testing data base where a unique incident identifier will be assigned to it, and

- fill out a form to advise the RP about the incident.

When the incident has been fixed,

- the RP will fill out a form to notify the Build Test Leader of the fix (e.g., change in documentation, change in code, change in the hardware),

- if software has been modified, the Build Test Leader will merge the modified software back into the Build Test software, and,

- the Build Testing data base will be updated.

The method in which software changes are made to Build Test software is described in the Software Configuration Management Plan.

4.6. ITEM PASS/FAIL CRITERIA

There are two major criteria for a Build Test to pass:

1. Unsolved Incidents

   If there are no unsolved incidents, then the Build will pass. If there are minor or improvement incidents still unsolved at the end of the Build, the Build will pass; however, the resolutions of the unsolved incidents must be included in the software tested in the next Build. If there are major or critical unsolved incidents at the end of the Build, the Build will not pass. Finally, if there are intermittent or unreproducible problems, the Build Test Leader will be responsible for deciding if the intermittent or unreproducible problem will need to be solved for the Build to pass.

2. Operation Of Functions

   By the end of the Build, the functions tested in the Build must execute as specified for the current stage of the BMSS for the Build to pass.

   There are two major criteria for a System Verification Test to pass:

   A. Unsolved Incidents

      If there are no unsolved incidents, then the SVT will pass. If there are minor or improvement incidents still unsolved at the end of the SVT, the Build Test Leader, with input from the Project Leader, will review all the unsolved minor and improvement incidents and determine on a case-by-case basis which ones must be solved for the SVT to pass. If there are major unsolved incidents at the end of the SVT, the SVT will not pass.

      Finally, if there are intermittent or unreproducible problems, the Build Test Leader, with input from the Project Leader, will determine the impact each intermittent or unreproducible problem has on the system. The SVT will not pass if there are any intermittent or unreproducible problems that have a major impact on the BMSS.

B. Operation of Functions

By the end of the SVT, the basic operational functions of the BMSS must execute as documented.

The Build Test Leader may make exceptions to the above Pass/Fail criteria; however, any exceptions made and the reason(s) for the exceptions will be documented in the summary for the Build or the SVT. Also, all exceptions to the Pass/Fail criteria will be approved by the Project Leader. An example of a possible exception is a major or critical incident found near the end of a Build. If there is not enough time to thoroughly investigate the incident so that a comprehensive, solid fix can be made before the end of the Build, then the Build may still pass, and the solution of the incident will be expected in the next Build.

4.7. SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

4.7.1. Suspension Criteria

Testing during a Build will be suspended if all test cases of the Build are dependent on incidents that are currently being fixed by RPs or software specialists.

4.7.2. Resumption Requirements

Testing during a Build will be resumed when one or more test cases are no longer dependent on incident fixes.

4.8. TEST DELIVERABLES

The Following documents will be generated for testing:

- BMSS Test Plan - this plan

- Build Test Specifications - one for each Build Test. Each specifies details on the test approach for a Build Test.

- Build Test Summaries - one for each Build Test. Each summarizes the results of a Build Test.

- System Verification Test Specifications - one for each SVT. Each specifies the test approach for the System Verification Tests.

- System Verification Tests Summaries - one for each SVT. Each summarizes the results of the System Verification Tests.

The BMSS Test Plan will be delivered after the Internal Design Review. Each Build Test Specification and Build Test Summary will be delivered at about the time that the Build Test is demonstrated to NCTR. The System Verification Test Specifications and the System Verification Test Summaries will be delivered before the Acceptance Test.

4.9. TESTING TASKS

Table 4.2. shows the major tasks of testing the BMSS. A general ordering of the tasks is also shown in the table. Unit Testing, Integration Testing, Beta Testing, and Acceptance Testing are included in the table to indicate their positions in the overall testing activities.

Table 4.2. Tasks Required to Test the NCTR BMSS

Task No.	Task Description	Predecessor Tasks	Responsibility
1	Identify the Test Bed hardware		ADPE Manager
2	Identify the Target System hardware		ADPE Manager
3	Prepare outline for Build Testing	Prepare System Specification	Build Test Leader and Project Leader
4	Prepare Test Plan	Task 3	Build Test Leader
5	Install the test bed hardware	Task 1	ADPE Manager
6	Develop software modules for Build Test n		all RPs
7	Document software modules for Build Test n		all RPs
8	Design Build Test n and prepare Build Test n Specification	Tasks 4 and 7	Test Team
9	Unit testing for Build Test n software	Task 6	all RPs
10	Integration testing for Build Test n software	Task 9	all RPs
11	Run Build Test n	Tasks 8 and 10, Completion of Previous Build Test	Test Team
12	Prepare Build Test n Summary	Task 11	Build Test Leader
13	Give demonstration of Build Test n software to NCTR	Task 11	Test Team and Project Leader
14	Design SVT I and prepare SVT I Specification	Task 4, Completion of final Build	Test Team
15	Run SVT I	Task 14	Test Team
16	Prepare SVT I Summary	Task 15	Build Test Leader

Table 4.2. (Continued)

Task No.	Task Description	Predecessor Tasks	Responsibility
17	Install the BMSS software at the operational site	Task 17	Build Test Leader, Project Leader, and all RPs
18	Design SVT II and prepare SVT II Specification	Task 15	Test Team
19	Run SVT II	Tasks 18 and 19	Test Team
20	Prepare SVT II Summary	Task 20	Build Test Leader
21	Run Beta-testing at the operational site	Task 20	NCTR
22	Design Acceptance Test and prepare Acceptance Test Specification	Prepare System Specification	NCTR
23	Run Acceptance Testing at the operational site	Task 23	NCTR

4.10. ENVIRONMENTAL NEEDS

4.10.1. Hardware

Existing software to be used for the BMSS fall into the categories of operating systems, communication, data base, and file management.

Two networked computer hardware systems, the VAX 11/785 and VAX 8600, will be used in testing. The two systems are the Test Bed System and the Target System which are located at the operational site. The Test Bed system will be used for all Build Testing and for the SVT I. The Target System will be used for the SVT II and any subsequent testing.

4.10.2. Software

4.10.2.1. Operating Systems

The DEC VMS operating system will be run on all the hardware nodes for the BMSS.

4.10.2.2. Communications Software

No communications software has been specified for accessing the BMSS. All connections to the BMSS will be through dedicated communication lines and the DCA communication multiplex at NCTR.

4.10.2.3. Data Base Software

ADABAS data base management software will be run on the VAX 8600 CPU.

4.10.2.4. File Management Software

TDMS will be used to perform screen data input into RMS files.

4.10.3. Tools

No special diagnostic tools will be used during testing. VMS monitor data will be used to gather performance information during testing.

4.10.4. Security

Security for the software on each processor will include password protection and read, write, and execute protection as provided by the VMS operating system.

4.10.5. Test Environment

Special login IDs to be used for testing will be set up on the Test Bed System and the Target System. These login IDs will be set up to simulate the logins of operators on the final operational system. Set up for these logins will include initialization of login, path specification, and placement of the login IDs in the proper groups. One of these testing logins will have special permissions and will be viewed as the "testing super user." The special permissions for this login ID will include super user privileges and permission to use the PRIV command under VMS.

4.11. RESPONSIBILITIES

4.11.1. Test Specialists

As members of the Test Team, the test specialists are responsible for designing Build and SVT test procedures and test cases, writing testing tools, running Build and SVT test procedures and test cases,

identifying incidents, isolating the incidents, contacting the PLs about the incidents, running the Build and SVT software demonstrations, and writing Build and SVT specifications.

4.11.2. Build Test Leader

The Build Test Leader is responsible for all of System Testing, which includes writing the test plan, supervising and coordinating testing activities, making policy and methodology decisions about testing, maintaining the testing data base, performing responsibilities of a test specialist, and writing the Build and SVT summaries. The Build Test Leader is also responsible for keeping track of incidents, merging software with incident fixes back into the Build Test software, and determining suspension and resumption of the testing.

4.11.3. Project Leader

The Project Leader is responsible for consulting with the Build Test Leader about BMSS requirements, testing requirements, and system schedules.

4.11.4. ADPE Manager

The ADPE Manager is responsible for procuring the hardware for the Test Bed configuration, installing the Test Bed hardware, installing new vendor supplied software on the Test Bed hardware, consulting with the Build Test Leader about hardware problems, maintaining the Test Bed hardware, procuring the Target System hardware for the operational site, and installing the Target System hardware at the operational site.

4.11.5. Responsible Programmers (RPs)

The Responsible Programmers are responsible for developing the software modules for each build test; documenting the software modules for a Build Test; running Unit Tests; running Integration Tests; placing the correct version of the software in the appropriate directories for the current Build Test; helping the Test Team prepare test data, simulators, or data generators for testing; consulting with the Build Test

Leader about possible problems in the software; fixing incidents that have been identified by the team; and assisting the Build Test Leader in merging software with incident fixes back into the Build Test software.

The RPs are also responsible for writing procedures to copy software from the development system to the test net system at the beginning of a Build and writing procedures for installing software at the operational site.

3.11.6 Software Specialists

The software specialists are responsible for performing the project functions assigned to them by their RPs. These functions may include coding and debugging software modules and fixing incidents that have been identified.

3.12 STAFFING NEEDS

At least one person will be assigned to the Test Team. This person will be designated as Build Test Leader and will work on system testing full time. Test specialists will be assigned to the Test Team according to the availability of manpower.

One person will be designated as the BMSS manager, and that person will spend at least one-third time on manager responsibilities.

Each module will have one RP. Software specialists will be assigned to modules according to the availability of manpower and the need and criticality of the functions.

3.13 SCHEDULE

Testing schedules and milestones can be found in the Test Schedule Document which is being developed.

4.14. RISKS AND CONTINGENCIES

If the system design or requirements are changed such that a Build Test has to be redesigned, the amount of time needed to complete the Build Test will increase by as much as the amount of time originally scheduled to design and run the Build.

Time constraints play a critical part in defining the thoroughness of testing. If schedule dates are set making complete testing of all the system functions unrealistic, a determination will have to be made of what will not be tested.

5.0. BMSS DESIGN APPROVALS

Approval of the internal design was provided by NCTR prior to initiating the next phase of the NCTR software development life cycle. This document represents the final submission of the internal design specifications to NCTR.

APPENDIX A

BMSS SCREENS

(Revisions to the External Design Document of 9/18/88)

PART A.1. BMSS Screens

PART A.2. Help Messages for the BMSS Screens

PART A.1. BMSS Screens

The new BIS will be entered through the NARSS main menu. The operator id is retrieved from the Operator Identification Table during the VAX logon. Four items are entered on the BIS identification screen.

1. Building and room - any BIS room may be accessed from any BIS terminal; a building and room card will be available to barcode; DEFAULT = 0

prompt: Enter Building and Room: XXX-XXX
response: building and room numbers
possible error message: Invalid Building and Room - Reenter
response: reenter building and room

2. Terminal ID; DEFAULT = 0

prompt: Enter Terminal ID: 999999
response: bar code or key in the terminal identification number
possible error message: Invalid Terminal ID - Reenter
response: reenter terminal id

3. Barcode ID; DEFAULT = 0

prompt: Enter Barcode ID: 999999
response: bar code or key in the barcode identification number
possible error message: Invalid Barcode ID - Reenter
response: reenter barcode id

4. Keyboard ID; DEFAULT = 0

prompt: Enter Keyboard ID: 999999
response: bar code or key in the keyboard identification number
possible error message: Invalid Keyboard ID - Reenter
response: reenter keyboard id

5. After all data are correctly entered; DEFAULT = blank

prompt: Enter Function or Cage: XXXX
response: bar code or key a cage number or press a function key
If a function key is pressed the prompt 'Enter cage: 9999' will appear and the cage number must be entered before the screen clears.
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

At this point all BIS breeding room functions are available. Note that if a cage number is entered here, the Display Cage screen (Fig. A.13) for that cage will appear next. The Display Cage screen will appear when a cage number is entered following any animal room function that is not cage-specific (such as Display Retirement Candidates).

Figure A.2. Ad hoc Report Selection Screen

```
==================================================================================
BIS Report Selection                                              DD MMM YY HH:MM
Operator 99999

----------------------------------------------------------------------------------
X Animal Birth                 from DD MMM YY to DD MMM YY
X Breeder Assignment           from DD MMM YY to DD MMM YY
X Average Number per Litter    from DD MMM YY to DD MMM YY
X Productivity Index           from DD MMM YY to DD MMM YY
X Transfer/Issue Cage Census
X Animal Orders

X Pup Weights at Weaning and Disposition for Exp# 9999-99
X Pup Weights                            for Exp# 9999-99
X Fertility from DD MMM YY to DD MMM YY  for Exp# 9999-99

X Target Trace for selected or range of (enter S or R): X
   (1) UIN(s) XX-9999999 XX-9999999 XX-9999999 XX-9999999 XX-9999999
   (2) Room(s) XXX-XXXX XXX-XXXX XXX-XXXX XXX-XXXX XXX-XXXX
       and Cage(s) 9999 9999 9999 9999 9999 9999 9999 9999 9999
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen; PF2 - Help; Gold 9 - Finished;

==================================================================================
```

Purpose:
To select any ad hoc report for printing and enter the selection criteria for those reports that require it.

This is a fill-in-the-blank screen. The leading 'X' fields will default to 'N'. If you want a report, put a 'Y' in that space. If the report requires a range of dates, etc, that data must be filled in as well. (Note that if they want the option of (1) printing a report, (2) storing the report in a file, or (3) both, that could be handled by allowing other values in this 'X' field. Of course, then the file name would have to be entered somewhere(?) on the screen or a standard name could be used, generated by concatenating the report number, date and time of request.)

It will be possible to correct data entered on this screen by 'backing up' with the Tab key (assuming TDMS).

This allows for three kinds of target trace: (1) by a set of selected UINs or room/cages each entered individually, (2) by a starting and ending UIN or room/cage (entering exactly two UIN's or two rooms), or (3) selection of an individual UIN or room/cage (which is a special case of (1)). It is assumed that these cases will cover most of what they want to do. There is no apparent reason that they would want to select on UIN and room/cage at the same time. This format doesn't allow for entry of very many UIN's, rooms, or cages. Those could be reformatted as scrolling columns if more were needed.

DEFAULTS: All Y/N fields default to N, all dates to today's date. Experiment number defaults to 0. Target trace selected or range defaults to S. UIN's, rooms, and cages all default to 0.

Figure A.3. Animal Allocation

```
==========================================================================
Alloc    Oper 99999 99999 99999 99999   Issue Cage 9999   DD MMM YY HH:MM
      Males              Females              Block        Original Block
   XX-9999999         XX-9999999           XX-9999999        XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999       Ct=99 Gen=9999    Ct=99 DD MMM YY
--------------------------------------------------------------------------
Exp 9999-99  Room XXX-XXXX  Rack 99  Side X   Sex X  Strain XX   #/Cage 9
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX  9999 XXXXX
XXXXX       XXXXX       XXXXX       XXXXX       XXXXX       XXXXX
XX-9999999 X/X 9999.9 grams - Allocated to cage 9999
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
--------------------------------------------------------------------------
PF2 - Help;
==========================================================================
```

Purpose:
To track animals as they are allocated to experiment cages.

Notes:

If animals are in an EB experiment, they must be removed from that experiment before they can be allocated to a second experiment.

A 10x10 rack will be displayed as four 5x5 racks. Whenever the actual rack size cannot fit on one screen, some sort of scrolling or shifting of the display will be required when accessing the cages on the edges.

'XXXXX' next to cage may be 'NONE', 'LEFT', etc. or a 5-digit animal id number.

The five 'X' fields below the cage number will be filled with 'N', 'L', etc. if a clip is used as a mark or with '1' if a 5-digit ID is used.

Note: When the Allocation function is invoked, if the balance has not previously been calibrated for this session, the INLIFE Balance Calibration screen will appear prior to the Allocation screen.

1. Multiple animal technicians; DEFAULT = 1
prompt: How many animal technicians on this allocation: 9
response: the number of technicians who will participate in this allocation; may be from 1 to 3
possible error message: May not be more that 3
response: reenter number

1a. if more than one; DEFAULT = 0
prompt: Enter operator id: 99999
response: VAX operator identifiers for each technician
The prompt will appear for 1,2, or 3 id's, as is appropriate.
possible error message: invalid operator id
response: reenter id

2. First cage on rack - barcoding the first cage on the first rack will bring up the experiment data. Allocation cannot occur if the experiment data has not been entered (presumably in Study Definition); DEFAULT = 0
prompt: Barcode first cage on rack: 9999
response: bar code or key in the number of the first cage for this allocation
possible error message: invalid cage number
response: reenter cage number

The experiment-test, building, room, rack, side, starting cage number, and sex (may be 'Both'), strain, and number of animals per cage are filled in.

If the rack is full, assume that only animal replacement is to be performed, and go to 8.

Weigh animals for weight-ranking - only if randomization algorithm requires weight-ranking.

2a. Issue cage; DEFAULT = 0
prompt: Enter issue cage: 9999
response: bar code or key in the issue cage number from which animals will be weighed
possible error message: invalid cage number
response: reenter cage number
possible error message: There are no animals in that cage
response: Reenter cage number
possible error message: Wrong strain/sex
response: Pick a different issue cage

2b. Issue cage weight; DEFAULT = 0.0
prompt: Weigh the full issue cage: 9999.9
response: place issue cage on balance and weigh; if weight is not automatically transferred to the screen, it may be keyed in.

possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Enter to accept value or reenter weight

May be keyed in or entered by placing the issue cage on the electronic balance and pressing the "WEIGH" key.

2c. Weigh animals; DEFAULT = 0.0
prompt: Weigh animal: 9999.9
response: Remove the animal from the issue cage and press the "W" key.
possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Enter to accept value or reenter weight

2d. New Issue cage; DEFAULT = 0
prompt: Enter new issue cage: 9999
response: bar code or key in the issue cage number into which this animal will be placed
possible error message: invalid cage number
response: reenter cage number
possible error message: There are animals in that cage
response: Reenter cage number

Repeat 2a-d until enough animals are weighed for the ranking; return to 2a if original issue cage is empty or to 2b if more animals are present. The number of animals weighed will be more than the number of animals required for allocation, to allow for replacement of animals culled or missing during allocation.

Note that some entries for 3 and 4 may be disallowed for individual allocations by the experiment protocol entered in Study Definition.

3. Subset of cages - select a subset of the cages on this rack; DEFAULT = blank prompt: Enter starting cage #, Row, Column, or Return: XXXX
response: bar code or key in the number of the first cage for this allocation or press R or C to select specific rows or columns or press the Return key to process the entire rack
possible error message: invalid cage number
response: reenter cage number

4a. If a cage number was entered in 3.
Ending cage; DEFAULT = last cage number on rack
prompt: Enter ending cage: 9999
response: bar code or key in the number of the last cage for this allocation
possible error message: invalid cage number
response: reenter cage number

4b. If R (for rows) or C (for columns) was entered in 3.

Row or Column Numbers; DEFAULT = 0
prompt: Enter row numbers: 9 9 9 9 9 9 9 9 9 99
- or -
prompt: Enter column numbers: 9 9 9 9 9 9 9 9 9 99
response: Enter the row(s) or column(s) to be processed
possible error message: invalid number -Reenter
response: reenter cage number
possible error message: That row/column does not exist on this rack
response: reenter number

--- Preliminary data entry finished; begin processing issue cages ---

A configuration of the specified rack side is displayed in the middle of the screen. If beginning and ending cage numbers were specified, only those cage numbers in that range are displayed, otherwise, all cage numbers for the rack side are displayed (up to 42). If any animals have previously been allocated to this rack side, their individual ear clip codes are displayed below the cages to which they have been allocated.

5. Issue cage - No issue cage is required, if an animal is to be weighed and sent to Teratology only; DEFAULT = 0
prompt: Enter issue cage: 9999
response: bar code or key in the issue cage number
possible error message: invalid cage number
response: reenter cage number
possible error message: There are no animals in that cage
response: reenter cage number
possible error message: Wrong strain/sex
response: reenter cage number

When an appropriate cage number is entered the issue cage number and its cage census is displayed.

6. Issue cage weight - only if this is not weight-ranked; DEFAULT = 0.0
prompt: Weigh the full issue cage: 9999.9
response: place issue cage on balance and weigh; if weight is not automatically transferred to the screen, it may be keyed in.
possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Return to accept value or reenter weight

May be keyed in or entered by placing the issue cage on the electronic balance and pressing the "WEIGH" key.

7. Animal weight with options; DEFAULT = Return for weight-ranking, W otherwise

As each animal is allocated, the allocation transaction is displayed on the transaction line. If the animal was weighed, a weight transaction is also displayed. The animal technician must place the animal in the cage

indicated on the transaction line (also indicated on the rack display by the animal id description next to the cage number).

If the experiment protocol does not allow certain litter mate configurations, a warning message is displayed when litter mates are about to be allocated to an inappropriate rack position: Litter mate Error; DEFAULT = blank
prompt: Enter cage or press return

Response: (1) Special Operator ID's allow the animal technician to override the warning by pressing the 'Enter' key and continuing with the allocation, i.e., allocating litter mates to the same column.

(2) An animal technician that does not have the permission just described, must enter a new issue cage as described for Step 5.

prompt: Weigh <SEX>, Return, End, Removal, or Issue cage: XXXXXX
response: W (weigh animal), 'Return' key, E (end), R (removal), or I (issue cage)

responses are processed as follows:

7a. Weigh - Not valid if this is a weight-ranked allocation.
prompt: Weigh animal: 9999.9
response: Remove the animal from the issue cage and press the "W" key.
possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Enter to accept value or reenter weight
If the weight is accepted, the allocation of that animal is recorded as indicated on the screen by an ear clip description (None, Left, Right, or Both) or by an id number next to the cage number. The following message is displayed; DEFAULT = blank

prompt: Clip, move animal, and verify: X - or -
Clip, move animal, and verify (must be <PHENOTYPE>): X
possible error message: Wrong clip
response: Reenter clip

Response: Verify the animal's ear clip (or other ID) and press the appropriate ear clip identification key or the digit 1, as appropriate. If ear clip is used the initial letter of that ear clip appears below the cage number. If another kind of id is used, the digit 1 appears below the cage number.

7b. Return - Press the 'Return' key if the animal being allocated is not to be weighed. This will always be the method of allocation for weight-ranked experiments. 'Return' will not be accepted as a valid choice if the experiment protocol requires weighing and this is not a weight-ranked allocation. The following message is displayed; DEFAULT = blank

prompt: Clip, move animal, and verify: X - or -
Clip, move animal, and verify (must be <PHENOTYPE>): X
possible error message: Wrong clip
response: Reenter clip

Response: Verify the animal's ear clip (or other ID) and press the appropriate ear clip identification key or the digit 1, as appropriate. If ear clip is used the initial letter of that ear clip appears below the cage number. If another kind of id is used, the digit 1 appears below the cage number.

7c. End - Signals a normal ending of the allocation process. Go to step 8.

7d. Removal - If an animal in the issue cage is going to be removed rather than allocated, the following prompts will appear:

Reason - not required if disposition is not blank; DEFAULT = blank
prompt: Enter reason: XXXXXXXX
response: D (dead) or C (culled)
possible error message: Must be D or C - Reenter
response: reenter

Disposition - not required; DEFAULT = blank
prompt: Enter disposition: XXXXXXXX
response: P (pathology) C (chemistry) or M (microbiology)
possible error message: Must be P C or M - Reenter
response: reenter
if both Reason and Disposition are blank the message
Must enter reason and/or disposition
will appear and processing will return to 'Reason'.

CID - not required; DEFAULT = 0
prompt: Enter CID: 99999999
response: bar code or key in the carcass identification number
possible error message: Invalid CID - Reenter
response: reenter CID
possible error message: CID already used - Reenter
response: enter a different CID

If this is a weight-ranked allocation, the 'excess' weighed animal with a weight that is closest to this animal's weight will be assigned to this experiment cage; DEFAULT = 0
prompt: Select issue cage <9999>; enter issue cage: 9999
response: Barcode or enter the issue cage required
No other cage will be allowed. (If this animal is also defective, it must be removed using the removal function, and then another animal can be selected.) Processing will continue at 7a. Otherwise, the allocation function is resumed as normal.

7e. Issue cage - If animals from a different issue cage are to be

processed, go to Step 5.

Step 7 is repeated until all appropriate animals in the issue cage have been allocated. Then processing either goes to step 5 to get a new issue cage or to step 8 if the allocation is complete.

---

8. When a rack side (or the selected range of cages) is finished or the animal technician signals that the allocation is ending (normally) the following prompt will appear; DEFAULT = N

Are any allocated animals missing? X

If the response is 'n', allocation is ended.

9. If the response is 'y', animals may be re-allocated and all allocation data on the original animal is removed. The original animal is given the final disposition 'Dead' rather than 'Allocated'. The sequence begins; DEFAULT = 0:
9a. prompt: Enter cage number of missing animal: 9999
response: bar code or key in the number of the experiment cage which has an animal missing
possible error message: No animals have been allocated to this cage
response: reenter cage number
9b. prompt: Enter clip or animal ID: XXXXX
If more that one animal was allocated to that cage; DEFAULT = blank:
response: enter animal ID or earclip
possible error message: No animal with that clip was allocated to this cage
response: reenter ID or earclip

9c. CID - not required; DEFAULT = 0
prompt: Enter CID: 99999999
response: bar code or key in the carcass identification number
possible error message: Invalid CID - Reenter
response: reenter CID
possible error message: CID already used - Reenter
response: enter a different CID

If current issue cage contains a suitable replacement perform steps 6 and 7, otherwise perform steps 5, 6, and 7.

When this animal is allocated the prompt; DEFAULT = 0;
Enter cage number of missing animal: 9999

appears again and the process may be repeated. If no cage number is entered (i.e., 'Return' is pressed), the process is assumed to be completed and allocation terminates normally.

Figure A.4. Enter/Fill/Display Animal Orders

```
Animal Orders                                                    DD MMM YY HH:MM
Operator 99999                                                     Room XXX-XXXX
          Date                    Strain    #    #                          Date
 Exp #  Required P.I./Contact      Sex Req Del Special Requirements        Filled
9999-99 MM/DD/YY XXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX MM/DD/YY
Room XXX-XXXX    XXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX
Date_Ordered= MM/DD/YY         XX X 999 999 XXXXXXXXXXXXXXXXXXXX

9999-99 MM/DD/YY XXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX MM/DD/YY
Room XXX-XXXX    XXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX
Date_Ordered= MM/DD/YY         XX X 999 999 XXXXXXXXXXXXXXXXXXXX
                                XX X 999 999 XXXXXXXXXXXXXXXXXXXX

9999-99 MM/DD/YY XXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX MM/DD/YY
Room XXX-XXXX   XXXXXXXXXXXXXX XX X 999 999 XXXXXXXXXXXXXXXXXXXX
Date_Ordered= MM/DD/YY         XX X 999 999 XXXXXXXXXXXXXXXXXXXX

Enter Function or Cage: XXXX
-------------------------------------------------------------------------------
Arrow keys - scroll text; Gold N - Erase Data, Restart Screen;
PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
===============================================================================
```

Purpose:
To track animal orders so that appropriate breeding levels can be maintained, and to ensure that orders are filled in a timely manner.

Notes:
All fields except 'Date Filled' and 'Number Delivered' will be entered by a person designated to perform this task. These data may be entered in any order and will be sorted for display by experiment number. After the order has been filled, the 'Date Filled' field will be entered and that order will be displayed only through Friday of the following week. This date will be filled automatically if the Issue process indicates that an order has been filled.

When entering orders, if an old experiment number is entered on a new line, the data previously entered for that experiment will be filled in as defaults for the new entry.

If the 'Exp #' is set to '0000-00', the order will be deleted from the system.

This screen will display either those orders that can be filled by the strains present in that room or all orders pending. The room will be displayed on the top of the screen only when the Display Animal Orders function is invoked.

Enter Animal Orders:

No prompts will appear for these data. They will be filled in from right to left and top to bottom as needed. As the screen is filled with orders, the data will scroll off the top of the screen. Error correction will be performed by 'backing up' with the Tab key. Note that the Experiment and Test + Date Requested must be unique for each order.

1. Experiment and test - The experiment the animals are order for; DEFAULT = 0
possible error message: Invalid experiment - Reenter
response: Reenter experiment and test

2. Date Requested - The date the animals are needed for the experiment; DEFAULT = today's date
possible error message: Invalid date - Reenter
response: Reenter date
possible error message: That seems unreasonable
response: Press Return to accept value or reenter date

3. Building and Room - The building and room to which the animals will be delivered; DEFAULT = 0
possible error message: Invalid room - Reenter
response: Reenter building and room

4. P.I/Contact - The principal investigator(s) for the experiment and/or the person(s) who should be contacted to answer questions about the order; DEFAULT = blank

5. Strain - The strain required for this order; DEFAULT = blank
possible error message: Invalid strain - Reenter
response: Reenter strain

6. Sex - The sex required for this order; DEFAULT = U
possible error message: Must be M F or U - Reenter
response: Reenter sex

7. Number Required - The number required of this strain and sex, for the special requirements to follow; DEFAULT = 0
possible error message: Invalid number - Reenter
response: Reenter number
possible error message: That seems unreasonable
response: Press Return to accept value or reenter number

8. Special Requirements - Any other requirements on the order (such as age, phenotype, etc.) This is a text string and no error checking is done on it; DEFAULT = blank

-- return to Step 1. Gold 9 returns the operator to the NARSS menu.

Fill Animal Orders
NOTE: not all animal handlers will be allowed to enter data on this

screen.

1. Experiment Number; DEFAULT = 0
prompt: Enter Experiment and Test: 9999-99
response: enter the experiment and test number for which an order is to be marked as filled. Enter zero to terminate filling of orders.
possible error message: Invalid experiment-test - Reenter
response: Reenter number
possible error message: No orders for that experiment-test - Reenter
response: Reenter number

2. Date Requested - only if there are more than one date for this experiment and test; DEFAULT = today's date
prompt: Enter Date Requested: MM/DD/YY
response: enter the date requested for which an order is to be marked as filled.
possible error message: Invalid date - Reenter
response: Reenter number
possible error message: No orders for that date- Reenter
response: Reenter number

3. Number delivered; DEFAULT = number previously issued
prompt: Enter the Number Delivered: 999
response: enter the number of animals actually delivered (which may not be the same as the number requested)
possible error message: Invalid number - Reenter
response: Reenter number
possible error message: That seems unreasonable
response: Renter number or press Return to accept value

4. Date filled; DEFAULT = today's date
prompt: Enter Date Filled: MM/DD/YY
response: enter the date on which an order is to be marked as filled.
possible error message: Invalid date - Reenter
response: Reenter date
possible error message: That seems unreasonable
response: Reenter date or press Return to accept value

5. After all data are correctly entered; DEFAULT = blank:
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
: There are no animals in that cage
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

6. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
Are all the data on this screen correct? X
If the response is 'n', data editing begins.

If the response is 'y', the new cage is displayed for the current function.

Display Animal Orders

1. Display for this room only or whole colony; DEFAULT = R
  prompt: Orders for this room (R) or whole colony (C): X
response: Enter R to display only those orders that can be filled from the strains present in this room. Enter C to display all orders.
    possible error message: Must be R or C
                response: reenter

If R is entered and no appropriate strains are present in this room, the message
'There are no strains in this room that can fill any pending orders'
will appear.

2. To end the display; DEFAULT = blank:
    prompt: Enter Function or Cage: XXXX
    response: bar code or key in the cage number or press a function key
    possible error message: Invalid cage - Reenter
                response: reenter cage number
    possible error message: There are no animals in that cage
                response: reenter cage number
    possible error message: Invalid function - Reenter
                response: reenter function

Figure A.5. Display Retirement Candidates

```
==========================================================================
Retirement Candidates                                        DD MMM YY HH:MM
Operator 99999                  Days      Age                  Room XXX-XXXX
  Cage   Female       Male   Established (Days)
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
  9999 XX-9999999 XX-9999999    999       999
Enter Function or Cage: XXXX
--------------------------------------------------------------------------

Arrow keys - scroll text; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==========================================================================
```

Purpose:
To display those breeders that have reached retirement age.

Notes:
Breeder animals in this room will appear on this screen if they are at or past the retirement age set for their strain in the strain table. The days established and age on this screen are those of the female.

If no breeder cages are present in this room, the message
'No breeders in this room' will appear.
If no breeders are retirement candidates in this room, the message
'No retirement candidates in this room' will appear.

1. End display; DEFAULT = blank;
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

Figure A.6. Display Unproductive Breeders

```
==================================================================================
Unproductive Breeders                                                DD MMM YY HH:MM
Operator 9999                          Days      Last Litter          Room XXX-XXXX
  Cage   Female       Male   Established  (Days)  No.
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
  9999 XX-9999999 XX-9999999    999        999    99
Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Arrow keys - scroll text; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To display those breeders that have not produced a litter within a set number of days.

Notes:
Breeder animals in this room will appear on this screen if they have not produced a litter for a length of time set for their strain in the strain table. The days established and last litter data on this screen are those of the female.

If no breeder cages are present in this room, the message
'No breeders in this room' will appear.
If no breeders are retirement candidates in this room, the message
'No unproductive breeders in this room' will appear.

1. End display; DEFAULT = blank;
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

Figure A.7. Display Feeders by Age Range

```
==================================================================================
Feeders by Age Range                                              DD MMM YY HH:MM
Operator 99999            Feeder Age: 999 to 999                    Room XXX-XXXX
          Cage     UIN      Sex  Age
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Arrow keys - scroll text; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To display those feeders that are between certain ages.

Notes:
Feeder animals in this room will appear on this screen if they have are at or between the ages entered by the terminal operator.

If no feeder cages are present in this room, the message
'No feeders in this room' will appear.
Note that there is no mechanism for correcting the minimum age after it has been entered. If it is incorrect, the maximum still must be entered and the screen displayed. The operator can then re-invoke the 'Display Feeders by Age Range Function'.

1. Minimum Age for display; DEFAULT = 0;
   prompt: Enter minimum age (days): 999
   response: enter minimum age in days of feeders to be displayed
   possible error message: Invalid age - Reenter
   response: reenter age

2. Maximum Age for display; DEFAULT = 999;
   prompt: Enter maximum age (days): 999
   response: enter maximum age in days of feeders to be displayed
   possible error message: Invalid age - Reenter
   response: reenter age
   possible error message: Must not be less than minimum - Reenter
   response: reenter age

If no feeders in this room are within the entered age range, the message
'No feeders in this age range in this room' will appear.

3. End display; DEFAULT = blank;
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

Figure A.8. Display Issue-Feeders by Age Range

```
==================================================================================
Issue/Feeders by Age Range                                        DD MMM YY HH:MM
Operator 99999            Feeder Age: 999 to 999
          Cage      UIN      Sex  Age
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
          9999  XX-9999999   X   999
Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Arrow keys - scroll text; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To display those feeders in issue cages that are between certain ages.

Notes:
Feeder animals in all issue cages will appear on this screen if they have are within the age range entered by the terminal operator.

If no feeder cages are present in issue cages, the message
'No feeders in any issue cages' will appear.
Note that there is no mechanism for correcting the minimum age after it has been entered. If it is incorrect, the maximum still must be entered and the screen displayed. The operator can then re-invoke the 'Display Feeders by Age Range Function'.

1. Minimum Age for display; DEFAULT = 0;
   prompt: Enter minimum age (days): 999
   response: enter minimum age in days of feeders to be displayed
   possible error message: Invalid age - Reenter
   response: reenter age

2. Maximum Age for display; DEFAULT = 999;
   prompt: Enter maximum age (days): 999
   response: enter minimum age in days of feeders to be displayed
   possible error message: Invalid age - Reenter
   response: reenter age
   possible error message: Must not be less than minimum - Reenter
   response: reenter age

If no feeders in issue cages are within the entered age range, the message
'No feeders in this age range in issue cages' will appear.

3. End display; DEFAULT = blank;
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

Figure A.9. Display Room Work Summary

```
==========================================================================================
Room Work Summary        Room XXX-XXXX                                  DD MMM YY HH:MM
Operator 99999           Issued this Room     Issued on System
   Strain Phen    Exp. #   M   F   U Total    M   F   U Total
     XX   XXXX   9999-99 999 999 999 999    999 999 999 999
     XX   XXXX   Feeder  999 999 999 999    999 999 999 999
     XX   XXXX O/9999-99 999 999 999 999    999 999 999 999

                         Terminal Totals
     Breeders Retired     Pairs    New    Num. of  Feeders Set Up
Strain  Phen  M    F      Set Up   Born   Litters  M   F   U   Tot
  XX    XXXX 999  999      999     999     999    999 999 999 999
  XX    XXXX 999  999      999     999     999    999 999 999 999
  XX    XXXX 999  999      999     999     999    999 999 999 999
Total        999  999      999     999     999    999 999 999 999

Removals:    Dead              Culled            Surplus           Survey
Strain M   F   U   Tot   M   F   U   Tot   M   F   U   Tot   M   F   U   Tot
  XX   999 999 999 999   999 999 999 999   999 999 999 999   999 999 999 999
  XX   999 999 999 999   999 999 999 999   999 999 999 999   999 999 999 999
  XX   999 999 999 999   999 999 999 999   999 999 999 999   999 999 999 999
Total  999 999 999 999   999 999 999 999   999 999 999 999   999 999 999 999

Enter Function or Cage: XXXX
------------------------------------------------------------------------------
Arrow keys - scroll text; Gold N - Erase Data, Restart Screen;
PF2- Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==========================================================================================
```

Purpose:
To display a summary of all work performed in a room on that day, or to display all work done in the colony on that day. Also, to be able to determine the number of issues still to be performed in this room by comparing the total number of issues on the system with the total number required.

Notes:
Only display issue lines with non-zero system totals.
Display all lines only for strains that are present in that room.

1. Display for this room only or whole colony; DEFAULT = R
prompt: Summary for this room (R) or whole colony (C): X
response: Enter R to display only work done in this room. Enter C to display all work done in the colony.
possible error message: Must be R or C
response: reenter

2. End display; DEFAULT = blank;
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

Figure A.10. New Animal Entry

```
==============================================================================
New Animal Entry          XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                       Room XXX-XXXX           ** MODE **
    Males XXXX          Females XXXX
     DD MMM YY            DD MMM YY            Block          Original Block
   XX-9999999  X        XX-9999999 XXX      XX-9999999          XX-9999999
  Ct=99 Gen=9999      Ct=99 Gen=9999      Ct=99 Gen=9999    Ct=99 DD MMM YY
------------------------------------------------------------------------------
Adult Males  Adult Females  Adults Unsexed  Litter Animals  Exp# 9999-99
   XXXXXXX        XXXXXXX        XXXXXXX          XXXXXXX   Status
  DD MMM YY      DD MMM YY      DD MMM YY        DD MMM YY  Date of Birth
  99 XXXXXX      99 XXXXXX      99 XXXXXX               99  Strain / Phen.
   99  9999       99  9999       99  9999         99  9999  Count / Gen.
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX M Vendor's ID
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX F Vendor's ID
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX U Vendor's ID
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX L Vendor's ID
Notes:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Enter Function or Cage: XXXX
-----------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen;
PF2 - Help; Gold T - Phen. Table; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==============================================================================
```

Purpose:
To enter new (purchased) animals into the colony.

Notes:
The Notes field applies only to the UIN's assigned on this screen.
If the cage into which the new animals are being entered already contains any animals, they will appear on the area at the top of the screen.

1. Enter sex or litter; DEFAULT = blank
   prompt: Enter sex or litter: X
   response: M F U L or blank (when finished entering new animals to this cage)
   possible error message: Must be M or F or L - Reenter
   response: reenter sex or litter

2. Status; DEFAULT = B
   prompt: Enter status: XXXXXXX
   response: B (breeder), F (feeder), or I (issue) for non-litter animals
   L or blank (litter) or I (issue) for litter animals
   possible error message: Must be B F L or I - Reenter
   response: reenter status
   possible error message: This is a Breeder/Feeder cage (see note)
   response: reenter status

Note: You may put any animals in an empty breeder/feeder cage; Animals that have been given breeder status may not be put in an Issue cage. Cage rules apply to non-empty cages.

Cage rules: Multiple feeder or issue animals put in one cage must be of the same sex. A breeder cage must contain zero or one male and must contain from 1 to 3 females. A litter that is currently with its mother can only be transferred AS A LITTER to another breeder cage if its mother is also transferred; this is only performed in the Move Cage function.

2a. Reason - for Issue status only; DEFAULT = blank
   prompt: Enter reason for issue: XXXXXXXX
   response: E (experiment) or F (feeder)
   possible error message: Must be E or F - Reenter
   response: reenter reason

2b. Experiment-test - for Issue Experiment only - may be blank - only one experiment may be entered for a cage; DEFAULT = 0
   prompt: Enter Experiment number: 9999-99
   response: experiment and test number
   possible error message: Invalid experiment - Reenter
   response: reenter experiment

3. Date of birth; DEFAULT = today's date
   prompt: Enter date of birth: DD MMM YY
   response: date of birth of this animal or litter
   possible error message: Invalid date - Reenter
   response: reenter date
   possible error message: That seems unreasonable
   response: Press Return to accept value or reenter date

4. Strain; DEFAULT = blank
   prompt: Enter strain: XX
   response: strain of this animal or litter

possible error message: Invalid strain - Reenter
response: reenter strain

Note: either 5a or 5b will appear, but not both.

5a. Phenotype - only appears when required by strain code; DEFAULT = blank
prompt: Enter phenotype: XXXXXX
response: H (haired) or N (nude) for strain 84
B (black) Y (yellow) or A (agouti) for strains 29 37 or 46
possible error message: Must be H N B Y or A - Reenter
response: reenter phenotype

5b. Phenotype - only appears when entering the second animal in a breeder pair and when required by strain code; DEFAULT = blank
prompt: Phenotype must be X or X; Enter phenotype: XXXXXX
response: H against N for strain 84
B against Y or A for strains 29 37 or 46
possible error message: Wrong phenotype - Reenter
response: if phenotype was entered incorrectly, reenter phenotype; if the animal is of the wrong phenotype, abandon the action and start over.

6. Count; DEFAULT = 1
prompt: Enter count: 99
response: number of animals of this type
possible error message: Invalid count - Reenter
response: reenter count
possible error message: That seems unreasonable
response: Press Return to accept value or reenter count

7. Generation; DEFAULT = 1
prompt: Enter generation: 9999
response: generation of this animal or litter
possible error message: Invalid generation - Reenter
response: reenter generation

8. Vendor's ID - optional; DEFAULT = blank
prompt: Enter vendor's ID: X(70)
response: any alphanumeric string that is supplied by the vendor and an ID for this animal or litter. No data checking is performed on this field.
--- return to Step 1 ---
---

9. Notes
After the operator has entered a blank at step 1, the cursor is positioned in the 'Notes' area. This is a 4 line scrolling field in which to enter any text relative to this new animal entry. Enter Cntl-Z to terminate the text entry. The field is initially blank.

10. After all data are correctly entered; DEFAULT = blank:
prompt: Enter Function or Cage: XXX

response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: There are no animals in that cage
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

11. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:

Are all the data on this screen correct? X

If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.11. Change Cage Status

```
==================================================================================
Change Cage Status      XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX          ** MODE **
    Males XXXX       Females XXXX
     DD MMM YY         DD MMM YY          Block         Original Block
   XX-9999999  X     XX-9999999 XXX     XX-9999999         XX-9999999
 Ct=99 Gen=9999    Ct=99 Gen=9999      Ct=99 Gen=9999    Ct=99 DD MMM YY
----------------------------------------------------------------------------------

 New Cage Status? XXXXXXX


Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To change the status of all animals in a cage without moving the cage.

Notes:
May be breeder to feeder or feeder to breeder or either of these to issue. The new status default will be selected based on the current cage status.

1. New status; DEFAULT = B for breeder cage and F for feeder cage
   prompt: Enter new status: XXXXXX
   response: B (breeder) or F (feeder)
   possible error message: Must be B or F - Reenter
   response: reenter status
   possible error message: Can't change status of an issue cage
   response: abandon function

2. After datum is correctly entered; DEFAULT = blank:
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   : There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter

3. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
   Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.12. Move Cage

```
==================================================================================
Move Cage                XXXXXXX Cage Number 9999                DD MMM YY HH:MM
Operator 99999                        Room XXX-XXXX               ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY           Block            Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999            XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999      Ct=99 Gen=9999      Ct=99 DD MMM YY
----------------------------------------------------------------------------------
New Cage 9999     Switch cages? X


Enter Function or Cage: XXXX
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To move a cage to another rack position in that room. This allows moving an entire breeding unit (male, female, litter) to a new location in the room.

Notes:
When a breed cage is moved, this must print a new breeder card with all the old cage information (except cage number) as it was.

1. New cage position; DEFAULT = 0
   prompt: Enter new cage number: 9999
   response: bar code or key in the cage number position
   possible error message: Invalid cage - Reenter
   response: reenter cage position
   possible error message: Can't move an issue cage using this screen
   response: abandon action

2. Switch cages; DEFAULT = Y
   prompt: Switch these two cages? X
   response: Y or N
   possible error message: Must be 'y' or 'n'
   response: reenter response

3. After all data are correctly entered; DEFAULT = blank:
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   possible error message: There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

4. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = Y:
   Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.13. Display Cage

```
==================================================================================
Display Cage            XXXXXXX Cage Number 9999            DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX             ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY          Block           Original Block
   XX-9999999  X     XX-9999999 XXX     XX-9999999          XX-9999999
 Ct=99 Gen=9999    Ct=99 Gen=9999     Ct=99 Gen=9999     Ct=9: DD MMM YY
----------------------------------------------------------------------------------

Enter Function or Cage: XXXX
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To display the number and status of the animals in a cage.

Notes:
To allow harem mating, two fields have been added on the top of the screen. The field next to male UIN is P (present in cage) or A (absent from cage). The three fields next to female UIN are the ear clips of the females present in the cage. These two fields will only be displayed on harem-mated cages.

Three other fields are new. The date under 'Males' and 'Females' is the date that individual was established as a breeder. The fields next to Males and Females are phenotype and will only appear for strains with more that one phenotype. The 'MODE' field on the right will display 'Seq Wean', 'Littered', or 'All' if one of the special processing aids has been selected.

'Block' is displayed when litter animals are present in the cage. 'Original Block' is displayed when a litter is born, and remains on that breeder cage until a new litter is born. If litter animals are placed in an issue cage, the corresponding 'Original Block' is displayed on that cage.

1. End display; DEFAULT = blank;
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

Figure A.14. Animal Assignment

```
==========================================================================
Assignment                XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                      Room XXX-XXXX            ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY          Block          Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999         XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999      Ct=99 Gen=9999    Ct=99 DD MMM YY
--------------------------------------------------------------------------
Current   Phen/     ---  New ---                      CAN or CID
Status Sex Obs  #  Status  Cage  Reason Disposition  or EXP  UIN Assigned
XXXXXXX X XXXX 99  XXXXXXX 9999                                XX-9999999
XXXXXXX X XXXX 99  XXXXXXX 9999                                XX-9999999
XXXXXXX X XXXX 99  XXXXXXX 9999                                XX-9999999
XXXXXXX X XXXX 99  XXXXXXX 9999                                XX-9999999
XXXXXXX X XXXX 99  XXXXXXX 9999                                XX-9999999


Enter Function or Cage: XXXX
--------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen;
PF2 - Help; Gold T - Phen. Table; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==========================================================================
```

Purpose:
To assign an animal a new status.

Notes:
Breeder assignments made with a Count greater than 1 must be sorted by new cage. UIN's assigned to litter animals appear automatically. UIN's are assigned to breeder pairs taking the numbers from the low end of the block for the males and from the high end of the block for the females.

Cage rules: Multiple feeder or issue animals put in one cage must be of the same sex. A breeder cage must contain zero or one male and must contain from 1 to 3 females. A litter that is currently with its mother can only be transferred AS A LITTER to another breeder cage if its mother is also transferred; this is only performed in the Move Cage function.

1. Current Status; DEFAULT = B for breeder cage, F for feeder cage, and I for issue cage
    prompt: Enter Status: XXXX
    response: B (breeder) F (feeder) L (litter) I (issue-feeder) or, when done, press the 'Enter' key
    possible error message: Must be B F L I or blank - Reenter
                response: reenter status
    possible error message: That status is not present in this cage
                response: reenter status

2. Sex; DEFAULT = B for breeder cage, U otherwise
    prompt: Enter Sex: X
    response: M F B (both) or U (unsexed; not valid for breeders).
    possible error message: Must be M F B or U - Reenter
                response: reenter sex

Note: either 3a or 3b may appear, but not both.

3a. Phenotype - only appears when required by strain code; DEFAULT = blank
    prompt: Enter Phenotype: XXXX
    response: H (haired) or N (nude) for strain 84
              B (black) Y (yellow) or A (agouti) for strains 29 37 and 46
    possible error message: Must be H N B Y or A - Reenter
                response: reenter phenotype
    possible error message: That phenotype is not present in this cage
                response: reenter phenotype

3b. Phenotype - only appears when entering the second animal in a breeder pair and when required by strain code; DEFAULT = blank
    prompt: Phenotype must be X or X; Enter phenotype: XXXXXX
    response: H against N for strain 84
              B against Y or A for strains 29 37 or 46
    possible error message: Wrong phenotype - Reenter
                response: if phenotype was entered incorrectly, reenter phenotype; if the animal is of the wrong phenotype, abandon the action and start over.

4. Count - only appears when allowed by sex, phenotype, and cage census; if sex = B this count is the number of pairs, not the number of individuals; DEFAULT = 1
    prompt: Enter Count: 99
    response: number of animals or breeder pairs to be assigned
    possible error message: Not that many of that sex
                response: reenter count
    possible error message: That seems unreasonable
                response: Press Return to accept value or reenter count

5. New Status; DEFAULT = blank
    prompt: Enter New Status: XXXXXX
    response: B (breeder) or F (feeder)
    possible error message: Must be B or F - Reenter

response: reenter status

6. New Cage; DEFAULT = 0
prompt: Enter New Cage: 9999
response: bar code or key in the new cage number
If no cage is entered and the cage census allows, the status of the current cage is changed.
If the new status is breeder and count is greater than one, then this is the first of 'Count' sequences cages to which breeder assignments will be made.
possible error message: Invalid cage - reenter
response: reenter cage
possible error message: Destination incompatible with status - reenter
response: reenter cage
possible error message: Can't assign to an issue cage
response: select new cage

--- Go back to Step 1 ---

7. After all data are correctly entered; DEFAULT = blank:
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
possible error message: There are no animals in that cage
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

If a valid function is entered, that function is begun for this cage.

8. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

If Return is pressed and no special processing aid has been selected, the message 'No special cage sequencing selected' appears and the data must be re-entered.

Figure A.15. Animal Removal

```
==================================================================================
Removal                    XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                        Room XXX-XXXX           ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY          Block          Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999          XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999      Ct=99 Gen=9999     Ct=99 DD MMM YY
----------------------------------------------------------------------------
Current   Phen/     ---  New ---                      CAN or CID
Status Sex Obs  #   Status  Cage  Reason Disposition  or EXP  UIN Assigned
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999



Enter Function or Cage: XXXX
----------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;

Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To cull or surplus an animal, to send it to be surveyed, or note its death.

Notes:
Previously a disposition and CID were required only if reason = Dead; However, since a 'missing' animal is defined to be 'dead', and a missing animal cannot get a CID or disposition, these two items must always be optional.

1. Current Status; DEFAULT = L for breeder cage, F for feeder cage, and I for issue cage
   prompt: Enter status: XXXXXX
   response: B (breeder) F (feeder) L (litter) I (issue-feeder) or press 'Enter' to end the function
   possible error message: Must be B F L I or blank - Reenter
                response: reenter status

2. Sex - only appears when more than one sex is present; DEFAULT = U
   prompt: Enter sex: X
   response: M F or U (unsexed)
   possible error message: Must be M F or U - Reenter
                response: reenter sex

3. Phenotype - only appears when allowed by strain code; DEFAULT = blank
   prompt: Enter phenotype
   response: H (haired) or N (nude) for strain 84
             B (black) Y (yellow) or A (agouti) for strains 29 37 or 46
   possible error message: Must be H N B Y or A - Reenter
                response: reenter phenotype

4. Count - only appears when required by sex, phenotype, and census; DEFAULT = 1
   prompt: Enter count: 99
   response: number of animals for this removal
   possible error message: Not that many of that sex
                response: reenter

5. Reason - required if disposition is blank; DEFAULT = blank
   prompt: Enter reason: XXXXXXXX
   response: D (dead) S (surplus) or C (culled)
   possible error message: Must be D S or C - Reenter
                response: reenter

6. Disposition - not required; DEFAULT = blank
   prompt: Enter disposition: XXXXXXXX
   response: P (pathology) C (chemistry) or M (microbiology)
   possible error message: Must be P C or M - Reenter
                response: reenter
   if both Reason and Disposition are blank the message
                Must enter reason and/or disposition
   will appear and processing will return to step 5.

7. CID - not required; DEFAULT = 0
   prompt: Enter CID: 99999999
   response: bar code or key in the carcass identification number
   possible error message: Invalid CID - Reenter
                response: reenter CID
   possible error message: CID already used - Reenter
                response: enter a different CID

8. After all data are correctly entered; DEFAULT = blank:

```
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
                response: reenter cage number
                        : There are no animals in that cage
                response: reenter cage number
possible error message: Invalid function - Reenter
                response: reenter function
```

If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:

Are all the data on this screen correct? X

If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.16. Animal Retirement

```
==========================================================================
Retire                    Breeder Cage Number 9999          DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX             ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY          Block          Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999         XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999     Ct=99 Gen=9999     Ct=99 DD MMM YY
--------------------------------------------------------------------------
Current   Phen/     ---  New ---                    CAN or CID
Status Sex Obs  #  Status  Cage  Reason Disposition  or EXP  UIN Assigned
        X                         Retired  XXXXXXXX 99999999999 XX-9999999


 Enter Function or Cage: XXXX
--------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==========================================================================
```

Purpose:
To retire one or both breeders in a cage or a sequence of cages.

1. Retire from an Individual Cage or a Sequence of Cages; DEFAULT = I
prompt: Sequential or Individual: X
response: I (individual) or S (sequential)
possible error message: Must be I or S - Reenter
response: reenter choice

If Sequential (each animal retired in each cage will appear as a line entry on the same screen; the cage numbers will appear in the New Cage field):

2a. Starting cage number; DEFAULT = current cage
prompt: Enter first cage to retire: 9999
response: bar code or key in the first cage number to retire or press 'Enter' to end the function
possible error message: Invalid cage number
response: reenter cage number

Ending cage number; DEFAULT = first cage entered above
prompt: Enter last cage to retire: 9999
response: bar code or key in the last cage number to retire
possible error message: Invalid cage number
response: reenter cage number

2b. If Individual use current cage.

3. Sex; DEFAULT = B
prompt: Enter sex
response: M F or B (both)
possible error message: Must be M F or B - Reenter
response: reenter sex

4. Count - only appears for females and harem matings; If count >1 and a sequence of cages are being retired, this is the maximum number of harem-mated females to be retired per cage; DEFAULT = 1
prompt: Enter count
response: 1, 2, or 3
possible error message: Must be 1 2 or 3 - Reenter
response: reenter count

5. Disposition - may be blank; DEFAULT = blank
prompt: Enter disposition
response: P (pathology) C (chemistry) or M (microbiology)
possible error message: Must be P C or M - Reenter
response: reenter disposition

6. CID - may be blank; DEFAULT = 0
will be repeated for each animal and cage
prompt: Enter CID
response: bar code or key in the carcass identification number
possible error message: Invalid CID - Reenter
response: reenter CID
possible error message: CID already used - Reenter

response: enter a different CID

--- If sequential, go to 2a ------

7. After all data are correctly entered; DEFAULT = blank:
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   : There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

8. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
   Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.17. Note Pregnant or Plugged

```
==================================================================================
Pregnant                  XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                      Room XXX-XXXX            ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY           Block          Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999          XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999      Ct=99 Gen=9999     Ct=99 DD MMM YY
----------------------------------------------------------------------------------
Current    Phen/     ---  New ---                        CAN or CID
Status Sex Obs   #  Status  Cage  Reason Disposition  or EXP  UIN Assigned
           PR                                                   XX-9999999


Enter Function or Cage: XXXX
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;

Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To note that a female is pregnant or that a plug has been observed.

Notes:
The code entered is 'PR' for pregnant or 'PL' for plugged.

possible error message: No breeder female in cage
response: select a new cage or new function

1. prompt: Enter Function or Cage: XXXX; DEFAULT = blank
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   : There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

2. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
   Are all the data on this screen correct? X
If the response is 'n', the observation is deleted.
If the response is 'y', the new cage is displayed for the current function.

Figure A.18. Note Litter

```
==================================================================================
Littered                  XXXXXXX Cage Number 9999                DD MMM YY HH:MM
Operator 99999                         Room XXX-XXXX               ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY            Block            Original Block
   XX-9999999  X      XX-9999999 XXX      XX-9999999            XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999       Ct=99 Gen=9999     Ct=99 DD MMM YY
----------------------------------------------------------------------------------
Current   Phen/     ---  New ---                          CAN or CID
Status Sex Obs  #   Status  Cage  Reason Disposition  or EXP  UIN Assigned
           L   99                                                XX-9999999

Is the number correct? X  Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To note that a female has littered and to note the number of litter animals.

Notes:
Original litter block remains on the screen until a new litter is born, and it is replaced by a new litter block, or the female is retired, and it is cleared.

Litter generation is the female generation plus 1, except for crossbreeding. In that case the generation is 1. The strain for a crossbred litter is read from a table.

The logic for determining the new litter's treatment/test will be provided by NCTR after discussions with the principal investigators. This problem pertains only to animals born on an experiment.

possible error message: No breeder female in cage
response: select a new cage or new function
possible error message: Litter already in cage, wean litter first
response: Assign, issue, or remove previous litter animals

1. Count; DEFAULT = 1
   prompt: Enter number of pups: 99
   response: number of pups observed
   possible error message: Invalid count - Reenter
   response: reenter count
   possible error message: That seems unreasonable
   response: Press Return to accept value or reenter count

2. After datum is correctly entered; DEFAULT = blank:
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   : There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

3. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
   Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.19. Issue Animal

```
==========================================================================
Issue                    XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX            ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY          Block          Original Block
   XX-9999999  X      XX-9999999 XXX     XX-9999999         XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999      Ct=99 Gen=9999    Ct=99 DD MMM YY
--------------------------------------------------------------------------
Current   Phen/     ---  New ---                        CAN or CID
Status Sex Obs  #  Status  Cage  Reason Disposition  or EXP  UIN Assigned
XXXXXXX X XXXX 99  Issue   9999  XXXXXXXX            9999-99     XX-9999999
                                                     99999999


Enter Function or Cage: XXXX
--------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen;
PF2 - Help; Gold T - Phen. Table; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==========================================================================
```

Purpose:
To issue animals to experiments, as feeders, or for other reasons.

Notes:
An issue may have and experiment number and a CAN, but CAN is not currently used.

Once an experiment/CAN is entered, it becomes the default value for the following issue screens.

'Issue other' is a final action.

Male and females may not reside in the same feeder or issue feeder cage. Must be only males, only females or a single litter.

1. Current Status; DEFAULT = B for breeder cage, F for feeder cage, and I for issue cage
   prompt: Enter status: XXXXXX
   response: B (breeder) F (feeder) L (litter) I (issue-feeder) or press 'Enter' to end the function
   possible error message: Must be B F L I or blank - Reenter
   response: reenter status

2. Sex - only appears when more than one sex in cage; DEFAULT = U
   prompt: Enter sex: X
   response: M F or U (unsexed)
   possible error message: Must be M F or U - Reenter
   response: reenter sex

3. Phenotype - only appears when allowed by strain code; DEFAULT = blank
   prompt: Enter phenotype: XXXX
   response: H (haired) or N (nude) for strain 84
   B (black) Y (yellow) or A (agouti) for strains 29 37 or 46
   possible error message: Must be H N B Y or A - Reenter
   response: reenter phenotype

4. Count - only appears when required by sex, phenotype, and cage census; DEFAULT = 1
   prompt: Enter count: 99
   response: number of animals for this issue
   possible error message: Not that many of that sex (and phenotype)
   response: reenter count

5. Reason; DEFAULT = blank
   prompt: Enter reason for issue: XXXXXXXX
   response: E (experiment) F (feeder) or O (other)
   possible error message: Must be E F or O - Reenter
   response: reenter reason

6. New cage - not allowed if reason is 'other'; DEFAULT = 0
   prompt: Enter new cage number: 9999
   response: bar code or key in the issue cage number
   possible error message: Not an issue cage
   response: reenter cage number

7. Experiment-test - may entered if reason = E or other; DEFAULT = 0
   prompt: Enter Experiment number: 9999-99
   response: experiment and test number
   possible error message: Invalid experiment - Reenter
   response: reenter experiment

8. CAN - may be blank. This prompt will appear if the reason is Other and if an experiment number is entered at 7; DEFAULT = 0
   prompt: Enter CAN: 99999999
   response: CAN number
   possible error message: Invalid CAN - Reenter

response: reenter CAN

9. After all data are correctly entered; DEFAULT = blank:
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
   response: reenter cage number
   : There are no animals in that cage
   response: reenter cage number
   possible error message: Invalid function - Reenter
   response: reenter function

10. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
   Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.20. Summary of Various Actions

```
==================================================================================
Retire                    Breeder Cage Number 9999                DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX                   ** MODE **
    Males XXXX        Females XXXX
     DD MMM YY          DD MMM YY            Block            Original Block
   XX-9999999  X      XX-9999999 XXX      XX-9999999          XX-9999999
 Ct=99 Gen=9999     Ct=99 Gen=9999       Ct=99 Gen=9999     Ct=99 DD MMM YY
----------------------------------------------------------------------------------
Current    Phen/     ---  New ---                          CAN or CID
Status Sex Obs   #  Status  Cage  Reason Disposition  or EXP  UIN Assigned
XXXXXXX X       99  XXXXXXX 9999                                XX-9999999
XXXXXXX X       99  XXXXXXX 9999                                XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99                XXXXXXXX XXXXXXXX 99999999999 XX-9999999
XXXXXXX X       99  Issue   9999  XXXXXXXX          9999-99     XX-9999999
                                                    999999999
XXXXXXX X       99  Issue   9999  XXXXXXXX          9999-99     XX-9999999

XXXXXXX X        9                Retired  XXXXXXXX 99999999999 XX-9999999
XXXXXXX X        9                Retired  XXXXXXXX 99999999999 XX-9999999

Enter Function or Cage: XXXX
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen;
PF2 - Help; Gold T - Phen. Table; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Notes:
This is the general appearance of the screen after two assignment, two removals (dead, etc.), two issues, and two retires.

All actions performed on this cage at this time will appear on this screen.

1. After all data are correctly entered; DEFAULT = :blank
   prompt: Enter Function or Cage: XXXX
   response: bar code or key in the cage number or press a function key
   possible error message: Invalid cage - Reenter
                response: reenter cage number
                        : There are no animals in that cage
                response: reenter cage number
   possible error message: Invalid function - Reenter
                response: reenter function

2. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:

   Are all the data on this screen correct? X

If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

Figure A.21. Weigh Pups

```
==================================================================================
Weigh                    XXXXXXX Cage Number 9999          DD MMM YY HH:MM
Operator 99999                     Room XXX-XXXX            ** MODE **
    Males XXXX       Females XXXX
     DD MMM YY         DD MMM YY          Block          Original Block
   XX-9999999  X     XX-9999999 XXX     XX-9999999         XX-9999999
 Ct=99 Gen=9999    Ct=99 Gen=9999      Ct=99 Gen=9999    Ct=99 DD MMM YY
----------------------------------------------------------------------------------
Current   Phen/
Status Sex Obs  #  Mark  UIN Assigned  Weight
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99
           L   99  XXXXX XX-9999999    9999.99

Is the number correct? X  Enter Function or Cage: XXXX
----------------------------------------------------------------------------------

Gold N - Erase Data, Restart Screen; PF2 - Help; Enter - Cont. sequence;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To weigh pups in litter.

Notes:
Original litter block remains on the screen until a new litter is born, and it is replaced by a new litter block, or the female is retired, and it is cleared.

Litter generation is the female generation plus 1, except for crossbreeding. In that case the generation is 1. The strain for a crossbred litter is read from a table.

possible error message: No breeder female in cage
response: select a new cage or new function
possible error message: No litter in cage
response: Renter cage number

1. Weighing cage;
prompt: Place Empty Weighing Cage on Scale
response: place empty cage to be used to weigh animals in on scale and hit enter
possible error message: Cage weight out of range
response: replace cage on scale and hit enter

May be keyed in or entered by placing the cage on the electronic balance and pressing the "WEIGH" key

2. Individual Pups or Litter weights; DEFAULT = (L)itter
prompt: Weighing Individual pups (I) or Entire Litter (L): X
response: Enter I for individual pups or L for entire litter;
possible error message: Invalid option
response: enter an L or I

3a.1 Enter Pup Mark or ID; DEFAULT = NONE
prompt: Enter Pup ID: XXXXX
response: Enter pup mark or ID.
possible error message: invalid ID for that litter
response: reenter ID
possible error message: Pup ID already weighed today
response: reenter ID

3a.2 Weigh Pup; DEFAULT = 0.0
prompt: Weigh Pup
response: Remove the animal from its litter cage and press the "W" key.
possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Enter to accept value or reenter weight

Note: Weight must be reasonable for the particular strain---check table of strain code with valid weights.

3b.1 Weigh All Pups; No default weight provided.
prompt: Weigh All Pups
response: Remove all pups from their litter cage, placing them in weighing cage and press the "W" key.
possible error message: invalid weight
response: reenter weight
possible error message: That seems unreasonable
response: Press Enter to accept value or reenter weight

3b.2. Count; DEFAULT = 1
prompt: Enter number of pups: 99
response: number of pups observed
possible error message: Invalid count - Reenter
response: reenter count
possible error message: That seems unreasonable
response: Press Return to accept value or reenter count

4. After datum is correctly entered; DEFAULT = blank:
prompt: Enter Function or Cage: XXXX
response: bar code or key in the cage number or press a function key
possible error message: Invalid cage - Reenter
response: reenter cage number
: There are no animals in that cage
response: reenter cage number
possible error message: Invalid function - Reenter
response: reenter function

5. If a valid cage number is entered or if Return is pressed, the following prompt appears; DEFAULT = N:
Are all the data on this screen correct? X
If the response is 'n', data editing begins.
If the response is 'y', the new cage is displayed for the current function.

PART A.2. Help Messages for the BMSS Screens

Figure A.1. Login / Identification

```
==================================================================================
Identification                                                    DD MMM YY HH:MM
Operator 99999
----------------------------------------------------------------------------------

  Building and Room number: XXX-XXXX

              Terminal ID: 999999

               Barcode ID: 999999

              Keyboard ID: 999999


Enter Function or Cage: XXXX
----------------------------------------------------------------------------------
Gold N - Erase Data, Restart Screen; PF2 - Help;
Alt W - Sequential Wean; Alt L - Littered Cages; Alt A - All Cages
==================================================================================
```

Purpose:
To identify the equipment to be used in this session.

HELP MESSAGES FOR BMSS DATA ENTRY SCREENS

Screen Help for Identification (A.1)

This screen is used to identify the computer equipment that will be used for this session. You will be prompted to bar code the identification number on each piece of equipment. If the bar code reader is not working, you can type in the numbers.

Field help messages for Identification (A.1)

1. Building and Room:
Barcode the building and room card or type in the building and room to which you what to log on.

2. Terminal ID:
Barcode the terminal identification number

3. Barcode ID:
Barcode the barcode reader identification number

4. Keyboard ID:
Barcode the keyboard identification number

5. Function or Cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for BMSS Report Selection (A.2)

This screen allows you to request ad hoc reports. Enter a Y next to each report that you meant to print. Additional data needed to generate the report, such as the range of dates covered or the experiment to be reported are entered on the line next to the report name. If these data are required for a report that you select, and you do not enter them, an error message will appear.

Target traces may be requested for one or more UIN's or room/cages. That is, you may enter either UIN's or rooms and cages, but not both. You can enter a list of UIN's or room/cages either by specifying each one individually or by entering a starting and an ending value of a range of numbers.

Press Gold 9 when you are finished.

Field help messages for BMSS Report Selection (A.2)

1. Animal Birth:
This report provides a complete list of litters born in the breeding colony during a specified time period.

2. from:
This is the earliest date from which data will be reported.

3. to:
This is the latest date from which data will be reported.

4. Breeder Assignment:
This report provides a list of breeders that were established during a specified time period.

5. Average Number per Litter:
This report provides a summary of new litter size by strain and litter number for litters born during a specified time period.

6. Productivity Index:
This report provides an index of productivity (the average number of animals weaned per week) for each strain during a specified time period.

7. Transfer/Issue Cage Census:
This report provides an inventory of the animals residing in the breeding colony's issue cages by sex and strain.

8. Animal Orders:
This report provides a list of the animals currently ordered for experiment.

9. Pup Weights at Weaning and Disposition:
This report provides the weight and disposition at weaning of pups born to animals in the experimental breeding system under a specific experiment.

10. for Exp#:
This is the experiment and test number for which the data will be reported.

11. Pup Weights:
This report provides the weight at weaning of pups born to animals in the experimental breeding system under a specific experiment.

12. Fertility:
This report provides data on the fertility of animals in the experimental breeding system under a specific experiment during a specified time period.

13. Target Trace:
This report provides a diagram that traces the progeny or ancestors of one or more root animals.

14. for selected or range of (enter S or R):
Press S if you want to enter a list of UIN's or room/cages (may be 1 number). Press R if you want to enter a starting and ending value only.

15. UIN(s):
If you are doing this trace by UIN, enter them here.

16. Room(s):
If you are doing this trace by room and cage, enter the room numbers here.

17. Cage(s):
If you are doing this trace by room and cage, enter the cage numbers here.

Screen Help for Allocation (A.3)

This function allows you to allocate animals from the breeding colony to experiments in the INLIFE system or to allocate animals in breeding experiments to other experiments. The function prompts will lead you through the process you must follow.

Field help messages for Allocation (A.3)

1. How many animal technicians on this allocation:
Enter the total number of technicians working on this allocation; may be from 1 to 4.

1a. Enter operator ID:
If technicians other than the person who signed on to the terminal are assisting with the allocation, enter their operator ID's here, one at a time.

2. Barcode first cage on rack:
Barcode a cage card or type in the cage number of the cage on the upper left corner of the rack. This number will key into the data needed for the allocation.

2a. Enter issue cage:
Barcode a cage card or type in the number of an issue cage containing animals to be allocated.

2b. Weigh the full issue cage:
Put the issue cage on the scale and press the control. If the weight is not transferred to the screen, type it in.

2c. Weigh animal:
Remove an animal from the issue cage and reweigh the cage. If the animal's weight is not transferred to the screen, type it in.

2d. Enter new issue cage:
Barcode a cage card or type in the number of an empty issue cage into which this animal will be placed.

3. Enter starting cage #, Row, Column, or Return:
Press Return to process the whole rack. Enter a starting cage # to select a cage range or R or C to denote that you will select rows or columns.

4a. Enter ending cage:
Enter the last cage in the range of cages that you allocate to. This cage may be on a different rack than the starting cage.

4b. Enter row numbers:
Enter the numbers of the rack rows to which you want to allocate animals.

4b. Enter column numbers:
Enter the numbers of the rack columns to which you want to allocate animals.

5. Enter issue cage:
Barcode a cage card or type in the number of an issue cage containing animals to be allocated.

6. Weigh the full issue cage:
Put the issue cage on the scale and press the control. If the weight is not transferred to the screen, type it in.

7. Weigh <SEX>, Return, End, Removal, or Issue cage:
Select an animal from the issue cage. If it can't be used, press R to remove it; otherwise press W or Return. If the cage is empty, barcode a new issue cage. Press E to end allocation for this session.

7a. Weigh animal:
Remove an animal from the issue cage and reweigh the cage. If the animal's weight is not transferred to the screen, type it in.

7b. Clip, move animal, and verify:
Verify the animal's ear clip or ID and place it in the experiment cage. Press the clip letter or 1 for any other type of ID.

7b. Clip, move animal, and verify (must be <PHENOTYPE>):
Verify the animal's ear clip or ID and phenotype and place it in the experiment cage. Press the clip letter or 1 for any other type of ID.

7d. Enter reason:
Enter D if the animal is dead (or missing). Enter C if the animal is alive but unusable. May be blank if a disposition is to be entered.

7d. Enter disposition:
Enter P for pathology, C for chemistry, or M for microbiology. May be blank if a reason was entered.

7d. Enter CID:
Barcode a CID card or type in a CID if one is needed. Each CID must be used only once.

7d. Select issue cage <9999>; enter issue cage:
The animal must be replaced with one of a similar weight. The animal in this issue cage has been selected to be used by the pre-allocation weighing.

8. Are any animals missing?:
Enter Y if any animals will be replaced at this time.

9a. Enter cage number of missing animal:
Barcode a cage card or type in the number of the cage in which the animal is missing.

9b. Enter clip or animal ID:
Enter N, L, R, B, or 1 to identify the animal.

9c. Enter CID:
Barcode a CID card or type in a CID if one is needed. Each CID must be used only once.

Screen Help for Animal Orders (A.4)

Order Entry:
This screen allows you to enter orders for animals. Fill in the data in each highlighted field. If you want to duplicate an order (and then change its date required or other information), you should enter that experiment number on the next blank line. When you are finished entering orders, press Gold 9 to return to the NARSS menu.

Filling Orders:
This screen allows you to mark the date an order has been filled and the number of animals that were delivered. The number of animals delivered may be different from the number of animals requested. Enter the experiment number and date requested for each order that you want to mark as filled. Enter zero for the experiment number when you are finished.

Displaying Orders:
This function will allow you to display the animal orders that can be filled from the strains that are present in this room. Certain personnel can also display all the orders on the system.

Field help messages for Animal Orders (A.4) - Order entry

1. Experiment and test:
Enter the experiment and test for which you want to enter an order. Press Gold 9 when you are finished entering orders.

2. Date requested:
Enter the date on which these animals are to be delivered to the experiment.

3. Building and room:
Enter the building and room to which these animals are to be delivered.

4. P.I./Contact:
Enter the name of the person to answer questions about this order and/or the principal investigator.

5. Strain:
Enter the 2-digit strain code of the animals ordered.

6. Sex:
Enter the sex (M, F, U) of the animals ordered.

7. Number required:
Enter the number of animals ordered that have the characteristics entered on this line.

8. Special requirements:
Enter any additional requirements (phenotype, age, etc.) on the animals ordered.

Field help messages for Animal Orders (A.4) - Fill Orders

1. Enter experiment and test:
Enter the experiment and test for which you want to fill an order. Press Gold 9 when you are finished filling orders.

2. Enter date requested:
Enter the date on which these animals were to be delivered to the experiment.

3. Enter the number delivered:
Enter the number of animals actually delivered on this order.

4. Enter date filled:
Enter the date on which these animals actually are delivered.

5. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

6. Are all data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Field help messages for Animal Orders (A.4) - Display Orders

1. Orders for this room only or whole colony:
Enter R to display only those orders that can be filled from the strains present in this room. Some operators may enter C to display all orders.

2. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Retirement Candidates (A.5)

This screen shows all the breeder pairs in this room that are at or past the retirement age for that animal's strain. If more that one screen full of information is present, use the PgDn and PgUp keys to see it.
Field help messages for Retirement Candidates (A.5)

1. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Unproductive Breeders (A.6)

This screen shows all the breeder pairs in this room that have not produced a litter for the number of days set for that strain. If more that one screen full of information is present, use the PgDn and PgUp keys to see it.

Field help messages for Unproductive Breeders (A.6)

1. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Feeders by Age Range (A.7)

This screen shows all the feeders assigned to this room that are at or between the ages entered. If more that one screen full of information is present, use the PgDn and PgUp keys to see it.

Field help messages for Feeders by Age Range (A.7)

1. Enter minimum age (days):
Enter the minimum feeder age in days to be displayed.

2. Enter Maximum age (days):
Enter the maximum feeder age in days to be displayed.

3. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Issue/Feeders by Age Range (A.8)

This screen shows all the feeders assigned to issue cages that are at or between the ages entered. If more that one screen full of information is present, use the PgDn and PgUp keys to see it.

Field help messages for Issue/Feeders by Age Range (A.8)

1. Enter minimum age (days):
Enter the minimum feeder age in days to be displayed.

2. Enter Maximum age (days):
Enter the maximum feeder age in days to be displayed.

3. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Work Room Summary (A.9)

This function allows you to display a summary of the work that was performed in this room on this day. It also includes all the animals issued on the system on this day whose strains are present in this room. Certain personnel can use this function to display a summary of all the work performed in the breeding colony on this day. If more that one screen full of information is present, use the PgDn and PgUp keys to see it.

Field help messages for Work Room Summary (A.9)

1. Summary for this Room or whole Colony:
Enter R to display only work done in this room. Some operators may enter C to display all work done in the colony.

2. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for New Animal Entry (A.10)

This screen allows you to introduce new (purchased) animals into the breeding colony. All animals that are introduced into one cage can be entered on one screen. Enter a blank for 'Sex' and the 'Cntl Z' out of the Notes field to signal that you are finished entering new animals to this cage. You can then perform other actions on this cage, or enter new animals to a different cage.

Field help messages for New Animal Entry (A.10)

1. Sex or litter:
Enter the sex (M, F, U) of the adult animals entered or L if a litter is being entered.

2. Status:
Enter the initial status these animals will have in the breeding colony. The status must be compatible with other animals in the cage and its type.

2a. Reason for issue:
Enter E for issue-experiment or F for issue-feeder.

2b. Experiment #:
Enter the experiment to which the animal is being issued.

3. Date of Birth:
Enter the date that the animal was born as (day month year).

4. Strain:
Enter the 2-digit strain code of this animal.

5a. Phenotype:
For strains 29, 37, or 46: black, yellow, or agouti; 84: haired or nude.

5b. Phenotype must be X or X; enter phenotype:
The phenotype of the second member of a pair must cross that of the original animal in the pair.

6. Count:
Enter the number of animals of this type.

7. Generation:
Enter the number of the generation of these animals or 0 if you don't know what it is.

8. Vendors ID:
If you need to keep track of the animal's identifier as supplied by the vendor, enter it here.

9. Notes:
Enter any other textual information pertaining to these new animals.

10. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

11. Are all the data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Change Cage Status (A.11)

This screen allows you to change a breeder cage to a feeder cage or a feeder cage to a breeder cage. You cannot change a breeder cage to a feeder cage if animals of both sexes are present. You also cannot change a breeder or feeder cage to an issue cage or vice versa.

Field help messages for Change Cage Status (A.11)

1. Enter new status:
Press B to change a feeder cage to a breeder cage. Press F to change a breeder cage to a feeder cage.

2. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Move Cage (A.12)

This screen allows you to change the positions of the breeder and feeder cages in a room. If you move a breeder cage, a new breeder card will be printed.

Field help messages for Move Cage (A.12)

1. Enter new cage number:
Barcode a cage card or type in the number of the cage that is currently in the position that this cage will be moved to.

2. Switch these two cages:
Enter Y if these two cages will switch positions. Enter N if the second cage will be moved someplace else.

3. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Display Cage (A.13)

This function displays the census of the cage number you have entered.

Field help messages for Display Cage (A.13)

1. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

Screen Help for Assignment (A.14)

This screen allows you to move animals from any cage to a breeder or feeder cage and change the status of the animals in the process. You can use this screen to set up breeder pairs, assign feeders, wean litter animals, etc.

Field help messages for Assignment (A.14)

1. Enter status:
Enter the current status of the animal to be given a new assignment, may be B, F, I, or L. Enter a blank when done assigning from this cage.

2. Enter sex:
Enter the sex of the animal to be given a new assignment, may be M, F, U, or B (to assign pairs).

3a. Enter phenotype:
Enter the phenotype of the animal to be given a new assignment, may be B - black, Y - yellow, A - agouti, H - haired, or N - nude.

3b. Phenotype must be X or X; Enter phenotype:
The phenotype of the second member of a pair must cross that of the original animal in the pair.

4. Enter count:
Enter the number of animals (or pairs) of this type to be assigned.

5. Enter new status:
Enter the animal's new status, may be B - breeder or F - feeder.

6. Enter new cage:
Barcode a cage card or type in the new cage that this animal will be placed in.

7. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

8. Are all the data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Removal (A.15)

This screen allows you to remove animals from the breeding colony that are not going to experiments. Dead, culled, or surplus animals are noted using this screen. Breeder retirement is a separate function.

Field help messages for Removal (A.15)

1. Enter status:
Enter the current status of the animal to be removed, may be B, F, L, or I. Enter a blank when done removing animals from this cage.

2. Enter sex:
Enter the sex of the animal to be removed, may be M, F, B (both), or U.

3. Enter phenotype:
Enter the phenotype of the animal to be removed, may be B - black, Y - yellow, A - agouti, H - haired, or N - nude.

4. Enter count:
Enter the number of animals of this type to be removed. If sex is B - both, this is the number of pairs removed.

5. Enter reason:
Enter D if the animal is dead (or missing). Enter C if the animal is alive but unusable. May be blank if a disposition is to be entered.

6. Enter disposition:
Enter P for pathology, C for chemistry, or M for microbiology. May be blank if a reason was entered.

7. Enter CID:
Barcode a CID card or type in a CID if one is needed. Each CID must be used only once.

8. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

9. Are all the data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Retire (A.16)

This screen allows you to retire breeder animals. Either a single cage or a sequence of cages may be retired. Either one or both animals in the cage may be retired.

Field help messages for Retire (A.16)

1. Sequential or Individual:
Enter I if you want to retire an individual cage. Enter S if you want to retire from a sequence of cage numbers.

2a. Enter first cage to retire:
Barcode a cage card or type in the number of the first cage that you want to retire. Enter 0 when finished retiring cages.

2a. Enter last cage to retire:
Barcode a cage card or type in the number of the last cage that you want to retire.

3. Enter sex:
Enter the sex of the animal that you want to retire, may be M, F, or B.

4. Enter count:
This is the maximum number of harem mated females that will be retired from any one cage.

5. Enter disposition:
Enter P for pathology, C for chemistry, or M for microbiology. May be blank.

6. Enter CID:
Barcode a CID card or type in a CID if one is needed. Each CID must be used only once.

7. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

8. Are all data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Pregnant (A.17)

This function allows you to note that a female is pregnant or that a plug has been observed.

Field help messages for Pregnant (A.17)

1. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

2. Are all data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Littered (A.18)

This function allows you to enter the number of animals counted in a new litter.

Field help messages for Littered (A.18)

1. Enter number of pups:
Enter the number of pups counted in this litter.

2. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

3. Are all data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Issue (A.19)

This screen allows you to place animals into issue cages. Animals may be issued without using a destination cage using 'issue other'.

Field help messages for Issue (A.19)

1. Enter status:
Enter the current status of the animal to be issued, may be B, F, L, or I. Enter a blank when done issuing animals from this cage.

2. Enter sex:
Enter the sex of the animal to be issued, may be M, F, or U.

3. Enter phenotype:
Enter the phenotype of the animal to be issued, may be B - black, Y - yellow, A - agouti, H - haired, or N - nude.

4. Enter count:
Enter the number of animals of this type to be issued.

5. Enter reason for issue:
Enter E - issue experiment, F - issue feeder, or O - issue other.

6. Enter new cage number:
Barcode a cage card or type in the number of the issue cage to which this animal will be issued.

7. Enter experiment number:
Enter the experiment-test to which this animal is being issued.

8. Enter CAN:
Enter the common account number under which this animal is being issued.

9. Enter function or cage:
Press a BMSS function key or barcode a cage card or type a cage number.

10. Are all data on this screen correct:
Enter Y to go on to the function or cage entered. Enter N to go back to correct data that was just entered here.

Screen Help for Pup Weighing (A.21)

To allow weighing of any number of pups in a single cage.

Field Help Messages for Pup Weighing (A.21)

1. Number Weighed: This is the number of animals to be weighed. Press enter without a number, if a new cage is to be weighed.

2. Weigh the full cage: Weigh the entire cage of animals corresponding to Number Weighed. Put the cage on the scale and press the control. If the weight is not transferred to the screen, type it in.

3. Weigh Animal(s): Remove the animals from the cage and reweigh the cage. If the animals weight is not transferred to the screen, type it in.

4. Units: If weighing units (for example, grams) are not transferred to the screen, type them in.

5. Enter a function or cage: Press a BMSS function key, bar-code a cage card, or type a cage number.

6. Are all data on this screen correct: Enter "Y" to go on to the cage or function entered. Enter "N" to go back and correct the data that were just entered here.

APPENDIX B
(Revisions to the External Design Document of 9/18/88)

REPORTS
PART B.1. Description of the Reports
PART B.2. Proposed Report Physical Layouts

NOTE: Parts with revisions are in bold type. Only revised sheets are included.

PART B.1. Description of the Reports
(Revisions to the External Design Document of 9/18/88)

B.1. DESCRIPTION OF BMSS REPORTS

Currently, output reports are generated by the VAX computer and transmitted to a high speed line printer. Reports or copies of printed reports are then reduced onto standard 8 1/2 x 11 inch paper and distributed to NCTR breeding colony managers and the experimental breeding staff. It has been suggeste suggested (Section 4.1, Page 39 of the External Design document/19 August 1988) that reports be transmitted to distributed laser printers and compressed print be used as necessary to improve report quality.

A recap is prepared daily to report all assignment, observation and removal transactions for the day. Weekly reports provide a synopsis of the more detailed information from these daily transactions, including summaries of the week's activity. Certain reports have been designated as "ad hoc" meaning that they fulfill a specific purpose and are generally issued upon request rather than on a specific date. Each report is described separately in this section. Sample physical layouts of each report are found in Section B.2.

1. Daily Recap Report - Assignments (BS-001)

This report provides a list of all animal assignments and issues made during the day. A distinction is made between assignments and issues. If an animal's status is changed such that it remains in the same room, it is considered an "assignment." This would include animals assigned as breeders or feeders. On the other hand, if an animal is assigned a standby status inside or outside of the room, it is considered an "issue." This would include animals issued to experiment or as feeders.

This report is issued daily. A section is prepared for each animal strain. Section detail is in sequence by source cage number. A summary is prepared as the last page of each section. Report detail is described below and a sample report is found in Section B.2.

Sort Sequence: Previous Cage Number

Frequency: Daily

Destination or Printer No.: To Be Announced (TBA)

a. Building room and NCTR strain code are identified on the title lines of each page. This issue is identified by the Gregorian date displayed below the report title.

b. A detail line is prepared for each assignment as follows:

(1) OPER - Operator: Number identifying the animal technician who performed this action.

(2) PREV. STATUS: Former status of the animal.

(3) UIN/COUNT: Strain code and unique identification number of the animal that was assigned or issued. "COUNT", if more than one, indicates either a pair of breeders or the number of animals treated as a block. For a block assignment this UIN is the first in a sequential series of UIN's. For a breeder pair this UIN is for the female.

(4) SEX: Sex of the animal indicated as:

F = Female, M = Male, or U = Unsexed

(5) BREEDING ANIMAL ID: Code that indicates the earclip or an optional code to distinguish females in harem mating. An earclip different from "N" could only apply to a single animal assignment.

(6) GENR - Generation: The generation number of this animal, breeder pair, or block of animals.

(7) PREV. CAGE: The number identifying the previous cage that this animal was assigned.

(8) CURRENT CAGE: Number identifying the cage to which this animal will be moved.

(9) CURRENT STATUS: Current status of the animal including final action (e.g., breeder, feeder, issue experiment, issue feeder, issue other, cull, dead, etc).

(10) CAN NO - Common Account Number: NCTR division account number assigned to animals that are issued to an experiment.

(11) EXP - Experiment: The experiment number and test the animal is issued to.

(12) MATE UIN: Unique identification number for the male animal assigned to this female breeder.

c. A summary page is prepared for each strain after the last detail page for that strain. Totals are displayed for the total number of male and female breeders, total number of male, female and unsexed feeders, and the total number of male, female and unsexed animals for all issues.

2. <u>Daily Recap Report - Pregnant, Plugged, Litter, and Newborn Observations (BS-002o)</u>

This report provides a list of pregnant, plugged, litter, and newborn observations recorded during the day.

This report is issued daily. A section is prepared for each animal strain. Section detail is in sequence by bldg-room, strain code, source cage, and unique identification number. A summary is prepared as the last page of each section. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Bldg-Room, Strain Code, Cage, and Unique Identification Number

Frequency: Daily

Destination and Printer No:

a. Building Room and NCTR strain code are identified on the title lines of each page. This report is identified by the Gregorian date displayed below the report title.

b. a detail line is prepared for each observation as follows:

(1) OPER - Operator: Number identifying the animal technician who performed this action.

(2) STATUS: The type of role currently assigned to this animal(s) such as breeder, feeder, or block animal.

(3) UIN/COUNT: Strain code and unique identification number of the animal or block to which this observation applies. If this is a "littered" observation the number of animals in the litter is indicated by "COUNT" which indicates the number of animals from the same cage that were observed.

(4) SEX: Sex of the animal indicated as:

F = Female, M = Male, or U = Unsexed

(5) BREEDING ANIMAL ID: Code that indicates the earclip or an optional code to distinguish females in harem mating. An earclip different from "N" could only apply to a single animal assignment.

(6) CAGE: Number identifying the cage for which this observation was made.

(7) OBSERVATIONS: A descriptive phrase for the condition observed.

(8) LITTER UIN: Strain code and unique identification number assigned to the new litter by the system. Only present for littered observations.

(9) LITTER-GEN: The placement of this litter within the succession of generations of this strain as developed at the NCTR breeder colony.

c. A summary page is prepared for each strain after the last detail page for that strain. Totals are displayed for the following observations:

Littered
Pregnant
Plugged
New Born Animals

3. <u>Daily Recap Report - Phenotype Observations (BS-002p)</u>

This report provides a list of all phenotype observations recorded during the day. These observations include normal conditions such as littered, yellow, black, haired and nude.

This report is issued daily. A section is prepared for each animal strain. Section detail is in sequence by bldg-room, strain code, source cage, and unique identification number. A summary is prepared as the last page of each section. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Bldg-Room, Strain Code, Cage, and Unique Identification Number

Frequency: Daily

Destination and Printer No:

a. Building Room and NCTR strain code are identified on the title lines of each page. This report is identified by the Gregorian date displayed below the report title.

b. a detail line is prepared for each observation as follows:

(1) OPER - Operator: Number identifying the animal technician who performed this action.

(2) STATUS: The type of role currently assigned to this animal(s) such as breeder, feeder, or block animal.

(3) UIN/COUNT: Strain code and unique identification number of the animal or block to which this observation applies. If this is a "littered" observation the number of animals in the litter is indicated by "COUNT" which indicates the number of animals from the same cage that were observed.

(4) SEX: Sex of the animal indicated as:

F = Female, M = Male, or U = Unsexed

(5) BREEDING ANIMAL ID: Code that indicates the earclip or an optional code to distinguish females in harem mating. An earclip different from "N" could only apply to a single animal assignment.

(6) PHENOTYPE: Visible properties of an animal produced by the interaction of the genotype and the environment.

(7) CAGE: Number identifying the cage for which this observation was made.

(8) OBSERVATIONS: A descriptive phrase for the condition observed.

(9) LITTER UIN: Strain code and unique identification number assigned to the new litter by the system. Only present for littered observations.

(10) LITTER-GEN: The placement of this litter within the succession of generations of this strain as developed at the NCTR breeder colony.

c. A summary page is prepared for each strain after the last detail page for that strain. Totals are displayed for the following observations:

Males Yellow
Females Yellow
Males Black
Females Black
Males Agouti
Females Agouti
Males Hair
Females Hair
Males Nude
Females Nude

4. <u>Daily Recap Report - Removals (BS-003)</u>

This report provides a list of all animals removed from the breeding colony during the day. Removals may be for any of the following reasons: dead, culled, surplused, retired, or for analysis by Micro, Chemistry or Pathology. A removal applies to one or more animals from the same cage, homogeneously identified, and removed for the same reason.

This report is issued daily. A section is prepared for each animal strain. Section detail is in sequence by bldg-room, strain code, source cage, and unique identification number. A summary is prepared as the last page of each section. Report detail is described below, and a sample report is in Section B.2.

Sort Sequence: Bldg-Room, Strain Code, Cage, and Unique Identification Number

Frequency: Daily

Destination or Printer No.: To Be Announced (TBA)

a. Building Room and NCTR strain code are identified on the title lines of each page. This issue is identified by the gregorian date displayed below the report title.

b. A detail line is prepared for each removal as follows:

(1) OPER - Operator: Number identifying the animal technician who made this removal.

(2) CURRENT STATUS: The assignment this animal was dedicated to prior to removal. See BS-001 for full definition.

(3) UIN/COUNT: Strain code and unique identification number of the animal that was removed. If more than one animal was removed, as indicated by "COUNT", this UIN is the first in a sequential series for the block of animals that were removed.

(4) SEX: Sex of the animal indicated as:

F = Female, M = Male, or U = Unsexed

(5) BREEDING ANIMAL ID: Code that indicates the earclip or an optional code to distinguish females in harem mating. An earclip different from "N" could only apply to a single animal assignment.

(6) CAGE: The number identifying the cage from which this animal was removed.

(7) REMOVAL REASON: The reason for the removal. Can also be "issue other" in which case the animal goes to an experiment for no further accountability by the breeding information system.

(8) REMOVAL DISPOSITION: The disposition of the animal only if sent to micro, chemistry, or pathology.

(9) EXPERIMENT: The experiment number and the test an animal is allocated to.

(10) CARCASS-ID: Carcass Identification (CID) number assigned to an animal that was sent to Micro, Chemistry or Pathology.

c. A summary page is prepared for each strain after the last detail page for that strain. Totals are displayed for the following categories:

Total number of each, male, female and unsexed animals culled.

Total number of each male, female and unsexed surplus animals.

Total number of each male, female animals retired.

Total number of each male, female and unsexed animals that were dead.

Total number of each male, female and unsexed animals that were sent to Pathology, Micro and Chemistry.

Total number of each male, female and unsexed animals that were removed as issue other.

5. **Female Breeder Card and Male Breeder Card (BS-004)**

Two cards are prepared for each breeder pair listed on the Breeder Assignment Report at the time established as breeders. Each set of cards is attached to a breeder cage and is used to log the birth and weaning of litters produced by that breeder pair and serves as a quick visual check for that cage. This information is also recorded on the data terminal.

Cards are prepared in cage number sequence. The information contained on each card is described below and a sample pair of cards is shown in Section b.2.

Sort Sequence: Source Cage Number

Frequency: Variable - At Time of Breeder Assignment

Destination or Printer No.: To Be Announced (TBA)

a. The top half of each card contains the same type of information as follows:

(1) UIN: Unique identification number for this breeder.

(2) MATED: Gregorian date that this breeder was assigned to its mate.

(3) STRAIN: Strain code for this breeder.

(4) GENERATION: The placement of this breeder within the succession of generations of this strain.

(5) SOURCE CAGE NO.: Number identifying the cage from which this breeder animal was taken.

(6) DOB FOR UIN: Birthdate of this breeder.

(7) BUILDING-ROOM NO.: Number identifying the breeder building and room in which this animal is located.

(8) The space in the center of the card contains the cage number for the cage assigned to this breeder pair.

b. The lower half of each card provides space for the animal technician to log in the birth of litters. This portion on the Male Breeder Card is not currently used. Information is logged on the Male and Female Breeder Card as follows:

(1) DATE FOUND PREG.: Date the female was found to be pregnant. Normally not used.

(2) DATE LITTER BORN: Gregorian date that the new litter will be recorded on the terminal. This may not be the actual date of birth due to weekend births or work schedules.

(3) NUMBER OF LITTER: Total number of animals born.

(4) NUMBER WEANED: Total number of surviving litter pups that were weaned.

(5) REMARKS: Normally used to record the number of males, females and dead animals, in that order.

(6) RETURN DATE: Date the animal is returned to the cage.

6. <u>Animal Birth Report (BS-005)</u>

This report provides a complete list of litters born in the breeding colony which may include more than one breeding room during a specified time period. Each litter is identified by its strain, unique identification number, litter size and litter number (rank), and generation number. Identity of the parents and cage location is also included. Totals are provided for all animals born of a specific strain and date and a total for that date. A summary is prepared as the final report page.

This report is issued weekly but can be requested for a longer period. It is prepared in sequence by birth date, room number, litter strain, and cage number. Report detail is described below and a sample report follows the description. This issue includes all animals born during the period specified by the dates displayed on the report title line.

Sort Sequence: Birth Date, Cage No., Bldg-Room, Litter Strain, and Cage Number

Frequency: Weekly, Ad hoc

Destination or Printer No.: To Be Announced (TBA)

a. A group of detail lines is prepared for all litters born of a specific strain, in a specific room and on a specific date. A line is prepared for each litter, however, birth date and wean date appear only on the first line of a group.

(1) BIRTH DATE: Gregorian date that the litters in this group were born.

(2) WEAN DATE: Gregorian date that the litters in this group are to be separated from their mothers which is determined by the species and strain and taken from the species and strain table.

(3) DATE ESTABLISHED: Gregorian date that a breeding relationship was established for the parents of this litter.

(4) CURRENT BUILDING-ROOM-CAGE: Current location of the cage containing the new litter.

(5) SIRE STRAIN-UIN: Strain code and unique identification number assigned to the father at birth.

(6) DAM STRAIN-UIN: Strain code and unique identification number assigned to the mother at her birth.

(7) LITTER STRAIN-UIN: Strain code and unique identification number assigned to the litter. This is the first number in a series (block) of numbers reserved for this litter. As each animal is weaned, or otherwise removed, it is assigned a number from this block.

(8) LITTER NUMBER: A chronological placement of this litter among other litters born to the same breeder pair.

(9) LITTER GENERATION: The placement of this litter within the succession of generations of this strain as developed at the NCTR breeder colony.

(10) LITTER SIZE: Total number of animals born in this litter.

b. A group total line is prepared after the last detail line for each group. The total number of animals born to this group is displayed, as indicated by strain number and room number.

c. A total line is prepared following the last group total line for the same birth date. It is identified by the phrase "TOTAL BORN THIS DATE" followed by the total number of animals.

d. RECAP OF BIRTHS: This is a report summary prepared at the end of the report. A summary line is prepared for each strain as follows:

(1) NCTR STRAIN CODE: Two character code identifying the strain.

(2) TOTAL LITTER: Total number of litters born of this strain during this period.

(3) TOTAL BORN: Total number of animals born of this strain during this period.

(4) AVERAGE LITTER SIZE: Average number of animals per litter for this strain during the time period.

(5) PROJECTED NO. TO BE WEANED: Two totals are displayed for the projected number of animals to be weaned - 90% and 95% of the total number of animals born of this strain.

7. Weaned Animal Report (BS-006)

This report provides a complete list of litters weaned during a specified time period. Each litter weaned is accounted for by its strain, unique identification number, litter size, and the disposition of its members. Identity of the parents and the cage location is also included. The total number of litters weaned is provided for each strain weaned on a specified date and for all litters weaned on that date. A summary is prepared as the final report page.

This report is issued weekly and is prepared in sequence by wean date, room number, litter strain and cage number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Wean Date, Room Number, Litter Strain, and Cage Number

Frequency: Weekly

Destination or Printer No.: To Be Announced (TBA)

This report includes all litters that were weaned during the period specified by the dates on the second line of each page.

a. A group of detail lines is prepared for all litters weaned on the same date and of the same strain and room location. The wean date and birth date appear only on the first detail line of a group and also on the first detail line of a new page when the group is continued. Detail is as follows:

(1) WEAN DATE: Gregorian date that the litters in this group were weaned.

(2) BIRTH DATE: Gregorian date that the litters in this group were born.

(3) CURRENT BUILDING-ROOM-CAGE: Room location of the cage that this litter was born in, and the cage number.

(4) SIRE UIN: Strain code and unique identification number assigned to the father at his birth.

(5) DAM UIN: Strain code and unique identification number assigned to the number at her birth.

(6) LITTER UIN: Strain code and unique identification number assigned to the litter at its birth.

(7) LITTER SIZE: Total number of animals in this litter.

(8) DEAD: Number of animals in this litter that were found dean.

(9) CULLED: Number of animals in this litter that were destroyed because of injury or deformity.

(10) SURPLUS: Number of animals in this litter that, although healthy, were removed because of no demand for them.

(11) SURVEY: Number of animals in this litter that were requested for a Micro, Chemistry, or Pathology examination.

(12) ISSUE: Number of animals in this litter that were issued to an experiment, feeder, or other.

(13) ASSIGN BREEDER: Number of animals that were established as breeders

(14) ASSIGN FEEDER: Number of animals established as feeders.

(15) GENERATION: The placement of this litter within the succession of generations of this strain raised at the NCTR breeding colony.

(16) ISSUE: Number of animals on "standby" for future assignment.

b. A group total line is prepared after the last detail line for the group. The total number of litters weaned is displayed for that strain number and room number.

c. A total line for all litters weaned on the same date is prepared following the last group total line for that date. The total number of animals for each detail column is displayed.

d. A summary is prepared on the final report page, identified by the heading "RECAP OF WEANED ANIMALS." A line is prepared for each strain as follows:

(1) NCTR STRAIN CODE: Strain code to which summary detail applies.

(2) TOTAL LITTERS: Total number of litters weaned of this strain.

(3) TOTAL BORN: Total number of animals weaned of this strain.

(4) AVG. NUMBER PER LITTER: Average number of animals per litter for this strain.

(5) DEAD: Total number of animals of this strain found dead and the percent.

(6) CULLED: Total number of animals culled of this strain and the percent.

(7) SURPLUS: Total number of surplus animals of this strain and the percent

(8) SURVEY: Total number of animals of this strain issued to Micro, Chemistry, or Pathology and the percent.

(9) ASSIGN BREEDER: Total number of animals of this strain assigned as breeders and the percent.

(10) ASSIGN FEEDER: Total number of animals of this strain assigned as feeders and the percent.

(10) ISSUE: Total number of animals of this strain issued to experiment, feeder, or other and the percent.

8. Breeder Assignment Report (BS-007)

This report provides a list of breeders that were established during a specified time period. The source of each individual breeder and the location of established pairs are provided. The total number of breeders is displayed for each strain and room, and for each established date. A summary recap displays the total number of assignments for each strain.

This report is issued weekly. It is prepared in sequence by established date, room number, strain and current cage number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Established Date, Room Number, Strain, and Current Cage Number.

Frequency: Weekly, Ad hoc

Destination or Printer No.: To Be Announced (TBA)

This report includes all animals assigned as breeders during the period specified by the dates displayed on the report title line.

a. A group of detail lines is prepared for all breeders established on a specific date and room. A line is prepared for each established pair, however, the established date appears only on the first line of a group.

(1) ESTABLISH DATE: Gregorian date that the breeders in this group were established as pairs.

(2) CURRENT CAGE: Identification number for the cage occupied by this breeder pair.

(3) CURRENT BUILDING ROOM: Number identifying the room in which this breeder pair is located.

(4) FEMALE: Individual information provided for the female member of the pair is as follows:

(a) BIRTH DATE: Gregorian date on which this animal was born.

(b) UIN: Strain code and unique identification number assigned to this animal at birth.

(c) PREV. CAGE: Identification number of the cage that contained this animal prior to this assignment.

(d) FROM BUILDING-ROOM: Number identifying the building and room this animal was in prior to this assignment.

(5) MALE: Individual information provided for the male member of the pair is as follows:

(a) BIRTH DATE: Gregorian date on which this animal was born.

(b) UIN: Strain code and unique identification number assigned to this animal at birth.

(c) PREV. CAGE: Identification number of the cage that contained this animal prior to this assignment.

(d) FROM BUILDING-ROOM: Number identifying the room this animal was in prior to this assignment.

b. A room total line is prepared after the last detail line for that room. The total number of animals assigned as breeders in this room, on this established date, is displayed.

c. A total line is prepared following the last room total line for the same established date. It is identified by the phrase "TOTAL THIS DATE", followed by the total number of animals assigned as breeders on this date.

d. BREEDER ASSIGNMENT RECAP: A separate page is prepared for the total number of animals, by strain, assigned as breeders.

(1) A total line is prepared for each strain as follows:

(a) STRAIN: The strain code to which this total applies.

(b) TOTAL: Total number of animals of this strain assigned as breeders during this period.

(2) A grand total line displays the total number of animals assigned as breeders during this period and is identified by the phrase "TOTAL THIS PERIOD."

9. Active Breeder Report (BS-008)

This report provides a complete list of active breeder pairs in the colony. It is used for taking inventory and also as a reference to obtain a pedigree trace on the female UIN.

This report is issued monthly and is in sequence by room number and cage number. This detail is described below and a sample report is in Section B.2.

Sort Sequence: Room Number, Cage Number

Frequency: Monthly

Destination or Printer No.: To Be Announced (TBA)

a. A detail line is prepared for each breeder pair as follows:

(1) CAGE: Number identifying the cage occupied by this breeder pair.

(2) BUILDING-ROOM: The number identifying the room where this cage is located. A group of detail lines is prepared for all cages in the same room. Therefore, the room number appears only on the first detail line.

(3) UIN: Strain code and unique identification number assigned to the animal at birth.

(4) SEX: Sex of the animal.

(5) GENERATION: The placement of this animal within the succession of generations of this strain raised at the NCTR breeding colony.

(6) DATE ESTABLISHED: Gregorian date on which this pair was established as breeders.

b. A building room total line is prepared following the last detail line for that room. The total number of breeder pairs in the room is displayed, followed by the phrase "TOTAL ACTIVE BREEDER PAIRS BUILDING ROOM", and the building room number.

10. Average Litter Size (BS-009)

This report provides a litter size analysis of all new litters, summarizing litters by animal strain and litter number. All litters reported on the Animal Birth Report are included on this report.

This report is issued weekly or as needed and features a separate page for each strain. Report detail is in sequence by strain and litter number. This detail is described below and a sample report is in Section B.2.

Sort Sequence: Strain Code and Litter Number

Frequency: Weekly, Ad hoc

Destination or Printer No.: To Be Announced (TBA)

This report includes all litters born during the period specified by the dates displayed on the report title line.

A summary line is prepared for each strain and litter number having births during this period.

a. NCTR STRAIN CODE: Strain code that applies to the litter numbers summarized on this page. This code is displayed only on the first summary line of a page.

b. LITTER NUMBER: The plurality of all litters, that, if taken individually, would be of the same ranking or chronological position within its family of litters.

c. NUMBER BORN: The total number of animals born of this litter number.

d. NUMBER LITTERS: The total number of litters born of this litter number.

e. AVERAGE NO. PER LITTER: The average number of animals per litter for this litter number.

11. Unproductive Breeder Report (BS-010)

This report provides a list of all breeder pairs the have been determined to be unproductive, as a result of not having a litter within the past 75 days or longer. Animals from this list may be disposed of at the discretion of management. The unproductive status of each pair listed is justified with dates. Pairs are listed by strain and room. Totals are displayed for the number of pairs in a room and the number of pairs by strain in a room. A summary prepared on the last report page displays the total number of unproductive pairs for each strain and a grand total.

This report is issued weekly and is in sequence by room number, female strain and cage number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Bldg-Room, Female Strain, and Cage Number

Frequency: Weekly

Destination or Printer No.: To Be Announced (TBA)

a. A group of detail lines is prepared for the unproductive pairs of the same female strain and room number as follows:

(1) BLDG-ROOM: Number identifying the building and room in which the breeder pair is located. The room number appears only on the first line of a group.

(2) CAGE: Number identifying the cage in which this breeder pair is kept.

(3) FEMALE UIN: Strain code and the unique identification number assigned to this female at birth.

(4) MALE UIN: Strain code and the unique identification number assigned to this male at birth.

(5) ESTABLISH

(a) DATE: Gregorian date that this breeder pair was established as mates.

(b) DAYS: Number of days that have elapsed since this breeder pair was established.

(6) LAST LITTER

(a) DATE: Gregorian date that the last litter was born to this breeder pair. Zeroes indicate total unproductivity.

(b) DAYS: Elapsed time in days since the last litter was born to this breeder pair. Zeroes indicate totally unproductive.

(c) LITTER NO.: The number of litters this breeder pair has produces.

(7) NEXT TO LAST DATE GAVE BIRTH: Gregorian date that the next to last litter was born to this breeder pair.

b. A strain total line is prepared after the last detail line for a group. It is identified by the phrase "TOTAL PAIRS FEMALE STRAIN" and followed by the strain number and total number of unproductive pairs of this strain.

c. A room total line is prepared after the last strain total line for the same room. It is identified by the phrase TOTAL PAIRS ROOM" and followed by the room number and total number of unproductive pairs in this room.

d. A separate page is prepared as a summary. It is identified by the heading "RECAP OF UNPRODUCTIVE BREEDER PAIRS". Totals are prepared as follows:

(1) STRAIN: The female strain code to which this total applies.

(2) PAIRS: Total number of unproductive breeder pairs for this strain.

(3) A grand total of all unproductive breeder pairs is displayed after the last strain total.

12. Retirement Candidates (BS-011)

This report lists all breeders that are eligible for retirement. Eligibility is established once a breeder has been on breeder status for a period of time predetermined for that species and strain. Once a breeder becomes a candidate, it will appear on subsequent issues of this report until it is actually retired.

This report is issued weekly and is in sequence by room number, strain code and cage number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Bldg-Room Number, Strain Code, and Cage Number

Frequency: Weekly

Destination or Printer No.: To Be Announced (TBA)

a. A group of detail lines is prepared for candidates of the strain and from the same room as follows:

(1) BLDG-ROOM: Number identifying the room in which this group of candidates is located. Room number appears only on the first detail line of a group.

(2) CAGE: Number identifying the cage in which this candidate is located.

(3) FEMALE UIN: Strain code and the unique identification number assigned to this female at birth.

(4) MALE UIN: Strain code and the unique identification number assigned to this male at birth.

(5) ESTABLISH:

DATE: Gregorian date that this animal was established as a breeder.

DAYS: Total number of elapsed days that this animal has been a breeder.

(6) FEMALE DATE OF BIRTH:

(a) DATE: Gregorian date that this animal was born.

(b) DAYS: Age of this animal in days.

13. Daily Allocations (BS-012)

This report lists all animals that were allocated to experiments during the day. It is prepared in sequence by experiment, room, rack, rack side and cage number. A new page is started for each rack side. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Experiment, Room, Rack, Rack Side, and Cage Number

Frequency: Daily

Destination or Printer No.: To Be Announced (TBA)

This report includes all animals allocated on the date displayed beneath the report title. The experiment, room, rack and rack side are identified in the heading of the report.

a. Detail information is prepared for each cage, with three cages to a line. Cage number sequence is from left to right. Information for each cage is as follows:

(1) OPER: Number identifying the animal technician who performed this action.

(2) EXPERIMENT: The experiment number and the test an animal is allocated to.

(3) BLDG-ROOM: Number identifying the building and room where the allocated animals will reside.

(4) RACK: The rack the animal cage resides on.

(5) SIDE: The side of the rack (A or B) the animal cage is on.

(6) CAGE: Number identifying the cage in the experiment animal room to which these animals were allocated. Cage number is shown only on the first line of detail pertaining to that cage.

(7) TREATMENT: Treatment number uniquely identifying the treatment within tests.

(8) ANIMAL START DATE: The date the animal was assigned to the treatment.

(9) BREEDING ANIMAL ID: Code that indicates the earclip or an optional code to distinguish females in harem mating. An earclip different from "N" could only apply to a single animal assignment.

(10) SEX: Sex code indicating the sex of this animal as:

F = Female, M = Male, or U = Unsexed

(11) UIN: Strain code and the unique identification number identifying the

animal allocated.

(12) ALLOCATION DATE: The date an animal is allocated to an experiment (part of the header information).

(13) ALLOCATION WEIGHT: The weight of the animal at the time of allocation.

(14) UNITS: In grams or kilograms.

(15) ISSUE CAGE: Cage where "issue" animal (animal on standby) resides.

(16) REPLACEMENT ANIMAL: Strain code and the unique identification number identifying the animal used to replace an allocated animal that is missing or dead.

(17) CARCASS-ID: Carcass Identification (CID) number assigned to an animal that was sent to Micro, Chemistry or Pathology.

b. A summary page is prepared as the last report page for each experiment. The experiment number is identified in the heading. The following totals are displayed:

(1) MALE ALLOCATIONS: Total number of male animals allocated to this experiment on the date of allocation.

(2) FEMALE ALLOCATIONS: Total number of female animals allocated to this experiment on the date of allocation.

(3) UNSEXED ALLOCATIONS: Total number of unsexed animals allocated to this experiment on the date of allocation.

(4) TOTAL ALLOCATIONS: Total number of animals allocated to this experiment on the date of allocation.

14. Female Breeder Status (BS-013)

This report provides the history of female breeders. First, each female breeder is described by its strain, unique identification number, date of birth, current bldg-room, the male paired with, status, generation, and number of litters. This is the initial header on the first page of the report and items are described below.

This report is issued weekly and is in sequence by female UIN, litter animal, birth date, and litter animal. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Female UIN, Litter Animal, Birth Date, and Litter Animal

Frequency: Weekly

Destination or Printer No.: To Be Announced (TBA)

FEMALE UIN: Strain code and unique identification number assigned to the number at her birth.

DATE OF BIRTH: Gregorian date when the female breeder was born.

CURRENT BLDG-ROOM/CAGE: Room location of the cage that this litter was born in, and the cage number.

MALE UIN: Strain code and unique identification number assigned to the father at his birth.

CURRENT STATUS: The destination of this animal as one of the following: breeder, feeder, issue experiment, issue feeder, or issue other.

GENERATION: The placement of this litter within the succession of generations of this strain raised at the NCTR breeding colony.

NUMBER OF LITTERS: The number of litters this female breeder has produced.

Next, the report provides a breakdown of each litter by litter size, and characterizes the disposition of the offspring within each litter by sex.

This report is issued weekly and is prepared in sequence by litter strain-uin, birth date, litter size, and litter number. The litter animal uin and the sex of each litter animal is given followed by its disposition. Report detail is described below and a sample report follows this description.

This report includes all female breeders specified by a certain strain.

a. A group of detail lines is prepared for all litter animals of a certain disposition. The litter strain-uin, birth date, litter size, and litter number appear only on the first detail

line of a group and also on the first detail line of a new page when the group is continued. Detail is as follows:

(1) LITTER UIN: Strain code and unique identification number assigned to the litter at its birth.

(2) BIRTH DATE: Gregorian date that the litters in this group were born.

(3) LITTER SIZE: Total number of animals in this litter.

(4) LITTER NUMBER: A chronological placement of this litter among other litters born to the same breeder pair.

(5) LITTER ANIMAL: The unique identification number and strain for each animal in a litter.

(6) SEX: Sex code indicating the sex of this animal as:

F = Female or M = Male

(7) DISPOSITION DATE: Date the animal was given a certain disposition (e.g., culled, surplused, dead, etc.)

(8) CULLED: Number of animals in this litter that were destroyed because of injury or deformity.

(9) SURPLUS: Number of animals in this litter that, although healthy, were removed because of no demand for them.

(10) DEAD: Number of animals in this litter that were found dean.

(11) BREEDER: Number of animals that were established as breeders

(13) FEEDER: Number of animals established as feeders.

(14) ISSUE: Number of animals in this litter that were issued to an experiment, feeder or other.

(15) SURVEY: Number of animals in this litter that were requested for a Micro, Chemistry, or Pathology examination.

b. A group total line is prepared after the last detail line for the group. The total number of animals of each disposition column is displayed for that litter number.

c. A total line for each sex (male, female, and unsexed) for the same litter number is then displayed following the totals for disposition.

15. Productivity Index Report (BS014)

This report provides an index of productivity (i.e., average number weaned weekly) for each strain within a specified range of dates. The report is produced as needed. Report detail is provided below and a sample report is in Section B.2.

Sort Sequence: Strain

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) NCTR STRAIN CODE: Strain code that applies to the breeders summarized on this page.

(2) PRODUCTIVITY INDEX: The productivity index is the average number of animals weaned weekly by a certain strain. It is calculated by dividing the total weaned by the total breeders of a particular strain.

(3) TOTAL BREEDERS: Total number of breeder animals by strain.

(4) TOTAL WEANED: Total number of animals weaned by strain.

16. Feeder Status Report (BS015)

This report provides the projected mating date by sex and strain for feeders in a certain bldg-room. The total of animals of each sex and strain is then provided for each bldg-room.

This report is issued weekly and is in sequence by room number, strain code, and cage number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Room Number, Strain Code, Cage Number

Frequency: Weekly

Destination or Printer No.: To Be Announced (TBA)

a. A group of detail lines is prepared for candidates of the strain and from the same room as follows:

(1) BLDG-ROOM: Number identifying the room in which this group of candidates is located. Room number appears only on the first detail line of a group.

(2) CAGE: Number identifying the cage in which this candidate is located.

(3) MALE UIN/COUNT: Strain code and the unique identification number of this male or block to which this observation applies. If more than one male, as indicated by "COUNT", this UIN is the first in a sequential series for the block of animals that are in feeder status.

(4) FEMALE UIN/COUNT: Strain code and the unique identification of this female or block to which this observation applies. If more than one female, as indicated by "COUNT", this UIN is the first in a sequential series for the block of animals that are in feeder status.

(5) UNSEXED UIN/COUNT: Strain code and the unique identification of this unsexed animal or block to which this observation applies. "COUNT" is the total number of males, females, and unsexed in the litter.

(6) DATE OF BIRTH: Gregorian date on which this animal was born.

(7) PROJECTED MATING DATE: The date the strain is expected to mate. This date is derived by adding a certain number of days to the date of birth (e.g., 21 days for some strains of mice).

b. A total line showing the number of males, females, and unsexed by strain for the same bldg-room is then displayed.

17. <u>Issue Cage Census Report (BS016)</u>

This report provides the number of animals by strain of each sex that were transferred or designated as issue animals.

This report is issued on request and is in sequence by strain, cage, and sex. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Strain, Cage, Sex

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) STRAIN: The strain code to which this animal or animals belongs.

(2) CAGE: Number identifying the cage where this animal or animals resided at the time of transfer or being designated as issue.

(3) UIN/COUNT: Strain code and unique identification number of the animal or block to which this action applies. If this is a "litter" action the number of animals in the litter as indicated by "COUNT" is the number of animals from the same cage that were transferred or issued.

(4) SEX: Sex code indicating the sex of this animal as:

F = Female, M = Male, or U = Unsexed

18. Animal Orders (BS017)

This report provides a list of the animals ordered for experiments. Each order is identified by operator, experiment, principal investigator, and the dates the orders were requested and filled. Information required for the animals includes the number ordered by strain and sex as well as their age and phenotype.

This report is issued weekly, monthly, monthly or as requested for certain date ranges for experiments requiring continuing delivery of animals. The sequence is by experiment number and principal investigator. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: Experiment Number, Principal Investigator

Frequency: Weekly, Monthly, Ad hoc with date range

Destination or Printer No.: To Be Announced (TBA)

a. The operator and the Gregorian date the order was placed are identified in the title lines of each page.

b. A detail line is prepared for each experiment number as follows:

(1) OPER: Number identifying the animal technician who placed this order.

(2) EXPERIMENT: The experiment number and the test an animal is allocated to.

(3) BUILDING-ROOM: Number identifying the building and room where the pups resided at the time of weighing and assignment.

(4) PI/CONTACT: The principal investigator (PI) or person ultimately responsible for the experiment and order.

(5) DATE REQUIRED: Date the animals are required to begin the experiment.

(6) STRAIN: The strain code to which this animal or animals belongs.

(7) SEX: Sex code indicating the sex of this animal as:

F = Female, M = Male, or U = Unsexed

(8) NUMBER REQUESTED: Number of animals of each strain requested for a specific experiment and principal investigator.

(9) NUMBER DELIVERED: Number of animals of each strain delivered for a specific experiment and principal investigator.

(10) DATE FILLED: Date the animals were received for the experiment.

(11) SPECIAL REQUIREMENTS: Special requirements for the order, if needed, are listed in this area.

c. A summary page is prepared monthly for each experiment number. Totals are displayed for the number requested and the number delivered by sex and strain for each experiment.

19. Target Trace (BS018)

This report provides a printed diagram that traces the genealogical history of a root or designated animal. Print options for the report include an option of reporting all the predecessors for a current UIN of interest or reporting the successors for an historical UIN.

The report is printed on request and the sort sequence is by strain, generation, and room number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: strain, generation, and room number

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) UIN: Strain code and unique identification number assigned at birth.

(2) GENERATION: The placement of this animal within the succession of generations of this strain as developed at the NCTR breeder colony.

(3) BUILDING-ROOM: Number identifying the current building and room where the litters reside.

(4) CAGE: Number identifying the cage where this animal or animals reside.

20. BS-019. Ad Hoc Reports

This report provides the ability for the terminal operator to generate ad hoc reports for viewing on the screen or printing. The Ad hoc Report Selection Screen (Figure A.2., Appendix A.1) shows the parameter selection menu used to specify parameters to generate one or more separate ad hoc reports. The Screen A.2 Module description in Appendix D describes the assumptions, process, etc. The parameters and data elements shown on the screen have been described in other reports. Reports will display the screen results.

21. BS-020. BMSSLOOK

This report provides the function of outputting all information for a particular UIN as shown in the physical layout for BS-019 (Appendix B.2.). The UIN selection criteria will be entered by the operator but the method of invocation has not yet been specified. The BS-019 Module description is found in Appendix E.

22. Pup Weights at Weaning and Disposition (EBS001)

This report provides the weight and disposition (surplus, allocation, breeder, etc.) at the time of weaning of pups born to parents who have undergone a specific experimental treatment. The report tracks the parents and specific litter information (UIN, generation, number) as well as the weight and disposition of the pup at the time of weaning.

This report is issued on request and is in sequence by experiment number, female treatment, litter UIN, litter generation, and litter number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: experiment number, female treatment, litter UIN, litter generation, and litter number

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) EXPERIMENT NO.: The experiment number and the test an animal is allocated to.

(2) OPER: Number identifying the animal technician who performed this action.

(3) FEMALE TMT: Treatment number indicating the initial chemical treatment administered to this female. The chemical, chemical concentration and the duration of the chemical dose is administered is identified by this number.

(4) FEMALE GENERATION: The placement of this female within the succession of generations of this strain raised at the NCTR breeding colony.

(5) MALE UIN: Strain code and unique identification number assigned at birth to the sire.

(6) FEMALE UIN: Strain code and unique identification number assigned to her at birth to the dam.

(7) LITTER BIRTH DATE: Gregorian date that this litter was born.

(8) LITTER UIN: Strain code and unique identification number assigned to the litter at its birth.

(9) LITTER GENERATION: The placement of this litter within the succession of generations of this strain raised at the NCTR breeding colony.

(10) LITTER NUMBER: A chronological placement of this litter among other litters born to the same breeder pair.

(11) LITTER WEIGHT: The combined weight of all the pups in the litter at

weaning.

(12) UNITS: In grams or kilograms.

(13) SURPLUS: Number of animals in this litter that, although healthy, were removed because of no demand for them.

(14) ALLOCATION: The number of pups chosen to be used in an experiment.

(15) BREEDER: Number of pups that were established as breeders

(16) SURVEY: Number of animals in this litter that were requested for a Micro, Chemistry, or Pathology examination.

(17) ISSUE: Number of pups on standby for future assignment.

(18) DEAD: Number of pups in this litter that were dead or missing.

(19) CULLED: Number of animals in this litter that were destroyed because of injury or deformity.

(20) BUILDING-ROOM: Number identifying the building and room where the pups resided at the time of weighing and assignment.

23. Pup Weights. (EBS002)

This report provides detailed weight information on the pups born to parents who have undergone a specific experimental treatment. The report the parents and specific litter information (UIN, generation, number) as well as the weight and age of the pup on a certain date.

This report is issued upon request and sequenced by room number, experiment number, litter UIN, and litter generation. Report detail is described below and a sample of the report is in B.2.

Sort Sequence: room number, experiment number, litter UIN, and litter generation

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) EXPERIMENT NO.: The experiment number and the test an animal is allocated to.

(2) OPER: Number identifying the animal technician who made this assignment.

(3) BUILDING-ROOM: Number identifying the building and room where the pups resided at the time of weighing and assignment.

(4) FEMALE GENERATION: The placement of this animal within the succession of generations of this strain raised at the NCTR breeding colony.

(5) MALE UIN: Strain code and unique identification number assigned at birth.

(6) FEMALE UIN: Strain code and unique identification number assigned to her at birth.

(7) LITTER BIRTH DATE: Gregorian date that this litter was born.

(8) LITTER ANIMAL: Strain code and unique identification number assigned to each animal in the litter at its birth.

(9) LITTER GENERATION: The placement of this litter within the succession of generations of this strain raised at the NCTR breeding colony.

(10) LITTER NUMBER: A chronological placement of this litter among other litters born to the same breeder pair.

(11) WEIGHT: The weight of the pup.

(12) UNIT: The unit of weight in grams or kilograms.

(13) NUMBER WEIGHED: Number of litter animals weighed.

(14) DATE WEIGHED: The Gregorian date the pup was weighed.

(15) AGE DAYS: The age in days of the pup at the time weighed. Calculated by subtracting the litter birth date from the date weighed.

24. Fertility (EBS003)

The purpose of this report is to provide information of the fertility of males and females undergoing experimental treatments. The report tracks the males and females by bldg-room, experiment, treatment, the generations and litters produced, and the disposition of the offspring for each generation and litter.

This report is issued on request for a selected range of dates and is in sequence by experiment number, female treatment, litter UIN, and litter number. Report detail is described below and a sample report is in Section B.2.

Sort Sequence: experiment number, female treatment, litter UIN, and litter number

Frequency: Ad hoc

Destination or Printer No.: To Be Announced (TBA)

(1) EXPERIMENT: The experiment number and the test an animal is allocated to.

(2) BUILDING-ROOM: Number identifying the building and room where the pups resided at the time of weighing and assignment.

(3) MALE GENR: The placement of this male within the succession of generations of this strain as developed at the NCTR breeder colony.

(4) MALE TMT: Treatment number indicating the initial chemical treatment administered to this male. The chemical, chemical concentration and the duration of the chemical dose is administered is identified by this number.

(5) MALE UIN: Strain code and unique identification number assigned at birth.

(6) FEMALE GENR: The placement of this female within the succession of generations of this strain as developed at the NCTR breeder colony.

(7) FEMALE TMT: Treatment number indicating the initial chemical treatment administered to this female. The chemical, chemical concentration and the duration of the chemical dose is administered is identified by this number.

(8) FEMALE UIN: Strain code and unique identification number assigned to her at birth.

(9) DATE EST: Gregorian date that a breeding relationship was established for the parents of this litter.

(10) PLUG DATE: Gregorian date that the female was observed plugged.

(11) OBSERVED PREGNANT: Gregorian date that the female was observed pregnant.

(12) LITTER BIRTH DATE: Gregorian date that the litters in this group were born.

(13) LITTER UIN: Strain code and unique identification number assigned to the litter at its birth.

(14) GENR: The placement of this litter within the succession of generations of this strain raised at the NCTR breeding colony.

(15) LITTER NUMBER: A chronological placement of this litter among other litters born to the same breeder pair.

(16) LITTER SIZE: Total number of animals born in this litter.

(17) OBSERVE DATE: The date the litter was observed or inspected.

(18) NO. M/F DAY: The ratio of males to females born to the observed litter that date.

(19) NO. L/D (LIVE/DEAD): Ratio of animals that were born alive to those born dead. Dead refers to those animals that were "stillborn".

(20) NO. MALES WEAN: Total number of males that were weaned on this date.

(21) NO. FEM WEAN: Total number of females that were weaned on this date.

(22) NO. DEAD WEAN: Total number of animals that died by the time they were to be weaned.

(23) WEAN DATE: Gregorian date that the litters in this group were separated from their mothers.

b. A summary page is prepared as the last report page for each experiment. The experiment number is identified in the heading. Totals are displayed for the following categories for each treatment:

Number of female breeders established.

Number of male breeders established.

Number of plugs observed.

Number of females observed pregnant.

Number of litters.

Average litter size.

Totals are presented for born, males, females, born live, born dead, males weaned, females weaned, and weaned.

Average number of litters per female.

Average duration between litters.

Duration in days from start to last litter.

Number of unproductive females.

PART B.2. Proposed Report Physical Layouts
(Revisions to the External Design Document of 9/18/88)

REPORT NO. BS002o NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY TIME = HH:MM

BLDG-ROOM XXX-XXXX BREEDING MULTIGENERATION SUPPORT SYSTEM PAGE ##

NCTR STRAIN CODE XX DAILY RECAP REPORT

PREGNANT, PLUGGED, LITTER, AND NEWBORN OBSERVATIONS

DATE OBSERVED MM-DD-YY

OPER	STATUS	UIN/COUNT	SEX	ANIMAL ID	PHENO-TYPE	CAGE	OBSERVATIONS	LITTER-UIN	LITTER-GEN
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXXX	XX-9999999	9999

REPORT NO. BS002os NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY TIME = HH:MM

BLDG-ROOM XXX-XXXX BREEDING INFORMATION SYSTEM PAGE ##

NCTR STRAIN.CODE XX DAILY RECAP REPORT SUMMARY

PREGNANT, PLUGGED, LITTER, AND NEWBORN OBSERVATIONS

MM-DD-YY

OBSERVATION TOTALS: TOTAL LITTERED..... 9999

TOTAL PLUGGED..... 9999

TOTAL PREGNANT..... 9999

TOTAL NEW BORN..... 9999

(*) INDICATES TRANSACTION DELETED

NOTE: SCREEN OUTPUT (WITH OPTION TO PRINT SCREEN) AS WELL AS HARDCOPY REPORT.

REPORT NO. BS002p NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY TIME = HH:MM

BLDG-ROOM XXX-XXXX BREEDING MULTIGENERATION SUPPORT SYSTEM PAGE ##

NCTR STRAIN CODE XX DAILY RECAP REPORT-PHENOTYPE OBSERVATIONS

DATE OBSERVED MM-DD-YY

OPER	STATUS	UIN/COUNT	SEX	ANIMAL ID	PHENO-TYPE	CAGE	OBSERVATIONS	LITTER-UIN	LITTER-GEN
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999
99999	XXXXXXXXXXXX	XX-9999999/99	X	XXXXX	XX	9999	XXXXXXXXXXXX	XX-9999999	9999

REPORT NO. BS002ps NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY TIME = HH:MM

BLDG-ROOM XXX-XXXX BREEDING INFORMATION SYSTEM PAGE ##

NCTR STRAIN.CODE XX DAILY RECAP REPORT SUMMARY -- PHENOTYPE OBSERVATIONS

MM-DD-YY

OBSERVATION TOTALS: TOTAL MALES YELLOW..... 9999

TOTAL FEMALES YELLOW..... 9999

TOTAL MALES BLACK..... 9999

TOTAL FEMALES BLACK..... 9999

TOTAL MALES AGOUTI..... 9999

TOTAL FEMALES AGOUTI..... 9999

TOTAL MALES HAIR..... 9999

TOTAL FEMALES HAIR..... 9999

TOTAL MALES NUDE..... 9999

TOTAL FEMALES NUDE..... 9999

(*) INDICATES TRANSACTION DELETED

NOTE: SCREEN OUTPUT (WITH OPTION TO PRINT SCREEN) AS WELL AS HARDCOPY REPORT.

FEMALE BREEDER CARD

UIN XX-9999999

MATED mm/dd/yy STRAIN (text field)

GENERATION 9999 SOURCE CAGE 9999

DOB FOR UIN mm/dd/yy BLDG-ROOM XXX-XXXX

CAGE NO 9999

DATE FOUND PREGN	DATE LITTER BORN	NUMBER OF LITTER	NUMBER WEANED	REMARKS

MALE BREEDER CARD

UIN XX-9999999

MATED mm/dd/yy STRAIN (text field)

GENERATION 9999 SOURCE CAGE 9999

DOB FOR UIN mm/dd/yy BLDG-ROOM XXX-XXXX

CAGE NO 9999

MOTHER'S NUMBER	DATE FOUND PREG'N	REMOVED TO CAGE	DATE RET'D TO MALE	REMARKS

REPORT NO. BS012 NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY PAGE ##

EXPERIMENT NO. 9999-99 BREEDING INFORMATION SYSTEM HH:MM:SS

BLDG-ROOM XXX-XXXX ALLOCATION mm/dd/yy

RACK 99 SIDE X

OPER	CAGE	TMT	ANIMAL START DATE	BREEDING ANIMAL ID	SEX	UIN	ALLOC WEIGHT	UNITS	ISSUE CAGE	REPLACEMENT ANIMAL	CARCASS-ID
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999
XXXXX	9999	999	mm/dd/yy	XXXXX	X	XX-9999999	999.9	XXX	9999	XX-9999999	99999999999

ISSUE CAGE CENSUS REPORT HH:MM:SS

STRAIN	CAGE	UIN/COUNT	SEX
XX	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X

TOTAL MALES.....9999 TOTAL FEMALES.....9999 TOTAL UNSEXED.....9999

XX	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X

TOTAL MALES.....9999 TOTAL FEMALES.....9999 TOTAL UNSEXED.....9999

XX	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X
	9999	9999999/99	X

TOTAL MALES.....9999 TOTAL FEMALES.....9999 TOTAL UNSEXED.....9999

REPORT NO. BS017 NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY TIME = HH:MM

OPER 99999 BREEDING MULTIGENERATION SUPPORT SYSTEM PAGE ##

RECAP REPORT -- ANIMAL ORDERS

MM-DD-YY

EXP. NO.	BLDG-ROOM	P.I./CONTACT	DATE REQUIRED	STRAIN	SEX	NUMBER REQUESTED	NUMBER DELIVERED	DATE FILLED	SPECIAL REQUIREMENTS
9999-99	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
9999-99	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
9999-99	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
	XXX-XXXX	XXXXXXXXXX	mm/dd/yy	XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX
				XX	X	999	999	mm/dd/yy	XXXXXXXXXXXXXXXXXXXXXXXXX

```
REPORT 5S018                                          TARGET TRACE ROOT ANIMAL XX-9999999
                                                                                                                        NOTE: PRINT OPTIONS FOR THE REPORT
                                                                  XX-9999999-9999                                             1.  CURRENT UIN, ALL PREDECESSORS
  VENDOR INFORMATION: (REQUEST FOR A 300-350                      XX-9999999-9999                                             2.  PAST UIN, ALL SUCCESSORS
         CHARACTER FIELD TO DESCRIBE THE                          XX-9999999-9999                                             3.  XX = NCTR STRAIN CODE
         VENDOR, DATE RECEIVED, ANY ANOMALIES                     XX-9999999-9999                                             4.  9999999 = UIN
         OR DESCREPANCIES,ETC.)                                   XX-9999999-9999                                             5.  9999 = GENERATION (adjacent to or above UIN's)
                                                                  XX-9999999-9999                                             6.  XXX-XXXX = ROOM NO. (BLDG-ROOM)
                                                                  XX-9999999-9999
                                                                  XX-9999999-9999
                                                                  XX-9999999-9999                                                                 Current
           * * * * * * * * * * * * * * * * * * * * * * * * * * * *XX-9999999-9999                                                              Room/Cage No.
           *
       9999              9999              9999             9999              9999             9999             9999             9999            9999
    XX-9999999       XX-9999999        XX-9999999       XX-9999999        XX-9999999       XX-9999999       XX-9999999                                      XXX-XXXX/9999
           *                                 *                 *                *                                  *
           *                                 *                 *                *                                  * * * * * XX-9999999                     XXX-XXXX/9999
           *                                 *                 *                *                                  *               *
           *                                 *                 *                *                                  *               * * * * * XX-9999999     XXX-XXXX/9999
           *                                 *                 *                *                                  *               *
           *                                 *                 *                *                                  *               * * * * * XX-9999999     XXX-XXXX/9999
           *                                 *                 *                *                                  *
           *                                 *                 *                *                                  * * * * * XX-9999999                     XXX-XXXX/9999
           *                                 *                 *                *
           *                                 *                 *                * * * * *  XX-9999999       XX-9999999                                      XXX-XXXX/9999
           *                                 *                 *                                                   *
           *                                 *                 *                                                   * * * * * XX-9999999                     XXX-XXXX/9999
           *                                 *                 *                                                   *
           *                                 *                 *                                                   * * * * * XX-9999999                     XXX-XXXX/9999
           *                                 *                 *                                                   *
           *                                 *                 *                                                   * * * * * XX-9999999                     XXX-XXXX/9999
           *                                 *                 *
           *                                 *                 * * * * *  XX-9999999       XX-9999999                                                       XXX-XXXX/9999
           *                                 *                                  *
           *                                 *                                  * * * * *  XX-9999999                                                       XXX-XXXX/9999
           *                                 *
           *                                 * * * * *  XX-9999999        XX-9999999       XX-9999999                                                       XXX-XXXX/9999
           *                                 *
           *                                 * * * * *  XX-9999999        XX-9999999       XX-9999999                                                       XXX-XXXX/9999
           *                                 *
           *                                 * * * * *  XX-9999999        XX-9999999       XX-9999999                                                       XXX-XXXX/9999
           *                                 *
           *                                 * * * * *  XX-9999999        XX-9999999       XX-9999999                                                       XXX-XXXX/9999
           *                                                                    *                 *
           *                                                                    *                 * * * * * XX-9999999                                      XXX-XXXX/9999
           *                                                                    *                 *
           *                                                                    *                 * * * * * XX-9999999                                      XXX-XXXX/9999
           *                                                                    *
           *                                                                    * * * * *  XX-9999999       XX-9999999                                      XXX-XXXX/9999
           *                                                                    *
           *                                                                    * * * * *  XX-9999999                                                       XXX-XXXX/9999
           *
           * * * * * XX-9999999        XX-9999999       XX-9999999        XX-9999999       XX-9999999       XX-9999999                                      XXX-XXXX/9999
                                                                                                  *                *
                                                                                                  *                * * * * * XX-9999999                     XXX-XXXX/9999
                                                                                                  *                *
                                                                                                  *                * * * * * XX-9999999                     XXX-XXXX/9999
                                                                                                  *
                                                                                                * * * * *   XX-9999999       XX-9999999                     XXX-XXXX/9999

                                                                  * * * * * 23 ANIMALS THIS GROUP * * * * *

NOTE: NO CAGE NUMBER WILL BE LISTED IF THE ANIMAL IS NOT ACTIVE
```

TARGET TRACE SELECTION CRITERIA

THIS TRACE WAS DONE USING THE FOLLOWING CRITERIA IN THE SEQUENCE SHOWN AT THE LEFT:

SELECTION SEQUENCE	CRITERIA	SELECTED OR RANGE
9	1. UIN(S)	9999999,9999999
9	2. BLDG-ROOM(S)	XXX-XXXX,XXX-XXXX
9	3. CAGE(S)	9999,9999
9	4. STRAIN(S)	XX,XX
9	5. BIRTHDATE(S)	mm/dd/yy,mm/dd/yy

NOTE: Selection sequence indicates the major to minor importance of the criteria and will vary depending on the query. For example, the person making the query may want to trace the genealogical history of animals within a building-room or by a combination of criteria such as a group of cages, or a strain within a building-room.

RECORD HEADER FOR UIN

UIN	9999999
STRAIN	AA
ASSIGNMENTS COUNTS	99
OBSERVATION COUNTS	99
REMOVAL COUNTS	99
ALLOCATION COUNTS	99
MATE COUNTS	99
LITTER INFORMATION COUNTS	99

DATA HEADER SEGMENT

GENERATION	9999
CURRENT BLDG-ROOM	AAA-AAAA
CURRENT CAGE	9999
BIRTH DATE	mm/dd/yy
PURCHASE REFERENCE NUMBER	999999
COUNT OF LITTERS DROPPED	99
MOTHER STRAIN	AA
MOTHER UIN	9999999
FATHER STRAIN	AA
FATHER UIN	9999999
LITTER UIN	9999999999
LAST ACTIVITY DATE	mm/dd/yy
CARCASS ID	99999999999
REMOVAL DATE	mm/dd/yy
SEX	A
MARK	X
STATUS	AAAAAAAAAAAA
REMOVAL REASON	AAAAAAAAAAAA
DISPOSITION	AAAAAAAAAAAA

ASSIGNMENT SEGMENT

SEX	A
MARK	X
OPERATOR	99999
DATE	mm/dd/yy
FIRST UIN	9999999
BLDG-ROOM	AAA-AAAA
COUNT	99
SOURCE CAGE	9999
DESTINATION CAGE	9999
ASSIGN TYPE	AAAAAAAAAAA
ASSIGN SUBTYPE	AAAAAAAAAAA
COMMON ACCOUNT NUMBER	99999999
EXPERIMENT NUMBER	9999-99
SEQUENCE NUMBER	99999
PREVIOUS STATUS	AAAAAA

REMOVAL SEGMENT

OPERATOR	99999
BLDG-ROOM	9999-99
DATE	mm/dd/yy
COUNT	99
SOURCE CAGE	9999
FIRST UIN	9999999
CARCASS ID	99999999999
SEQUENCE NUMBER	99999

NOTE: The report continues listing the assignments, removals,etc. for the UIN.

EBS003 NATIONAL CENTER FOR TOXICOLOGICAL RESEARCH DD-MMM-YY PAGE ##

EXPERIMENTAL BREEDING SYSTEM HH:MM:SS

EXPERIMENT NO. 9999-99 FERTILITY REPORT FROM DATE MM/DD/YY TO MM/DD/YY

BLDG-ROOM	MALE GENR	MALE TMT	MALE UIN	FEM GENR	FEM TMT	FEMALE UIN	DATE EST	PLUG DATE	OBSERVED PREGNANT	LITTER BIRTH DATE	LITTER UIN	GENR	LITT. NO.	LITT. SIZE	OBSERVE DATE	NO. M/F DAY	NO. L/D	NO. MALES WEAN	NO. FEM WEAN	NO. DEAD WEAN	WEAN DATE
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy
XXX-XXXX	9999	999	XX-9999999	9999	999	XX-9999999	mm/dd/yy	mm/dd/yy	mm/dd/yy	mm/dd/yy	XX-9999999	9999	99	99	mm/dd/yy	99/99	99/99	99	99	99	mm/dd/yy

APPENDIX C

DATA BASE FILE STRUCTURES
(ADABAS VERSION)

Appendix C...Data Base File Structures (ADABAS version)

File: (EB-1) ANIMAL

Primary Key: ANIM-BIS-UIN
Projected Growth Rate: 2000 records per month
Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	ANIM-UNIQUE-ANIMAL-ID	A	11.0	F	U	From INLIFE
1	AB	ANIM-SEX	A	1.0	F		From INLIFE
1	AC	ANIM-STRAIN-CODE	N	5.0			From INLIFE
1	AE	ANIM-ANIMAL-IN-CAGE-ID	A	5.0			From INLIFE
G 1	AF	ANIM-BIS-UIN					From INLIFE
2	AG	ANIM-BIS-UIN-STRAIN-CODE	A	2.0	N	D	From INLIFE
2	AH	ANIM-BIS-UIN-SEQ-NUM	N	7.0	N	D	From INLIFE
1	AI	ANIM-BIRTH-DATE	N	6.0	N	D	From INLIFE
1	AJ	ANIM-CARCASS-ID	N	11.0	N	D	From INLIFE
1	AK	ANIM-LITTER-UIN	A	9.0	F	D	
1	AL	ANIM-PHENOTYPE	N	5.0	F	D	Coded
1	AM	ANIM-BREEDING-ID	A	5.0	F		Coded
1	AN	ANIM-ON-EXPERIMENT	A	1.0	F		Derivable
1	AO	ANIM-CAN	A	11.0			
1	X1	ANIM-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	ANIM-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	ANIM-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	ANIM-CREATE-SESSION-DATE	N	6.0	N		
1	C0	ANIM-CREATE-DATE	N	6.0		D	Derivable
G 1	AP	ANIM-NOTES-GROUP					
2	AQ	ANIM-VENDORS-ID	A	80.0	N		
M 2	AR	ANIM-NOTES	A	80.0	N		2 occurrences
2	X3	ANIM-NOTES-APPL-PREFIX-CODE	A	2.0	N		
2	X4	ANIM-NOTES-SESSION-NUM	N	8.0	N		
2	Y3	ANIM-NOTES-OPERATOR-CODE	N	5.0	N		
2	Y4	ANIM-NOTES-SESSION-DATE	N	6.0	N		
G 1	AS	ANIM-REMOVAL-GROUP					
2	AT	ANIM-REMOVAL-DESTINATION	N	5.0	F		Coded
2	AU	ANIM-REMOVAL-DATE	N	6.0		D	
2	X5	ANIM-REMOVAL-APPL-PREFIX-CODE	A	2.0	N		
2	X6	ANIM-REMOVAL-SESSION-NUM	N	8.0	N		
2	Y5	ANIM-REMOVAL-OPERATOR-CODE	N	5.0	N		
2	Y6	ANIM-REMOVAL-SESSION-DATE	N	6.0	N		
G 1	AV	ANIM-WEAN-GROUP					
2	AW	ANIM-WEAN-WEIGHT	N	6.0			
2	AX	ANIM-WEAN-WEIGHT-UNIT	N	5.0			Coded
2	X7	ANIM-WEAN-APPL-PREFIX-CODE	A	2.0	N		
2	X8	ANIM-WEAN-SESSION-NUM	N	8.0	N		
2	Y7	ANIM-WEAN-OPERATOR-CODE	N	5.0	N		
2	Y8	ANIM-WEAN-SESSION-DATE	N	6.0	N		
P 1	AY	ANIM-STATUS-GROUP					
2	AZ	ANIM-STATUS	N	5.0		D	Coded
2	BA	ANIM-BLDG-ROOM	A	7.0		D	
2	BB	ANIM-CAGE	N	4.0		D	

```
  2 BC ANIM-STATUS-CHANGE-DATE            N   6.0     D
  2 X9 ANIM-STATUS-APPL-PREFIX-CODE       A   2.0   N
  2 X0 ANIM-STATUS-SESSION-NUM            N   8.0   N
  2 Y9 ANIM-STATUS-OPERATOR-CODE          N   5.0   N
  2 Y0 ANIM-STATUS-SESSION-DATE           N   6.0   N
G 1 BD ANIM-CURRENT-STATUS-GROUP
  2 BE ANIM-CURRENT-STATUS                N   5.0     D  Coded
  2 BF ANIM-CURRENT-EXPERIMENT-NUM        N   6.0   F D
  2 BG ANIM-CURRENT-BLDG-ROOM             A   7.0     D
  1 BH ANIM-LATEST-RETIRE-DATE            N   6.0     D
  0 SU ANIM-BIS-UIN-SP                    A   9.0     S
*      SU = AG(1,2),AH(1,7)
```

File: (EB-2) - DATE-ESTABLISHED

Primary Key: DE-DAM-UIN
Projected Growth Rate: 100 records per month
Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	DE-DAM-UIN	A	9.0	F	D	
1	AB	DE-SIRE-UIN	A	9.0	F	D	
1	AC	DE-DATE-ESTABLISHED	N	6.0		D	
1	AD	DE-DATE-PAIR-DISSOLVED	N	6.0		D	Derivable
1	X1	DE-CREATE-APPL-PREFIX-CODE	A	2.0			
1	X2	DE-CREATE-SESSION-NUM	N	8.0			
1	Y1	DE-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	DE-CREATE-SESSION-DATE	N	6.0	N		
1	X3	DE-DISSLV-APPL-PREFIX-CODE	A	2.0	N		
1	X4	DE-DISSLV-SESSION-NUM	N	8.0	N		
1	Y3	DE-DISSLV-OPERATOR-CODE	N	5.0	N		
1	Y4	DE-DISSLV-SESSION-DATE	N	6.0	N		
P 1	AE	DE-PAIR-OBSERVATION-GROUP					
2	AF	DE-LITTER-OBSERVATION	N	5.0			Coded
2	AG	DE-LITTER-OBSERVATION-DATE	N	6.0		D	
2	X5	DE-LITOBS-APPL-PREFIX-CODE	A	2.0	N		
2	X6	DE-LITOBS-SESSION-NUM	N	8.0	N		
2	Y5	DE-LITOBS-OPERATOR-CODE	N	5.0	N		
2	Y6	DE-LITOBS-SESSION-DATE	N	6.0	N		
1	AH	DE-PROGENY-STRAIN	A	2.0		D	
1	AI	DE-EXPERIMENT-NUM	N	6.0	F	D	

File: {EB-3} - LITTERS

Primary Key: LIT-LITTER-UIN
Projected Growth Rate: 300 records per month
Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	LIT-LITTER-UIN	A	9.0	F	D	
1	AB	LIT-DAM-UIN	A	9.0	F	D	
1	AC	LIT-SIRE-UIN	A	9.0	F	D	
1	AD	LIT-BIRTH-DATE	N	6.0		D	
P 1	AE	LIT-NUMBER-DETAIL-GROUP					Occ1=Total
2	AF	LIT-NUMBER-IN-LIT	N	2.0			Occ2,3,4 =
2	AG	LIT-NUMBER-BORN-DEAD	N	2.0			U,F,M
1	AH	LIT-GENERATION	N	4.0			
1	AI	LIT-REMARKS	A	80.0			
1	AJ	LIT-WEAN-DATE	N	6.0		D	Derivable
1	AK	LIT-LITTER-NUM	N	2.0			Derivable
G 1	AL	LIT-NOTES-GROUP					
M 2	AM	LIT-LITTER-NOTES	A	80.0			2 occ
2	AN	LIT-VENDOR-ID	A	80.0			
1	X1	LIT-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	LIT-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	LIT-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	LIT-CREATE-SESSION-DATE	N	6.0	N		
P 1	AO	LIT-PUP-WEIGHING-GROUP					
2	AP	LIT-PUP-WEIGHT	N	6.0			
2	AQ	LIT-PUP-WEIGHT-UNITS	N	5.0			Coded
2	AR	LIT-NUMBER-PUPS-WEIGHED	N	2.0			
2	AS	LIT-DATE-PUPS-WEIGHED	N	6.0			
2	X3	LIT-WEIGH-APPL-PREFIX-CODE	A	2.0	N		
2	X4	LIT-WEIGH-SESSION-NUM	N	8.0	N		
2	Y3	LIT-WEIGH-OPERATOR-CODE	N	5.0	N		
2	Y4	LIT-WEIGH-SESSION-DATE	N	6.0	N		
1	AT	LIT-EXPERIMENT-NUM	N	6.0	F	D	
0	S1	LIT-BIS-STRAIN-SB	A	2.0		S	
*		S1 = AA(1,2)					

File: (EB-4) - ALLOCATION

Primary Key: ALC-UIN
Projected Growth Rate: 2000 records per month
Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	ALC-UIN	A	9.0	F	D	
1	AB	ALC-ALLOCATE-DATE	N	6.0		D	
1	AC	ALC-EXPERIMENT-NUM	N	6.0	F	D	
1	AD	ALC-BLDG-ROOM	A	7.0		D	
1	AE	ALC-CAGE	N	4.0			
1	AF	ALC-RACK	N	2.0			
1	AG	ALC-SIDE	A	1.0			
1	AH	ALC-TREATMENT	N	3.0			
1	AI	ALC-WEIGHT	N	6.0			
1	AJ	ALC-WEIGHT-UNIT	N	5.0			Coded
1	X1	ALC-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	ALC-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	ALC-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	ALC-CREATE-SESSION-DATE	N	6.0	N		
0	S1	ALC-BIS-STRAIN-SB	A	2.0		S	

* S1 = AA(1,2)

File: {EB-5} - ORDERS

Primary Key: ORD-EXPERIMENT-NUMBER
Projected Growth Rate: 100 records per month
Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	ORD-EXPERIMENT-NUMBER	N	6.0	F	D	
1	AB	ORD-NUMBER-REQUESTED	N	3.0			
1	AC	ORD-NUMBER-DELIVERED	N	3.0			
1	AD	ORD-DATE-ORDERED	N	6.0			
1	AE	ORD-DATE-REQUIRED	N	6.0		D	
1	AF	ORD-DATE-FILLED	N	6.0		D	
1	AG	ORD-STRAIN	A	2.0			Coded SEEBELOW
1	AH	ORD-SEX	A	1.0			Coded
1	AI	ORD-CONTACT	A	13.0			
1	AJ	ORD-BLDG-ROOM	A	7.0		D	
1	AK	ORD-SPECIAL-REQUIREMENTS	A	20.0			
1	X1	ORD-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	ORD-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	ORD-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	ORD-CREATE-SESSION-DATE	N	6.0	N		

Efforts will be made to allow for use of both the BIS 2-character strain code and the five-digit TDMS strain code.

File: {EB-6} - CAGE

Primary Key: CAG-CAGE-NUMBER
Projected Growth Rate: Not Applicable...About 7000 records total Access Method: ADABAS NATURAL (historical)

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	CAG-CAGE-NUMBER	N	4.0	F	D	
1	AB	CAG-BLDG-ROOM	A	7.0		D	
1	AC	CAG-STATUS	N	5.0		D	Coded
1	AD	CAG-ORIGINAL-UIN	A	9.0	F	D	
1	AE	CAG-ORIGINAL-COUNT	N	2.0			
1	AF	CAG-ORIGINAL-DATE	N	6.0			
1	AG	CAG-BLOCK-UIN	A	9.0	F		
1	AH	CAG-BLOCK-COUNT	N	2.0			
1	AI	CAG-BLOCK-GENERATION	N	4.0			
P 1	AJ	CAG-ANIMAL-GROUP					
2	AK	CAG-ANIMAL-UIN	A	9.0	F	D	
2	AL	CAG-ANIMAL-SEX	A	1.0			
2	AM	CAG-ANIMAL-PHENOTYPE	N	5.0			Coded
2	AN	CAG-ANIMAL-GENERATION	N	4.0			
2	AO	CAG-ANIMAL-FLAG	A	1.0			P,A,N,L,R,B

File: {EB-7} - AUD-ANIMAL

Primary Key: ANIM-BIS-UIN
Projected Growth Rate: About 200 records per month or 10% of the ANIMAL file
Access Method: ADABAS NATURAL

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	ANIM-UNIQUE-ANIMAL-ID	A	11.0	F	U	From INLIFE
1	AB	ANIM-SEX	A	1.0	F		From INLIFE
1	AC	ANIM-STRAIN-CODE	N	5.0			From INLIFE
1	AE	ANIM-ANIMAL-IN-CAGE-ID	A	5.0			From INLIFE
G 1	AF	ANIM-BIS-UIN					From INLIFE
2	AG	ANIM-BIS-UIN-STRAIN-CODE	A	2.0	N	D	From INLIFE
2	AH	ANIM-BIS-UIN-SEQ-NUM	N	7.0	N	D	From INLIFE
1	AI	ANIM-BIRTH-DATE	N	6.0	N	D	From INLIFE
1	AJ	ANIM-CARCASS-ID	N	11.0	N	D	From INLIFE
1	AK	ANIM-LITTER-UIN	A	9.0	F	D	
1	AL	ANIM-PHENOTYPE	N	5.0	F	D	Coded
1	AM	ANIM-BREEDING-ID	A	5.0	F		Coded
1	AN	ANIM-ON-EXPERIMENT	A	1.0	F		Derivable
1	AO	ANIM-CAN	A	11.0			
1	X1	ANIM-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	ANIM-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	ANIM-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	ANIM-CREATE-SESSION-DATE	N	6.0	N		
1	C0	ANIM-CREATE-DATE	N	6.0		D	Derivable
G 1	AP	ANIM-NOTES-GROUP					
2	AQ	ANIM-VENDORS-ID	A	80.0	N		
M 2	AR	ANIM-NOTES	A	80.0	N		2 occurrences
2	X3	ANIM-NOTES-APPL-PREFIX-CODE	A	2.0	N		
2	X4	ANIM-NOTES-SESSION-NUM	N	8.0	N		
1	Y3	ANIM-NOTES-OPERATOR-CODE	N	5.0	N		
1	Y4	ANIM-NOTES-SESSION-DATE	N	6.0	N		
G 1	AS	ANIM-REMOVAL-GROUP					
2	AT	ANIM-REMOVAL-DESTINATION	N	5.0	F		Coded
2	AU	ANIM-REMOVAL-DATE	N	6.0		D	
2	X5	ANIM-REMOVAL-APPL-PREFIX-CODE	A	2.0	N		
2	X6	ANIM-REMOVAL-SESSION-NUM	N	8.0	N		
2	Y5	ANIM-REMOVAL-OPERATOR-CODE	N	5.0	N		
2	Y6	ANIM-REMOVAL-SESSION-DATE	N	6.0	N		
G 1	AV	ANIM-WEAN-GROUP					
2	AW	ANIM-WEAN-WEIGHT	N	6.0			
2	AX	ANIM-WEAN-WEIGHT-UNIT	N	5.0			Coded
2	X7	ANIM-WEAN-APPL-PREFIX-CODE	A	2.0	N		
2	X8	ANIM-WEAN-SESSION-NUM	N	8.0	N		
2	Y7	ANIM-WEAN-OPERATOR-CODE	N	5.0	N		
2	Y8	ANIM-WEAN-SESSION-DATE	N	6.0	N		
P 1	AY	ANIM-STATUS-GROUP					
2	AZ	ANIM-STATUS	N	5.0		D	Coded
2	BA	ANIM-BLDG-ROOM	A	7.0		D	

```
  2 BB ANIM-CAGE                          N   4.0     D
  2 BC ANIM-STATUS-CHANGE-DATE            N   6.0     D
  2 X9 ANIM-STATUS-APPL-PREFIX-CODE       A   2.0  N
  2 X0 ANIM-STATUS-SESSION-NUM            N   8.0  N
  2 Y9 ANIM-STATUS-OPERATOR-CODE          N   5.0  N
  2 Y0 ANIM-STATUS-SESSION-DATE           N   6.0  N
G 1 BD ANIM-CURRENT-STATUS-GROUP
  2 BE ANIM-CURRENT-STATUS                N   5.0     D  Coded
  2 BF ANIM-CURRENT-EXPERIMENT-NUM        N   6.0 F   D
  2 BG ANIM-CURRENT-BLDG-ROOM             A   7.0     D
  2 BH ANIM-LATEST-RETIRE-DATE            N   6.0     D
G 1 ZA ENTITY-AUDIT-DATA
  2 ZB EA-AUDIT-DATE                      N   6.0     D
  2 ZC EA-AUDIT-TIME                      N   6.0
  2 ZD EA-AUDIT-ENTRY-OPER-ID             N   5.0
P 1 ZE EA-AUDIT-CORRECT-REASON-PE
  2 ZF EA-CORRECT-REASON-PE-ELEMENT       A  80.0  N
G 1 ZG CREATE-AUDIT-DATA
  2 ZH CA-AUDIT-CREATION-DATE             N   6.0     D
  2 ZI CA-AUDIT-STATUS                    A   1.0     D
  2 ZJ CA-AUDIT-ENTRY-CODE                A   6.0  F
  0 SU ANIM-BIS-UIN-SP                    A   9.0     S
*      SU = AG(1,2),AH(1,7)
  0 SE EA-DATE-ENTRY-OPER-ID              N  11.0     S
*      SE=ZB(1,6),ZD(1,5)
  0 SF CA-DATE-STATUS-ENTITY-CODE         A  13.0     S
*      SF=ZH(1,6),ZI(1,1),ZJ(1,6)
```

File: (EB-8) - AUD-DATE-ESTABLISHED

Primary Key: DE-DAM-UIN
Projected Growth Rate: About 10 records per month or 10% of the DATE-ESTABLISHED file
Access Method: ADABAS NATURAL

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	DE-DAM-UIN	A	9.0	F	D	
1	AB	DE-SIRE-UIN	A	9.0	F	D	
1	AC	DE-DATE-ESTABLISHED	N	6.0		D	
1	AD	DE-DATE-PAIR-DISSOLVED	N	6.0		D	Derivable
1	X1	DE-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	DE-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	DE-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	DE-CREATE-SESSION-DATE	N	6.0	N		
1	X3	DE-DISSLV-APPL-PREFIX-CODE	A	2.0	N		
1	X4	DE-DISSLV-SESSION-NUM	N	8.0	N		
1	Y3	DE-DISSLV-OPERATOR-CODE	N	5.0	N		
1	Y4	DE-DISSLV-SESSION-DATE	N	6.0	N		
P 1	AE	DE-PAIR-OBSERVATION-GROUP					
2	AF	DE-LITTER-OBSERVATION	N	5.0			Coded
2	AG	DE-LITTER-OBSERVATION-DATE	N	6.0		D	
2	X5	DE-LITOBS-APPL-PREFIX-CODE	A	2.0	N		
2	X6	DE-LITOBS-SESSION-NUM	N	8.0	N		
2	Y5	DE-LITOBS-OPERATOR-CODE	N	5.0	N		
2	Y6	DE-LITOBS-SESSION-DATE	N	6.0	N		
1	AH	DE-PROGENY-STRAIN	A	2.0		D	
1	AI	DE-EXPERIMENT-NUM	N	6.0			
G 1	ZA	ENTITY-AUDIT-DATA					
2	ZB	EA-AUDIT-DATE	N	6.0		D	
2	ZC	EA-AUDIT-TIME	N	6.0			
2	ZD	EA-AUDIT-ENTRY-OPER-ID	N	5.0			
P 1	ZE	EA-AUDIT-CORRECT-REASON-PE					
2	ZF	EA-CORRECT-REASON-PE-ELEMENT	A	80.0	N		
G 1	ZG	CREATE-AUDIT-DATA					
2	ZH	CA-AUDIT-CREATION-DATE	N	6.0		D	
2	ZI	CA-AUDIT-STATUS	A	1.0		D	
2	ZJ	CA-AUDIT-ENTRY-CODE	A	6.0	F		
0	SE	EA-DATE-ENTRY-OPER-ID	N	11.0		S	
*		SE=ZB(1,6),ZD(1,5)					
0	SF	CA-DATE-STATUS-ENTITY-CODE	A	13.0		S	
*		SF=ZH(1,6),ZI(1,1),ZJ(1,6)					

File: {EB-9} - AUD-LITTERS

Primary Key: LIT-LITTER-UIN
Projected Growth Rate: About 30 records/month or 10% of LITTERS file
Access Method: ADABAS NATURAL

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	LIT-LITTER-UIN	A	9.0	F	D	
1	AB	LIT-DAM-UIN	A	9.0	F	D	
1	AC	LIT-SIRE-UIN	A	9.0	F	D	
1	AD	LIT-BIRTH-DATE	N	6.0		D	
P 1	AE	LIT-NUMBER-DETAIL-GROUP					Occ1=Total
2	AF	LIT-NUMBER-IN-LIT	N	2.0			Occ2,3,4 =
2	AG	LIT-NUMBER-BORN-DEAD	N	2.0			U,F,M
1	AH	LIT-GENERATION	N	4.0			
1	AI	LIT-REMARKS	A	80.0			
1	AJ	LIT-WEAN-DATE	N	6.0		D	Derivable
1	AK	LIT-LITTER-NUM	N	2.0			Derivable
G 1	AL	LIT-NOTES-GROUP					
M 2	AM	LIT-LITTER-NOTES	A	80.0			2 occ
2	AN	LIT-VENDOR-ID	A	80.0			
1	X1	LIT-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	LIT-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	LIT-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	LIT-CREATE-SESSION-DATE	N	6.0	N		
P 1	AO	LIT-PUP-WEIGHING-GROUP					
2	AP	LIT-PUP-WEIGHT	N	6.0			
2	AQ	LIT-PUP-WEIGHT-UNITS	N	5.0			Coded
2	AR	LIT-NUMBER-PUPS-WEIGHED	N	2.0			
2	AS	LIT-DATE-PUPS-WEIGHED	N	6.0			
2	X3	LIT-WEIGH-APPL-PREFIX-CODE	A	2.0	N		
2	X4	LIT-WEIGH-SESSION-NUM	N	8.0	N		
2	Y3	LIT-WEIGH-OPERATOR-CODE	N	5.0	N		
2	Y4	LIT-WEIGH-SESSION-DATE	N	6.0	N		
1	AT	LIT-EXPERIMENT-NUM	N	6.0	F	D	
G 1	ZA	ENTITY-AUDIT-DATA					
2	ZB	EA-AUDIT-DATE	N	6.0		D	
2	ZC	EA-AUDIT-TIME	N	6.0			
2	ZD	EA-AUDIT-ENTRY-OPER-ID	N	5.0			
P 1	ZE	EA-AUDIT-CORRECT-REASON-PE					
2	ZF	EA-CORRECT-REASON-PE-ELEMENT	A	80.0	N		
G 1	ZG	CREATE-AUDIT-DATA					
2	ZH	CA-AUDIT-CREATION-DATE	N	6.0		D	
2	ZI	CA-AUDIT-STATUS	A	1.0		D	
2	ZJ	CA-AUDIT-ENTRY-CODE	A	6.0	F		
0	S1	LIT-BIS-STRAIN-SB	A	2.0		S	
*		S1 = AA(1,2)					
0	SE	EA-DATE-ENTRY-OPER-ID	N	11.0		S	
*		SE=ZB(1,6),ZD(1,5)					
0	SF	CA-DATE-STATUS-ENTITY-CODE	A	13.0		S	
*		SF=ZH(1,6),ZI(1,1),ZJ(1,6)					

File: {EB-10} - AUD-ALLOCATION

Primary Key: ALC-UIN
Projected Growth Rate: About 200 records per month or 10% of the ALLOCATION file
Access Method: ADABAS NATURAL

TYL	DB	NAME	F	LENG	S	D	Remarks
1	AA	ALC-UIN	A	9.0	F	D	
1	AB	ALC-ALLOCATE-DATE	N	6.0		D	
1	AC	ALC-EXPERIMENT-NUM	N	6.0	F	D	
1	AD	ALC-BLDG-ROOM	A	7.0		D	
1	AE	ALC-CAGE	N	4.0			
1	AF	ALC-RACK	N	2.0			
1	AG	ALC-SIDE	A	1.0			
1	AH	ALC-TREATMENT	N	3.0			
1	AI	ALC-WEIGHT	N	6.0			
1	AJ	ALC-WEIGHT-UNIT	N	5.0			Coded
1	X1	ALC-CREATE-APPL-PREFIX-CODE	A	2.0	N		
1	X2	ALC-CREATE-SESSION-NUM	N	8.0	N		
1	Y1	ALC-CREATE-OPERATOR-CODE	N	5.0	N		
1	Y2	ALC-CREATE-SESSION-DATE	N	6.0	N		
G 1	ZA	ENTITY-AUDIT-DATA					
2	ZB	EA-AUDIT-DATE	N	6.0		D	
2	ZC	EA-AUDIT-TIME	N	6.0			
2	ZD	EA-AUDIT-ENTRY-OPER-ID	N	5.0			
P 1	ZE	EA-AUDIT-CORRECT-REASON-PE					
2	ZF	EA-CORRECT-REASON-PE-ELEMENT	A	80.0	N		
G 1	ZG	CREATE-AUDIT-DATA					
2	ZH	CA-AUDIT-CREATION-DATE	N	6.0		D	
2	ZI	CA-AUDIT-STATUS	A	1.0		D	
2	ZJ	CA-AUDIT-ENTRY-CODE	A	6.0	F		
0	S1	ALC-BIS-STRAIN-SB	A	2.0		S	
*		S1 = AA(1,2)					
0	SE	EA-DATE-ENTRY-OPER-ID	N	11.0		S	
*		SE=ZB(1,6),ZD(1,5)					
0	SF	CA-DATE-STATUS-ENTITY-CODE	A	13.0		S	
*		SF=ZH(1,6),ZI(1,1),ZJ(1,6)					

File: BRP-REPORT-PARAMS (Breeding Report Parameters)

Primary key: BRP-REPORT-NAME + BRP-DATE-RUN + BRP-TIME-RUN
Projected Growth Rate: 200 records per month
Access Method: ADABAS NATURAL

TYL	DB	NAME	F	LENG	S	D	Remarks
	1	BRP-REPORT-NAME	A	10.0		D	
	1	BRP-REPORT-SEQUENCE-NUMBER	N	5.0		D	
	1	BRP-SUBMIT-INTERVAL	A	1.0		D	D,W,M, or A
	1	BRP-INPUT-ONE	N	6.0			date or exp#
	1	BRP-INPUT-TWO	N	6.0			date or exp#
	1	BRP-INPUT-THREE	N	6.0			date or exp#
M	1	BRP-INPUT-UIN	A	9.0	F		uin input
M	1	BRP-INPUT-CAGE	N	4.0			cage # input
	1	BRP-OPERATOR	N	5.0			
	1	BRP-DATE-RUN	N	6.0		D	
	1	BRP-TIME-RUN	N	6.0		D	
	1	BRP-APPL-PREFIX-CODE	A	2.0			
	1	BRP-SESSION-NUM	N	8.0			

Explanation:

BRP-REPORT-NAME	holds the file name of the program that will generate the desired report.
BRP-SEQUENCE-NUMBER	holds a sequential number assigned to the report for identification purposes.
BRP-SUBMIT-INTERVAL	is a one byte field that holds information about whether the desired report is submitted daily (D), weekly (W), monthly (M), or ad hoc (A).
BRP-INPUT-ONE	can hold a date or an experiment number since both have the same definition.
BRP-INPUT-TWO	see BRP-INPUT-ONE above.
BRP-INPUT-THREE	see BRP-INPUT-ONE above.
BRP-INPUT-UIN	used in report BS018; there may be up to 5 UINs specified.
BRP-INPUT-CAGE	used in report BS018; there may be up to 5 cages specified.
BRP-OPERATOR	operator that submitted this report if the report is ad hoc. Regularly scheduled reports are submitted automatically, so this field will remain blank.

BRP-DATE-RUN	date this report was run
BRP-TIME-RUN	time this report was run

NOTE: Report BS019 requires "UIN selection criteria" as input. The exact format of the UIN selection criteria has not yet been decided, and if criteria other than UIN and cage number are required, this file may need revision.

```
DEFINE RECORD ANIMAL_REC.

ANIM_UNIQUE_ANIMAL_ID                 DATATYPE TEXT 11 CHARACTERS.
ANIM_SEX                              DATATYPE TEXT  1 CHARACTER.
ANIM_STRAIN_CODE                      DATATYPE UNSIGNED NUMERIC  5 DIGITS.
ANIM_ANIMAL_IN_CAGE_ID                DATATYPE TEXT   5 CHARACTERS.

ANIM_BIS_UIN STRUCTURE.
   ANIM_BIS_STRAIN_CODE               DATATYPE TEXT  2 CHARACTERS.
   ANIM_BIS_UIN_SEQ_NUM               DATATYPE UNSIGNED NUMERIC  7 DIGITS.
END ANIM_BIS_UIN STRUCTURE.

ANIM_BIRTH_DATE                       DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ANIM_CARCASS_ID                       DATATYPE UNSIGNED NUMERIC 11 DIGITS.
ANIM_LITTER_UIN                       DATATYPE TEXT  9 CHARACTERS.
ANIM_PHENOTYPE                        DATATYPE UNSIGNED NUMERIC  5 DIGITS.
ANIM_BREEDING_ID                      DATATYPE TEXT  5 CHARACTERS.
ANIM_ON_EXPERIMENT                    DATATYPE TEXT  1 CHARACTER.
ANIM_CAN                              DATATYPE TEXT 11 CHARACTERS.

ANIM_CREATE_APPL_PREFIX_CODE          DATATYPE TEXT  2 CHARACTERS.
ANIM_CREATE_SESSION_NUM               DATATYPE UNSIGNED NUMERIC  8 DIGITS.
ANIM_CREATE_OPERATOR_CODE             DATATYPE UNSIGNED NUMERIC  5 DIGITS.
ANIM_CREATE_SESSION_DATE              DATATYPE UNSIGNED NUMERIC  6 DIGITS.

ANIM_CREATE_DATE                      DATATYPE UNSIGNED NUMERIC  6 DIGITS.

ANIM_NOTES_GROUP STRUCTURE.
   ANIM_VENDORS_ID                    DATATYPE TEXT 80 CHARACTERS.
   ANIM_NOTES                         OCCURS  2 TIMES
                                      DATATYPE TEXT 80 CHARACTERS.
   ANIM_NOTES_APPL_PREFIX_CODE        DATATYPE TEXT  2 CHARACTERS.
   ANIM_NOTES_SESSION_NUM             DATATYPE UNSIGNED NUMERIC  8 DIGITS.
   ANIM_NOTES_OPERATOR_CODE           DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_NOTES_SESSION_DATE            DATATYPE UNSIGNED NUMERIC  6 DIGITS.
END ANIM_NOTES_GROUP STRUCTURE.

ANIM_REMOVAL_GROUP STRUCTURE.
   ANIM_REMOVAL_DESTINATION           DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_REMOVAL_DATE                  DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   ANIM_REMOVAL_APPL_PREFIX_CODE      DATATYPE TEXT  2 CHARACTERS.
   ANIM_REMOVAL_SESSION_NUM           DATATYPE UNSIGNED NUMERIC  8 DIGITS.
   ANIM_REMOVAL_OPERATOR_CODE         DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_REMOVAL_SESSION_DATE          DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END ANIM_REMOVAL_GROUP STRUCTURE.

ANIM_WEAN_GROUP STRUCTURE.
   ANIM_WEAN_WEIGHT                   DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   ANIM_WEAN_WEIGHT_UNIT              DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_WEAN_APPL_PREFIX_CODE         DATATYPE TEXT  2 CHARACTERS.
   ANIM_WEAN_SESSION_NUM              DATATYPE UNSIGNED NUMERIC  8 DIGITS.
```

```
   ANIM_WEAN_OPERATOR_CODE          DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_WEAN_SESSION_DATE           DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END ANIM_WEAN_GROUP STRUCTURE.

ANIM_STATUS_GROUP STRUCTURE         OCCURS 50 TIMES.
   ANIM_STATUS                      DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_BLDG_ROOM                   DATATYPE TEXT  7 CHARACTERS.
   ANIM_CAGE                        DATATYPE UNSIGNED NUMERIC  4 DIGITS.
   ANIM_STATUS_CHANGE_DATE          DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   ANIM_STATUS_APPL_PREFIX_CODE     DATATYPE TEXT  2 CHARACTERS.
   ANIM_STATUS_SESSION_NUM          DATATYPE UNSIGNED NUMERIC  8 DIGITS.
   ANIM_STATUS_OPERATOR_CODE        DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_STATUS_SESSION_DATE         DATATYPE UNSIGNED NUMERIC  6 DIGITS.
END ANIM_STATUS_GROUP STRUCTURE.

ANIM_CURRENT_STATUS_GROUP STRUCTURE.
   ANIM_CURRENT_STATUS              DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   ANIM_CURRENT_EXPERIMENT_NUM      DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   ANIM_CURRENT_BLDG_ROOM           DATATYPE TEXT  7 CHARACTERS.
   ANIM_LATEST_RETIRE_DATE          DATATYPE UNSIGNED NUMERIC  6 DIGITS.
END ANIM_CURRENT_STATUS_GROUP.

END ANIMAL_REC.
```

```
DEFINE RECORD DATE_ESTABLISHED_REC.

DE_DAM_UIN                              DATATYPE TEXT   9 CHARACTERS.
DE_SIRE_UIN                             DATATYPE TEXT   9 CHARACTERS.
DE_DATE_ESTABLISHED                     DATATYPE UNSIGNED NUMERIC  6 DIGITS.
DE_DATE_PAIR_DISSOLVED                  DATATYPE UNSIGNED NUMERIC  6 DIGITS.

DE_CREATE_APPL_PREFIX_CODE              DATATYPE TEXT   2 CHARACTERS.
DE_CREATE_SESSION_NUM                   DATATYPE UNSIGNED NUMERIC  8 DIGITS.
DE_CREATE_OPERATOR_CODE                 DATATYPE UNSIGNED NUMERIC  5 DIGITS.
DE_CREATE_SESSION_DATE                  DATATYPE UNSIGNED NUMERIC  6 DIGITS.

DE_DISSLV_APPL_PREFIX_CODE              DATATYPE TEXT   2 CHARACTERS.
DE_DISSLV_SESSION_NUM                   DATATYPE UNSIGNED NUMERIC  8 DIGITS.
DE_DISSLV_OPERATOR_CODE                 DATATYPE UNSIGNED NUMERIC  5 DIGITS.
DE_DISSLV_SESSION_DATE                  DATATYPE UNSIGNED NUMERIC  6 DIGITS.

DE_PAIR_OBSERVATION_GROUP STRUCTURE OCCURS 30 TIMES.
   DE_LITTER_OBVSERVATION               DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   DE_LITTER_OBSERVATION_DATE           DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   DE_LITOBS_APPL_PREFIX_CODE           DATATYPE TEXT   2 CHARACTERS.
   DE_LITOBS_SESSION_NUM                DATATYPE UNSIGNED NUMERIC  8 DIGITS.
   DE_LITOBS_OPERATOR_CODE              DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   DE_LITOBS_SESSION_DATE               DATATYPE UNSIGNED NUMERIC  6 DIGITS.
END DE_PAIR_OBSERVATION_GROUP STRUCTURE.

DE_PROGENY_STRAIN_CODE                  DATATYPE TEXT   2 CHARACTERS.
DE_EXPERIMENT_NUM                       DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END DATE_ESTABLISHED_REC.
```

```
DEFINE RECORD LITTERS_REC.

LIT_LITTER_UIN                          DATATYPE TEXT  9 CHARACTERS.
LIT_DAM_UIN                             DATATYPE TEXT  9 CHARACTERS.
LIT_SIRE_UIN                            DATATYPE TEXT  9 CHARACTERS.
LIT_BIRTH_DATE                          DATATYPE UNSIGNED NUMERIC  6 DIGITS.

LIT_NUMBER_DETAIL_GROUP STRUCTURE       OCCURS  4 TIMES.
   LIT_NUMBER_IN_LIT                    DATATYPE UNSIGNED NUMERIC  2 DIGITS.
   LIT_NUMBER_BORN_DEAD                 DATATYPE UNSIGNED NUMERIC  2 DIGITS.
END LIT_NUMBER_DETAIL_GROUP STRUCTURE.

LIT_GENERATION                          DATATYPE UNSIGNED NUMERIC  4 DIGITS.
LIT_REMARKS                             DATATYPE TEXT 80 CHARACTERS.
LIT_WEAN_DATE                           DATATYPE UNSIGNED NUMERIC  6 DIGITS.
LIT_LITTER_NUM                          DATATYPE UNSIGNED NUMERIC  2 DIGITS.

LIT_NOTES_GROUP STRUCTURE.
   LIT_LITTER_NOTES                     OCCURS  2 TIMES
                                        DATATYPE TEXT 80 CHARACTERS.
   LIT_VENDOR_ID                        DATATYPE TEXT 80 CHARACTERS.
END LIT_NOTES_GROUP STRUCTURE.

LIT_CREATE_APPL_PREFIX_CODE             DATATYPE TEXT  2 CHARACTERS.
LIT_CREATE_SESSION_NUM                  DATATYPE UNSIGNED NUMERIC  8 DIGITS.
LIT_CREATE_OPERATOR_CODE                DATATYPE UNSIGNED NUMERIC  5 DIGITS.
LIT_CREATE_SESSION_DATE                 DATATYPE UNSIGNED NUMERIC  6 DIGITS.

LIT_PUP_WEIGHING_GROUP STRUCTURE        OCCURS 99 TIMES.
   LIT_PUP_WEIGHT                       DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   LIT_PUP_WEIGHT_UNITS                 DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   LIT_NUMBER_PUPS_WEIGHED              DATATYPE UNSIGNED NUMERIC  2 DIGITS.
   LIT_DATE_PUPS_WEIGHED                DATATYPE UNSIGNED NUMERIC  6 DIGITS.
   LIT_WEIGH_APPL_PREFIX_CODE           DATATYPE TEXT  2 CHARACTERS.
   LIT_WEIGH_SESSION_NUM                DATATYPE UNSIGNED NUMERIC  8 DIGITS.
   LIT_WEIGH_OPERATOR_CODE              DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   LIT_WEIGH_SESSION_DATE               DATATYPE UNSIGNED NUMERIC  6 DIGITS.
END LIT_PUP_WEIGHING_GROUP STRUCTURE.
LIT_EXPERIMENT_NUM                      DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END LITTERS_REC.
```

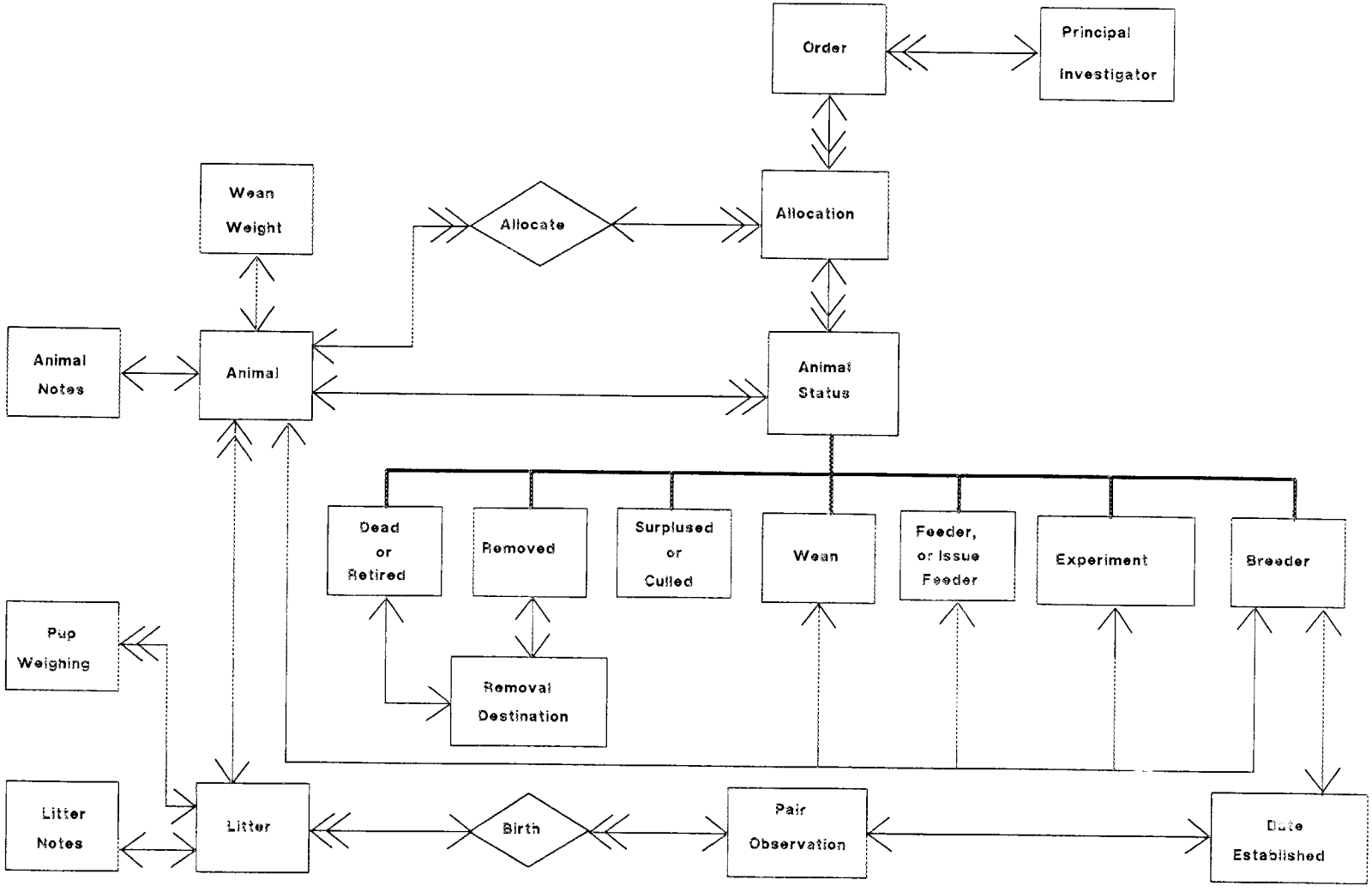
```
DEFINE RECORD ALLOCATION_REC.

ALC_UIN                          DATATYPE TEXT   9 CHARACTERS.
ALC_ALLOCATE_DATE                DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ALC_EXPERIMENT_NUM               DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ALC_BLDG_ROOM                    DATATYPE TEXT   7 CHARACTERS.
ALC_CAGE                         DATATYPE UNSIGNED NUMERIC  4 DIGITS.
ALC_RACK                         DATATYPE UNSIGNED NUMERIC  2 DIGITS.
ALC_SIDE                         DATATYPE TEXT   1 CHARACTER.
ALC_TREATMENT                    DATATYPE UNSIGNED NUMERIC  3 DIGITS.
ALC_WEIGHT                       DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ALC_WEIGHT_UNIT                  DATATYPE UNSIGNED NUMERIC  5 DIGITS.

ALC_CREATE_APPL_PREFIX_CODE      DATATYPE TEXT   2 CHARACTERS.
ALC_CREATE_SESSION_NUM           DATATYPE UNSIGNED NUMERIC  8 DIGITS.
ALC_CREATE_OPERATOR_CODE         DATATYPE UNSIGNED NUMERIC  5 DIGITS.
ALC_CREATE_SESSION_DATE          DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END ALLOCATION_REC.
```

```
DEFINE RECORD ORDERS_REC.

ORD_EXPERIMENT_NUMBER                DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ORD_NUMBER_REQUESTED                 DATATYPE UNSIGNED NUMERIC  3 DIGITS.
ORD_NUMBER_DELIVERED                 DATATYPE UNSIGNED NUMERIC  3 DIGITS.
ORD_DATE_ORDERED                     DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ORD_DATE_REQUIRED                    DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ORD_DATE_FILLED                      DATATYPE UNSIGNED NUMERIC  6 DIGITS.
ORD_STRAIN                           DATATYPE TEXT  2 CHARACTERS.
ORD_SEX                              DATATYPE TEXT  1 CHARACTER.
ORD_CONTACT                          DATATYPE TEXT 13 CHARACTERS.
ORD_BLDG_ROOM                        DATATYPE TEXT  7 CHARACTERS.
ORD_SPECIAL_REQUIREMENTS             DATATYPE TEXT 20 CHARACTERS.

ORD_CREATE_APPL_PREFIX_CODE          DATATYPE TEXT  2 CHARACTERS.
ORD_CREATE_SESSION_NUM               DATATYPE UNSIGNED NUMERIC  8 DIGITS.
ORD_CREATE_OPERATOR_CODE             DATATYPE UNSIGNED NUMERIC  5 DIGITS.
ORD_CREATE_SESSION_DATE              DATATYPE UNSIGNED NUMERIC  6 DIGITS.

END ORDERS_REC.
```

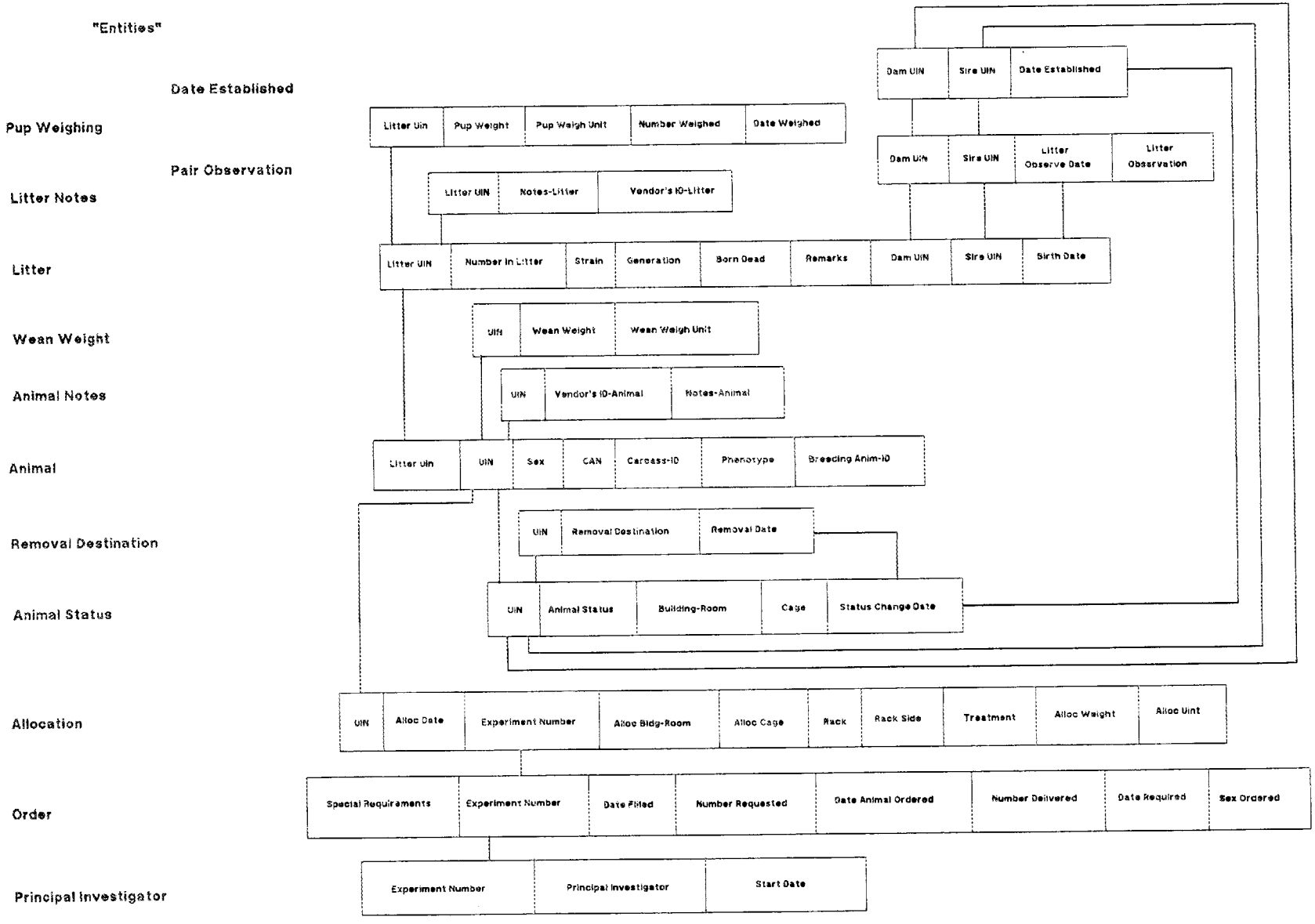
```
DEFINE RECORD CAGE_REC.

CAG_CAGE_NUMBER                    DATATYPE UNSIGNED NUMERIC  4 DIGITS.
CAG_BLDG_ROOM                      DATATYPE TEXT  7 CHARACTERS.
CAG_STATUS                         DATATYPE UNSIGNED NUMERIC  5 DIGITS.

CAG_ORIG_UIN                       DATATYPE TEXT  9 CHARACTERS.
CAG_ORIGINAL_COUNT                 DATATYPE UNSIGNED NUMERIC  2 DIGITS.
CAG_ORIGINAL_DATE                  DATATYPE UNSIGNED NUMERIC  6 DIGITS.

CAG_BLOCK_UIN                      DATATYPE TEXT  9 CHARACTERS.
CAG_BLOCK_COUNT                    DATATYPE UNSIGNED NUMERIC  2 DIGITS.
CAG_BLOCK_GENERATION               DATATYPE UNSIGNED NUMERIC  4 DIGITS.

CAG_ANIMAL_GROUP STRUCTURE         OCCURS  4 TIMES.
   CAG_ANIMAL_UIN                  DATATYPE TEXT  9 CHARACTERS.
   CAG_ANIMAL_SEX                  DATATYPE TEXT  1 CHARACTER.
   CAG_ANIMAL_PHENOTYPE            DATATYPE UNSIGNED NUMERIC  5 DIGITS.
   CAG_ANIMAL_GENERATION           DATATYPE UNSIGNED NUMERIC  4 DIGITS.
   CAG_ANIMAL_FLAG                 DATATYPE TEXT  1 CHARACTER.
END CAG_ANIMAL_GROUP_STRUCTURE.

END CAGE_REC.
```

```
DEFINE RECORD BREEDING_RPT_REC.

BRP-REPORT-NAME                    DATATYPE TEXT 10 CHARACTERS.
BRP-REPORT-SEQUENCE-NUMBER         DATATYPE UNSIGNED NUMERIC 5 DIGITS.
BRP-SUBMIT-INTERVAL                DATATYPE TEXT 1 CHARACTERS.
BRP-INPUT-ONE                      DATATYPE UNSIGNED NUMERIC 6 DIGITS.
BRP-INPUT-TWO                      DATATYPE UNSIGNED NUMERIC 6 DIGITS.
BRP-INPUT-THREE                    DATATYPE UNSIGNED NUMERIC 6 DIGITS.
BRP-INPUT-UIN                      OCCURS 5 TIMES
                                   DATATYPE TEXT 9 CHARACTERS.
BRP-INPUT-CAGE                     OCCURS 5 TIMES
                                   DATATYPE UNSIGNED NUMERIC 4 DIGITS.
BRP-OPERATOR                       DATATYPE UNSIGNED NUMERIC 5 DIGITS.
BRP-DATE-RUN                       DATATYPE UNSIGNED NUMERIC 6 DIGITS.
BRP-TIME-RUN                       DATATYPE UNSIGNED NUMERIC 6 DIGITS.
BRP-APPL-PREFIX-CODE               DATATYPE TEXT 2 CHARACTERS.
BRP-SESSION-NUM                    DATATYPE UNSIGNED NUMERIC 8 DIGITS.

END BREEDING_RPT_REC.
```

ENTITY DIAGRAM FOR BMSS



LOGICAL STRUCTURE FOR BMSS



APPENDIX D

MODULE SPECIFICATIONS FOR SCREENS

Module Name: Screen A.1 (Logon)

Descriptive Title: Logon to BREEDING Screens

Method of Invocation: User Select

Inputs: Operator ID, Driver Required

Outputs: Session Information

Called Modules: Post-Logon

Function: To prompt the operator for certain session parameters required in other BREEDING processing.

Assumptions:

None made.

Process:

(1) Input the operator ID and name of the required screen driver program.

(2) Display the Building and Room prompt and accept the operator input. Validate the input value and proceed or prompt again with an error message.

(3) Display the Terminal prompt and accept operator input. Validate the input value and proceed or prompt again with an error message.

(4) Display the Keyboard ID prompt and accept operator input. Validate the input value and proceed or prompt again with an error message.

(5) After all data are correctly entered, call the POST-LOGON module, passing control to it.

Note: Some provision for an abort of this screen must be made so that the operator cannot be stranded in this screen. The convention selected should be extended to all screens where an abort seems like a possible action.

Module Name: Screen A.2

Descriptive Title: Ad Hoc Report Selection Screen

Method of Invocation: Called from NARSS Main Menu

Inputs: Operator ID and date report requested

Outputs: One record per each operator validated report request (which will contain the information necessary to run zero or one report for all Ad Hoc reports except the Target Trace Report - there are 6 BREEDING and 3 EB reports that may be selected here, as well as the target trace report.)

Called Modules: None

Function: To allow the operator to specify parameters required to run one or more separate ad hoc reports.

Assumptions:

1. If two versions of the same ad hoc report are desired (for example, running the birth report for two separate date ranges), then the operator must enter and confirm each separately (except for the target trace report, where several UINs, Rooms, or Cages may be specified).

2. Requests are written to a disk or data base file for access by the nightly transaction run. The reports are queued for execution here, but actually submitted by a separate process that occurs after census data have been applied to the Historical data base.

3. All reports execute using the Historical data base data.

Process:

(1) Display the input screen allowing report parameter entry.

(2) Accept and verify the data entered (make certain that dates are valid and that experiments referenced actually exist).

(3) If the operator has entered the data for any report, write one record to the BMSS Report Parameters file containing all the report specifications.

Module Name: ALLOCATE

Descriptive Title: Allocation Screen

Method of Invocation: Called from CAGE-DRIVER

Inputs: No initial inputs

Outputs: Changes to ANIMAL and CAGE records, new ALLOCATION file records, INLIFE initial entries for newly allocated records.

Called Modules: Read_Allocate_Data, Determine_Next_Cage, Update_Inlife_Record

Function: To allow the allocation of animals to be performed based on the Study Definition allocation parameters, the current contents of the selected rack, and the current contents of the issue cages.

Assumptions:

1. Manual allocation is allowed, but must be specified in the Study Definition data if it is to be used.
2. Animals in issue cages may be found to be unusable. Such an animal may be eliminated via prompts similar those used in the Removal screen.

Process 1:

(1) Examine the session data and determine if the balance has been calibrated. If not, prompt for calibration information.

(2) Obtain the number of animal technicians.

(3) For the number of animal technicians entered, obtain each individual technician ID via screen prompts.

(4) Prompt for the first cage number on the rack to be filled and allow the operator to enter or barcode the cage number.

(5) Verify the cage number by attempting to find it in the Study Definition allocation data (this data is currently assumed to have been loaded to the INLIFE files).

(6) If (5) results in no records found, then issue an error message and reprompt.

(7) If (5) results in one record found, then read that record by calling READ_ALLOCATE_DATA. The record read should contain

the FIRST-CAGE-NUMBER, EXP-TEZST, BLDG-ROOM, RACK, SIDE, SEX (may be B(oth)), STRAIN, and NUMBER-ANIMALS-CAGE.

(8) Read through the cages on the rack and set up a structure which defines the locations into which animals may be placed. Note that some animal requirements may be handled here (such as clip or number of animals per cage). Also, using the allocation parameters, determine the number of animals required to fill this rack -- the possibility of culls and loss of animals during allocation should be taken into account here by estimating a value greater than the actual number of animals required.

(9) Determine if weight ranking is required, and if so:

(10) For the number of animals required (from step 8):

(11) Prompt for and obtain an issue cage number.

(12) Weigh the entire issue cage and verify its weight against any Study Definition imposed cage-weight limits. For certain Study Definition restrictions, the cage census should immediately be checked (for example, if only females are allowed in this allocation).

(13) For the number of animals in this cage:

(14) Weigh the animal

(15) If this is not the last animal in the issue cage, then prompt for a new issue cage number. Animals for which an individual weight exists should be kept in separate cages.

(16) Store the weight for the animal (probably in a temporary structure to be used only during allocation -- but this information eventually needs to be stored in the INLIFE or EB files)

(17) Increment the ACTUAL-ANIMAL-COUNT by 1.

(18) If the ACTUAL-ANIMAL-COUNT equals the NUMBER-OF-ANIMALS-REQUIRED, then enough animals have been weighed, escape this loop and go to (19).

(19) If allowed by Study Definition, prompt for the cage subset. Some allocation requirements may force randomization across an entire rack, so this option may not always be allowed.

(20) Accept and check the cage subset information...the data entered here may be Row, Column, or a starting cage number.

(21) Depending on the data entered and verified in (20) prompt for additional information (list or rows or columns, or ending cage number) and verify that the data makes sense based on the current rack selected.

Note: For now, it is assumed that if starting and ending cage numbers are to be allowed then no cage number between those values is located outside this rack. If the location of cage numbers in a rack is not sequential, then Study Definition will need to make the rack configuration available to Allocate. Also, note that the weighing of animals for a subset of cages should really be done after the subset is known, not before. This may not be an issue because weight randomization may require allocation to any rack location, and not allow a subset.

(22) Move existing rack information to temporaries used to allow scrolling of information (up and down, left and right -- the rack may be 10x10, but the on-screen display will allow only 6x7 (maximum))...the information used to indicate rack contents and indicate the next allocation cage location will have to be buffered for output.

Note: If only certain cages, rows, or columns have been selected, then it is asusmed that only these locations should be displayed, even if other locations on the rack have contents.

(23) Start a loop that will execute for the number of animals required to fill the empty rack locations:

(24) Prompt for the issue cage and allow the operator to enter the cage number.

(25) Determine the contents of the issue cage and display information at the top of the screen (this will use a modified version of Display Cage (A.13)).

(26) If the cage or animal weight is required and it was not done in a complete weighing above, then prompt for cage weight.

(27) For each animal in the issue cage:

(28) If this is a weight ranked allocation, then there is only one animal in the issue cage and its weight was already recorded, but all code leads to (29) for

consistency -- the operator will then hit RETURN to accept the previously recorded weight.

(29) But if it was not weighed and the weight is required, then prompt to weigh:

(30) At this point, the operator must handle the animal, so it is an appropriate point to determine if the animal is suitable for use on the experiment. Prompt for WEIGH, RETURN, END, REMOVAL, or ISSUE CAGE:

(31) If WEIGH, then weigh the animal.

(32) If RETURN, then no weighing is to be done, so if a previous weighing was made, use its data.

(33) If END, then terminate the Allocation process.

(34) If REMOVAL, then prompt for and accept the REASON, DISPOSITION, and CID, verifying each as appropriate.

Note: During ranked (age or weight) allocations, the removal of an animal is assumed to cause each animal ranked lower in the list to move up one spot. This ties into the fact that more animals will be available to use than are expected to be needed. However, because some animals may escape or expire during a ranked allocation (and not be noticed until later) it is necessary to establish a reserve pool. It is suggested that this pool be established by skipping every 10th animal or so in the ranking. These animals may then be used during missing corrections to fit into a rank gap as close as possible. An alternative would be to have a separate pool of weighed animals.

(35) Issue BREEDING transactions to indicate that this animal has been removed...it is assumed that this can be done immediately without additional confirmation.

(36) If ISSUE CAGE, escape the current processing loop and return to (25)

(37) Call DETERMINE-NEXT-CAGE. It uses the animal from the issue cage, the current rack configuration and the Study Definition rules to determine a suitable location on the rack for this animal. If the Study Definition parameters indicate that this is a manual allocation, then it prompts the operator to enter an allocation cage number.

Note: rules like "No Litter Mates in Cage" and "Required Sex = F" are handled in DETERMINE-NEXT-CAGE.

(38) If no valid cage exists for this animal, then issue a warning message, and allow certain operators to override the allocation cage selection. However, when such an override is encountered, force the operator to enter the reason for this action.

Note: This override information should be stored in the database somewhere for output and future reference.

(39) If a valid cage exists, then this animal is ready to actually be Allocated:

(40) Prompt the operator to clip, move the animal to the indicated allocation cage, and verify.

Note: Because of the removal mechanism outlined above, culls and removals are not allowed here...if an animal becomes unusable between the prompt allowing removal and here, that will have to be handled in the error correction phase below.

(41) Generate the ALLOCATION record.

(42) Update the cage contents (in BREEDING for the issue cage and in INLIFE or EB for the allocation cage, use UPDATE_INLIFE_RECORD for the INLIFE or EB data).

(43) Increment the ANIMALS-ALLOCATED count. Use it to determine if all animals of a particular earclip are entered. If so, make visual and audible alarm to alert the operator that the next animal will have a different earclip.

(44) If all animals required are allocated, then terminate this allocation process...finished.

(45) Termination Processing: in all termination cases, ask the user if there are any missing animals and if YES:

(46) Get the cage and clip for the missing animal:

(47) Remove and deallocate that animal.

(48) Reenter processing at (24) to obtain the new issue cage and to pass thruough the process by which an animal is weighed, clipped, and assigned an allocation cage.

(49) If there are no missing animals, then exit the Allocation process and return to the calling program.

Module Name: Screen A.4a

Descriptive Title: Enter Animal Orders

Method of Invocation: User Inputs Enter Animal Orders Function

Inputs: User Input Only

Outputs: Update the Orders File with User Data

Called Modules: None

Function: To track animal orders so that appropriate breeding levels can be maintained, and to ensure that orders are filled in a timely manner.

Assumptions:

1. Upon completion of entering orders, the user presses the Gold 9 key sequence to return to the NARSS menu. Before exiting this function all orders entered must be complete. All required fields should be entered. Before completing a new order, the user may blank out all fields in the current line and terminate the function or reenter data.

2. Allow entry of Date Filled and zero Number Delivered. ** NCTR approved **

3. The user enters data into the appropriate fields on the screen without being prompted.

4. Only one operator at a time can enter orders.

Process:

(1) Display Operator, Date, Time, Building, and Room in screen header. Display order header.

(2) Require the user to enter a combination of experiment number and date requested and perform the following data check:

Search the Order file for the Experiment Number and Date Request entered.

If the record exists, display the existing data in the current line. Allow the user to change any data in the line. Verify that this does not create duplicate records and go to step 9. If the user enters "0000-00" for Experiment Number, mark the order as deleted. Do not

delete the record. If the user enters "CANCEL" in the Date Filled field, mark the order as cancelled.

If the record does not exist, allow further entry of data to add the record upon successful completion of entry and confirmation.

(3) Require the entry of Building and Room and perform the following data check:

Search the Validation Tables for Building and Room. If the combination is valid, allow further entry.

If the combination is not valid, prompt the user to reenter the data.

(4) Require the entry of at least one name for P.I./Contact. At most, two names may be entered. The names are not validated.

(5) Require the entry of Strain and perform the following data check:

Search the Strain Table for this strain. If the strain is found in the table, continue data entry. If the strain is not found in the table, display error and prompt for reentry.

(6) Require the entry of Sex and verify that "M", "F", or "U" was entered. If data entered was not correct, display error message. If the data entered was correct, continue with data entry.

(7) Require the entry of Number Required for this strain and sex and verify that the data entered is numeric.

(8) Allow entry of Special Requirements. This is a text string, and no error checking is performed. This field is optional.

(9) Update the order file with all the data entered plus date ordered and go to step 2.

Module Name: Screen A.4b

Descriptive Title: Fill Animal Orders

Method of Invocation: User Inputs Fill Animal Orders Function

Inputs: User Input Only

Outputs: Update the Orders File with User Data

Called Modules: None

Function: To track animal orders so that appropriate breeding levels can be maintained and to ensure that orders are filled in a timely manner.

Assumptions:

1. There will be a table of people authorized to perform the Fill Animal Orders function that will be used to determine valid users for this function (Operators Functions Table). ** NCTR approved **

2. Only one operator at a time can fill animal orders.

Process:

(1) Display Operator, Date, Time, Building, and Room in screen header.

(2) Verify that the operator is allowed access to this function. Search the Operators Functions Table for this operator and function. If the operator is allowed access, continue with this function. If the operator is not allowed access, display error and prompt the user to enter a function or cage.

(3) Prompt the user to enter Experiment Number. If the Experiment Number is not 0000-00, search the Order file for the record(s) with this Experiment Number.

If only one record is found, display the order information (Experiment Number, Building, Room, Date Requested, P.I./Contact, Strain, Sex, Number Requested, and Special Requirements) and move today's date to the Date Filled field.

If more than one record is found for this experiment, prompt the user to enter Date Requested to determine which order the user is to fill.

If the Experiment Number is 0000-00, terminate filling orders by going to step 7.

(4) Prompt the user to enter Number Delivered. Check the number for validity. Display error if invalid. If valid, continue.

(5) Prompt the user to enter Date Filled. If the user presses Return, accept default of today's date. If the user enters a date, check for validity. If valid, continue. If invalid, repeat step 5.

(6) Go to step 3.

(7) Prompt the user to enter a function or cage.

(8) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(9) If the cage is invalid, display error message and reprompt the user for input.

(10) If a valid cage is entered, prompt the user to enter a function or cage (go to step 7).

(11) When a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(12) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(13) Call the module determined in step 11.

Module Name: Screen A.4c

Descriptive Title: Display Animal Orders

Method of Invocation: User inputs Display Animal Orders function.

Inputs: User input only.

Outputs: Update the Orders file with user data.

Called Modules: None.

Function: To track animal orders so that appropriate breeding levels can be maintained and to ensure that orders are filled in a timely manner.

Assumptions:

1. If there are any animals in this room that are of the strain required in the order, display the order. The <u>number of animals required by an order</u> is not used to determine the display of orders.

2. <u>The user enters a blank to terminate this function.</u>

Process:

(1) Display Operator, Date, Time, Building, and Room in screen header.

(2) Prompt the user to enter room (R) or colony (C). Display "R" initially. Verify that "R", "C", or blank is entered. Display error message if anything but "R", "C", or blank is entered.

(3) If the user enters "R", search the Orders file for orders not yet filled (Date-Filled = ' ') and sort by Experiment Number and Date Required.

For each order found, search the Animal file for those animals in this building and room that are of the same strain and sex as required in the order record and are not currently allocated to an experiment.

Display the detail line for this order (Experiment Number, Date Required, P.I./Contact, Strain, Sex, Number Requested, and Special Requirements.)

If no orders can be filled, display message.

(4) If the user enters "C", verify that this Operator is allowed access to this function.

If access is not allowed, display error message and go to step 6.

If access is allowed, search the Orders file for orders not yet filled (Date-Filled = ' ') and sort by Experiment Number and Date Required.

For each order found, search the Animal file for those animals that are of the same strain and sex as required in the order record and are not currently allocated to an experiment.

Display the detail line for this order (Experiment Number, Date Required, P.I./Contact, Strain, Sex, Number Requested, and Special Requirements.)

If no orders can be filled, display message.

(5) If the user enters a blank, go to step 6.

(6) Prompt the user to enter a function or cage.

(7) If a cage is entered, then do the following:

Determine if the cage is valid by reading the Validation Tables.

(8) If the cage is invalid, display error message and reprompt the user for input.

(9) If the cage is valid, prompt the user to enter a function or cage (go to step 6).

(10) When a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(11) Call the module determined in step 10.

Module Name: Screen A.5

Descriptive Title: Display Retirement Candidates

Method of Invocation: User Inputs Display Retirement Candidates Function

Inputs: None

Outputs: Data Displayed to the Screen Only

Called Modules: None

Function: To display those breeders that have reached retirement age.

Assumptions:

1. Only breeder pairs are included in this report. There may be animals in breeder status that are not paired, and these are not included.

Process:

(1) Display the Operator, Date, Time, Building, and Room in the screen header.

(2) Read the Cage file to determine whether there are female breeders in this Room using Cage Status, Building-Room, and Animal Sex. Sort the Cage file by Cage. If there are breeders in this room, do the following for each Cage record found:

(3) Read the Animal Group Periodic Group in the Cage file. If Sex is female then do the following:

(4) Find the record in the Date Established file corresponding to the female's UIN where the Date Pair Dissolved is blank.

(5) Using the strain of the female (the first two characters of Dam UIN), search the Strain Table for this strain and the number of days at which to retire animals of this strain.

(6) Find the animal record with a UIN the same as the female's UIN.

(7) Calculate the age of the female using today's date and Animal Birth Date from the Animal file.

(8) If the age of the female is greater than or equal to the number of days at which to retire animals of this strain, do the following:

(9) Calculate Days Established from Date Established and today's date.

(10) Display Cage (from Cage file), Dam UIN (from Date Established file), Sire UIN (from Date Established file), Days Established (calculated), and the age of the female (calculated).

(11) If there are no retirement candidates in this room, display message.

(12) If there are no breeder cages present in this room, display message.

(13) Prompt the user to enter a function or cage.

(14) If a cage is entered, then do the following:

Determine if the cage is valid by reading the Validation Tables.

(15) If the cage is invalid, display error message and reprompt the user for input.

(16) If the cage is valid, go to step 13.

(17) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(18) Call the function determined in step 17.

Module Name: Screen A.6

Descriptive Title: Display Unproductive Breeders

Method of Invocation: User Inputs Display Unproductive Breeders Function

Inputs: None

Outputs: Data Displayed to Screen Only

Called Modules: None

Function: To display those breeders that have not produced a litter within the number of days set for their strain in the Strain Table.

Assumptions:

1. Breeders are found in breeder cages, and may be paired or single. Single breeders are unproductive by default.

2. The Days Established and litter data on this screen are those of the female.

Process:

(1) Display Operator, Date, Time, Building, and Room in the screen header.

(2) Read the Cage file to determine if there are female breeders in this Room using Cage Status, Building-Room, and Animal Sex. Sort the Cage file by Cage.

(3) If there are breeders in this room, do the following for each Cage record found (4 - 17):

(4) Read the Animal Group Periodic Group in the Cage file.

(5) For each animal, find the corresponding record in the DATE-ESTABLISHED file.

(6) If there is no record, then the animal is a single breeder and is unproductive by default. Store appropriate information, and go to the next animal.

(7) If there is a record, then do the following (8 - 16):

(8) If Sex is male, do nothing. Males are counted with the corresponding female.

(9) If Sex is female then do the following (10 - 16):

(10) Using the female and male UINs found in the Animal Group, store the date the pair was established (Date Established).

(11) Calculate the number of days the pair has been established (Days Established), using today's date and Date Established.

(12) Read the Pair Observation Group Periodic Group to determine the date of the last littered observation (Observation Date).

(13) Using the Observation Date and today's date, calculate the number of days since the last litter was born.

(14) Search the Strain Table for the female's strain. Compare the unproductive time in the Strain Table to the number of days since the last litter was born. If the days since the last litter was born is greater than or equal to the unproductive time read from the Strain Table, do the following:

(15) Using the Dam UIN, Sire UIN, and Observation Date, find the record in the Litter file for the last litter born. Retrieve the Litter Number.

(16) Display Cage, Female UIN, Male UIN (all from Cage file), Days Established (calculated), number of days since the last litter was born (calculated), and Litter Number (from Litter file).

(17) If there are no unproductive breeders in this room, display message.

(18) If there are no breeder cages present in this room, display message.

(19) Prompt the user to enter a function or cage.

(20) If a cage is entered, then do the following:

Determine if the cage is valid by reading the Validation Tables.

(21) If the cage is invalid, display error message and reprompt the user for input.

(22) If the cage is valid, go to step 19.

(23) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(24) Call the function determined in step 23.

Module Name: Screen A.7

Descriptive Title: Display Feeders by Age Range

Method of Invocation: User Inputs Display Feeders by Age Range Function

Inputs: User Enters Age Range for Animals to Display

Outputs: Data Displayed to Screen Only

Called Modules: None

Function: To display feeder animals in this room that are between the ages entered by the Operator.

Assumptions:

1. Feeders will always be in Feeder Cages.

Process:

(1) Prompt the user to enter the minimum age of the range. If the age is invalid, display message and reprompt. If the age is valid, continue.

(2) Prompt the user to enter the maximum age of the range. If the age is invalid, display message and reprompt. If the age is valid, continue.

(3) Display in the screen header the Operator, Date, Time, Building, Room, and the age range entered by the Operator.

(4) Using Cage Status and Building Room, search the Cage file for Feeder cages in this Building Room. For each cage found do the following:

Read the Animal Group Periodic Group to retrieve Animal UIN.

For each Animal UIN in the group do the following:

Find the record corresponding to this Animal UIN in the Animal file. Retrieve the Birth Date of this animal.

Using today's date and Birth Date, calculate the age of the animal.

If the age of the animal is between the ages in the range entered by the user, display Cage, UIN, Sex (all from the Cage file), and the age of the animal (calculated).

(5) If there are no feeders of this age range in this room, display message.

(6) Prompt the user to enter a function or cage.

(7) If a cage is entered, then do the following:

Determine if the cage is valid by reading the Validation Tables.

(8) If the cage is invalid, display error message and reprompt the user for input.

(9) If the cage is valid, go to step 6.

(10) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(11) Call the function determined in step 10.

Module Name: Screen A.8

Descriptive Title: Display Issue-Feeders by Age Range

Method of Invocation: User Inputs Display Issue-Feeders by Age Range Function

Inputs: User Enters Age Range for Animals to Display

Outputs: Data Displayed to Screen Only

Called Modules: None

Function: Display feeders with ages that fall in the age range entered by the Operator and that are in issue cages.

Assumptions:

1. All Issue-Feeders will be in Issue-Feeder cages. It is sufficient to check the status of the cage to determine the status of the animals in the cage.

Process:

(1) Prompt the user to enter the minimum age of the range. If the age is invalid, display message and reprompt. If the age is valid, continue.

(2) Prompt the user to enter the maximum age of the range. If the age is invalid, display message and reprompt. If the age is valid, continue.

(3) Display in the screen header the Operator, Date, Time, Building, Room, and the age range entered by the Operator.

(4) Using Cage Status and Building Room, search the Cage file for Issue-Feeder cages in this Building Room. For each cage found do the following:

Read the Animal Group Periodic Group to retrieve Animal UIN.

For each Animal UIN in the group do the following:

Find the record corresponding to this Animal UIN in the Animal file. Retrieve the Birth Date of this animal.

Using today's date and Birth Date, calculate the age of the animal.

If the age of the animal is between the ages in the range entered by the user, display Cage, UIN, Sex (all from the Cage file), and the age of the animal (calculated).

(5) If there are no Issue-Feeders in this room that are in the proper age range, display message.

(6) Prompt the user to enter a function or cage.

(7) If a cage is entered, then do the following:

Determine if the cage is valid by reading the Validation Tables.

(8) If the cage is invalid, display error message and reprompt the user for input.

(9) If the cage is valid, go to step 6.

(10) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(11) Call the function determined in step 10.

*** Note *** This is a totally rewritten screen.

Module Name: Screen A.9

Descriptive Title: Display Room Work Summary

Method of Invocation: User Inputs Room Work Summary Function

Inputs: User Input

Outputs: Display to Screen Only

Called Modules: None

Function: To display a summary of all work performed in a given room today, or to display all work performed in the colony today.

Assumptions:

1. Latest Retirement Date, Current Status, Experiment Number, and Current Building Room, have been added to the Animal file structure in order to greatly simplify this function.

2. Strain has been added to the Date Established file. This will be the strain of the progeny of a pair. In this function, information for pairs of different phenotypes will be displayed under the strain of the progeny.

3. Because there is no phenotype associated with a litter, litter information is displayed in the first line of information displayed for each strain.

4. Survey animals consist of those that are removed to Chemistry, Pathology, or Microbiology.

Process:

(1) Display Operator, Date, Time, Building, and Room in the screen header.

(2) Prompt the user to enter "R" (room) or "C" (colony).

(3) If the user enters "R", then do the following:

Determine the animals issued today by locating records in the Animal file with a Current Status of Issue Feeder, Issue Breeder, Issue Experiment, or Issue Other and a latest Status Change Date equal to today's date. The

latest Status Change Date is the last occurrence in the Animal Status Periodic Group. Sort the records found by Strain, Phenotype, Status and Experiment Number.

(4) Determine what to display in the Exp. # field according to Current Status. If Current Status is Issue Experiment, display the experiment number. If Current Status is Issue Feeder, display Feeder. If Current Status is Issue Breeder, display Breeder. If Current Status is Issue Other and no experiment number is entered, display Other. If Current Status is Issue Other and an experiment number is entered, display O/ and the experiment number.

(5) Accumulate system totals for males, females, and unsexed according to the strain, phenotype, and data displayed in the Exp. # field on the screen (e.g., keep totals for males, females, and unsexed animals that are on a particular experiment and are of a particular strain and phenotype).

(6) If Current Building Room in the Animal record is the same as the building room in which the user is working, do the following:

(7) If sex is male, add one to the number of males counted for this room.

(8) If sex is female, add one to the number of females counted for this room.

(9) If unsexed, add one to the number of unsexed animals for this room.

(10) At break of Strain, Phenotype and Exp. #, display the detail line for Strain, Phenotype and status as indicated by the contents of the Exp. # field. The detail line displays Strain, Phenotype, the contents of the Exp. # field determined in step 4, room totals as calculated in steps 6-9, the sum of values calculated in steps 7-9, the totals for the system as calculated in step 5, and the sum of the values calculated in step 5.

(11) If the user enters "C", determine the animals issued in the entire colony. The logic to perform this function is the same as steps 3-10 except step 6 is omitted. Accumulate totals regardless of the building room.

(12) To determine terminal totals for retired breeders, pairs set up, newborn animals, number of litters, and feeders set up, do the following:

(13) The Strain Table is used to drive this section (steps 13-2 4) of the Room Work Summary. For each strain in the Strain Table perform the following:

(14) To determine the retired breeders, search the Animal file for animals of the same strain as read from the Strain Table and Latest Retirement Date equal to today's date. If the user entered "R", read the last occurrence of Building Room in the Status Group Periodic Group to determine the current room for this animal. If the user entered "C", Building Room logic is not needed. Sort the records found by phenotype. If the Latest Retirement Date is today's date, do:

(15) If the sex of the animal is female, add one to the female count for this strain and phenotype.

(16) If the sex of the animal is male, add one to the male count for this strain and phenotype.

(17) To determine the breeder pairs set up, search the Date Established file for the same strain as read from the Strain Table and Date Established equal to today's date. If the user entered "R", using the Dam UIN in the Date Established file, find the corresponding record in the Animal file. If the last occurrence of the Building Room in the Status Group Periodic Group is the same as the room in which the operator is working, count this animal in the Pairs Set Up count. If the user entered "C", disregard Building Room and count the number of records found in the Date Established file with Date Established equal to today's date.

(18) If the user entered "C", determine the number of newborn animals by searching the Litter file for the same strain as read from the Strain Table and Birth Date equal to today's date. If the user entered "R", in addition to finding litters for a strain, use the Dam UIN from the Litter file to locate the corresponding record in the Animal file and determine the Building Room for each litter from the last occurrence of the Status Group Periodic Group. For each record found determine the number of animals in the litter by the following:

(19) Read the first occurrence of the Number Detail Group. Number-in-Lit is the total number of animals in this litter. Add Number-in-Lit to the accumulated total for this strain.

(20) The number of litters is the total number of records found for this strain with Birth Date equal to today's date.

(21) If the user entered "C", determine the number of animals set up as feeders by searching the Animal file for Strain equal to the strain read from the Strain Table, Current Status equal to Feeder, and a latest Status Change Date equal to today's date. The latest Status Change Date is found in the last occurrence of the Status Group Periodic Group. If the user entered "R", in addition to the above processing, determine the Building Room for each animal from the last occurrence of the Status Group Periodic Group.

(22) Accumulate totals for female, male, and unsexed feeders.

(23) At the break of Strain and Phenotype do the following:

(24) Display detail line by strain and phenotype. For the first line of a strain, display newborn and number of litters. Subsequent lines for the same strain will not have litter information displayed. Litter data cannot be determined by phenotype.

(25) If the user entered "C" or "R", determine animal removals by searching the Animal file for Current Status equal to Dead, Culled, or Surplus. If the user entered "R", in addition, read the last occurrence of the Status Group Periodic Group to determine the Building Room for each animal. Sort the records found by Strain.

(26) If Current Status is Dead, perform the following:

(27) If Removal Destination is Chemistry, Pathology, or Microbiology, add one to the totals for males, females, or unsexed for this strain depending on Sex in this record.

(28) If Removal Destination is not Chemistry, Pathology, or Microbiology, add one to the total males, females, or unsexed of this strain that are dead.

(29) If Current Status is Culled, add one to culled count for males, females, or unsexed for this strain, depending on Sex in the Animal file.

(30) If Current Status is Surplus, add one to surplus count for males, females, or unsexed for this strain according to Sex in the Animal file.

(31) At the break of Strain, print the detail line for a strain and accumulate grand totals for all strains.

(32) At the end of the report, display grand totals for males, females, and unsexed that are dead, culled, surplused, or sent to survey.

Module Name: Screen A.10

Descriptive Title: New Animal Entry

Method of Invocation: User Inputs New Animal Entry Function

Inputs: Cage Number, Flag for Special Processing Mode

Outputs: In the Cage File: updates to the ORIGINAL group and BLOCK group if a litter, updates to ANIMAL group if not a litter

In the Litter File: new record created if new animals are a litter

In the Animal File: new record created for each new animal that is not in a litter

In the Date-Established File: new record created, if new dam and sire are estabished as breeder pair

Called Modules: Display Cage

Function: To enter new (purchased) animals into the colony.

Assumptions:

1. Cage status can be Breeder, Feeder, or Issue. A cage has a status even if empty. The only assignments that can be made to that cage is for animals of compatible type. Cage Status cannot be changed by this function.

2. A litter cannot be entered into a cage that already contains animals.

3. A male cannot be entered into a Breeder cage that already contains a male. A Breeder cage can contain zero males. (This assumption is given in the screen layout draft).

4. An animal cannot be entered into a Feeder or Issue cage that contains animals of the opposite sex. (This assumption is given in the screen layout draft).

5. A female cannot be entered into a Breeder cage that contains more than two females. There must be at least one female in a Breeder cage. (This assumption is given in the screen layout draft.)

6. A female cannot be entered into a Breeder cage that contains a dam with a litter.

7. A litter that is currently with its dam can only be transferred AS A LITTER to another Breeder cage if the dam is also transferred; this is only performed in the Move Cage function. (This assumption is given in the screen layout draft.)

8. You may put any animals in an empty Breeder/Feeder cage; animals that have been given breeder status may not be put in an Issue cage. (This assumption is given in the screen layout draft.)

9. If a new female (and possibly male) is (are) entered into a cage with a new litter, then the new animal(s) should be the dam (and sire) of the litter.

Process:

(1) User has entered the New Animal Entry function, a cage or range of cages, and/or a processing mode. As a result, this function is called and supplied with a cage number and knowledge of whether or not a special processing mode is in effect. Perform the following for each cage (2-35):

(2) Display the Cage data (call Display Cage subroutine).

(3) Prompt the user to enter the sex or litter. If input is in error, display message and reprompt. If blank is entered, this means the user is finished entering new animals. In this case, proceed to step 19. If correct (and not a blank), proceed to step 4.

(4) Test to see if the sex of the animal is compatible with animals already in the cage (if there are any). Also test to see if the new animal is compatible with other new animals that may be entered into the same cage at the current time.

(5) Test to see if new animal is being entered into a cage where a new litter is also being entered. If so, assume the new animal is the dam or sire of the new litter, and save this information for storage into that litter's Litter record. Also, save information for creating a new animal record for the dam or sire. Finally, if both dam and sire are present, save information for creating a new Date-Established record.

(6) If compatible according to the assumptions listed above, continue. If not, print an error message and go to step 3.

(7) If there is a litter, test to see if there are any other new animals to be entered into the current cage. If so, consider the other new animals to be the dam or sire of the litter, depending on the sex. Save information for creating a Date-Established record, if a breeder pair is in the cage. If other animals in the current cage are not compatible with the litter to be entered, proceed to step 3.

(8) Prompt the user to enter animal status. If the status is invalid, display an error message and reprompt. If correct, continue.

(9) Test to see if the status is compatible with the status of the cage, the animals already in the cage, and/or the other new animals being added to the cage. If the status is compatible according to the assumptions listed above, do the following (10 - 11). If not, print an error message, set the Error flag, and go to step 3.

(10) If status entered is Issue, prompt the user to enter the reason for the issue. If reason is invalid, display an error message and reprompt. If correct, continue.

(11) If the reason for Issue is "experiment", prompt the user to enter the experiment number. If number is invalid, display an error message and reprompt. If correct, continue.

(12) Prompt the user to enter the date of birth. Test to see if date entered is valid. If not, display an error message and reprompt. If correct, continue.

(13) Prompt the user to enter the strain of this animal or litter. Test to see if the strain is a valid one. If not, display error message and reprompt. If correct, continue.

(14) If required by the strain code (possibly for the second animal in a breeder pair) prompt the user to enter the phenotype for the animal. Test to see if the user input is a valid phenotype for that strain. If valid, continue. If not, print error message and reprompt the user. If the animal is of the wrong phenotype for introduction into the current cage, display an error message and go to step 3.

(15) Prompt the user to enter the number of animals to be introduced to the current cage. If an invalid number is entered, display an error message and reprompt the user. If a seemingly unreasonable number is entered, notify the user, then give the user the option of accepting the current number or correcting it. If a valid number is entered, continue.

(16) Prompt the user to enter the generation of the animal or litter. Test to see if the user input is valid. If valid, continue, else display an error message and reprompt the user.

(17) Prompt the user to enter the Vendor's ID. Any alphanumeric string is accepted, as is an empty field.

(18) Return to step 3.

(19) After the user inputs a blank in step 3, prompt the user to enter any notes about the new animal entries into the current cage. No data tests are done on this field. The user enters a <CTRL/Z> to terminate the text entry.

(20) If a special processing mode is not in effect, do the following (21 - 25):

(21) Prompt the user to enter a function or cage, then accept user input.

(22) If user input was more than just pressing the Enter key, do the following (23 - 25):

(23) Check to see if the user input a valid cage by reading the Validation Tables.

(24) If not a valid cage, then check to see if user input was a valid function by checking the Function Tables.

(25) If not a valid cage or valid function, display an error message and return to step 21.

(26) Whether or not a special processing mode is in effect, if any new animal information has been stored, do the following (27 - 34):

(27) Display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(28) For each animal or litter, do the following (29 - 34):

(29) If the status is litter, do the following (30 - 31):

(30) In the Cage file, update the Original and Block information to reflect the new litter's Litter UIN, count, and birth date.

(31) In the Litter file, make a new entry that reflects the new litter information.

(32) If the status is not littered, do the following: (33 - 34):

(33) In the Cage file, make an entry into the Animal Group that reflects the new animal information.

(34) In the Animal file, make a new record that reflects the new animal information.

(35) If information exists for creating a Date-Established record, then do so.

(36) If a valid new cage was entered in step 21, clear the screen and go to step 2.

(37) If a valid new function was entered in step 21, return to the driver module with the number of the cage just processed and the name of the new function.

(38) If Enter was pressed in step 21, or if a special processing mode is in effect, return to the driver module with no cage or function.

Module Name: Screen A.11

Descriptive Title: Change Cage Status Screen

Method of Invocation: User Inputs Change Cage Status Function

Inputs: Cage Number, Flag for Special Processing Mode

Outputs: In the Animal File: new animal statuses for animals in the current cage

In the Cage File: an update of the status for the current cage

Called Modules: Display Cage

Function: To change the status of all animals in a cage and to change the status of the cage itself.

Assumptions:

1. New breeder cards will be printed at the end of the day when the assignment report is printed.

2. Correct animals have been put in the cage, and the Cage file has been updated. The Animal file has not yet been updated. This function will update the animal's status in the Animal file, and the Cage status in the cage file.

3. Restrictions on cages to be changed to breeder status:

   a. animals in the cage must be sexed
   b. there can be zero or one males
   c. there can be zero to three females

4. Restrictions on cages to be changed to feeder status:

   all animals in the cage must be of the same sex.

Process:

(1) User has entered the Change Cage Status function, a cage or a range of cages, and/or a special processing mode. As a result, this function is called and supplied with a cage number and knowledge of whether or not a special processing mode is in effect. Perform the following for each cage chosen:

(2) Display the Cage data (call Display Cage subroutine).

(3) Check the status of the current cage in the Cage file. If the status is issue, display error message, set an Error flag, then go to step 8.

(4) Check the status of the animals in the current cage. If the cage status is Breeder and there is an unweaned litter present, the litter must be weaned before this function can proceed. Display an error message, set the Error flag, and go to step 8.

(5) Prompt user to enter new cage status. Display the default, then the new cage status. Validate the new cage status according to the following rules:

(6) If the new cage status is Breeder, find the record corresponding to the current cage in the Cage file. Search through the animals in the Cage file to make sure that (a) all animals are sexed, (b) there is not more than one male present, and (c) there are not more than three females present. If correct, go to step 7; else display error message, set the Error flag, and go to step 8.

(7) If the new cage status is Feeder, find the record corresponding to the current cage in the Cage file. Search through the animals in the Cage file to make sure that all animals in the cage are of the same sex. If correct, go to step 7; else display error message, set Error flag, and go to step 8.

(8) If a special processing mode is not in effect, do the following (9 - 13):

(9) Prompt the user to enter a function or cage, then accept user input.

(10) If user input was more than just pressing the Enter key, do the following (11 - 13):

(11) Check to see if user input was a valid cage by reading the Validation Tables.

(12) If not a valid cage, then check to see if user input was a valid function.

(13) If not a valid cage or valid function, display error message and return to step 9.

(14) Whether or not a special processing mode is in effect, if the Error flag is not on, do the following (15 - 17):

(15) Display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(16) Change the Cage record's status field to reflect the new cage status.

(17) For each animal UIN listed in the current Cage record, find the corresponding record in the Animal file. Make a new entry in that Animal record's status group that contains (a) the animal's new status, based on the new cage status; (b) the building-room, based on the Cage building room; (c) the cage; (d) the status change date, which will be the current date; (e) the prefix code; and (f) the session number.

(18) If a valid cage is entered, go to step 2.

(19) If a new function is entered, return to the driver module with the number of the cage just processed, and the name of the new function.

(20) If Enter was pressed, or if a special processing mode is in effect, return to the driver module with no cage or function.

Module Name: Screen A.12

Descriptive Title: Move Cage

Method of Invocation: User Inputs Move Cage Function

Inputs: Cage Number, Flag for Special Processing Mode

Outputs: Male and Female Breeder Cards (if a breeder cage is moved)

In the Animal File: new Animal Status Group entries that reflect the new position of the cage

In the Cage File: an update of the position of the cage being moved

Called Modules: Display Cage

Function: To move a cage to another rack position in the same room. This allows moving an entire breeding unit (male, female, litter) to a new location in the room.

Assumptions:

1. This function does not include issue cages, and an error will occur if (a) an attempt is made to move an issue cage or (b) the destination cage is an issue cage.

2. There is a cage (empty or not) in every cage position in the room.

3. This function is only used to move cages within a room.

Process:

(1) User has entered the Move Cage function, a cage or a range of cages, and/or a processing mode. As a result, this function is called and supplied with a cage number and knowledge of whether or not a special processing mode is in effect. Perform the following for each cage chosen:

(2) Display the Cage data (call Display Cage subroutine).

(3) Check the status of the current cage in the Cage file. If the status is issue, display error message, set an Error flag, then go to step 13.

(4) Prompt the user to enter the new cage number. Validate the new cage number according to the following rules (5 - 6):

(5) If the new cage is an issue cage, display error message and return to step 4.

(6) If the new cage does not exist, display error message and return to step 4.

(7) If the new cage is valid and the question "Switch cages?" has not yet been asked for this set of cages, prompt the user with "Switch cages?"

(8) If the answer is not "Y" or "N", display error message and return to step 7.

(9) If the answer is "Y" or "N", keep up with all source cages and corresponding destination cages in parallel arrays (for source and destination) or other similar data structures. No changes should be made to the data base until all cages that are to be removed from their source positions have a valid destination position.

(10) If the answer is "Y", the source cage (from this point on, called CAGE-1) is stored in the source array, and the destination cage (from this point on, called CAGE-2) is stored in the corresponding position in the destination array. CAGE-2 is then stored in the next position in the source array, and CAGE-1 is stored in the corresponding position in the destination array. Proceed to step 12.

(11) If the answer is "N", CAGE-1 is stored in the source array, and CAGE-2 in the destination array. CAGE-2 becomes CAGE-1 (a source cage with no destination). Return to step 4 to get the destination for new CAGE-1.

(12) When all source cages to be moved have valid destinations, do the following (13 - 27):

(13) If a special processing mode is not in effect, do the following (14 - 18):

(14) Prompt the user to enter a function or cage, then accept user input.

(15) If user input was more than just pressing the Enter key, do the following (16 - 18):

(16) Check to see if user input was a valid cage by reading the Validation Tables.

(17) If not a valid cage, then check to see if user input was a valid function.

(18) If not a valid cage or valid function, display error message and return to step 14.

(19) Whether or not a special processing mode is in effect, if the Error flag is not on, do the following (20 - 24):

(20) Display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(21) For each cage number in the data structure, do the following (22 - 24):

(22) Assign the destination cage number to the source cage in the Cage file.

(23) If the newly written cage record has a status of breeder in the Cage file, make new breeder cards for each animal, retaining all the old information but filling in the new cage number.

(24) For each animal in the newly written cage record, update the corresponding Animal record by adding a new entry to the Status Group. The new entry in the Status Group should be identical to the old one, except for the change of cage number.

(25) If a valid new cage was entered in step 14, go to step 2.

(26) If a new function was entered in step 14, return to the driver module with the number of the cage just processed and the name of the new function.

(27) If Enter was pressed in step 14, or if a special processing mode is in effect, return to the driver module with no cage or function.

Module Name: Screen A.13

Descriptive Title: Display Cage

Method of Invocation: User Inputs Cage Number

Inputs: Cage Number

Outputs: Display to Screen Only

Called Modules: None

Function: Display cage data at the top of the screen.

Assumptions:

1. The Animal Sex field in the Animal Group Periodic Group of the Cage file will be blank for unsexed animals, "M" for males, and "F" for females in each cage.

2. If the male is not present in the cage, his UIN is still present in the Animal Group Periodic Group.

3. If there is a litter present in the cage, there will not be a male present but the Dam could be present.

4. If data is entered in the Block UIN of the Cage file, this cage contains a litter.

Process:

(1) User enters Cage Number and/or a processing mode while in a particular Building Room. If a mode is entered, move "Seq. Wean", "Littered", or "All" to a user-defined variable (#Mode) for display purposes.

(2) Read the Cage file to find the Cage entered for this Building Room.

(3) Read each occurrence in the Animal Group Periodic Group, moving relevant data to user-defined variables that will be written to the screen.

(4) If Animal Sex is Male then do the following:

Increment the male counter (#Male Count).

(5) If Phenotype is entered for this animal, then move Phenotype to a user-defined variable (#Male Phenotype).

(6) Move the generation to a user-defined variable (#Male Generation).

(7) If Animal Flag is "P" or "A", then this is a harem-mated cage that may or may not contain a male at this time. For harem-mated cages, the male's UIN and the "present or absent" flag is written to the screen as follows:

(8) Move Animal Flag to a user-defined variable (#Present Absent).

(9) Move Animal UIN to a user-defined variable (#Male UIN).

(10) If this is a breeder cage (Cage Status indicates Breeder), then do the following:

(11) Find the record in the Date Established file that corresponds to this male's UIN.

(12) Move Date Established to a user-defined variable (#Male DE).

(13) If Animal Sex is Female then do the following:

Increment the female counter (#Female Count).

(14) If Phenotype is entered for this animal, then move Phenotype to a user-defined variable (#Female Phenotype).

(15) Move the generation to a user-defined variable (#Female Generation).

(16) If this Animal UIN is less than the previous occurrence of Animal UIN, move it to a user-defined variable (#Female UIN). If there is a block of females, display the lowest UIN.

(17) If Animal Flag is equal to "N", "L", "R", or "B" (this female has an ear clip), then concatenate this ear clip value to the user-defined variable, #Ear Clip. The value of #Ear Clip will be a sequence of "NLRB", depending on the number of females present in the cage.

(18) If Block UIN is not blank, there is a litter present and it is processed as follows:

(19) Move the Cage Block data to user-defined variables (Block UIN, #Block Count, #Block Generation).

(20) Move the Cage Original Block data to user-defined variables (#Original UIN, #Original Count, #Original Date).

(21) Display the cage data at the top of the screen as follows:

Display Line 1: "Display Cage", Cage Status, "Cage Number", Cage Number, System Date, and Time

Display Line 2: "Operator", Operator ID, "Room", Building Room, and #Mode

Display Line 3: "Males," #Male Phenotype, "Females," and #Female Phenotype

Display Line 4: #Male DE, #Male DE, "Block", and "Original Block"

Display Line 5: #Male UIN, #Present Absent, #Female UIN, #Ear Clip, #Block UIN, and #Original UIN

Display Line 6: #Male Count, #Male Generation, #Female Count, #Female Generation, #Block Count, #Block Generation, #Original Count, and #Original Date

(22) Prompt the user to enter a function or cage.

(23) If a cage is entered, validate it and go to step 2.

(24) If a function is entered, read the Function Table to determine which module to call. If the function is valid, call it. If the function is not valid, go to step 22.

Module Name: Screen A.14

Descriptive Title: Animal Assignment

Method of Invocation: User Inputs Animal Assignment Function

Inputs: Cage Number

Outputs: Updates Cage and Animal Records with the Status as Entered by the User and possibly creates a Date-Established record.

Called Modules: None

Function: To assign an animal a new status of either Breeder or Feeder. Animals may be weaned in this screen.

Assumptions:

1. Current Status is the Cage Status (Breeder, Feeder, Litter, Issue).

2. Actions that may be performed are assigning litter animals as Breeder or Feeder, changing Breeders to Feeders, changing Feeders to Breeders, or changing Issue animals to Breeders or Feeders.

3. In the case of assigning littered animals to a status, the Count entered by the user can only be checked to determine that the user has not attempted to assign more animals than are in the current cage.

4. UINs may be assigned to littered animals in this function.

5. If a pair is assigned to Breeder status, the female UIN is displayed on the screen under UIN Assigned.

6. The user can assign "n" pairs to "n" sequential cages.

Process:

(1) User has entered the Animal Assignment function, a cage or range of cages, and/or a processing mode (Sequential Wean, Littered Cages, All Cages). If the Enter key is pressed without entering a cage or function, go to step 12; else perform the following for each cage chosen:

(2) Display the Cage data (call subroutine to display Display Cage screen).

(3) Prompt the user to enter Cage Status. If error, display message and reprompt. If correct, continue.

(4) Prompt the user to enter Sex or accept default of "B" for Breeder and "U" otherwise. If Sex is not "M", "F", "B", or "U", display error message and reprompt. If correct, continue.

(5) Using Strain from the Cage file, read the valid phenotypes for this strain from the Validation Tables. If there are phenotypes for this strain, prompt the user to enter the phenotype. If this is a breeder pair, prompt the user to enter the phenotype for the second animal in the pair. If incorrect data is entered, reprompt the user for correct data.

(6) Prompt the user to enter Count (the number of animals to be assigned). If Sex is "B" (Both), the Count indicates the number of breeder pairs to assign to cages. If Sex is not "B", Count is the number of individuals to be assigned the new status.

(7) Prompt the user to enter New Status. Verify that it is "B" (Breeder) or "F" (Feeder). Display error message and reprompt the user if New Status is not "B" or "F".

(8) If New Status is Breeder and Count is greater than one, this is the first of the "Count" sequences for cages to which breeder assignments will be made. Each breeder pair or harem is assigned to a separate cage. The new cage numbers are assigned sequentially by the system. One line will be displayed on the screen to show the first of the "Count" cages and the first UIN assigned to the first cage. Prompt the user to enter ear clips for each female in harem-mated cages.

(9) If sequential cages are not to be automatically assigned, prompt the user to enter the New Cage. If the New Cage is entered, do the following:

(10) Verify that the New Cage is available and is of the proper type to receive the animals by checking the Validation Tables and Cage file (e.g., assign breeders to Breeder cages and feeders to Feeder Cages).

(11) If the current cage is a Breeder cage, determine whether there is a litter present (Block UIN will contain data). If there is a litter in the cage, use the Block Count to determine the number of animals from this cage that can be assigned a status. Check that

the user does not assign more animals than are available.

(12) If the cage contains individuals with UINs already assigned (not a litter) then do the following:

(13) If New Status is Feeder, verify that the animals to be assigned to the new cage are of the same sex (e.g., that "M" or "F" was entered in Sex).

(14) If New Status is Breeder, verify that, at most, one male is assigned to the new cage and from zero to three females are assigned. At least one animal must be assigned to the new cage.

(15) If New Cage is not entered, do the following:

(16) Assume the user intends to change the status of the current cage. If New Status is Feeder, verify that the animals present in the current cage are of the same sex (e.g., that "M" or "F" was entered in Sex). If not, display error message and prompt the user to enter a new cage.

(17) If New Status is Breeder, verify that, at most, one male is present in the current cage and from zero to three females are present. At least one animal must be present in the current cage. If not, display error message and prompt the user to enter a new cage.

(18) Prompt the user to enter a function or cage.

(19) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(20) If the cage is invalid, display error message and reprompt the user for input.

(21) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(22) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(23) Using the Cage file, determine the UINs of the animals assigned from the current cage. Find the animals with the corresponding UIN in the Animal file and add an occurrence in the Animal Status Group for each animal in this cage as required by the user input (e.g., store Status in Animal Status and store the new Cage in Animal Cage).

(24) If the assignment was performed on a littered cage, then store new animal records in the Animal file with UINs corresponding to the UINs assigned.

(25) If assignment was to breeder status, check to see if their is a Date-Established record for this pair (or pairs). If a Date-Established record exists, check to see if the Dissolve-Date is blank. If there is no Date_Established record or if the existing record has a Dissolve-Date, create a new Date-Established record.

(26) If a cage is entered, go to step 2.

(27) If a function is entered, call the module as determined in the Function Table.

Module Name: Screen A.15

Descriptive Title: Animal Removal

Method of Invocation: User Inputs Animal Removal Function

Inputs: User has previously entered a cage, selected a range of cages, or specified all cages; user enters Status, Sex, Phenotype, Count, Reason for removal, Disposition (Destination), and CID.

Outputs: Data Input by User is Displayed to the Screen, Along with the Cage Data Header

Updates to files Cage and Animal

Called Modules: None

Function: To cull or surplus an animal, send it to be surveyed, or note its death.

Assumptions:

1. If more than one sex is present in a cage, the cage must be a breeder cage.

2. How to validate the CIDs is still a question that NCTR is reviewing.

Process:

(1) User has entered the Animal Removal function, a cage or range of cages, and/or a processing mode (Sequential Wean, Littered Cages, All Cages). If the Enter key is pressed without entering a cage or function, go to step 16; else perform the following for each cage chosen:

(2) Display the Cage data (call subroutine to display "Display Cage" screen).

(3) Prompt the user to enter Cage Status. If error, display message and reprompt. If correct, continue.

(4) If the cage's current status is Breeder and both sexes are present in the cage, do the following: prompt the user to enter Sex. If input is in error, display message and reprompt. If correct, continue. Check to see if the animal to be removed is part of a breeder pair with no

Dissolve-Date. If so, disallow animal removal until pair is retired. Display an error message and go to step 16.

(5) Using the strain of the female, read the Strain Table to determine the phenotypes (if any) for this strain. Prompt the user to enter phenotypes, if there are any, for this strain.

(6) Prompt the user to enter count when required by sex, phenotype, and census. For example, if there is more than one animal of the sex to be removed, Count determines how many of that sex will be removed. All combinations of sex and phenotype that are possible during the execution of this function need to be determined and properly handled.

(7) Prompt the user to enter Reason for removal. If blank, continue. If entered, check that it is "D", "S", or "C", and prompt for reentry if invalid. Continue if entry is valid.

(8) Prompt the user to enter Disposition. If entered, check that it is "P", "C", or "M", and prompt for reentry if invalid. Continue if entry is valid.

(9) If Reason and Disposition are both blank, prompt the user to enter Reason or Disposition and return to step 7.

(10) Prompt the user to enter a CID. If entered, then determine whether this CID has already been used by reading the Validation Tables. If the value is not found, continue. If the value is found, display error message and reprompt (repeat step 10).

(11) Display UIN of the individual removed. If a block of animals is removed, display the first UIN of the block.

(12) If a range of cages is to be removed, have the user re-enter the range of cages to verify correctness.

(13) Prompt the user to enter a function or cage.

(14) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(15) If the cage is invalid, display error message and reprompt the user for input.

(16) If a function is entered, read the Function Table to determine which module to call. If the function is not in

the table, display an error message and reprompt the user to enter a valid function.

(17) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "Y", continue. If the user enters "N", allow editing of the data just entered.

(18) Using the Cage file, determine the UINs of the animals removed from this cage. Find the animals with the corresponding UIN in the Animal file and add an occurrence in the Animal Status Group and Animal Removal Group for each animal in this cage as required by the user input (e.g., store Reason in Animal Status and store Disposition in Animal Removal Destination). In the cage file, decrement the original count, and remove information about this animal from the Periodic Group.

(19) If a cage is entered, go to step 2.

(20) If a function is entered, call the module as determined in the Function Table.

Module Name: Screen A.16

Descriptive Title: Animal Retirement

Method of Invocation: User Inputs Animal Retirement Function

Inputs: Cage Number

Outputs: Data Input by User is Displayed to the Screen, Along with the Cage Data Header

Called Modules: None

Function: To retire breeders individually or in pairs.

Assumptions:

1. One line will be output to the screen for each CID.

2. The sequential processing option needs clarification with regard to prompting the user with "Are all data on the screen correct?" and when to update the census data files. Answer: when a range of cages is entered, prompt the user to verify the correct range and update census immediately if user verifies that the range is correct.

Process:

(1) Display the Cage data (call subroutine to display Display Cage screen) for the Cage Number entered by the user.

(2) Prompt the user to enter "S" (Sequential) or "I" (Individual) cages.

(3) If the user enters "S", then do the following:

(4) Prompt the user to enter the first cage number in the sequence (#Starting Cage). Validate the cage number for this Building Room using the Validation Tables. If the cage is valid, continue. If the cage is invalid, display an error message and prompt the user to reenter.

(5) Prompt the user to enter the last cage number in the sequence (#Ending Cage). Validate the cage number for this Building Room using the Validation Tables. If the cage is valid, continue. If the cage is invalid, display an error message and prompt the user to reenter.

(6) Prompt the user to enter Sex.

(7) Validate Sex. If Sex is "Both", check that Cage Status is Breeder. If Cage Status is not Breeder, display error and reprompt the user. If Cage Status is Breeder, continue. If Sex is not "Both", verify that it is "M"(Male) or "F" (Female). If not, reprompt the user.

(8) If this is a harem-mated cage, prompt the user to enter Count. Verify that Count is 1, 2, or 3. If not, reprompt the user. If Count is 1, 2, or 3, continue. This count is the number of females in the cage that are to be retired.

(9) Prompt the user to enter Disposition. If anything is entered, verify that it is "P", "C", or "M". If invalid, reprompt the user. If Disposition is blank or valid, continue.

(10) Prompt the user to enter a CID. If entered, then determine whether this CID has already been used by reading the Validation Tables. If the value is not found, continue. If the value is found, display an error message and reprompt the user.

(11) Display the prompt "Are all data on this screen correct?" If the user enters "N", allow editing of the data just entered. If the user enters "Y", then do the following:

(12) If the user entered a range of cages to retire, prompt the user to confirm that the range is correct.

(13) For each cage to be retired, find the Cage record that corresponds to this cage.

(14) For each animal to be retired, do the following:

(15) Read each occurrence in the Cage Animal Group. Using the male and female UINs in the periodic group, locate the corresponding Date Established record.

(16) Update the Date Pair Dissolved field to today's date.

(17) Update the Status and Latest Retirement Date in the Animal file to "Retired" and today's date, respectively, for each UIN retired. Also, for each UIN retired, update Removal Destination in the Animal file with the Disposition entered by the user in step 9, and update the CID in the Animal file with the CID entered in step 10. Display Sex, Reason, Disposition, CID, and UIN for each animal retired.

(18) If the user entered an individual cage, then do the following:

(19) Prompt the user to enter a function or cage.

(20) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(21) If the cage is invalid, display an error message and reprompt the user for input.

(22) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display an error message and reprompt the user to enter a valid function.

(23) Call the module determined in step 22.

Module Name: Screen A.17

Descriptive Title: Note Pregnant or Plugged

Method of Invocation: User Inputs Note Pregnant or Plugged Function

Inputs: Cage Number

Outputs: Data Input by User is Displayed to the Screen, Along with the Cage Data Header

Called Modules: None

Function: To note that a female is pregnant or that a plug has been observed.

Assumptions:

None made.

Process:

(1) Display the Cage data (call subroutine to display Display Cage screen) for the Cage Number entered by the user.

(2) Find the record in the Cage file that corresponds to this Building Room and Cage.

(3) If Cage Status is not Breeder, then reprompt the user to enter another cage and go to step 1.

(4) If Cage Status is Breeder, then do the following:

(5) Read each occurrence of the Cage Animal Group Periodic Group searching for the first female UIN.

(6) If there is no female in the cage, display an error message to prompt the user to reenter a cage and go to step 1.

(7) If there is at least one female in the cage, then do the following:

(8) Prompt the user to enter the observation (PR or PL). Verify that only "PR" or "PL" was entered. If one of these is not entered, display an error message and reprompt the user. If entered correctly, continue.

(9) Prompt the user to enter a function or cage.

(10) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(11) If the cage is invalid, display an error message and reprompt the user for input.

(12) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display an error message and reprompt the user to enter a valid function.

(13) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "N", allow editing of the data just entered. If the user enters "Y", then do the following:

(14) For each female in the cage do the following:

(15) Using the female and male UINs, locate the record in the Date Established file that corresponds to this pair.

(16) Maintaining the proper order in the Pair Observation Group Periodic Group, add an occurrence to the group. Litter Observation should be the code for littered, and Litter Observation Date is today's date.

(17) If the user entered a valid cage in step 9, go to step 1.

(18) If the user entered a valid function in step 9, call the module as determined by the Function Table in step 12.

Module Name: Screen A.18

Descriptive Title: Note Litter

Method of Invocation: User Inputs Note Litter Function

Inputs: Cage Number

Outputs: Data Input by User is Displayed to the Screen, Along with the Cage Data Header

Called Modules: None

Function: To note that a female has littered and to note the number of litter animals.

Assumptions:

1. It is possible to have two litters in one cage. Force weaning of first litter before observing the second litter. ** NCTR approved **

2. The strain of a litter resulting from crossbreeding will be determined from a table. ** NCTR approved **

3. Maximum litter size is 23. ** NCTR approved **

Process:

(1) Display the Cage data (call subroutine to display Display Cage screen) for the Cage Number entered by the user.

(2) Find the record in the Cage file that corresponds to this Building Room and Cage.

(3) If Cage Status is not Breeder, then reprompt the user to enter another cage and go to step 1.

(4) If Cage Status is Breeder, then do the following:

(5) Read each occurrence of the Cage Animal Group Periodic Group searching for the first female UIN.

(6) If there is no female in the cage, display an error message to prompt the user to reenter a cage and go to step 1.

(7) If there is a female in the cage, then do the following:

(8) If Cage Block UIN is not blank, there is a litter already observed in this cage. Display error message and go to step 21.

(9) If Cage Block UIN is blank (the litter in the cage has not been counted and assigned a block of UINs), do the following:

(10) Prompt the user twice to enter the number of pups in the litter (Count). If count is outside the range of 1 to 23, display error message and reprompt the user. If valid, continue.

(11) Determine the next available UIN.

(12) Using Count and the available UIN, determine the block of UINs to be used by this litter.

(13) Display "L" as the observation, Count, and the first of the block of UINs.

(14) Read the generation of the female from the Animal Generation field in the Animal Group Periodic Group.

(15) To determine whether there has been cross-breeding, find the Date Established record that corresponds to this female's UIN.

(16) Retain the Sire UIN (#Sire UIN).

(17) If the strain of the Sire is different from the strain of the female, do the following:

(18) Set the litter generation equal to one.

(19) Read the Crossbreeding Table to determine the strain of the litter.

(20) If the strains are the same, the litter generation is set to one greater than the female's generation.

(21) Prompt the user to enter a function or cage.

(22) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(23) If the cage is invalid, display an error message and reprompt the user for input.

(24) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display error message and reprompt the user to enter a valid function.

(25) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "N", allow editing of the data just entered. If the user enters "Y", then do the following:

(26) Update the Cage record with the Original UIN (same as the new Block UIN), Original Count (entered by the user), Original Date (today's date), Block UIN, Block Count (entered by the user), Block Generation (calculated in step 18 or step 20).

(27) To determine the litter number for this litter, find the latest litter record with the same Dam and Sire as the current litter and increment by one (#Litter Number).

(28) Store a new litter record in the Litter's file. Litter UIN is the Block UIN determined in step 12, Dam UIN is the female UIN (Animal UIN from Cage file), Sire UIN is from the Date Established file, Birth Date is today's date, Number in Litter is Count, and Litter Number is calculated in step 27.

(29) If the user entered a valid cage in step 21, go to step 1.

(30) If the user entered a valid function in step 21, call the module as determined by the Function Table in step 24.

Module Name: Screen A.19

Descriptive Title: Issue Animal

Method of Invocation: User Inputs Issue Animal Function

Inputs: Cage Number

Outputs: Data Input by User is Displayed to the Screen, Along with the Cage Data Header

Called Modules: None

Function: To issue animals to experiments, as feeders, or for other reasons. Animals may be weaned in this screen.

Assumptions:

1. Animals may be issued without using a destination cage by using Issue Other.

2. Breeders, Feeders, Litter animals, or other Issued animals may be issued.

3. The possible issue statuses are Issue Feeder, Issue Breeder, Issue Experiment, and Issue Other.

4. Animal UINs may be assigned in this screen if there are litter animals in the cage that have not been weaned. The Animal record will be created for these animals.

5. Where to store the experiment number entered by the user needs to be clarified. Store the experiment number to which animals are issued in the Animal file.

6. Experiment numbers can be validated with the Orders file.

Process:

(1) Display the Cage data (call subroutine to display "Display Cage" screen) for the Cage Number entered by the user.

(2) Find the record in the Cage file that corresponds to this Building Room and Cage.

(3) Prompt the user to enter Status. Status must be "B", "F", "L", "I", or blank. If not, reprompt the user. Verify that the status is entered is the same as the Cage Status read from the Cage file. If not, reprompt the user.

(4) If Status is blank, the user is finished with this cage. Go to step 12.

(5) If there is more than one sex in the cage, prompt the user to enter Sex. Verify that it is "M", "F", or "U". If not, reprompt the user.

(6) If allowed by the strain code (locate the strain in the Strain Table to determine phenotypes), prompt the user to enter the phenotype. If invalid, reprompt the user. If valid, continue.

(7) If required by sex, phenotype, and cage census, prompt the user for the Count (number of animals to issue).

(8) Prompt the user to enter the reason for issue (Reason). If anything but "E", "R", or "O" is entered, reprompt the user. If one of these is entered, continue.

(9) If Reason is not "O" (Other), prompt the user to enter New Cage. Verify that this is an issue cage by consulting the Validation Tables. If this is not an issue cage, reprompt the user. If this is a valid issue cage, continue.

(10) If Reason is "E" (Experiment), prompt the user to enter Experiment Number. Verify that this is a valid experiment by consulting the Order file. If this experiment is not found in the unfilled orders in the Orders file, reprompt the user. If the Experiment Number is valid, continue.

(11) If Reason is "O" and an Experiment Number was entered, prompt the user to enter CAN. Check that the CAN is valid by consulting the Validation Tables.

(12) Prompt the user to enter a function or cage.

(13) If a cage is entered, then do the following:

Determine whether the cage is valid by reading the Validation Tables.

(14) If the cage is invalid, display an error message and reprompt the user for input.

(15) If a function is entered, read the Function Table to determine which module to call. If the function is not in the table, display an error message and reprompt the user to enter a valid function.

(16) If a valid cage or function is entered or the Enter key is pressed, display "Are all the data on this screen correct?" If the user enters "N", allow editing of the data just entered. If the user enters "Y", then do the following:

(17) If Block UIN of the current Cage record is not blank, there are animals in this cage that are not weaned. Perform the following:

(18) Using Count, Original UIN, and Block UIN, assign the animals individual UINs. Give male UINs from the lowest UIN and female UINs from the highest UIN.

(19) For each UIN assigned, create a record in the Animal file for this animal. Store the Animal UIN (as calculated in step 18), Litter UIN (Original Block UIN), Sex (entered by the user), Phenotype (entered in this screen by the user), Birth Date (today's date), CAN (may be entered in this screen), Current Status (new status entered by user), Current Building Room, and Experiment Number. In the Status Group Periodic Group, store Building Room, Cage (New Cage), Status, and Status Change Date (today's date). Store weight information in the Animal Wean group. Update the Wean Date and the Pup Weighing Group in the Litter file. (Note: the Issue Breeder status is created by moving a breeder animal to an issue cage.)

(20) Reset the Block UIN and Block Count fields in the Cage record.

(21) If Reason is "O" (Other), this is a final action and the Removal Group in the Animal file must be updated. Prompt user for removal destination.

(22) If all animals in the cage have been weaned, add a new occurrence for each animal in the Animal Status Group in the Animal file by performing the following:

(23) Using the sex entered by the user in step 5, read each occurrence in the Cage Animal Group.

(24) If the occurrence read is for the same sex entered by the user, find the Animal record in the Animal file that corresponds to this animal's UIN.

(25) Maintaining the proper order of the Animal Status Group, add a new occurrence for the status entered by the user. Store Status (entered by user), Building Room (current building room), Cage (New Cage), and Status Change Date (today's date).

(26) If the user entered a valid cage in step 12, go to step 1.

(27) If the user entered a valid function in step 12, call the module as determined by the Function Table in step 15.

*** Note *** This is a new screen.

Module Name: Screen A.21

Descriptive Title: Pup Weighing

Method of Invocation: User Selection

Inputs: Cage
Files ..... CAGE

Outputs: Files ..... LITTER

Called Modules: None

Function: To record the total weight of pups in a given cage.

Assumptions:

1. A litter has been observed.

2. The Dam is still in the cage.

3. The weighing device is calibrated the same as in INLIFE and is attached to the user terminal.

4. The user may enter a number between 1 and 23, inclusive, for the number of animals to be weighed. Individual and litter weights are accomodated in this function.

Process:

(1) User has entered the Pup Weighing function, a cage or range of cages, and/or a processing mode. If the Enter key is pressed without entering a cage or function, go to step 23 else perform the following for each cage chosen:

(2) Check for a litter in this cage.

(3) Using female UIN from CAGE file, locate the corresponding record in the LITTERS file.

(4) Display cage header at the top of the screen.

(5) Prompt user to weigh the empty cage. If the value does not appear on the screen, enter it.

(6) Prompt the user to enter whether this is entire litter (L) or individual animal (I).

(7) Find the corresponding record in the Litter file. Retrieve the number in the litter from the corresponding record.

(8) If this is a full litter weighing, weigh the entire litter as follows. Otherwise repeat the following for each animal in the litter.

(9) Prompt user to place the animal(s) to be weighed in the empty cage.

(10) Receive #WEIGHT and #UNITS (grams, pounds, etc.) from weighing device. Verify that weight is reasonable for this strain.

(11) Return to Step 9 and perform another weighing on the current cage, if individual weighing.

(12) If a special processing mode is not in effect, do the following (13-17):

(13) Prompt the user to enter a function or cage. Then accept user input.

(14) If user input was more than just pressing the enter key, then do the following (15-17):

(15) Check to see if the user input was a valid cage by reading the validation tables.

(16) If not a valid cage, then check to see if user input was a valid function by checking the function tables.

(17) If not a valid cage or valid function, display an error message and return to step 13.

(18) Whether or not a special processing mode is in effect, if any new animal information has been stored, do the following (19-20):

(19) Display "Are all the data on this screen correct?". If the user enters "Y", continue. If the user enters "N", allow the editing of the data just entered.

(20) Add an occurrence to the Pup Weighing Group for each animal or animals weighed in the LITTERS file and record #WEIGHT, #UNITS, #NUM-WEIGHED, #DATE, and

session data. Number weighed will be the entire litter or the number 1.

(21) If a valid new cage was entered in step 13, clear the screen and go to step 2.

(22) If a valid new function was entered in step 13, return to the driver module with the number of the cage just processed and the name of the new function.

(23) If enter was pressed in step 1, or if a special processing mode is in effect, return to the driver module with no cage or function.

APPENDIX E

MODULE SPECIFICATIONS FOR REPORTS

General Assumptions for Reports:

(1) In the ANIMAL file, the most recent status is located at the bottom of the PE group.

(2) Dates are stored in the format YYMMDD.

(3) Weaned is not a valid status.

(4) If a report is run for historical purposes (as opposed to the time the report is normally run), the report should be re-created exactly as it would have looked if the historical date was the current date.

(5) Breeder pairs are not necessarily established for life. A breeder pair may be established, then dissolved, then later reestablished. Breeder pairs may also be physically separated at some point while remaining established.

(6) Breeder pairs will always remain in the same room, even if they are placed in separate cages.

Module Name: BS001, BS004

Descriptive Title: Daily Recap Reports - Assignments
Male and Female Breeder Cards

Method of Invocation: Autosubmit Daily or by User Selection

Inputs: #TODAY .... The given run date
Files ..... ALLOCATE
ANIMAL
DATE-ESTABLISHED
LITTER

Outputs: Reports BS001
BS001s (summary)
Male Breeder Card
Female Breeder Card

Called Modules: None

Function: Records all assignments made during the given day. Creates Breeder Cards for all animals who were assigned Breeder status during the given day.

Assumptions:

1. For Report BS004, assume that in a harem, each female has a separate Female Breeder Card.

2. Source cage is the cage the animal is in as of #TODAY, i.e., the current cage. **NCTR APPROVED**

Process Summary:

BS001/BS004 is composed of four separate parts. Process 1 obtains the data fields required for both BS001 and BS004. Process 2 sorts the data in two files to allow for proper generation of BS001 and BS004 report output. Process 3 prints out BS001, BS001s, and BS004. Process 4 deletes work files.

Process 1:

(1) Find all ANIMAL records that have a status has changed on the given date. This find initiates a loop that will end after step (14) below. Each iteration of the loop will process one record. Do the following for each record (2 - 16):

(2) Store the first two characters of the animal's UIN in #STRAIN. Store information found in the current animal record in #OPERATOR, #UIN, #SEX, #CAN-NO, #BREEDING-ID, and #BIRTH-DATE.

(3) Search through the current animal's Status Change records <u>from the bottom (most recent) toward the top, quitting at the bottom of the loop when a status change date prior to #TODAY is encountered.</u> Do the following for each status changed on the given date (4 - 15):

(4) Store information found in the current status record in #CURRENT-STATUS, #CURRENT-CAGE, and #BLDG-ROOM.

(5) Add one to the animal's #ORDER-OF-STATUS. This variable keeps track of the number of times in a day an animal's status changes, with one being the most recent status change on the <u>given</u> day.

(6) Look at the status record preceding the current status record to find the previous status and previous cage. Store these values in #PREV-STATUS and #PREV-CAGE. If there is no preceding status record, these fields will be left blank in BS001.

(7) If the current animal is on experiment <u>(this information can be found in the ANIMAL record)</u>, find this animal's record in the ALLOCATE file to obtain the animal's most recent experiment number. Store this value in #EXP-NUM. If the animal is not on experiment, the experiment field in BS001 will be left blank.

(8) Use the current animal's LITTER-UIN to find the generation in the LITTER file. Store this value in #GENERATION.

(9) If the current animal is a male breeder, look for a record in the DATE-ESTABLISHED file corresponding to this male where the establish date is the given date.

(10) If no record exists in the DATE-ESTABLISHED file, then this male has been assigned breeder status on the given date, but no corresponding female(s) have yet been assigned. Write the following information about this male to work file-one: #BLDG-ROOM, #STRAIN, #CURRENT-CAGE, #ORDER-OF-STATUS, #UIN, #OPERATOR, #PREV-STATUS, #PREV-CAGE, #SEX, #BREEDING-ID, #GENERATION, #CURRENT-STATUS, #CAN-NO, #EXP-NUM, and #MATE-UIN (will be blank in this case). Store blanks in the variable #ESTABLISH-DATE, for use in (13) below.

(11) If a record exists, then the male is paired with a female breeder. In this case, do not store anything in work file-one since this male will be accounted for on the line in BS001 corresponding to his mate(s). Store the establish date in #ESTABLISH-DATE for use in (13) below.

(12) If the current animal is a female breeder, look for a record in the DATE-ESTABLISHED file corresponding to this female where the establish date is #TODAY. If a record exists, get the corresponding male's UIN from DATE-ESTABLISHED into #MATE-UIN, and the corresponding establish date into #ESTABLISH-DATE. If no record exists, set #MATE-UIN and #ESTABLISH-DATE to blanks. Write all information about this female to work file-one [see step (10)].

(13) For all breeders encountered, write to work file-two: #CURRENT-CAGE, #UIN, #ESTABLISH-DATE, #STRAIN, #PREV-CAGE, #BIRTH-DATE, #BLDG-ROOM, and #SEX.

(14) If animal is not a breeder, write all information about this animal to work file-one [see step (9)].

(15) Return to (3) above to find the current animal's next status change on the current day.

(16) Return to (1) above to find and store information about the next animal whose status changed during the given day.

Process 2:

(1) Sort work file-one by #BLDG-ROOM, #STRAIN, #CAGE, #ORDER-OF-STATUS, and #UIN.

(2) Sort work file-two by CAGE.

Process 3:

(1) Set #COUNT to zero. This counter keeps track of how many members are in a block. Set all summary total-counters to zero. These counters keep track of information to be stored on the summary page after every strain or building room change.

(2) Read one line from work file-one into the following variables: #BLDG-ROOM, #STRAIN, #CURRENT-CAGE, #ORDER-OF-STATUS, #UIN, #OPERATOR, #PREV-STATUS, #PREV-CAGE, #SEX, #BREEDING-ID, #GENERATION, #STATUS, #CAN-NO, #EXP-NUMBER, and

#MATE-UIN. This read initiates a loop that processes one animal. Do the following for each animal (3 - 17):

(3) Each time a new line of variables is read from the work file, if the #CAGE or #ORDER-OF-STATUS changes, do the following (4 - 6):

(4) If the temporary storage variables contain information about a female breeder, and the #MATE-UIN corresponding to this [group of] female(s) is not empty, then count the corresponding male whose UIN is located in #MATE-UIN by adding one to #COUNT.

(5) Write out all temporary storage variables to BS001.

(6) Set #COUNT to zero.

(7) Each time a new line of variables is read from the work file, if the #BLDG-ROOM or #STRAIN changes, do the following (8 - 10):

(8) Do everything that is required whenever #CAGE or #ORDER-OF-STATUS changes [see step (3)].

(9) Go to a new page of the report and write out new page and column headers.

(10) Write out a separate summary page that shows the results of all summary total-counters. Set all summary total-counters to zero.

(11) Add one to #COUNT to count the current animal.

(12) If #COUNT is greater than one, **and the new information differs** from the information stored in the temporary storage variables (by SEX, non-consecutive UINs, or LITTER-UINs), then the new information is not part of the block already stored in the temporary storage variables. Do the following to write out the temporary storage variables to BS001 (13 - 16):

(13) If the entries in the temporary storage variables describe a female breeder who has a mate, count the mate by adding one to #COUNT.

(14) Write out all temporary storage variables to BS001, including #COUNT-1 (cannot count the new information just read in from the work file).

(15) Add the information found in the temporary storage variables to the summary total-counters.

(16) Set #COUNT back to one to account for the new information read in from the work file. Proceed directly to (17) below.

(17) If #COUNT has the value one, copy the new information read in from the work file into the temporary storage variables. It should not be written to the report until it is determined that subsequent lines in the work file do not belong to the same block. Step (12) above makes this determination. Note in (12) above: If the count had been greater than one but the new line from the work file was determined to be part of the same block already stored in temporary storage variables, the program would have counted the new line but not stored it.

(18) After reading all lines from the work file, write out the last summary page for report BS001.

(19) Read a record from work file-two into the following variables: #CAGE, #UIN, #MATED, #STRAIN, #GENERATION, #PREV-CAGE, #BIRTH-DATE, #BLDG, and #SEX. This read initiates a loop that ends after (22). For each record, do the following (20 - 22):

(20) Begin a new breeder card.

(21) If animal is a female breeder, write out all necessary information to her breeder card.

(22) If animal is a male breeder, write out all necessary information to his breeder card. Different information is required for the male breeder than for the female breeder.

Process 4:

(1) Delete work file-one and work file-two.

*** Note *** This is a totally rewritten report.

Module Name: BS002o

Descriptive Title: Daily Recap Report - Pregnant, Plugged, Litter, and Newborn Observations

Method of Invocation: Autosubmit Daily or by User Selection

Inputs: #TODAY .... the given run date
Files ..... ANIMAL
DATE-ESTABLISHED

Outputs: Report BS002o
Report BS002os (summary)

Called Modules: None

Function: Report observations made during the given day.

Assumptions:

1. Source Cage, building, strain, and status are the location of the animal as of #TODAY, i.e., the current cage. **NCTR APPROVED**

2. A newborn litter (block) of animals gets a sex of "U" assigned (see Process 1, #6).

Process Summary:

BS002o is composed of four separate parts. Process 1 obtains the required data to report on all observations made in the current day. Process 2 sorts the data. Process 3 prints out information regarding observations and supplies the summary report, BS002os. Process 4 deletes the work files.

Process 1:

(1) Find all records in DATE-ESTABLISHED that have ANY litter observation date equal to the given date. This returns all records fulfilling this criteria but does not give the corresponding litter observation. This find initiates a loop that ends after step (6) below and processes one DATE-ESTABLISHED record per iteration. Do the following for each record (2 - 6):

(2) Find the (first) corresponding ANIMAL record for the dam in the current DATE-ESTABLISHED record. This find initiates a loop that ends after step (6) below, and processes one dam per iteration. Do the following for each dam (3 - 6):

(3) Store in variables for future printing: #STRAIN-CODE, #OPERATOR, #UIN, #SEX, and #BREEDING-ID.

(4) Search through this dam's animal status records from the bottom (most recent) toward the top for the most recent change date on or before #TODAY to breeder status. Quit the loop at the bottom when a match is found. Store the information found in #STATUS, #BLDG-ROOM, and #CAGE for printing.

(5) Search through the current dam's litter-observations in DATE-ESTABLISHED to find all litter observations made on the given date. When found, write the following to work file-one: #BLDG-ROOM, #STRAIN-CODE, #CAGE, litter observation, #UIN, #OPERATOR, #STATUS, #SEX, #BREEDING-ID, '(blanks for litter UIN)', and '(blanks for litter generation)'.

(6) If the current litter observation is 'Littered', then find the corresponding litter in the LITTERS file. Write the following to work file-one: the building-room where the litter was born, litter strain, litter birth cage, 'Newborn' for observation, litter UIN, operator (for litter), blanks for status, litter UIN, blanks for sex, or U, blanks for breeding-ID, litter UIN, and litter generation.

Process 2:

(1) Work file-one is sorted in the following order: #BLDG-ROOM, #STRAIN, #SOURCE-CAGE, #OBSERVATION, and #UIN.

Process 3:

(1) Read a line from work file-one into #BLDG-ROOM, #STRAIN, #CAGE, #OBSERVATION, #UIN, #OPERATOR, #STATUS, #SEX, #BREEDING-ID, #LITTER-UIN, and #LITTER-GEN. This read initiates a loop that ends after step (10) below, and processes one line of date from work file-one per iteration. Do the following for each line (2 - 10):

(2) At the first read (start of data), print out new page and column headers; zero out #COUNT, which counts the members of the current block; and zero out the summary-counters.

(3) Whenever the current strain or the current building-room changes, write out all summary counters to a new summary page, zero out summary counters, and write page and column headers on a new report page.

(4) Whenever the current cage changes, write out the block from the previous cage to the report, and set #COUNT to zero (the first entry into the new cage will be counted immediately below).

(5) Add one to #COUNT to count the new entry read in at step (1) above.

(6) If #COUNT is greater than one, **and the new information differs** from the information already stored in the temporary storage variables (by SEX, non-consecutive UINs, or LITTER-UINs), then the new information is not part of the block already stored in the temporary storage variables. Do the following to write out the temporary storage variables to BS002o (7 - 9):

(7) Write all temporary variables to the report, including #COUNT-1.

(8) Add to summary counters.

(9) Set #COUNT to one to account for the new information just read in. Proceed directly to (10) below.

(10) If #COUNT has the value one, copy the new information read from work file-one into temporary storage variables. The information should not be written to the report until it is determined that subsequent lines in work file-one do not belong to the same block. Step (6) above makes this determination. Note in (6) above, that if the count had been greater than one but the new line from the work file was determined to be part of the same block already stored in the temporary storage variables, the program would have counted the new line but not stored it.

(11) Write out the last summary report using the results from the summary counters.

Process 4:

(1) Delete work file-one.

*** Note *** This is a totally rewritten report.

Module Name: BS002p

Descriptive Title: Daily Recap Report - Phenotype Observations

Method of Invocation: Autosubmit Daily or by User Selection

Inputs: #TODAY .... the given run date
Files ..... ANIMAL

Outputs: Report BS002p
Report BS002ps (summary)

Called Modules: None

Function: Report phenotype observations of newly weaned animals made during the given day.

Assumptions:

1. Source Cage, building, strain, and status are location of the animal as of #TODAY.

Process Summary:

BS002p is composed of four separate parts. Process 1 obtains necessary information to report on phenotypes. Process 2 sorts the data. Process 3 prepares the BS002p report and the summary report, BS002ps. Process 4 deletes the work file.

Process 1:

(1) Read ANIMAL records that were created on the given date. This read initiates a loop that ends after step (5) below and processes one record each time through the loop.

(2) If the animal whose record was just read has a phenotype, do the following (3 - 5):

(3) Store #STRAIN, #UIN, #OPERATOR, #SEX, #BREEDING-ID, AND #PHENOTYPE from the ANIMAL record.

(4) Search through the animal status group from the bottom (most recent) toward the top to find the first occurrence of a status change date equal to #TODAY. By searching from the bottom to the top and stopping at the first encounter, the latest status change for #TODAY will be found. Quit the loop at the bottom when

a match is found. Most of the time, #TODAY is the current date, and a match will be found the first try (at the bottom of the status group). Store values in #BLDG-ROOM, #STATUS, and #CAGE.

(5) Write the following information for that animal to work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #UIN, #OPERATOR, #STATUS, #SEX, #BREEDING-ID, and #PHENOTYPE.

Process 2:

(1) Sort work file-one in the following order: #BLDG-ROOM, #STRAIN, #SOURCE-CAGE, and #UIN.

Process 3:

(1) Read a line from work file-one into #BLDG-ROOM, #STRAIN, #CAGE, #UIN, #OPERATOR, #STATUS, #SEX, #BREEDING-ID, and #PHENOTYPE. This read initiates a loop that ends after step (10) below and processes one line of date from work file-one per iteration. Do the following for each line (2 - 10):

(2) At the first read (start of data) print out new page and column headers; zero out #COUNT, which counts the members of the current block; and zero out the summary-counters.

(3) Whenever the current strain or the current building-room changes, write out all summary counters to a new summary page, zero out summary counters, and write page and column headers on a new report page.

(4) Whenever the current cage changes, write out the block from the previous cage to the report and set #COUNT to zero (the first entry into the new cage will be counted immediately below).

(5) Add one to #COUNT to count the new entry read in at step (1) above.

(6) If #COUNT is greater than one, **and the new information differs** from the information already stored in the temporary storage variables (by SEX, non-consecutive UINs, or LITTER-UINs), then the new information is not part of the block already stored in the temporary storage variables. Do the following to write out the temporary storage variables to BS002p (7 - 9):

(7) Write all temporary variables to the report, including #COUNT - 1.

(8) Add to summary counters

(9) Set #COUNT to one to account for the new information just read in. Proceed directly to step (10) below.

(10) If #COUNT has the value one, copy the new information read from work file-one into temporary storage variables. The information should not be written to the report until it is determined that subsequent lines in work file-one do not belong to the same block. Step (6) above makes this determination. Note in (6) above, that if the count had been greater than one but the new line from the work file was determined to be part of the same block already stored in the temporary storage variables, the program would have counted the new line but not stored it.

(11) Write out the last summary report using the results from the summary counters.

Process 4:

(1) Delete work file-one.

Module Name: BS003

Descriptive Title: Daily Recap Report - Removals

Method of Invocation: Autosubmit Daily or by User Selection

Inputs: #TODAY .... the desired run date
Files ..... ALLOCATE
ANIMAL

Outputs: BS003
BS003s (summary)

Called Modules: None

Function: This report provides a list of all animals removed from the breeding colony during the day. Removals may be for any of the following reasons: dead, culled, surplused, retired, or for analysis by Micro, Chemistry, or Pathology. A removal applies to one or more animals from the same cage, homogeneously identified, and removed for the same reason.

Assumptions:

1. There can be more than one status change in a day. **NCTR APPROVED**

2. The most recent experiment number as of the end of #TODAY is reported. **NCTR APPROVED**

3. If the removal date is #TODAY, then there will be a status change date equal to #TODAY.

Process Summary:

BS003 is composed of four separate parts. Process 1 gathers the necessary data for the report. Process 2 sorts the data. Process 3 provides BS003 and BS003s in report form. Process 4 deletes the work file.

Process 1:

(1) Read in all ANIMAL files with a removal date equal to #TODAY. This read will initiate a loop that will process one animal. For each animal, do the following (2 - 10):

(2) Store information for this animal in #OPERATOR, #UIN, #SEX, #BREEDING-ID, #REMOVAL-DEST, and #CID.

(3) Search through the animal's status change dates from the bottom (most recent) toward the top to find a status change date equal to #TODAY and with a status equal to a legitimate removal status. This will usually be the first record encountered since #TODAY is usually the current date. Exit at the bottom of the loop when a match is found. Store this status in #REMOVAL-REASON.

(4) Find the animal's status prior to the removal, and store the status and cage in #CURRENT-STATUS and #CAGE.

(5) Locate the experiment number that the animal was on #TODAY by searching through ALLOCATE in the following way:

(6) Set #DATE to the value zero.

(7) Find all ALLOCATE records where ALC-UIN is equal to the animal UIN. This initiates a loop that processes steps (8) and (9) for each ALC-UIN:

(8) If the allocation date is greater than #DATE and less than or equal to #TODAY, then #EXP is assigned the corresponding experiment number from ALLOCATE.

(9) Set #DATE equal to the allocation date.

(10) Write the following information for this animal to work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #REMOVAL-REASON, #UIN, #OPERATOR, #CURRENT-STATUS, #SEX, #BREEDING-ID, #REMOVAL-DEST, #EXP, and #CID.

Process 2:

(1) Sort work file-one on #BLDG-ROOM, #STRAIN, #SOURCE-CAGE, #REMOVAL-REASON, and #UIN.

Process 3:

(1) Read the following from work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #REMOVAL-REASON, #UIN, #OPERATOR, #CURRENT-STATUS, #SEX, #BREEDING-ID, #REMOVAL-DEST, #EXP, and #CID. This read initiates a loop that processes one line from work file-one. For each line, do the following (2 - 9):

(2) At the start of data, print new page and column headers.

(3) Whenever the building-room or strain changes, do the following:

Write out the values in the temporary variables to the report.

Set #COUNT to zero. The new entry will be counted below.

Write out a summary page for the previous building-room and strain.

Zero out the summary counters.

Write new page and column headers

(4) Add one to #COUNT to count the new information read in above.

(5) If #COUNT is greater than one, **and the new information differs** from the information already stored in the temporary storage variables (by SEX, non-consecutive UINs, different REMOVAL DESTINATIONs, or LITTER-UINs), then the new information is not part of the removal record stored in the temporary variables. Do the following to write out the temporary storage variables to BS003 (6 - 8):

(6) Write all temporary variables to the report, including #COUNT - 1.

(7) Add to summary counters.

(8) Set #COUNT to one to account for the new information just read in. Proceed directly to step (9) below.

(9) If #COUNT has the value one, copy the new information read from work file-one into temporary storage variables. This information should not be written to the report until it is determined that subsequent lines in work file-one do not belong to the same removal status. Step (5) above makes this determination. Note in (5) above, that if the count had been greater than one but the new line from the work file was determined to be part of the same removal block already stored in the temporary storage variables, the program would have counted the new line but not stored it.

(10) Write out the last summary report using the results from the summary counters.

Process 4:

(1) Delete work file-one.

Module Name: BS005

Descriptive Title: Animal Birth Report

Method of Invocation: Autosubmit Weekly or by User Selection

Inputs: #TODAY ..... the given run date
#FROM-BIRTH-DATE ... beginning of range of birth dates to be reported
#TO-BIRTH-DATE ..... end of range of birth dates to be reported
Files ..... ANIMAL
LITTERS

Outputs: BS005
BS005s (summary)

Called Modules: None

Function: Provides a complete list of litters born in the breeding colony.

Assumptions:

1. The current cage is the cage the dam is in as of #TODAY, even if the litter is weaned. If the dam is dead, write 'dead' in the cage column.

2. If animals are weaned, the actual wean date is given. If not weaned (the usual case since this is a birth report), the proposed wean date is determined from the Strain Table. Note that this could cause some ambiguity ONLY if the birth date range was historical dates instead of the current range of dates. If a litter was not weaned within the number of days the Strain Table suggests, the "proposed wean date" would be less than the actual wean date. This may, in some cases, suggest that a strain was already weaned when it actually had not been.

Process Summary:

BS005 is composed of six separate parts. Process 1 gathers data pertaining to all births during the given period. Process 2 sorts the data. Process 3 generates BS005. Process 4 sorts the summary work file. Process 5 generates BS005s. Process 6 deletes the work files.

Process 1:

(1) Read records from LITTERS by birth date in the range beginning with #FROM-BIRTH-DATE and ending with #TO-BIRTH-DATE. This read initiates a loop that ends after step (7) below and processes one litter record per iteration. For each litter record, do the following (2 - 7):

(2) Store information in variables for future printing: #BIRTH-DATE, #SIRE-UIN, #DAM-UIN, #LITTER-UIN, #STRAIN, #BLDG-ROOM, #LITTER-NUM, #LITTER-GEN, and #LITTER-SIZE.

(3) If a wean-date is given in the current litter record, store in #WEAN-DATE. If a wean-date is not given, look up the number of days from birth to wean in the Strain Table. Add this value to the birth date, and store the result in #WEAN-DATE.

(4) Match the DAM-UIN given in the current litter record to DAM-UINs in the DATE-ESTABLISHED file. Find the most current date that this dam was given Breeder status and store in #ESTABLISH-DATE.

(5) Find the dam's ANIMAL record.

(6) Search through the dam's animal status group to find the cage and building-room corresponding to the most recent status change. Store in #CAGE and #BLDG-ROOM.

(7) Write the following to work file-one: #BIRTH-DATE, #STRAIN, #BLDG-ROOM, #CAGE, #WEAN-DATE, #ESTABLISH-DATE, #SIRE-UIN, #DAM-UIN, #LITTER-UIN, #LITTER-NUM, #LITTER-GEN, and #LITTER-SIZE.

Process 2:

(1) Sort work file-one by #BIRTH-DATE, #STRAIN, #BLDG-ROOM, and #CAGE.

Process 3:

(1) Read from work file-one: #BIRTH-DATE, #STRAIN, #BLDG-ROOM, #CAGE, #WEAN-DATE, #ESTABLISH-DATE, #SIRE-UIN, #DAM-UIN, #LITTER-UIN, #LITTER-NUM, #LITTER-GEN, and #LITTER-SIZE. This read initiates a loop that ends after step (12) below and processes one line from work file-one per iteration. For each line, do the following (2 - 12):

(2) At the first line of data (at start of data) write out new page and column headers and set #BIRTH-FLAG and #WEAN-FLAG to one. When these flags are on, the current birth

date and/or wean date is to be printed out on report BS005. When they are off, blanks are printed.

(3) Whenever the birth date or wean date changes, do the following (4 - 5):

(4) If the birth date changed, write the total born on the previous birthday, and set the total-born counter back to zero. Go to a new page and write new page and column headers. Set #BIRTH-FLAG to one so the new birth date will be printed. Store the new birth date in #TEMP-BIRTH-DATE.

(5) Whether or not the birthday changed, the wean date should be printed. Set #WEAN-FLAG to one, and store the new wean date in #TEMP-WEAN-DATE.

(6) Whenever the building-room or strain changes, do the following (7 - 8):

(7) Write out the total number of this strain, the previous building-room, and the litter count of the strain. Set #COUNT-LITTER-SIZES to zero. This counts the total animals in the current building with the current strain. If this is a new strain, write to work file-two: #TEMP-STRAIN, #TOT-LIT-THIS-STRAIN, and #TOT-BORN-THIS-STRAIN. Set the two #TOT- counters to zero. Store new strain in #TEMP-STRAIN.

(8) Whether or not the strain changed, store the new building-room in #TEMP-BLDG.

(9) If #BIRTH-FLAG is off, set #BIRTH-DATE to ' ' so blanks will be printed. If #WEAN-FLAG is off, set #WEAN-FLAG to ' '.

(10) Increment #COUNT-LITTER-SIZES by #LITTER-SIZE.

(11) Write to report BS005: #BIRTH-DATE, #WEAN-DATE, #ESTABLISH-DATE, #CAGE, #SIRE-UIN, #DAM-UIN, #LITTER-UIN, #LITTER-NUM, #LITTER-GEN, and #LITTER-SIZE.

(12) Turn off #BIRTH-FLAG and #WEAN-FLAG.

Process 4:

(1) Sort work file-two by strain.

Process 5:

(1) Read from the summary-work file: #STRAIN, #TOT-LIT-THIS-STRAIN, and #TOT-BORN-THIS-STRAIN. This read initiates a loop that ends after step (9) below and processes one line from the summary-work file per iteration. For each line, do the following (2 - 9):

(2) Copy #STRAIN to #TEMP-STRAIN to save it. Several lines of data from the summary-work file may have the same strain.

(3) At the start of data, write out summary page and column headers. Set counters #TOT-LIT and #TOT-BORN to zero.

(4) Whenever the strain changes, do the following (5 - 7):

(5) Figure average litter size from #TOT-LIT and #TOT-BORN. Figure projected number to be weaned if 90% are weaned. Figure projected number to be weaned if 95% are weaned.

(6) Write out to the summary report: #STRAIN, #TOT-LIT, #TOT-BORN, average litter size, 90%-projection, and 95%-projection.

(7) Reset #TOT-LIT and #TOT-BORN to zero.

(8) Increment #TOT-LIT by #TOT-LIT-THIS-STRAIN.

(9) Increment #TOT-BORN by #TOT-BORN-THIS-STRAIN.

Process 6:

(1) Delete work file-one and work file-two.

Module Name: BS006

Descriptive Title: Weaned Animal Report

Method of Invocation: Autosubmit Weekly or by User Selection

Inputs: #TODAY .... the desired run date
#FROM-WEAN-DATE ... the beginning of the range of wean dates to be reported
#TO-WEAN-DATE ..... the end of the range of wean dates to be reported
Files ..... ANIMAL
LITTER

Outputs: BS006
BS006s (summary)

Called Modules: None

Function: Prints a complete list of litters weaned during a specified time period.

Assumptions:

1. Current cage is the birth cage, current building-room is the room where the litter was born. **Documented in External Design, p. B-11**

2. Status reported in this report is the status given at wean time. If this report is used for historic purposes, the status given will reflect the status at wean time, not necessarily the current status. **NCTR APPROVED**

Process Summary:

BS006 is composed of six separate parts. Process 1 gathers data about animals weaned during the given period. Process 2 sorts the data. Process 3 prints out report BS006. Process 4 sorts the summary information. Process 5 writes out the summary information. Process 6 deletes the work files.

Process 1:

(1) Read LITTER records by litter wean date in the range beginning with #FROM-WEAN-DATE and ending with #TO-WEAN-DATE. This read initiates a loop that will end after step (7) below, and processes one litter record per iteration. For each litter record, do the following (2 - 7):

(2) Copy information from this file to #WEAN-DATE, #BIRTH-DATE, #SIRE-UIN, #DAM-UIN, #LITTER-UIN, #LITTER-SIZE, and #STRAIN.

(3) Find the DAM's ANIMAL record. Search through her status records <u>from the bottom record (most recent) toward the top</u> to find her cage at the time the litter was born. <u>Quit searching after a match has been found.</u> This should be stored in #BIRTH-CAGE and #BLDG-ROOM.

(4) Read ANIMAL records that have a litter UIN equal to the litter UIN in step (1) above. This read initiates a loop that will end after step (7) below, and processes one animal record per iteration. For each animal record, do the following (5 - 7):

(5) <u>Store the top status (the earliest one) in #STATUS.</u>

(6) If the animal's REMOVAL-STATUS-DATE indicates that the animal has been removed at/by the time of weaning, and the removal destination is Micro, Pathology, or Chemistry, then the #CURRENT-STATUS is assigned 'Survey'.

(7) Write the following to work file-one for each animal: #WEAN-DATE, #BLDG-ROOM, #STRAIN, #BIRTH-CAGE, #CURRENT-STATUS, #BIRTH-DATE, #DAM-UIN, #SIRE-UIN, #LITTER-UIN, and #LITTER-SIZE.

Process 2:

(1) Sort work file-one by #WEAN-DATE, #BLDG-ROOM, #STRAIN, #CAGE, and #STATUS.

Process 3:

(1) Read from work file-one: #WEAN-DATE, #BLDG-ROOM, #STRAIN, #CAGE, #STATUS, #BIRTH-DATE, #DAM-UIN, #SIRE-UIN, #LITTER-UIN, and #LITTER-SIZE. This read initiates a loop that processes one line of data from the work file. For each line of data, do the following (2 - 14):

(2) At the start of data, do the following: Write new page and column headers. Set #BIRTH-FLAG and #WEAN-FLAG to one. These flags indicate whether the birth date and wean date should be printed on a particular line of the report. Set all counters to zero. Store #WEAN-DATE, #BIRTH-DATE, #BLDG-ROOM, and #STRAIN in temporary-storage variables

#TEMP-WEAN-DATE, #TEMP-BIRTH-DATE, #TEMP-BLDG, and #TEMP-STRAIN.

(3) If the birth date or wean date changes, do the following (4 - 5):

(4) If the wean date changes, do the following: Do everything that occurs whenever a building-room or strain changes. Write out totals for number dead, number culled, number survey, etc., for this date. Set the above total counters to zero. Turn the wean flag on so that the new wean date will be printed. Store the new wean date in temporary-storage variable #TEMP-WEAN-DATE.

(5) Whether or not the wean date changes, turn the birth flag on and store the new birth date in a temporary storage variable #TEMP-BIRTH-DATE.

(6) If the building-room or strain changes, do the following (7 - 8):

(7) Do everything that occurs whenever a cage changes. Write out total animals for this building-room, this strain. Set the #TOT-ANIMAL counter to zero. Store the new building in temporary storage variable #TEMP-BLDG.

(8) If the strain changes, write to work file-two: #TEMP-STRAIN, #TOT-LIT-THIS-STRAIN, #TOT-BORN-THIS-STRAIN, #TOT-DEAD, #TOT-CULLED, #TOT-SURPLUS, #TOT-SURVEY, #TOT-ASSIGN-FEEDER, #TOT-ASSIGN-BREEDER, and #TOT-ISSUED.

Zero out the above counters.

Store the new strain in temporary storage.

(9) If the cage changes, do the following (10 - 11):

(10) Write to the report: #WEAN-DATE, #BIRTH-DATE, #TEMP-CAGE, #TEMP-SIRE-UIN, #TEMP-DAM-UIN, #TEMP-LITTER-UIN, #TEMP-LITTER-SIZE, #DEAD, #CULLED, #SURPLUS, #SURVEY, #ASSIGN-BREEDER, #ASSIGN-FEEDER, and #ISSUE.

(11) Add above cage counters to #TOT- counters. Zero out the cage counters. Turn off #BIRTH-FLAG and #WEAN-FLAG if they are on.

(12) Add to one of the following to count the current status: #DEAD, #CULLED, #SURPLUS, #SURVEY, #ASSIGN-BREEDER, #ASSIGN-FEEDER, or #ISSUE.

(13) If #WEAN-FLAG is off, set #WEAN-DATE to ' ' so blanks will print out on the report.

(14) If #BIRTH-FLAG is off, set #BIRTH-DATE to ' ' so blanks will print out on the report.

Process 4:

(1) Sort work file-two by strain.

Process 5:

(1) Read from work file-two: #STRAIN, #TOT-LITTERS, #TOT-BORN, #TOT-DEAD, #TOT-CULLED, #TOT-SURPLUS, #TOT-SURVEY, #TOT-ASSIGN-FEEDER, #TOT-ASSIGN-BREEDER, and #TOT-ISSUE. This read initiates a loop that processes one line from the work file per iteration. For each line, do the following (2 - 10):

(2) At the start of data, write new page and column headers to the summary report.

(3) Copy #STRAIN to #TEMP-STRAIN to save it. Several lines of data from the summary-work file may have the same strain.

(4) At the start of data, write out summary page and column headers. Set the following counters to zero: #T-LITTERS, #T-BORN, #T-DEAD, #T-CULLED, #T-SURPLUS, #T-SURVEY, #T-ASSIGN-FEEDER, #T-ASSIGN-BREEDER, and #T-ISSUE.

(5) Whenever the strain changes, do the following (6 - 9):

(6) Figure the average number per litter of the previous strain, and store in #AVG-PER-LITTER.

(7) Figure the percent dead from #T-DEAD/#BORN and store in #PER-DEAD. Figure similarly for culled, surplus, survey, assign-breeder, assign-feeder, and issue, storing the results in #PER-CULLED, #PER-SURPLUS, #PER-SURVEY, #PER-A-BREEDER, #PER-A-FEEDER, and #PER-ISSUE.

(8) Write out to the summary report: #STRAIN, #T-LITTERS, #T-BORN, #AVG-PER-LITTER, #T-DEAD, #PER-DEAD, #T-CULLED, #PER-CULLED, #T-SURPLUS, #PER-SURPLUS, #T-SURVEY, #PER-SURVEY, #T-ASSIGN-BREEDER, #PER-A-BREEDER, #T-ASSIGN-FEEDER, #PER-A-FEEDER, #T-ISSUE, and #PER-ISSUE.

(9) Reset #T- counters to zero.

(10) Increment #T- counters by the #TOT counters read in for each line of data.

Process 6:

(1) Delete work file-one and work file-two.

*** Note *** This is a totally rewritten report.

Module Name: BS007

Descriptive Title: Breeder Assignment Report

Method of Invocation: Autosubmit Weekly or by User Select

Inputs: #START-DATE ..... The beginning of the range of dates for which this report is made
#END-DATE ..... The end of the range of dates
Files ..... ANIMAL
DATE-ESTABLISHED

Outputs: BS007
BS007s (summary)

Called Modules: None

Function: To identify the breeder pairs established during a date range.

Assumptions:

1. Any animal in breeder status is included in this report, whether or not the animal is established as part of a breeder pair. **NCTR APPROVED**

2. Total Strain is considered to be the strain of the progeny of the breeder or breeder pair. If both dam and sire are of the same strain, or if there is a single breeder, then the progeny will have the same strain. If the strain of the dam and sire differ, they will be listed once under the (new) strain of the progeny. **NCTR APPROVED**

3. The number in the Total Strain entry is the number of breeder pairs relating to that strain. A pair counts as one. A single breeder also counts as one.

4. Establish date is the date a pair is established if there is a pair. If there is a single breeder, the establish date is the date the animal was established as a breeder.

5. Current cage is the cage the breeder pair or single breeder occupied as of the #END-DATE. Previous cage is the entry in the animal status group immediately preceding the entry for the current cage and room. It is possible that current cage and previous cage could be the same if the animal was not moved when changed to Breeder status.

6. A new column header will be required in BS007 to report the sire's current cage if it differs from the dam's.

Process Summary:

BS007 is composed of six separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 resorts the data to allow for generation of the strain summary. Process 5 prepares the strain summary page. Process 6 deletes the work files.

Process 1:

(1) Input the #START-DATE and #END-DATE.

(2) Find all ANIMALS who were changed to breeder status in the date range from #START-DATE to #END-DATE. This instruction initiates a loop that ends after step (23) below and processes one animal per iteration. For each animal, do the following (3 - 23):

(3) Find this animal's entry in DATE-ESTABLISHED, then do the following (5 - 13):

(4) If the animal is male, do nothing. This male will be processed with the corresponding female(s) in the breeder pair.

(5) If the animal is female, do the following (7 - 13):

(6) Store the #DAM-BIRTH-DATE and #UIN (the female's UIN) from the ANIMAL file.

(7) Read through the status group in the ANIMAL file from newest to oldest to determine the #CURRENT-CAGE and #CURRENT-ROOM (from the first Status element where the status change date is less than or equal to the #END-DATE).

(8) Continue reading through the status group to find the #DAM-PREV-CAGE and #DAM-PREV-ROOM.

(9) Store the #ESTABLISH-DATE and #ALT-UIN (the male's UIN) from the DATE-ESTABLISHED file.

(10) If the strain (first two characters) of the #UIN are the same as the first two characters of the #ALT-UIN, then assign these two characters to

#STRAIN. Else find the strain of the breeder pair's progeny from the Strain Table and store in #STRAIN.

(11) Find the ANIMAL record corresponding to male's UIN.

(12) Store the #SIRE-BIRTH-DATE.

(13) Search through the status group to find the #ALT-CURRENT-CAGE, #ALT-CURRENT-ROOM, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM, as in steps (8 - 9) above.

(14) If animal's entry was not found in DATE-ESTABLISHED, do the following (15 - 22):

(15) Search through the animal's status group as in steps (7 - 8) above to find the animal's status at the time of the #END-DATE.

(16) Store #UIN, #CURRENT-CAGE, #CURRENT-ROOM, and the corresponding status change date in #ESTABLISH-DATE.

(17) If the animal is female, do the following (18):

(18) Store #DAM-BIRTH-DATE, #DAM-PREV-CAGE, and #DAM-PREV-ROOM. Store blanks in #SIRE-BIRTH-DATE, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM.

(19) If the animal is male, do the following (20):

(20) Store #SIRE-BIRTH-DATE, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM. Store blanks in #DAM-BIRTH-DATE, #DAM-PREV-CAGE, and #DAM-PREV-ROOM.

(21) Store the first two characters of #UIN in #STRAIN.

(22) Store blanks in #ALT-UIN, #ALT-CURRENT-CAGE, and #ALT-CURRENT-ROOM.

(23) Write to work file-one: #ESTABLISH-DATE, #CURRENT-ROOM, #STRAIN, #CURRENT-CAGE, #UIN, #DAM-BIRTH-DATE, #DAM-PREV-CAGE, #DAM-PREV-ROOM, #SIRE-BIRTH-DATE, #ALT-UIN, #ALT-CURRENT-CAGE, #ALT-CURRENT-ROOM, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM.

Process 2:

(1) Sort work file-one by #ESTABLISH-DATE, #CURRENT-ROOM, #STRAIN, #CURRENT-CAGE, and #UIN.

Process 3:

(1) At the top of each page of output, write page and column headers.

(2) Read from work file-one: #ESTABLISH-DATE, #CURRENT-ROOM, #STRAIN, #CURRENT-CAGE, #UIN, #DAM-BIRTH-DATE, #DAM-PREV-CAGE, #DAM-PREV-ROOM, #SIRE-BIRTH-DATE, #ALT-UIN #ALT-CURRENT-CAGE, #ALT-CURRENT-ROOM, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM.

(3) Establish whether this is a breeder pair, a single male, or a single female. Assign to #STAT a corresponding P, M, or F.

(4) Each time the date established changes, write out the total number of pairs and/or single breeders established on that date and clear the counter for this total.

(5) Each time a new #STRAIN appears, write the strain to the next position of an array that records strains and corresponding counters for breeder pairs and/or singles. Each time a new record is read from the work file for the same strain, increment the counter.

(6) Each time the Building-Room changes, do the following (7 - 9):

(7) Write out the total for each strain in this room on this date.

(8) Write #STRAIN and the corresponding number of pairs/singles to work file-two.

(9) Clear the array that keeps track of strain totals.

(10) For each current cage, write out the detail line for the breeder pair or single in this cage in the following way (11 - 14):

(11) Suppress the establish date on output except at the top of the page and the first time that establish date is encountered.

(12) If #STAT equals "P", AND the current cage is the same as the alternate current cage, AND the current room is the alternate current room, then set #ALT-CURRENT-CAGE and #ALT-CURRENT-ROOM to blanks to suppress printing.

(13) If #STAT equals "P" or "F", then write the following to report BS007: #ESTABLISH-DATE, #CURRENT-CAGE, #DAM-BIRTH-

DATE, #UIN, #DAM-PREV-CAGE, #DAM-PREV-ROOM, #ALT-CURRENT-CAGE, #ALT-CURRENT-ROOM, #SIRE-BIRTH-DATE, #ALT-UIN, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM.

(14) Else if #STAT equals "M", then write the following to report BS007: #ESTABLISH-DATE, #ALT-CURRENT-CAGE, #DAM-BIRTH-DATE, #ALT-UIN, #DAM-PREV-CAGE, #DAM-PREV-ROOM, #CURRENT-CAGE, #CURRENT-ROOM, #SIRE-BIRTH-DATE, #UIN, #SIRE-PREV-CAGE, and #SIRE-PREV-ROOM.

Process 4:

(1) Sort work file-two by the Strain.

Process 5:

(1) Read #STRAIN and #NUMBER from work file-two. This read initiates a loop that ends after step (6) below and processes one record per iteration.

(2) Each time the strain changes, do the following (3 - 5):

(3) Write out the old strain code and the total number of pairs/singles

(4) Increment the grand-total counter by #STRAIN-COUNTER.

(5) Zero out the #STRAIN-COUNTER for individual strains.

(6) Increment the #STRAIN-COUNTER by #NUMBER.

(7) Write out the grand total counter.

Process 6:

(1) Delete work file-one and work file-two.

*** NOTE *** This is a totally rewritten report.

Module Name: BS008

Descriptive Title: Active Breeder Report

Method of Invocation: Autosubmit Monthly (possible ad hoc)

Inputs: #TODAY .... Breeder Inventory Date
Files ..... ANIMAL
DATE-ESTABLISHED
LITTERS

Outputs: BS008

Called Modules: None

Function: To provide a complete list of active breeders as of a given date (usually the date the report is run).

Assumptions:

1. "Active breeder" includes any animal in breeder status on #TODAY. **NCTR APPROVED**

2. Any animal in breeder status on #TODAY is included in this report, whether or not it is currently paired. **NCTR APPROVED**

3. Breeder pairs should be listed together even if their cages differ. In this case, the male will be listed after the corresponding female, and the cage of the male will be noted for reference.

4. Even if breeder pairs are separated into different cages, they will ALWAYS remain in the same room.

5. For breeder pairs, #ESTABLISH-DATE is the date the breeder pair was established. For single breeders, it is the date the animal was established as a breeder.

6. At the bottom of each page, the total number of breeder pairs is to be displayed. This number will include both breeder pairs, which count as one, and breeder singles, which also count as one.

Process Summary:

BS008 is composed of four separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work file.

Process 1:

(1) Find all ANIMAL records where the animal is or has been in breeder status. This find initiates a loop that processes one animal per iteration. For each animal, do the following (2 - 20):

(2) Search through the animal's status group from the bottom toward the top to find where the animal was in breeder status as of #TODAY. If such an entry is found, do the following (3 - 20):

(3) Find the animal's entry in DATE-ESTABLISHED where the date established is less than or equal to #TODAY and the dissolve date is blank or after #TODAY. If an entry is found, do the following (4 - 14):

(4) Set #STAT to "P" to indicate that this animal is a member of a pair on #TODAY.

(5) If the animal is male, do nothing. This animal will be processed when the corresponding female is.

(6) If the animal is female, do the following (6 - 14):

(7) Store #ESTABLISH-DATE from the DATE-ESTABLISHED record.

(8) Store the female's UIN in #UIN, and the male's UIN in #ALT-UIN (found in the DATE-ESTABLISHED record.)

(9) Find the first ANIMAL record corresponding to the #SIRE-UIN.

(10) Search through the male's ANIMAL status group from bottom toward the top to find his entry into breeder status immediately prior to #TODAY. Store the male's cage in #ALT-CAGE.

(11) Find the first LITTER record corresponding to the male's LITTER-UIN. Store the male's litter generation in #ALT-LIT-GENERATION.

(12) Find the first LITTER record corresponding to the female's LITTER-UIN [found in the female's

animal record, see step (1) above]. Store the female's litter generation in #LIT-GENERATION.

(13) Store the female's cage, found in step (2) above in #CAGE.

(14) Store the female's room, found in step (2) above in #BLDG-ROOM.

(15) If no entry was found for the current animal in DATE-ESTABLISHED, then this animal is a single breeder (not associated with a breeder pair). Do the following for each animal (16 - 19):

(16) Store the following from steps (1 and 2) above: #UIN, #ESTABLISH-DATE (from Status Change Date), #CAGE, and #BLDG-ROOM.

(17) Find the first LITTER record corresponding to the animal's #LITTER-UIN [found in step (1) above]. Store the animal's litter generation in #LIT-GENERATION.

(18) If the animal is female, set #STAT to "F", else set it to "M".

(19) Blank out #ALT-UIN, #ALT-CAGE, and #ALT-LIT-GENERATION.

(20) Write the following to work file-one: #BLDG-ROOM, #CAGE, #UIN, #STAT, #ESTABLISH-DATE, #LIT-GENERATION, #ALT-UIN, #ALT-CAGE, and #ALT-LIT-GENERATION.

Process 2:

(1) Sort work file-one by #BLDG-ROOM, #CAGE, and #UIN.

Process 3:

(1) Print out new page and column headers. Set #TOT-COUNT equal to zero.

(2) Read one record from work file-one: #BLDG-ROOM, #CAGE, #UIN, #STAT, #ESTABLISH-DATE, #LIT-GENERATION, #ALT-UIN, #ALT-CAGE, and #ALT-LIT-GENERATION. This read initiates a loop that processes one record per iteration. For each record, do the following (3 - 10):

(3) Each time the BLDG-ROOM changes, write out a totals line for that room. Go to a new page and print page and column headers for the next room. Set #TOT-COUNT to zero.

(4) Each time the #CAGE changes, skip a line in the report and set a flag to print out the new cage (otherwise, the cage should be suppressed).

(5) Add one to #TOT-COUNT for each record.

(6) If #STAT is equal to "M" or "F", print #CAGE (if not suppressed), #UIN, #STAT, #LIT-GENERATION, and #ESTABLISH-DATE.

(7) If #STAT is equal to "P", do the following (8 - 10):

(8) Print #CAGE (if not suppressed), #UIN, "F", #LIT-GENERATION, and #ESTABLISH-DATE.

(9) If #ALT-CAGE is equal to #CAGE, set #ALT-CAGE to blanks.

(10) Print #ALT-CAGE, #ALT-UIN, "M", #ALT-LIT-GENERATION, and #ESTABLISH-DATE.

Process 4:

(1) Delete work file-one.

Module Name: BS009

Descriptive Title: Average Litter Size Report

Method of Invocation: Autosubmit Weekly or by User Select

Inputs: #START-DATE .. beginning of range for birth dates
#END-DATE .... end of range for birth dates
Files ..... LITTERS

Outputs: BS009

Called Modules: None

Function: To average litter size for each litter number within each NCTR strain.

Assumptions:

None made.

Process Summary:

BS009 is composed of four separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work file.

Process 1:

(1) Input the #START-DATE and #END-DATE.

(2) Find all litters in the LITTERS file that have the date of birth between the #START-DATE and #END-DATE of the date range (inclusive).

(3) For each such litter, write the #STRAIN, #LITTER-NUMBER, and #NUMBER-BORN to work file-one.

Process 2:

(1) Sort work file-one by #STRAIN and #LITTER-NUMBER.

Process 3:

(1) Write out page and column headers.

(2) Read from work file-one: #STRAIN, #LITTER-NUMBER, and #NUMBER-BORN.

(3) At break of #STRAIN, skip a line on the report.

(4) At break of #STRAIN*LITTER-NUMBER, calculate the average litter size.

(5) Write out the #STRAIN, #LITTER-NUMBER, total born to that strain and litter number, total number of litters in that strain and litter number, and the average litter size. STRAIN should be suppressed except at the break of page and for the first line written for each strain. Clear the totals after printing.

(6) For each record, add together the number born and increment the litter count.

Process 4:

(1) Delete work file-one.

*** NOTE *** This is a totally rewritten report.

Module Name: BS010

Descriptive Title: Unproductive Breeder Report

Method of Invocation: Autosubmit Weekly or by User Select

Inputs: #TODAY .... the Target Date
Files ..... ANIMAL
DATE-ESTABLISHED
LITTER

Outputs: BS010
BS010s (summary)

Called Modules: None

Function: To identify all breeder pairs and single breeders that are determined to be unproductive based on the strain-specific litter inactivity interval.

Assumptions:

1. Any animal in breeder status is included in this report, whether or not the animal is established as part of a breeder pair. **NCTR APPROVED**

2. All single breeders are considered to be unproductive by default, and therefore are included in this report. **NCTR APPROVED**

3. Total Strain is considered to be the strain of the progeny of the breeder or breeder pair. If both dam and sire are of the same strain, or if there is a single breeder, then the progeny will have the same strain. If the strain of the dam and sire differ, they will be listed once under the (new) strain of the progeny. **NCTR APPROVED**

4. The number in the Total Strain entry is the number of the breeder pairs or singles relating to the strain. A pair counts as one. A single breeder also counts as one.

5. Establish date is the date a pair is established if there is a pair. If there is a single breeder, the establish date is the date the animal was established as a breeder.

6. The cage is the cage of the breeder pair or single breeder as of #TODAY. If the cage of the sire and dam differ, the report will be sorted by the dam's cage, and the sire's cage will be

noted in a new column for reference. A new column header will be required for the sire's cage.

Process Summary:

BS010 is composed of five separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 re-sorts the data to allow for generation of the strain summary. Process 5 prepares the strain summary page. Process 6 deletes the work file.

Process 1:

(1) At start of data, assign blanks to #STRAIN. Reset the first two characters of #UIN to #CURRENT-STRAIN.

(2) Find all ANIMAL records where the animal has ever been in breeder status. This find initiates a loop that processes one animal per iteration. For each animal, do the following (3 - 27):

(3) Search through the animal's status group from the bottom toward the top to (a) find the #CAGE and #BLDG-ROOM of the animal as of #TODAY, and (b) find if the animal was in breeder status as of #TODAY. If animal was in breeder status, do the following (4 - 27):

(4) Find the animal's entry in DATE-ESTABLISHED where the establish date is less than or equal to #TODAY and the dissolve date is blank or after #TODAY. If an entry is found, do the following (5 - 21):

(5) Set #STAT to "P" to indicate a pair.

(6) If the animal is male, do nothing. This animal will be processed when the corresponding female is.

(7) If animal is female, do the following (8 - 21):

(8) Store the dam's UIN in #UIN. Store the corresponding sire's UIN in #ALT-UIN.

(9) If #STRAIN differs from #CURRENT-STRAIN, then do the following (10 - 11):

(10) If #UIN is equal to #ALT-UIN, then store #CURRENT-STRAIN in #STRAIN. Else consult the Strain Table to find the (new) strain of the progeny and store this value in #STRAIN.

(11) Find the number of days of inactivity for this strain, which indicates non-productivity. Store in #INACTIVE.

(12) Store the date the pair was established in #ESTABLISH-DATE. Calculate the number of days the pair has been established by subtracting the establish date from #TODAY. Store in #DAYS-EST.

(13) Find all LITTER records with a dam UIN equal to the dam UIN from the DATE-ESTABLISHED record, sorted by litter number (descending order).

(14) Set the following to zeros: #LAST-DOB, #LITTER-NUM, #PREV-DOB, and #DAYS-SINCE-LITTER.

(15) If this is the first pass through the find loop, then do the following (16 - 17):

(16) Store #LAST-DOB (date of birth) and #LITTER-NUM.

(17) Calculate the number of days since the last litter and store in #DAYS-SINCE-LITTER.

(18) If this is the second pass through the loop, then store #PREVIOUS-DOB.

(19) If the #DAYS-SINCE-LITTER is less than or equal to #INACTIVE, then ESCAPE this loop (ignore this pair).

(20) Find the first ANIMAL record with a UIN equal to #ALT-UIN (the sire's UIN). Do the following for this animal (21):

(21) Read through the status group from the bottom to top to determine the sire's #ALT-CAGE (selecting the values from the first element in which the status change date is less than or equal to the target date).

(22) If no entry is found in DATE-ESTABLISHED (see step 4), then the animal is a single breeder and, by default, unproductive. Do the following for single breeders (23 - 26):

(23) Store #UIN from step (1).

(24) Set #STAT to "F" for female, "M" for male.

(25) Store #ESTABLISH-DATE from step (2). Calculate #DAYS-EST by subtracting #ESTABLISH-DATE from #TODAY.

(26) Store blanks in #LAST-DOB, #LITTER-NUM, #DAYS-SINCE-LITTER, #PREVIOUS-DOB, and #ALT-CAGE.

(27) Write #BLDG-ROOM, #STRAIN, #CAGE, #UIN, #ALT-UIN, #ALT-CAGE, #DAYS-EST, #ESTABLISH-DATE, #LAST-DOB, #LITTER-NUM, #DAYS-SINCE-LITTER, #PREVIOUS-DOB, and #STAT to work file-one.

Process 2:

(1) Sort work file-one data by #BLDG-ROOM, #STRAIN, and #CAGE.

Process 3:

(1) Read from work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #UIN, #ALT-UIN, #ALT-CAGE, #DAYS-EST, #ESTABLISH-DATE, #LAST-DOB, #LITTER-NUM, #DAYS-SINCE-LITTER, #PREVIOUS-DOB, and #STAT.

(2) At break of building-room, write the total number of unproductive pairs, skip a line, and reset the number of unproductive pairs counters.

(3) At break of strain, write out a strain total, skip a line, and reset the strain within room counter.

(4) For each record read, accumulate strain and building-room totals and write out the cage detail.

Process 4:

(1) Sort work file-one by #STRAIN.

Process 5:

(1) Read work file-one and select the #STRAIN.

(2) At break of #STRAIN, write the strain and strain total.

(3) For each record read, increment the strain total and grand total.

(4) Write the grand total.

Process 6:

(1) Delete work file-one.

Module Name: BS011

Descriptive Title: Retirement Candidates

Method of Invocation: Autosubmit Weekly

Inputs: Files ..... DATE-ESTABLISHED

Outputs: BS011
BS011s (summary)

Called Modules: None

Function: This report lists all breeders that are eligible for retirement. Eligibility is established once a breeder has been on breeder status for a period of time predetermined for that species and strain. Once a breeder becomes a candidate, it will appear on subsequent issues of this report until it is actually retired.

Assumptions:

None made.

Process Summary:

BS011 is composed of six separate parts. Process 1 obtains the required data fields. Process 2 sorts the work file for BS011. Process 3 writes hardcopy detail for BS011. Process 4 sorts summary data. Process 5 generates the summary report, BS011s. Process 6 deletes the work files.

Process 1:

(1) Find the records in the DATE-ESTABLISHED file where the dissolve date is blank. This find initiates a loop that processes one DATE-ESTABLISHED record (with dissolve date blank) for each iteration. If the dissolve date is blank, do the following (2 - 6):

(2) Look in the Strain Table for the number of days until retirement for this strain.

(3) Calculate the number of days the current pair has been established. If this number is equal to or greater than the number of days until retirement (for the strain), do the following (4 - 6):

(4) Locate the building-room and cage using the dam UIN.

(5) Calculate the age of the female using her birthday and the current date.

(6) Write the following to work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #DAM-UIN, #SIRE-UIN, #ESTABLISH-DATE, number of days established, birth date of dam, and age of dam.

Process 2:

(1) Sort work file-one by #BLDG-ROOM, #STRAIN, and #CAGE.

Process 3:

(1) Read the following from work file-one: #BLDG-ROOM, #STRAIN, #DAM-UIN, #SIRE-UIN, #ESTABLISH-DATE, #DAYS-ESTABLISHED, #DAM-BIRTH-DATE, #DAM-AGE. This read initiates a loop that processes one line from the work file per iteration. For each line, do the following (2 - 7):

(2) Count the candidate pairs per #STRAIN.

(3) Count the candidate pairs per #BLDG-ROOM.

(4) Print #BLDG-ROOM, #CAGE, #DAM-UIN, #SIRE-UIN, and #ESTABLISH-DATE.

(5) Whenever the #STRAIN changes, do the following (6 - 8):

(6) Write the previous strain and the total for that strain to work file-two.

(7) Write total for #STRAIN.

(8) Write total for #BLDG-ROOM.

Process 4:

(1) Sort work file-two by #STRAIN.

Process 5:

(1) Read a record from work file-two, selecting #STRAIN and #TOTAL.

(2) Whenever strain changes, do the following (3 - 5):

(3) Write out previous strain and #STRAIN-TOTAL for that strain.

(4) Increment #GRAND-TOTAL by #STRAIN-TOTAL.

(5) Set #STRAIN-TOTAL to zero.

(6) For each record, increment #STRAIN-TOTAL by #TOTAL.

(7) At end of data, write out the #GRAND-TOTAL.

Process 6:

(1) Delete work file-one and work file-two.

Module Name: BS012

Descriptive Title: Daily Allocations

Method of Invocation: Autosubmit Daily

Inputs: #TODAY ..... Target Date
Files ..... ALLOCATE
ANIMAL

Outputs: BS012
BS012s (summary)

Called Modules: None

Function: To report all animals allocated on the specified date.

Assumptions:

None made.

Process Summary:

BS012 is composed of four separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work file.

Process 1:

(1) Input the target date (usually the date the report is run).

(2) Find all ALLOCATION records where the allocation date is equal to the target date.

(3) Obtain the #OPERATOR, #UIN, #EXPERIMENT-NUM, #BLDG-ROOM, #RACK, #RACK-SIDE, #CAGE, #WEIGHT, and #WEIGHT-UNITS from the allocation record.

(4) Find the ANIMAL record with a UIN equal to the UIN from the allocate record. Obtain the #BREEDING-ID from this record.

(5) Read through the status group to determine the #ISSUE-CAGE-NUM to which the animal was assigned prior to allocation.

(6) Write out to work file-one: #BLDG-ROOM, #RACK, #RACK-SIDE, #CAGE, #UIN, #BREEDING-ID, #OPERATOR, #EXPERIMENT-NUM, #WEIGHT, #WEIGHT-UNITS, and #ISSUE-CAGE-NUM.

Process 2:

(1) Sort work file-one by #BLDG-ROOM, #RACK, #RACK-SIDE, and #CAGE.

Process 3:

(1) Read the following from work file-one: #BLDG-ROOM, #RACK, #RACK-SIDE, #CAGE, #UIN, #BREEDING-ID, #OPERATOR, #EXPERIMENT-NUM, #WEIGHT, #WEIGHT-UNITS, and #ISSUE-CAGE-NUM.

(2) At break of experiment, write the allocation totals for that experiment to a new page and clear the experiment totals.

(3) At break of building-room, rack, or rack side, force a page break.

(4) For each individual record:

(5) Write the cage detail, but for animals allocated to the same cage, write the cage number only on the first line of output for that cage.

(6) Collect totals for males, females, and unsexed allocated.

Process 4:

(1) Delete work file-one.

*** NOTE *** This is a totally rewritten report.

Module Name: BS013

Descriptive Title: Female Breeder History

Method of Invocation: Autosubmit Weekly (possible ad hoc)

Inputs: #START-DATE ..... beginning of the range
#END-DATE ..... end of the range
Files ..... ANIMAL
DATE-ESTABLISHED
LITTERS

Outputs: BS013

Called Modules: None

Function: This report provides the litter history of female breeders.

Assumptions:

1. The current cage is assumed to be the cage in which the female breeder was located on the #END-DATE. The cage of the male is not noted, even if different from the female.

Process Summary:

BS013 is composed of four separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work files.

Process 1:

(1) Input the #START-DATE and #END-DATE.

(2) Find all ANIMAL records where the animal is female, AND has been in breeder status, AND whose removal date is blank or after the #END-DATE. For each animal, do the following (3 - 20):

(3) Search through the status group from the bottom toward the top to find whether the animal was put in breeder status on or before the #END-DATE. If so, do the following (4 - 20):

(4) Find the #CAGE, #BLDG-ROOM, and #STATUS of the female as of the #END-DATE. (May still be breeder, or may have changed statuses.)

(5) Store the #DAM-UIN, and #DAM-DOB (date of birth).

(6) Find the first LITTER record where the litter UIN is equal to the dam's litter UIN (found in ANIMAL). Store the #DAM-GENERATION.

(7) Find the DATE-ESTABLISHED record(s) corresponding to this female where the establish date is on or before the #END-DATE. If such record(s) exist, do the following (8 - 18):

(8) Set #NUM-OF-LITTERS to zero.

(9) Get the #SIRE-UIN.

(10) Find LITTER records where the litter dam UIN is equal to the #DAM-UIN and the litter sire UIN is equal to the #SIRE-UIN. For each litter, do the following (11 - 17):

(11) Add one to #NUM-OF-LITTERS.

(12) Store #LITTER-UIN, #LITTER-BIRTHDATE, #NUM-IN-LITTER, and #LITTER-NUM.

(13) Find the ANIMAL records where the animal litter UIN is equal to #LITTER-UIN. For each animal, do the following (14 - 16):

(14) Read through the status group to find the first status that has a status change date less than the #END-DATE to determine the status of that animal as of the end of the date range. Use that date as the actual #DISPOSITION-DATE. Store the status in #DISPOSITION.

(15) Store the animal UIN in #LITTER-ANIMAL-UIN. Store the sex in #LITTER-ANIMAL-SEX.

(16) Write to work file-two: #DAM-UIN, #LITTER-UIN, #LITTER-BIRTHDATE, #NUM-IN-LITTER, #LITTER-NUM, #LITTER-ANIMAL-UIN, #LITTER-ANIMAL-SEX, #DISPOSITION, and #DISPOSITION-DATE.

(17) If no ANIMAL records exist, then animals have not yet been weaned. Write to work file-two:

#DAM-UIN, #LITTER-UIN, #LITTER-BIRTHDATE, #NUM-IN-LITTER, #LITTER-NUM, blanks for litter animal uin, blanks for litter animal sex, blanks for disposition, and blanks for disposition date.

(18) If no LITTER record exists, then this dam-sire pair has had no litters. #NUM-OF-LITTERS will still be zero as a result of step (8) above.

(19) If no DATE-ESTABLISHED record exists, then at the time of the #END-DATE this was a single female breeder. Set #SIRE-UIN to blanks. Set #NUM-OF-LITTERS to blanks.

(20) Write to work file-one: #DAM-UIN, #SIRE-UIN, #DAM-DOB, #CAGE, #BLDG-ROOM, #STATUS, #GENERATION, and #NUM-OF-LITTERS

Process 2:

(1) Sort the work file-one by DAM-UIN, then SIRE-UIN.

(2) Sort work file-two by #DAM-UIN, #LITTER-BIRTHDATE, and #LITTER-ANIMAL-UIN.

Process 3:

(1) Read from work file-one: #DAM-UIN, #SIRE-UIN, #DAM-DOB, #CAGE, #BLDG-ROOM, #STATUS, #GENERATION, and #NUMBER-OF-LITTERS.

(2) At break of #DAM-UIN*#SIRE-UIN, go to new page and write page and column headers.

(3) Write to report BS013: #DAM-UIN, #DAM-DOB, #BLDG-ROOM, #CAGE, #SIRE-UIN, #STATUS, #GENERATION, and #NUMBER-OF-LITTERS.

(4) If #NUMBER-OF-LITTERS is greater than zero, read from work file-two those records with the same dam UIN as found in step (1) above. Read the following: #2-DAM-UIN, #LITTER-UIN, #LITTER-BIRTHDATE, #NUM-IN-LITTER, #LITTER-NUM, #LITTER-ANIMAL-UIN, #LITTER-ANIMAL-SEX, #DISPOSITION, and #DISPOSITION-DATE. For each record, do the following (5 - 7):

(5) At break of #LITTER-UIN, print totals from last litter if last #LITTER-ANIMAL-UIN was not blank (blank indicates not weaned). Set #SUPPRESS to zero for false

so that new litter information will not be suppressed.

(6) If new #LITTER-ANIMAL-UIN is blank, print out #LITTER-UIN, #LITTER-BIRTHDATE, #LITTER-SIZE, and #LITTER-NUMBER.

(7) Else (i.e., #LITTER-ANIMAL-UIN is not blank) print out #LITTER-UIN, #LITTER-BIRTHDATE, #LITTER-SIZE, #LITTER-NUMBER, #LITTER-ANIMAL-UIN, #LITTER-ANIMAL-SEX, #DISPOSITION-DATE, and an "X" in the appropriate column to align with one of the accepted dispositions.

Process 4:

(1) Delete work file-one and work file-two.

*** NOTE *** This is a totally rewritten report.

Module Name: BS014

Descriptive Title: Productivity Index

Method of Invocation: User Selection

Inputs:
#START-DATE .. beginning of date range for which productivity is measured
#END-DATE .... end of date range for which productivity is measured
Files ..... ANIMAL
DATE-ESTABLISHED
LITTERS

Outputs: BS014

Called Modules: None

Function: To determine the productivity index (number weaned per week) for each of the strains maintained.

Assumptions:

1. The productivity index is normalized to a weekly value if the date range is for any period other than a single week. **NCTR APPROVED**

2. In the Total Breeders column of the report, a breeder pair counts as one. A single breeder also counts as one. Single breeders are unproductive by default.

3. NCTR strain code is considered to be the strain of the progeny of the breeder or breeder pair. If both dam and sire are of the same strain, or if there is a single breeder, then the progeny will have the same strain. If the strain of the dam and sire differ, they will be listed once under the (new) strain of the progeny. **NCTR APPROVED**

Process Summary:

BS014 is composed of four separate parts. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work file.

Process 1:

(1) Input the #START-DATE and #END-DATE.

(2) Determine the number of days in the date range and compute the #PRODUCTIVITY-FACTOR (7/number of days). Write this value to work file-one.

(3) Find all ANIMALs who were breeders at any time during the given date range.

(4) For each animal, find the record in DATE-ESTABLISHED. If a record exists, do the following (5 - 9):

(5) If the animal is male, do nothing. This animal will be processed with the corresponding female.

(6) If the animal is female, do the following (7 - 9):

(7) If the dam and sire strain are the same, then store this strain in #STRAIN. If they differ, then find the strain of the progeny from the Strain Table and store in #STRAIN.

(8) Set #NUM-WEANED to zero.

(9) Find the LITTER records corresponding to this dam-sire pair that have a wean date within the given date range. For each litter record, subtract number born dead from number born, and add the resulting value to #NUM-WEANED.

(10) If no DATE-ESTABLISHED record exists, then this is a single breeder and is unproductive by default. Store the strain of this animal in #STRAIN, and store zero in #NUM-WEANED.

(11) Write out the #STRAIN and #NUM-WEANED to work file-one.

Process 2:

(1) Sort work file-one by #STRAIN.

Process 3:

(1) Read the #PRODUCTIVITY-FACTOR from work file-one. This should be a single read and not the beginning of a loop, since this value is only listed once.

(2) Read #STRAIN and #NUM-WEANED from work file-one. This read begins a loop that processes one record per iteration.

(3) At break of #STRAIN do the following (4 - 5):

(4) Calculate the #PRODUCTIVITY-INDEX as the number weaned for this strain divided by #TOTAL-BREEDERS (see below) times the #PRODUCTIVITY-FACTOR.

(5) Print the strain summary and reset the strain totals and #TOTAL-WEANED.

(6) For each record do the following (7 - 8):

(7) Increment the #TOTAL-BREEDERS counter. One pair or one single breeder counts as an entry, one entry per record.

(8) Add #NUM-WEANED to #TOTAL-WEANED.

Process 4:

(1) Delete work file-one.

Module Name: BS015

Descriptive Title: Feeder Status Report

Method of Invocation: Autosubmit Weekly

Inputs: Files ..... ANIMAL

Outputs: BS015

Called Modules: None

Function: This report provides the projected mating date by sex and strain for feeders in a certain building-room. The total animals of each sex and strain is then provided for each building-room.

Assumptions:

None made.

Process Summary:

BS015 is composed of four separate processes. Process 1 obtains the required data fields. Process 2 sorts the data to allow for proper generation of report output. Process 3 provides the hardcopy report detail. Process 4 deletes the work file.

Process 1:

(1) Find the ANIMAL records with a current status of Feeder and that have no removal date. This find initiates a loop that processes one animal record per iteration. For each animal record, do the following (2 - 17):

(2) At the start of data, save the litter UIN for the first animal in #OLD-LITTER-UIN.

(3) Go through the current animal's status group. For each entry in the animal status group, do the following (4 - 17):

(4) If the Animal status is Feeder and the status change date is less than or equal to the current date, do the following (5 - 17):

(5) If the #OLD-LITTER-UIN is not equal to the current litter UIN, do the following (6 - 7):

(6) Calculate the projected mating date from the current animal's date of birth and the Strain Table for that animal. Write the following to work file-one: #OLD-BLDG-ROOM, #OLD-STRAIN, #OLD-CAGE, #OLD-SEX, #OLD-UIN, #COUNT, #OLD-DATE-OF-BIRTH, and #PROJECTED-MATING-DATE.

(7) Reset the #COUNT and the #PROJECTED-MATING-DATE.

(8) Otherwise (i.e., step 5 not true), do the following (9 - 16):

(9) Add one to #COUNT. If #COUNT is one, this is the first animal in a block of animals with the same LITTER-UIN or an individual alone in a cage.

(10) If #COUNT is one, do the following (11 - 16):

(11) Store the current building-room in #OLD-BLDG-ROOM.

(12) Store the current strain in #OLD-STRAIN.

(13) Store the cage in #OLD-CAGE.

(14) Store the sex in #OLD-SEX.

(15) Store the UIN in #OLD-UIN.

(16) Store the date of birth in #OLD-DATE-OF-BIRTH.

(17) Once an animal with the correct status and removal date has been found, break out of this loop.

Process 2:

(1) Sort work file-one by #BLDG-ROOM, #STRAIN, and #CAGE.

Process 3:

(1) Read from work file-one: #BLDG-ROOM, #STRAIN, #CAGE, #SEX, #UIN, #COUNT, #BIRTH-DATE, and #PROJECTED-MATING-DATE. This read initiates a loop that processes one line from the work file per iteration. For each line, do the following (2 - 5):

(2) If SEX is male, write the detail to the report for the male block. Add one to the total males in this building-room.

(3) If SEX is female, write the detail to the report for the female block. Add one to the total females in this building-room.

(4) If SEX is unsexed, write the detail to the report for the unsexed block. Add one to the total unsexed in this building-room.

(5) Whenever the building-room or strain changes, write the totals for male, female, and unsexed in this building-room.

Process 4:

(1) Delete work file-one.

Module Name: BS016

Descriptive Title: Transfer/Issue Cage Report

Method of Invocation: User Selection

Inputs: Allow for date range???
Files ..... ANIMAL

Outputs: BS016

Called Modules: None

Function: This report provides the number of animals (by strain of each sex) that were transferred or designated as issue animals.

Assumptions:

None made.

Process Summary:

BS016 consists of four separate processes. Process 1 obtains the data fields required. Process 2 sorts the data. Process 3 provides hardcopy detail for the report. Process 4 deletes the work file.

Process 1:

(1) Find the ANIMALs with no removal date that have a status of issue feeder, issue breeder, issue experiment, or issue other, sorted in order by the LITTER-UIN. This find initiates a loop that processes one animal record per iteration. For each animal record, do the following (2 - 9):

(2) At the start of data, save the LITTER-UIN in #OLD-LITTER-UIN.

(3) Go through the entries in the current animal status group. For each entry, do the following (4 - 9):

(4) If the status is issue feeder, issue breeder, issue experiment, or issue other, AND the status change date is less than or equal to the given date, do the following (5 - 8):

(5) If #OLD-LITTER-UIN is not equal to the current litter-UIN, write the following to work file-one: #OLD-STRAIN, #OLD-CAGE, #OLD-UIN, #COUNT, and #OLD-SEX. Add one to #COUNT.

(6) Else (i.e., #OLD-LITTER-UIN equals the current litter-UIN) do the following (7 - 8):

(7) Add one to #COUNT

(8) If #COUNT is one, store current information in #OLD-STRAIN, #OLD-CAGE, #OLD-UIN, and #OLD-SEX.

(9) Once the correct entry in the animal status group is found, leave the loop that began in step (3).

Process 2:

(1) Sort work file-one by #STRAIN, #CAGE, and #SEX.

Process 3:

(1) Read from work file-one: #STRAIN, #CAGE, #UIN, #COUNT, and #SEX. This read initiates a loop that processes one line from work file-one per iteration. For each line of data, do the following (2 - 6):

(2) If #SEX is male, add one to the total males.

(3) If #SEX is female, add one to the total females.

(4) If #SEX is unsexed, add one to the total unsexed.

(5) Print #STRAIN, #CAGE, #UIN/#COUNT, and #SEX detail.

(6) Whenever STRAIN changes, print out total males, total females, and total unsexed.

Process 4:

(1) Delete work file-one.

Module Name: BS017

Descriptive Title: Animal Orders

Method of Invocation: Autosubmit Weekly, Month, User Selection

Inputs: #START-DATE .. The beginning of the date range for which the report is run
#END-DATE .... The end of the date range for which the report is run
Files ..... ORDER

Outputs: BS017
BS017s

Called Modules: None

Function: This report provides a list of the animals ordered for experiments. Each order is identified by operator, experiment, Principal Investigator, and the dates the orders were requested and filled. Information required for the animals includes the number ordered by strain and sex, as well as their age and phenotype.

Assumptions:

1. Operator comes from session data on ORDER file.

2. Principal Investigator comes from Study Definition and can be retrieved by experiment number.

Process Summary:

BS017 consists of two separate processes. Process 1 obtains the data and provides the hardcopy detail for report BS017. Process 2 obtains the data and provides the hardcopy detail for report BS017s.

Process 1:

(1) Find records in the ORDER file that have the date the order was placed as greater than the #START-DATE and less than the #END-DATE. These records should be sorted by the Operator and the experiment number (#EXP-NUM). This find initiates a loop that processes one ORDER record per iteration. For each record, do the following (2 - 5):

(2) Find the Principal Investigator (#PI) in Study Definition by the experiment number.

(3) At the top of each new report page, write the title.

(4) If the current experiment number is the first occurrence of that number, it should be written in step (5). Do not repeat the same experiment number.

(5) Write to the report: #EXP-NUM, #BLDG-ROOM, #PI, date required, strain, sex ordered, number requested, number delivered, date filled, and special requirements.

Process 2:

(1) Find records in the ORDER file that have the date the order was placed as greater than the #START-DATE and less than the #END-DATE. These records should be sorted by the experiment number (#EXP-NUM), strain, and sex. This find initiates a loop that processes one Order record per iteration. For each record, do the following (2 - 7):

(2) Add the current number delivered to an accumulator, #NUM-DELIVERED.

(3) Add the current number requested to an accumulator, #NUM-REQUESTED.

(4) Whenever the experiment number, sex, or strain changes, do the following (5 - 7):

(5) Determine when to print the experiment number.

(6) Print strain, sex, and totals.

(7) Reset #NUM-REQUESTED and #NUM-DELIVERED.

Module Name: BS018

Descriptive Title: Target Trace Report

Method of Invocation: User Selected

Inputs: one UIN
Files ..... ANIMAL
LITTERS

Outputs: BS018

Called Modules: Not yet resolved

Function: To output the family tree for one or several UINs.

Assumptions:

1. Only female UINs are output in the report. **NCTR APPROVED**

2. Animals who are not living and who are located at the end of a branch in the tree are to be printed with no cage number. **NCTR APPROVED**

3. An animal's current status will be added to the report next to the animal's cage number. **NCTR APPROVED**

Process:

(1) Input the UIN.

(2) Find the ANIMAL record with the corresponding UIN and do the following:

(3) Find the LITTER record with the LITTER UIN from the animal record. Obtain the generation from this LITTER record. Copy the ANIMAL-UIN to the SEED-UIN.

(4) For I = 1 to the number of generations:

(5) Find the ANIMAL record with UIN equal to the SEED-UIN.

(6) Find the LITTER record with the LITTER UIN from step (5). From this record, obtain the DAM-UIN. Write the DAM-UIN to a work file with the GENERATION number from the same record.

(7) /* the previous four steps, find the chain of female ancestors for the given UIN */

(8) The method to be used to develop the descendant's tree has not yet been fully determined, but the important elements are outlined below:

The size of the box needed to contain the family tree for an individual UIN is fully determined if the number of living (female) descendants and the number of generations to the most remote descendant are both known. The number of rows in the descendant box corresponds to the number of living (female) descendants, while the number of columns corresponds to the number of generations.

For each animal born to an individual, the same values can be calculated, determining the size of the box needed to contain its portion of the full descendant box. The sum of the number of rows required in generation N+1 will equal the total number of rows required for generation N, so the actual row location of each direct descendant can be determined. The number of columns required for each descendant in generation N+1 may differ, but at least one will contain just one less column than the parent box required (others may require even fewer). The right side of each descendant box will never end further to the right than the parent box.

Because the size of each box can be determined, the entire UIN list can be formatted.

The process of determining the list of UINs in each generation and the corresponding box sizes has not yet been fully resolved. This program would lend itself to a recursive solution, but NATURAL does not allow direct recursion. Programs could be written to take a UIN list and obtain all descendants. The resulting UIN list could be saved on ADABAS WORK using a named ISN list, and the program could fetch itself with that list as a parameter. The process would eventually terminate, because eventually the last generation in the strain would be evaluated. The result would be a series of named ISN lists corresponding to each descendant generation. By counting the number of ISN lists and living (female) individuals for each ISN in each list, the full box parameters can be derived.

However, the methods explored to date require unusual or difficult processing and storage of intermediate parameters. Further investigation will be conducted to provide a program that lends itself to simpler coding and maintenance.

(9) Example, descendant boxes only:

```
UIN101d -- UIN201a -- UIN301a -- UIN401a
           |           |          UIN402a
           |           |          UIN403a
           |           |
           |           -- UIN302a
           |
           -- UIN202d -- UIN303a -- UIN404a
```

UIN101 (4 generations, 8 living)
UIN201 (3 generations, 6 living)
UIN202 (3 generations, 2 living)
UIN301 (2 generations, 4 living)
UIN302 (1 generation, 1 living)
UIN303 (2 generations, 2 living)
UIN401 (1 generation, 1 living)
UIN402 (1 generation, 1 living)
UIN403 (1 generation, 1 living)
UIN404 (1 generation, 1 living)

UIN101 box must be 4 columns by 8 rows, etc.

```
-----------------------------------------------------------------
            col1      col2      col3      col4

row 1      UIN101....UIN201                          cage xxxx
             .         .
row 2        .         ........UIN301                cage xxxx
             .         .         .
row 3        .         .         ........UIN401      cage xxxx
             .         .         .
row 4        .         .         ........UIN402      cage xxxx
             .         .         .
row 5        .         .         ........UIN403      cage xxxx
             .         .
row 6        .         ........UIN302                cage xxxx
             .
row 7        .......UIN202.....UIN303                cage xxxx
                                .
row 8                           ........UIN404      cage xxxx
-----------------------------------------------------------------
```

Note that if an animal has no descendants and is dead, then its box size is one column by zero rows. For example, UIN302 might be dead. If so, it could be placed in any of the rows 3-5. Depending on the width of the columns, this could lead to confusion. It is suggested that no box ever be allowed to have zero rows. Terminal UINs that are not living would merely have no cage number listed.

Module Name: BS019

Descriptive Title: UIN Summary Report (BMSSLOOK)

Method of Invocation: Not Yet Specified

Inputs: UIN Selection Criteria
Files ..... ANIMAL
ALLOCATION
DATE-ESTABLISHED
LITTERS

Outputs: BS019

Called Modules: None

Function: To output all information for a particular UIN.

Assumptions:

1. The BREEDING segment offsets previously printed are no longer applicable and are omitted.

2. The output format of this report will require some revision to take the new data structure into account.

Process:

(1) Find all UINs in the ANIMAL file that match the input criteria. For each such UIN, do the following (2 - 23):

(2) Determine the number of status elements for this UIN.

(3) Find all DATE-ESTABLISHED records with this UIN (in either the dam UIN or sire UIN, depending on animal sex), thereby obtaining the number of pair counts. For each record, do the following (4):

(4) Count the number of pair observations made in each of these records and sum them up.

(5) Determine whether the animal has been removed.

(6) Determine the number of ALLOCATION records with this UIN.

(7) Determine the number of LITTERS with this UIN (in either the dam UIN or sire UIN, depending on animal sex).

(8) Write out the header information for this UIN, including the UIN, STRAIN, STATUS COUNTS, OBSERVATION COUNTS, REMOVAL COUNT (zero or one), ALLOCATION COUNTS, and MATE COUNTS.

(9) Obtain this animal's litter record, and from it determine the mother and father UIN and STRAINs. Also determine the GENERATION and BIRTHDATE.

(10) From the ANIMAL record, obtain the CARCASS-ID, REMOVAL DATE, REASON, and DESTINATION (if any). Also obtain the SEX. For each record, do the following (11 - 12):

(11) Read through the animal status group and obtain the last activity date and current STATUS.

(12) Obtain the VENDOR'S ID (if any).

(13) Write the data obtained in steps (9 - 12) as the DATA HEADER information.

(14) Re-read the status group and do the following (15 - 16):

(15) Obtain the OPERATOR, STATUS-CHANGE-DATE, STATUS, CAGE, and BUILDING-ROOM.

(16) Write out the information, including the PREVIOUS CAGE and STATUS, and retain the current CAGE and STATUS for the next PREVIOUS CAGE and STATUS.

(17) Find each DATE-ESTABLISHED record with this UIN (in dam or sire, depending on sex) and do the following (18 - 19):

(18) Write out the pair information (mate UIN and sex, establish date, and date dissolved, if any).

(19) Read through the observations group and write out the individual observations and corresponding dates for this pair.

(20) Find all LITTER records with this UIN (in dam or sire UIN, depending on sex) and do the following (21):

(21) Write out the litter information for each litter.

(22) Find all ALLOCATE records with this UIN and do the following (23):

(23) Write out the allocation fields for this record.

Module Name: EBS001

Descriptive Title: Pup Weights at Weaning and Disposition

Method of Invocation: User Selection

Inputs: #EXP-NUM .. Experiment Number (entered by the user)
Files ..... ALLOCATION
ANIMAL
LITTER

Outputs: EBS001

Called Modules: None

Function: This report provides the weight and disposition (surplus, allocation, breeder, etc.) at the time of weaning of pups born to parents that have undergone a specific experimental treatment. The report tracks the parents and specific litter information (UIN, generation, number), as well as the weight and disposition of the pup at time of weaning.

Assumptions:

None made.

Process:

(1) Find the records in the ALLOCATION file with the experiment number equal to #EXP-NUM. This find initiates a loop that processes one Allocation record per iteration. For each record, do the following (2 - 9):

(2) Find the first ANIMAL record (i.e., the only animal record) with a UIN number equal to the one found in step (1), and a sex of Female. If an Animal record is found that fits this criteria, do the following (3 - 9):

(3) Determine the female's generation by matching her litter UIN to the LITTER file.

(4) Find litters born to this female since she was allocated to this experiment by finding litter records where (a) the dam UIN matches her UIN and (b) the litter birth date is greater than the female's allocation date. This find initiates a loop that

processes one litter per iteration. For each litter, do the following (5 - 9):

(5) Find the wean weight of the pups in this litter from the pup-weighing PE group in the LITTER file.

(6) Determine the number of pups of each status.

(7) Find the first ANIMAL record created for each pup. The first occurrence in the animal status PE group will be the wean status, and the second status stored will be the status immediately following weaning. The PE group is ordered chronologically from one to n, where n is the first status stored.

(8) Locate the next-to-last occurrence of status.

If the status is surplus, add one to #SURPLUS.

If the status is allocated, add one to #ALLOCATE.

If the status is breeder, add one to #BREEDER.

If the status is survey, add one to #SURVEY.

If the status is issue, add one to #ISSUE.

If the status is dead, add one to #DEAD.

If the status is cull, add one to #CULL.

(9) Print a line to the report for this litter.

Module Name: EBS002

Descriptive Title: Pup Weights

Method of Invocation: User Selection

Inputs: #EXP-NUM ..... Experiment Number (entered by the user)
#PUP-DETAIL-DATE
Files ..... ALLOCATION
ANIMAL
LITTERS

Outputs: EBS002

Called Modules: None

Function: This report provides detailed weight information on the pups born to parents that have undergone a specific experimental treatment. The report shows the parents and specific litter information (UIN, generation, number), as well as the weight and age of the pup on a certain date.

Assumptions:

1. The header "LITTER ANIMAL" appears in this report, but the weighing information appears to relate to an entire litter. The report was therefore coded under the assumption that litter information is being reported. If the intent is to report individual animal weight, then data must be derived from files other than those indicated here.

Process:

(1) Find the records in the ALLOCATION file with the experiment number equal to #EXP-NUM. This find initiates a loop that processes one record per iteration. For each record, do the following (2 -9):

(2) Find the first ANIMAL record (i.e., the only animal record) with a UIN number equal to the one found in step (1), and a sex of female. If an Animal record is found that fits this criteria, do the following (3 - 9):

(3) Determine the female's generation by matching her litter UIN to the LITTER file.

(4) Find litters born to this female since she was allocated to this experiment by finding litter records where (a) the dam UIN matches her UIN and (b) the litter birth date is greater than the female's allocation date. This find initiates a loop that processes one litter per iteration. For each litter, do the following (5 - 6):

(5) Determine the weight of the animal as of the desired date from the pup weighing group; if the animal was not weighed on the desired date, then select the previous weight and retain the date of the weighing. Use the number of pups from the element selected.

(6) Print a line to the report for this litter.

Module Name: EBS003

Descriptive Title: Fertility

Method of Invocation: User Selection

Inputs:

#EXP-NUM	Experiment Number, input by the user
#START-DATE ..	beginning of the date range
#END-DATE	end of the date range
Files	ALLOCATION ANIMAL DATE-ESTABLISHED LITTER INLIFE files

Outputs: EBS003

Called Modules: None

Function: The purpose of this report is to provide information on the fertility of males and females undergoing experimental treatments. The report tracks the males and females by building-room, experiment, treatment, generations and litters produced, and disposition of the offspring for each generation and litter.

Assumptions:

None made.

Process 1:

(1) Find records in the ALLOCATION file that have an experiment number equal to #EXP-NUM. These records should be sorted by operator, building-room, and treatment. This find initiates a loop that processes one allocation record per iteration. For each record, do the following (2 - 17):

(2) Whenever the operator changes, go to a new page of the report and print the titles for that page.

(3) Whenever the experiment changes, do the following (4 - 6):

(4) Determine the summary totals for this treatment. All totals are calculated here to avoid the use of an

array. To calculate the number of female breeders established:

Find the ANIMAL records for the females on this experiment.

Search the status PE group for breeder status within the date range.

If found, increment the female counter.

(5) To calculate the number of male breeders established:

Find the ANIMAL records for males on this experiment.

Search the status PE group for breeder status within the date range.

If found, increment the male counter.

(6) To calculate all other totals, use the same logic as in this report (e.g., find all litters for a female on this experiment and accumulate totals.) Also, retrieve the INLIFE data in the same way as in the remainder of this report.

(7) Find the sex of the animal read from the ALLOCATION file. If the animal is a female, find her generation and the litters born to her during the date range input by the user.

(8) Find the first ANIMAL record (i.e., the only record) that has a UIN equal to the UIN in the current ALLOCATION record, and a sex of female. If such a record exists, do the following (9 - 17):

(9) Save the generation in the current ALLOCATION record in #FEMALE-GENERATION.

(10) Find the litters born to this female after the beginning of the experiment by matching the current female's UIN to dam UINs in the LITTER file, where the litter birth date is later than the experiment start date and before the experiment end date. These files should be sorted by litter UIN and litter number. This find initiates a loop that processes one litter record per iteration. For each litter, do the following (11 - 17):

(11) Read data from INLIFE files to include in born live, born dead, males weaned, females weaned, total weaned, and observe date.

(12) Retrieve the male UIN data using the sire UIN for this litter. Find the sire's ANIMAL record to locate this male's litter UIN.

(13) Find the first LITTER record where the litter UIN matches the male's litter UIN. From this LITTER record, get the male's #GENERATION.

(14) Save the male's UIN in #MALE-UIN.

(15) Find the DATE-ESTABLISHED record with the current female and current male, such that they have not yet retired.

(16) Look through the PE group in the current DATE-ESTABLISHED record for the date littered that is the same as the birth date of the litter. Begin with this occurrence and check the two previous occurrences for pregnant and plugged observations. These observations may or may not be stored. If they are not stored, the previous occurrence will be "littered", if it exists.

(17) Write the current information to one line in report EBS003.

INTERNAL DISTRIBUTION

1.	A. Sjoreen
2-11.	Rod Strand
12.	E. T. Rogers
13.	W. Fulkerson
14.	R. B. Shelton
15.	T. J. Wilbanks
16.	R. B. Honea
17.	D. E. Reichle
18.	Central Research Library
19.	Document Reference Section
20-22.	Laboratory Records
23.	Laboratory Records - RC
24.	F. P. Baxter
25.	G. R. Hadder
26.	Glen Harrison
27.	R. S. Loffman
28.	George T. Privon
29.	Angela K. Sexton
30.	Frances L. Sexton
31.	M. M. Stevens
32.	Patent Office

EXTERNAL DISTRIBUTION

33.	P. Y. Bengtson, 19526 40th Place, NE, Seattle, WA 98155
34-43.	John Appleget, National Center for Toxicological Research, HFT-200, Jefferson, AR 72079
44.	Mr. John J. Cuttica, Vice President of Research & Development, Gas Research Institute, 8600 W. Bryn Mawr Avenue, Chicago, IL 60631
45.	Dr. Joseph P. Kalt, Professor of Economics, Kennedy School of Government, Harvard University, 79 John F. Kennedy Street, Cambridge, MA 02138
46.	Dr. Denton E. Morrison, Professor of Sociology, Michigan State University, 201 Berkey Hall, East Lansing, MI 48824-1111
47.	Dr. Richard L. Perrine, Professor, Engineering and Applied Sciences, Civil Engineering Department, Engineering I, Room 2066, University of California, Los Angeles, CA 90024
48.	Office of Assistant Manager for Energy, Research and Development, DOE-ORO, Oak Ridge, TN 37930
49-58.	OSTI, U. S. Department of Energy, P. O. Box 62, Oak Ridge, TN 37831

☆U.S. GOVERNMENT PRINTING OFFICE:1989 -648-119/ 00037