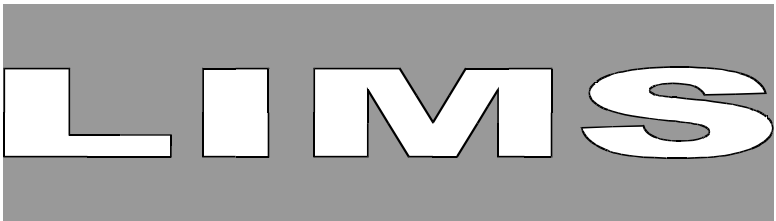


# Livestock Information Management System

Database for Livestock  
Performance Data  
Version 1.2



*By:*  
*Thomas Metz and Michael D. Asfaw*

Computer and Biometrics Section  
ILRI-Ethiopia

Copyright © 1999  
International Livestock Research Institute  
P.O. Box 30709, Nairobi, Kenya

## **Preface**

The current version of the LIMS system was developed by ILCA (now, the International Livestock Research Institute (ILRI)) between January 1990 and June 1992. Its novel design stems from the experience gained with IDEAS (ILCA's Data Entry and Analysis System), an earlier software package for the management of animal performance data.

Thomas Metz is responsible for the system design and Michael Asfaw for the programming. Thomas Metz and Michael Asfaw worked out the details of the system development. Thomas Metz wrote the system documentation.

We thank the KARI-ILCA team at Mtwapa Research Station and the manager of Kilifi Plantations Ltd for starting to use LIMS at an early stage of its development and providing helpful feedback during the development phase. We also thank all ILCA scientists who made valuable comments during several presentations and demonstrations of the LIMS system.

Further information about the distribution and support of LIMS can be obtained from:

Biometrics Unit  
ILRI  
P.O. Box 5689  
Addis Ababa  
Ethiopia

Tel.: (251-1) 61 32 15  
Telex: 21207 ILRI ET  
Fax: (251-1) 61 18 92  
E-mail: [ILRI-ETHIOPIA@cgiar.org](mailto:ILRI-ETHIOPIA@cgiar.org)

Michael D. Asfaw  
2101 Crystal Plaza, Suite 237  
Arlington, VA 22202  
USA  
Fax: (703) 415-0403

Thomas Metz  
In der Galgendell 24  
D-6600 Saarbrücken 1  
Germany  
Tel.: (681) 52934

The International Livestock Centre for Africa (ILCA, now the International Livestock Research Institute (ILRI)), established in 1974, is an autonomous, non-profit-making research training and information centre with a mandate to improve livestock production throughout sub-Saharan Africa. The activities and publications of the centre are financed by the Consultative Group on International Agricultural Research (CGIAR). The CGIAR members that have funded ILCA to date include the African Development Bank, the European Economic Community, the Ford Foundation, the International Development Research Centre, the International Fund for Agricultural Development, the Organization of Petroleum Exporting Countries, the United Nations Development Programme, the World Bank, and the governments of Australia, Austria, Belgium, Canada, China, Denmark, Finland, France, Germany, India, Ireland, Italy, Japan, The Netherlands, Nigeria, Norway, Sweden, Switzerland, the United Kingdom and the United States of America.

© International Livestock Research Institute 1999.

All rights reserved. No part of this publication may be reproduced in any form or by any means without written permission from the copyright owner.

Metz T. and Michael D. Asfaw

LIMS: Database for Livestock Performance Data. Version 1.2. Computer Program with System Documentation 178 pp.

Published by the International Livestock Research Institute, P.O. Box 30709, Nairobi, Kenya.

ISBN 92-9053-257-2

CLIPPER® is a registered trademark of Nantucket Corporation.  
dBASE® and dBASE III® are registered trademarks of Ashton-Tate.  
Hercules® is a registered trademark of Hercules Computer Technology.  
IBM® and AT® are registered trademarks of International Business Machines, Inc.  
MS-DOS® is a registered trademark of Microsoft® Corporation.

# Table of Contents

## Introduction to LIMS

|   |      |
|---|------|
| <b>1 Overview</b> .....                           | 1-1  |
| <b>2 Hardware and Software Requirements</b> ..... | 1-2  |
| <b>3 The LIMS Software</b> .....                  | 1-3  |
| <b>4 Installing LIMS</b> .....                    | 1-6  |
| <b>5 Starting LIMS</b> .....                      | 1-8  |
| <b>6 The LIMS Software Interface</b> .....        | 1-10 |
| 6.1 Display Areas .....                           | 1-10 |
| 6.2 Record Edit Area .....                        | 1-11 |
| 6.3 Function Key Edit Window .....                | 1-13 |
| 6.4 Menus .....                                   | 1-14 |
| 6.5 Selection Lists .....                         | 1-15 |
| 6.6 Lookup Lists .....                            | 1-16 |
| 6.7 Dialogue Boxes .....                          | 1-17 |
| 6.8 Prompts .....                                 | 1-18 |
| 6.9 Messages .....                                | 1-18 |
| <b>7 The LIMS Generic Data Set</b> .....          | 1-19 |
| 7.1 System Configuration .....                    | 1-20 |
| 7.2 File Definition .....                         | 1-21 |
| 7.3 Data Definition .....                         | 1-24 |
| 7.4 Code Definition .....                         | 1-25 |
| 7.5 Breed Definition .....                        | 1-25 |
| <b>8 Runtime Error System</b> .....               | 1-26 |
| <b>9 Equivalent Key Sequences</b> .....           | 1-28 |

# SETUP module

|  |      |
|--|------|
| <b>1 Introduction</b> .....              | 2-1  |
| <b>2 Starting the SETUP Module</b> ..... | 2-4  |
| <b>3 Main Menu</b> .....                 | 2-6  |
| 3.1 System Configuration .....           | 2-7  |
| 3.2 File Definition .....                | 2-11 |
| 3.2.1 Add a New File .....               | 2-12 |
| 3.2.2 Edit a File .....                  | 2-15 |
| 3.2.3 Delete a File .....                | 2-15 |
| 3.2.4 Activate a File .....              | 2-16 |
| 3.2.5 Deactivate a File .....            | 2-16 |
| 3.3 Data Definition .....                | 2-16 |
| 3.3.1 Select a Data File .....           | 2-17 |
| 3.3.2 Add a New Field .....              | 2-18 |
| 3.3.3 Edit a Field .....                 | 2-27 |
| 3.3.4 Delete a Field .....               | 2-28 |
| 3.3.5 Preview Fields .....               | 2-28 |
| 3.4 Code Definition .....                | 2-29 |
| 3.4.1 Select a Code Field .....          | 2-30 |
| 3.4.2 Add a New Code .....               | 2-30 |
| 3.4.3 Edit a Code .....                  | 2-30 |
| 3.4.4 Delete a Code .....                | 2-31 |
| 3.5 Breed Definition .....               | 2-32 |
| 3.5.1 Add a New Breed Rule .....         | 2-33 |
| 3.5.2 Delete a Breed Rule .....          | 2-33 |
| 3.6 Create Database Files .....          | 2-34 |
| 3.7 Detach Databae Files .....           | 2-35 |
| 3.8 Print Definitions .....              | 2-36 |
| 3.8.1 Print to Printer .....             | 2-36 |
| 3.8.2 Print to File .....                | 2-36 |

## **UPDATE module**

|  |      |
|--|------|
| <b>1 Introduction</b> .....                | 3-1  |
| <b>2 Starting the UPDATE Module</b> .....  | 3-3  |
| <b>3 Main Menu</b> .....                   | 3-5  |
| 3.1 Update a Data File .....               | 3-5  |
| 3.2 Change ID Numbers .....                | 3-6  |
| 3.3 Sort & Pack Records .....              | 3-10 |
| 3.4 Pack Records Only .....                | 3-10 |
| 3.5 Re-Index All Files .....               | 3-11 |
| 3.6 System Configuration .....             | 3-12 |
| <b>4 File Selection List</b> .....         | 3-13 |
| <b>5 Data Processing Menu.</b> .....       | 3-14 |
| 5.1 Add a New Record .....                 | 3-15 |
| 5.1.1 Parturition Record .....             | 3-17 |
| 5.1.2 Documentation Record .....           | 3-20 |
| 5.2 Edit a Record .....                    | 3-22 |
| 5.2.1 General Animal Record .....          | 3-22 |
| 5.2.2 Parturition Record .....             | 3-22 |
| 5.3 Delete a Record .....                  | 3-23 |
| 5.3.1 General Animal Record .....          | 3-24 |
| 5.3.2 Parturition Record .....             | 3-24 |
| 5.4 Mark Data Fields .....                 | 3-26 |
| 5.5 Sort & Pack Records .....              | 3-27 |
| 5.6 Pack Records Only .....                | 3-27 |
| 5.7 Re-Index Data File .....               | 3-27 |
| <b>6 Data Processing Sub-Systems</b> ..... | 3-28 |
| 6.1 Record Retrieval System .....          | 3-28 |
| 6.2 Assist System [F2] .....               | 3-30 |
| 6.2.1 Breed Lookup .....                   | 3-30 |
| 6.2.2 Code Lookup .....                    | 3-31 |
| 6.2.3 Value Lookup .....                   | 3-32 |
| 6.2.4 Date Lookup .....                    | 3-32 |
| 6.2.5 Mating Lookup .....                  | 3-32 |
| 6.2.6 Parity Lookup .....                  | 3-33 |
| 6.3 Validation .....                       | 3-34 |
| 6.3.1 Field Validation .....               | 3-34 |
| 6.3.2 Record Validation .....              | 3-35 |
| 6.4 Function Key Edit .....                | 3-36 |

## **VALIDATION module**

|   |      |
|---|------|
| <b>1 Introduction</b> .....                   | 4-1  |
| <b>2 Starting the VALIDATION Module</b> ..... | 4-2  |
| <b>3 Main Menu</b> .....                      | 4-4  |
| 3.1 Validation Configuration .....            | 4-4  |
| 3.2 Validate Data Files .....                 | 4-6  |
| 3.3 Print Data Validation .....               | 4-7  |
| 3.3.1 Print to Printer .....                  | 4-7  |
| 3.3.2 Print to File .....                     | 4-7  |
| <b>4 Data Validation</b> .....                | 4-8  |
| 4.1 Basic Field Check .....                   | 4-8  |
| 4.2 Basic Animal Data Check .....             | 4-11 |
| 4.3 Environment Data Check .....              | 4-15 |
| 4.4 Parturition Data Check .....              | 4-16 |
| 4.5 Estrus Data Check .....                   | 4-23 |
| 4.6 Mating Data Check .....                   | 4-23 |
| 4.7 Pregnancy Diagnosis Data Check .....      | 4-24 |
| 4.8 Milk Data Check .....                     | 4-25 |
| 4.9 Weight Data Check .....                   | 4-26 |

## **REPORTING module**

|   |      |
|---|------|
| <b>1 Introduction</b> .....               | 5-1  |
| <b>2 Starting the REPORT Module</b> ..... | 5-2  |
| <b>3 Main Menu</b> .....                  | 5-3  |
| 3.1 Report Definition .....               | 5-3  |
| 3.1.1 Add a New Report .....              | 5-4  |
| 3.1.2 Modify a Report .....               | 5-4  |
| 3.1.3 Delete a Report .....               | 5-4  |
| 3.2 Report Generation .....               | 5-4  |
| 3.3 Report Printing .....                 | 5-5  |
| 3.3.1 Print to Printer .....              | 5-5  |
| 3.3.2 Print to File .....                 | 5-5  |
| <b>4 Report Types</b> .....               | 5-6  |
| 4.1 Population Structure .....            | 5-6  |
| 4.2 Population Dynamics: Entry .....      | 5-8  |
| 4.3 Population Dynamics: Disposal .....   | 5-11 |
| 4.4 Action List: Age .....                | 5-13 |



## **EXTRACTION module**

|   |      |
|---|------|
| <b>1 Introduction</b> .....                     | 6-1  |
| <b>2 Starting the EXTRACT Module</b> .....      | 6-2  |
| <b>3 Main Menu</b> .....                        | 6-3  |
| 3.1 Extract Definition .....                    | 6-3  |
| 3.2 Extract Generation .....                    | 6-4  |
| 3.3 File Conversion .....                       | 6-4  |
| 3.4 Conversion Type Menu .....                  | 6-5  |
| 3.4.1 SDF Format .....                          | 6-5  |
| 3.4.2 Delimited Format .....                    | 6-6  |
| 3.4.3 Report Format .....                       | 6-6  |
| <b>4 Extraction Types</b> .....                 | 6-7  |
| 4.1 Individual Animal Performance by Age .....  | 6-7  |
| 4.2 Individual Animal Performance by Date ..... | 6-11 |
| 4.3 Parturition and Lactation Performance ..... | 6-16 |

# 1 Overview

The LIMS software package is designed to facilitate the recording and processing of livestock data in developing countries. The package is not specialised for use with any particular species or production system, but can be used with data for any mammal species for a variety of purposes, such as research, extension and commercial production.

The objective is to provide users with a computerised tool that allows them to manage complex livestock performance data set, and to provide help in livestock management without relying on specialist computer and data management support.

The LIMS software package has been designed to address the following six key issues in management of livestock performance data:

- Balance of flexibility and standardisation of data definition
- Documentation and definition of data sets
- Assistance in deriving non-observed data (breed, parity, mating-parturition connection)
- Data validation and error correction
- Reporting for animal management
- Data extraction and calculation of standard performance traits for statistical analysis.

In order to address these various aspects of livestock data management, the LIMS software has a modular design, with different modules addressing each of the above mentioned issues.

A generic data set is distributed along with the LIMS software, consisting of the definitions of essential and useful data files and variables. This generic data set can be individually configured by the user to suit personal requirements by changing existing data files and variables, as well as by adding new data files and variables. The possibility to individually configure the data structure allows users to accommodate a large variety of different livestock-related data.

LIMS is developed in the CLIPPER language. All data in a LIMS data set are stored in dBASE III+ compatible data files. This allows easy access to the data by other software packages that are able to read files in dBASE standard format. Using the **File Conversion** option of the EXTRACTION module the raw data stored in dBASE format can also be converted into three different ASCII formats.

The LIMS software can be run on a standard IBM compatible microcomputer without any additional software.

## 2 Hardware and Software Requirements

The LIMS software is developed in the CLIPPER version of the dBASE language. The different modules are distributed as executable programs (.EXE). No additional software, except the operating system, is required to run the LIMS software.

The following hardware and software are minimum requirements and/or recommendations in order to run the LIMS package:

|                       |   |
|-----------------------|---|
| Computer:             | IBM compatible, XT or AT<br>AT (80286 processor) is recommended       |
| Random Access Memory: | 640 KB  |
| Hard Disk Drive:      | required;<br>capacity according to the size of the data set(s)        |
| Video System:         | monochrome (MGA, Hercules) or colour (CGA, EGA, VGA) graphics adaptor |
| Operating System:     | MS-DOS or PC-DOS version 3.3 or higher is required                    |

It is recommended that the LIMS software be run on a micro-computer with an 80286 micro-processor (AT) having 640 KB of random access memory in order to attain an acceptable execution speed. LIMS is designed to make use of the colour capabilities of a colour graphics adaptor, but can also be run on a monochrome video system. Since the LIMS software accesses many files at the same time, it cannot run properly within the restrictions imposed by an operating system older than version 3.3.

### 3 The LIMS Software

The LIMS software consists of the executing programs, a generic LIMS data set and a set of colour definition files.

These are the executing programs:

|              |  |
|--------------|--|
| LSETUP.EXE   | SETUP module (MODULE1)<br>used to define a user data set   |
| LUPDATE.EXE  | UPDATE module (MODULE2)<br>used to add and edit data in a user data set                                      |
| LVALID.EXE   | VALIDATION module (MODULE3)<br>used to validate data in a user data set                                      |
| LREPORT.EXE  | REPORTING module (MODULE4)<br>used to create reports from a user data set                                    |
| LEXTRACT.EXE | EXTRACTION module (MODULE5)<br>used to extract data and calculate<br>performance traits from a user data set |

The definition of the generic data set, which is used to derive a user data set, is stored in the following five files:

|             |   |
|-------------|---|
| CFG_REF.MEM | System Configuration,<br>dBASE standard memory variable file. |
|-------------|---|

This file contains the settings for the system configuration of the generic LIMS data set.

|             |   |
|-------------|---|
| FDL_REF.DBF | File Definition Library, dBASE standard<br>database file. |
|-------------|---|

This file contains the generic file definitions. There is one record for each file defined in the generic data set.

|             |   |
|-------------|---|
| DDL_REF.DBF | Data Definition Library, dBASE standard<br>database file. |
|-------------|---|

This file contains the generic field (variable) definitions. There is one record for each field in each file belonging to the generic LIMS data set.

|             |  |
|-------------|--|
| CDL_REF.DBF | Code Definition Library, dBASE standard<br>database file. This file is empty in the generic<br>data set, but contains the code definitions for<br>categorical variables in a user data set. Each<br>code value has one record. |
|-------------|--|

**BDL\_REF.DBF** Breed Definition Library, dBASE standard database file.

This file is empty in the generic data set, but contains the breed rules in a user data set. A breed rule defines an offspring breed as a combination of the maternal and paternal breed codes. There is one record for each breed rule.

During the SETUP procedure, users create their own versions of these 5 data set definition files in a separate subdirectory for each user data set. These files contain the definitions of a user-defined data set. The generic data files are renamed in the following way (xxxx = user-defined data set name):

| generic LIMS data set definition files | user-defined LIMS data set definition files |
|--|---|
| CFG_REF.MEM                            | xxxxCFG.MEM                                 |
| FDL_REF.DBF                            | xxxxFDL.DBF                                 |
| DDL_REF.DBF                            | xxxxDDL.DBF                                 |
| CDL_REF.DBF                            | xxxxCDL.DBF                                 |
| BDL_REF.DBF                            | xxxxBDL.DBF                                 |

Three different files store the default settings for different monochrome video standards. By default, the LIMS system starts in colour mode, but can also be run in three different monochrome modes, the default settings for which are stored in the following files:

**CLRHERC.MEM** Hercules mode defaults, dBASE standard memory variable. This file contains the monochrome default settings for use with a Hercules graphics adaptor.

**CLRMONO.MEM** Monochrome mode defaults, dBASE standard memory variable.

This file contains the monochrome default settings for use with a colour adaptor (CGA, EGA, VGA) and a monochrome monitor.

**CLRLCD.MEM** LCD mode defaults, dBASE standard memory variable. This file contains the monochrome default settings for a liquid crystal or gas plasma display.

The settings in these video default files are loaded when video mode is specified. This is done by giving a parameter when calling one of the LIMS modules.

By default, the LIMS system loads the video settings from the file CLRNORM.MEM, if such a file exists. If there is no such file, the system will start in colour mode, unless a video parameter is specified when starting a LIMS module.

In order to make the LIMS system automatically start in a video mode other than the standard colour mode, the file containing the appropriate video settings must be copied to the file CLRNORM.MEM.

For example:

```
COPY CLRHERC.MEM CLRNORM.MEM
```

The above DOS command copies the file with the Hercules video settings to the file CLRNORM.MEM. All LIMS modules now start automatically in Hercules mode, without the need for specifying the video parameter /H.

## 4 Installing LIMS

The LIMS software is distributed on 5.25", double sided, double density diskettes. The software files are compressed in self-extracting files having the name MODULEx.EXE, where x is the module number. Sample data sets are also distributed as compressed files having the name xxxx.EXE where xxxx is the LIMS data set name. For more information on the DOS system configuration described below, see the DOS User's guide.

Use the following procedure to install the LIMS software:

1) Loading the LIMS software:

- a) Create a subdirectory on the hard disk which will contain the LIMS software with the DOS command MD (make directory). The recommended subdirectory name is: **\LIMS**.  
DOS command: MD \LIMS
- b) Make the subdirectory created under the current subdirectory, using the DOS command CD (change directory).  
DOS command: CD \LIMS
- c) Insert the diskette named MODULE1 into disk drive A:. Copy the LIMS software files from the diskette in drive A: to the current subdirectory on the hard disk using the following command:  
**A:\MODULE1**  
Repeat the above command, with different module numbers, for the other diskettes named MODULEx.  
If a floppy disk drive other than A: is used, change the disk drive designator in the command accordingly.

2) Making changes in the AUTOEXEC.BAT file:

- a) Create or change the PATH command to include the LIMS software directory (recommended name: \LIMS) with the full path and drive designator in the AUTOEXEC.BAT file.

For example (assuming that the LIMS software directory is C:\LIMS):

old PATH statement: PATH=C:\;C:\DOS;

new PATH statement: PATH=C:\;C:\DOS;C:\LIMS;

This change enables you to call the LIMS modules from any drive or subdirectory.

- b) Include the following statement in the AUTOEXEC.BAT file:

**SET CLIPPER=F25**

This allows the LIMS software to use a maximum of 25 files at the same time, unless the system default is less than 25. The system default is defined by the FILES setting in the CONFIG.SYS file (see below).

Changes made to the AUTOEXEC.BAT file become effective only after rebooting the computer.

3) Making changes in the CONFIG.SYS file:

- a) The setting for FILES in the CONFIG.SYS should be at least 25. Add the FILES statement below to the CONFIG.SYS file if it does not already exist, increase the number to 25 if it is less, but do not decrease the number if it is set to a value higher than 25.

**FILES = 25** (or greater)

- b) The setting for BUFFERS in the CONFIG.SYS should have a value of at least 20. Add the BUFFERS statement to the CONFIG.SYS file if it does not already exist, increase the number to 20 if it is less, but do not decrease the number if it is set to a value higher than 20.

**BUFFERS = 20** (or greater)

Changes made to the CONFIG.SYS file become effective only after rebooting the computer.

4) Loading the supplied sample LIMS data set:

- a) Create a subdirectory on the hard disk which will contain the sample LIMS data set. Example: **\SCH1**
- b) Make the subdirectory created under a) the current subdirectory, using the DOS command CD (change directory).
- c) Insert the diskette named DATA SET into the floppy disk drive A:. Extract and copy the sample LIMS data set files from the diskette in drive A: to the current subdirectory on the hard disk using the following command:

**A:\SCH1**

If a floppy disk drive other than A: is used, change the disk drive designator in the command accordingly. Other supplied LIMS data sets are copied in the same manner, replacing the above LIMS data set name SCH1 with the actual data set name indicated on the diskette.



## 5 Starting LIMS

Before calling a LIMS module, change the current subdirectory to the subdirectory in which the data set that you want to work with is located, using the DOS command CD (change directory). For the distributed sample data set the subdirectory is **\SCH1**.

If a new data set is to be created, first create a subdirectory having the same name as the new data set using the DOS command MD (make directory). Certain restrictions apply to the data set name and these are specified below. Change the current subdirectory to the newly created subdirectory which will contain the new data. Start the SETUP module using the syntax given below.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

To start one of the LIMS modules, change the current subdirectory to the subdirectory in which the data set to be used is located. Start the module as described below.

The following syntax is used to start the LIMS modules:

```
LSETUP      [xxxx] [/y]  (for the SETUP module)
LUPDATE     [xxxx] [/y]  (for the UPDATE module)
LVALID      [xxxx] [/y]  (for the VALIDATION module)
LREPORT     [xxxx] [/y]  (for the REPORT module)
LEXTRACT    [xxxx] [/y]  (for the EXTRACTION module)
```

The two items in square brackets ([ ]) are optional parameters that can be specified when calling a LIMS module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling a LIMS module, a prompt on the initial screen will request the input of the data set name.

/y This is the video mode option. By default, LIMS modules will attempt to start in colour mode.  
/H sets a special monochrome display which is optimised for

Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be selected. This mode uses the video settings contained in the file CLRHERC.MEM.

/M sets monochrome mode, even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor. This mode uses the video settings contained in the file CLRMONO.MEM.

/L sets a special monochrome display which is optimised for liquid crystal and gas plasma displays. If your computer has a display of one of these types, you should select this mode. This mode uses the video settings contained in the file CLRLCD.MEM.

If the data set name is given as a parameter when a LIMS module is started, the initial screen will display a number of messages, which will disappear after a fixed amount of time. If no name is given, a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

## 6 The LIMS Software Interface

All modules of the LIMS software have a standard way of communicating with the user. The collection of standard elements described in this chapter is called the LIMS software interface. Since the LIMS system can be individually customised, some of the sample displays shown in this chapter may differ from the actual displays of a particular user data set.

The LIMS system produces a character-based screen output 25 lines long and 80 characters wide. Depending on which video configuration is loaded when starting LIMS, different combinations of colours, shades and intensity levels are used to display the various elements of the software interface.

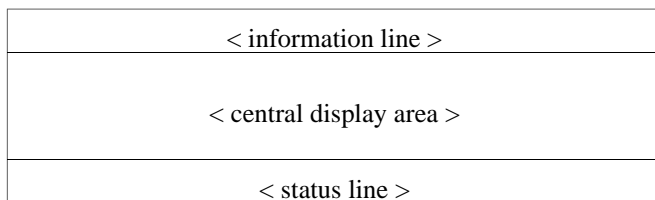
Four up-to-date video configurations for different types of display adaptors and screens are provided. These video configurations are not user-configurable.

### 6.1 Display Areas

The 25 lines of the LIMS screen display are divided into the following areas:

- 1) Information Line (line 1)  
This line displays the name of the currently running LIMS module, the name and label of the currently used data set and information about how to access the HELP system.
- 2) Central Display Area (lines 2 to 24)  
This is the main communication and data entry area. All menus, selection lists, look-up lists, information display and data entry screens appear in this area.
- 3) Status Line (line 25)  
This line is mainly used to display information, warning and error messages as well as user prompts. During data processing, this line shows the name of the currently active data file, the name of the currently active process and a record pointer.

On the screen, the LIMS screen display areas look like this:



The above description of screen areas and their functions applies to all the modules of the LIMS system.

## **6.2 Record Edit Area**

After a file has been selected from the File Selection List of the UPDATE module, the central display area will contain the Data Processing Menu in the foreground and an empty record of the currently selected file in the background. The file label is displayed above the empty record.

A record of the currently selected data file is displayed in a column on the left-hand side of the central display area, one field per row, in all up to 19 fields. If there are more than 19 displayed fields in a data file, the remaining fields will be shown in a column on the right-hand side of the central display area. The index fields of a data file are always displayed as the first fields.

Each field is displayed with the corresponding field label and the data entry area. The space between the field label and the data entry area is taken up by the 'underline' character. The data entry area is normally empty for character variables whereas for numeric fields it contains the default missing value. Some fields may contain characters that are part of the field value but do not have to be typed. The slashes in the date fields are an example.

When the Record Retrieval System is active the index fields are displayed in a different colour. Similarly, during the Edit Process the currently active Field input area is indicated by a different colour.

The following table shows the keys used to edit variable values:

|              |   |  |
|--------------|---|--|
| Leftarrow    | - | moves the cursor one character to the left                         |
| Rightarrow   | - | moves the cursor one character to the right                        |
| Uparrow      | - | leaves the current field, activates the previous field for editing |
| Dnarrow      | - | leaves the current field, activates the next field for editing     |
| Home         | - | moves the cursor to the beginning of a field                       |
| End          | - | moves the cursor to the end of a word/field                        |
| Ctrl-Y       | - | deletes from the cursor to the end of a field                      |
| Ctrl-U       | - | restores the previous field value                                  |
| Del          | - | deletes the character above the cursor                             |
| Backspace    | - | deletes the character before the cursor                            |
| <character>  | - | types <character> (if permissible in that field and position)      |
| Return       | - | leaves the present field, activates the next field for editing     |
| Ins          | - | toggles between insert and overwrite mode                          |
| F2           | - | invokes the <b>Assist System</b>                                   |
| F3/4/5/6/7/8 | - | invokes the <b>Function Key Edit</b>                               |
| F9           | - | jumps to the next marked field                                     |
| F10          | - | saves edited record  |
| Esc          | - | aborts edited record, returns to the <b>Data Processing Menu</b>   |

On the screen, the LIMS record edit area has the following appearance:

| WEIGHT DATA     |       |        |
|-----------------|-------|--------|
| ID Number       | _____ | -      |
| Weighing Date   | _____ | / /    |
| Weight          | _____ | -999.9 |
| Weighing Method | _____ |        |
| Body Score      | _____ |        |

The above example shows an empty record belonging to the Weight data file as it would appear in the central display area. Note the hyphen which in this example has been defined as a separator in the animal identification field, and the default missing value (-999.9) for the field Weight.

### 6.3 Function Key Edit Window

The Function Key Edit system is available when a data record is displayed and one of the data fields is active during the **Add a New Record** and **Edit a Record** processes.

Using the Function Key Edit system it is possible to view and edit a record belonging to another data file than the one active, simply by pressing one of the assigned function keys. The record of the other file is displayed in a scrolled window that overlays the record of the main edit file.

Within the active Function Key Edit window, cursor movement and editing are carried out using the same keys as for the Record Edit Area, except for the function keys of the Function Key Edit system itself.

In order to protect data integrity, not all fields in the active Function Key Edit window can be edited. Fields that can be edited are preceded by a triangular marker.

Fields in the Function Key Edit window can be active in two modes, Browse mode and Edit mode. Fields that cannot be edited can only be active in Browse mode, whereas fields that can be edited may be active in both modes.

The Function Key Edit window has the following appearance on the screen:

| GENERAL ANIMAL DATA: (70-01) |                 |
|------------------------------|-----------------|
| ID Number _____              | 70-01           |
| Birth Date _____             | 01/08/70        |
| Sex of Animal _____          | F               |
| Sire ID Number _____         | -               |
| Dam ID Number _____          | -               |
| Animal Breed _____           | BO              |
| Breed Purity _____           | P               |
| Parity _____                 | -9              |
| Birth Type _____             | 0               |
| Birth difficult _____        |                 |
| Rearing Type _____           | -9              |
| Castration Ind. _____        |                 |
| Current Mode                 | - Browse Record |

The above example shows the Function Key Edit window displaying part of a General Animal Record. The centre part of the Function Key Edit window can be scrolled if there are more fields in a record than fit on the screen. The bottom line indicates the current mode of the active field.

Pressing the RETURN key in Browse mode switches the active field to edit mode in which the current field value can be changed (provided that the field can be edited). In Edit mode, pressing the RETURN key saves the edited value, leaves the current field and activates the next field that can be edited in Browse mode. The F9 key (jump key) activates the next editable field in the Browse mode.

## **6.4 Menus**

Menus allow you to access the different levels and processes available in the LIMS modules. A menu appears in a separate window which displays a fixed number of options of which one can be selected. The currently selected option is shown in a different colour. When a lower level process terminates, it returns to the next higher level, which is usually a menu.

The following keys are generally used with an active menu:

| Key      | Action  |
|----------|---|
| Spacebar | selects the following option                      |
| Dnarrow  | selects the following option                      |
| Uparrow  | selects the preceding option                      |
| <number> | selects and activates the option with that number |
| Return   | activates the selected option                     |
| Esc      | returns to the higher level process               |

LIMS menus have the following appearance on the screen:

| MAIN MENU             |                       |
|-----------------------|-----------------------|
| 1.                    | System Configuration  |
| 2.                    | File Definition       |
| 3.                    | Data Definition       |
| 4.                    | Code Definition       |
| 5.                    | Breed Definition      |
| 6.                    | Create Database Files |
| 7.                    | Detach Database Files |
| 8.                    | Print Definitions     |
| Esc - Exit the system |                       |

The above example shows the Main Menu of the SETUP module. The lower level processes that can be accessed from a menu may be data entry screens (1) sub-menus (2,3,4,5,8) selection lists (6,7) or processes that do not require a new display (for example, **Sort & Pack Records** in the Main Menu of the UPDATE module).

## 6.5 Selection Lists

The selection lists of the LIMS software interface allow the user to select one item, usually a file or a field, from a list with a variable number of items. Selection lists function in much the same way as menus. A selection list appears as a separate scrolled window which displays a variable number of items of which one can be selected. The currently selected item is shown in a different colour.



The following keys are generally used with an active selection list:

| Key         | Action   |
|-------------|--|
| Spacebar    | selects the following item                       |
| Dnarrow     | selects the following item                       |
| Uparrow     | selects the preceding item                       |
| PgDn        | scrolls the selection list downward              |
| PgUp        | scrolls the selection list upward                |
| <character> | selects the next item beginning with <character> |
| Return      | activates a process on the selected item         |
| Esc         | returns to the higher level process              |

On the screen, selection lists look like this:

| SELECTED FILE LIST        |
|---------------------------|
| GEN - General Animal Data |
| ENV - Environment Data    |
| PAR - Parturition Data    |
| Esc - return to main menu |

The above example shows the File Selection List of the UPDATE module. If there are more items in the selection list than can be displayed, arrows on the frame of the selection list indicate that it can be scrolled.

## 6.6 Lookup Lists

The lookup lists of the LIMS software interface enable you to access available information on an active field. Lookup lists are generated by the Assist System, which is invoked by pressing the F2 function key. Lookup lists appear very similar to selection lists, but function in a different manner. Selection lists are separate levels within the hierarchy of processes of the LIMS system, whereas lookup lists are optional displays of information. A lookup list appears in a separate scrolled window which displays a variable number of items of which one can be selected. The currently selected item is shown in a different colour.

The following keys are generally used with an active lookup list:

| Key         | Action   |
|-------------|--|
| Spacebar    | selects the following item                         |
| Dnarrow     | selects the following item                         |
| Uparrow     | selects the preceding item                         |
| PgDn        | scrolls the selection list downward                |
| PgUp        | scrolls the selection list upward                  |
| <character> | selects the next item beginning with <character>   |
| Return      | carries the selected item over to the active field |
| Esc         | returns to the higher level process                |

A LIMS lookup list has the following appearance on the screen:

| BREED LOOKUP LIST     |                   |
|-----------------------|-------------------|
| BO                    | - Boran           |
|                       | -                 |
| BO-HF                 | - BO x HF         |
| HF                    | - Holst. Friesian |
| HF-BO                 | - HF x BO         |
| Esc - abort selection |                   |

The above example shows the Breed Lookup List, which can be displayed when the field BREED is active during data processing. If there are more items in the lookup list than fit on the screen, arrows on the frame of the lookup list indicate that the list can be scrolled.

## 6.7 Dialogue Boxes

The dialogue boxes of the LIMS software interface allow the user to supply additional information to the program. Usually a default value is provided.

LIMS dialogue boxes have the following appearance:

Enter File Name :

The above example shows a dialogue box requesting the user to supply a filename for a LIMS output file that will be sent to a printer or to an ASCII file.

## **6.8 Prompts**

The status line displays prompts requesting the user to make certain decisions or choose between alternatives of processes. A default answer is usually supplied indicating the expected standard answer. Pressing the RETURN key will select the standard default answer, other choices can be made by pressing one of displayed keys.

A LIMS prompt may look like this:

ARE ALL DETAILS CORRECT? (Y/N/A) Y

The above example shows the prompt that appears after a data record has been added or edited.

## **6.9 Messages**

The status line is also used to display messages. There are three types of messages: general information, warnings and error messages. Warnings and error messages are normally displayed for 3 seconds or until any key is pressed.

LIMS messages can have the following appearance:

INDEXING DATABASE FILE - SCH1GEN  
WARNING - CODE WAS NOT FOUND IN THE CODE  
DEFINITION LIBRARY!  
ERROR - ANIMAL ID (id) DOES NOT EXIST IN THE  
GENERAL FILE!

## 7 The LIMS Generic Data set

As a part of the LIMS package, a generic data set definition is supplied together with the LIMS software. This generic data set defines the standard data files and variables for animal performance data. In addition to using the standard data files and variables, users can define additional (= user-defined) variables and files.

A LIMS data set is defined on 5 levels. Standard definitions and default values are supplied for the first 3 levels (data set, file, variable).

### 1) Data set

On the data set level, configurations, options and defaults can be set that apply to an entire LIMS data set. This configuration is done with the **System Configuration** option of the SETUP module.

Some examples are:

- date display format
- animal identification length and format
- the code used to denote animal sex

### 2) Files

On the file level, users can select predefined files and add user-defined files to the data set. Selection and definition is done with the **File Definition** option of the SETUP module. All LIMS data sets must include the three predefined files: General Animal Data, Environment Data and Parturition Data.

### 3) Variables

On the variable level, users can select or configure certain predefined variables and add user-defined variables. Certain restrictions can be imposed on the range of values that variables can have. This selection and definition is done with the **Data Definition** option of the SETUP module.

### 4) Codes

No predefined codes are distributed with the LIMS package. Users must define all the codes that will be used in a data set, including those that are defined at the data set level. Definition is done with the **Code Definition** option of the SETUP module.

### 5) Breed Rules

The same as codes, no predefined breed rules are supplied with the LIMS package. Users must define all breed rules that are to be used in a data set, using only defined breed codes. Each breed rule defines an offspring breed code by the combination of the parental breed codes. This definition is done using the **Breed Definition** option of the SETUP module.

Users can add a label and description to each file, variable and code used in a LIMS data set. User-specified labels are displayed whenever possible instead of the internally used names. The user-supplied information is used for error checking during data entry and validation.

## 7.1 System Configuration

The option **System Configuration** of the SETUP module allows users to set configurations, options and default values which apply to the entire data set. This configuration is stored in a dBASE memory variable file with the name xxxxCFG.MEM (xxxx = user-defined data set name).

The default configuration screen has the following appearance, displaying the default settings supplied with the generic LIMS data set:

Data Set Code \_\_\_\_\_ SCH1  
Data Set Title \_\_\_\_\_ Sample Cattle Herd Data Set 1

Code for <Yes> \_\_\_\_\_ Y                    ID Format \_\_\_\_\_ 99–NNNN  
Code for <No> \_\_\_\_\_ N                    Maximum Litter Size \_\_\_\_\_ 02  
Code for <Abort> \_\_\_\_\_ A                Record Dead Animals \_\_\_\_\_ F  
    Epoch Year \_\_\_\_\_ 1940

Mate Lookup Period \_\_\_\_\_ 310  
Warnings Enabled \_\_\_\_\_ T                Male \_\_\_\_\_ M  
Confirm Enabled \_\_\_\_\_ T                Female \_\_\_\_\_ F  
Date Display Format \_\_\_\_\_ DD/MM/YY    InterSex \_\_\_\_\_ X  
    Genetic Active \_\_\_\_\_ G

Function Key 6 File \_\_\_\_\_                Was Genetic Active \_\_\_\_\_ W  
Function Key 7 File \_\_\_\_\_                Entry Due to Birth \_\_\_\_\_ 01  
Function Key 8 File \_\_\_\_\_

The codes and options of the System Configuration are described in further detail in the chapter **System Configuration** in the SETUP manual.

## 7.2 File Definition

This chapter lists the default file definitions for the predefined data files of the supplied sample data set. Each file definition is stored in one record of the file definition library xxxxFD.LDB (xxxx = user-defined data set name).

The file definition information is listed in a format similar to that produced by the **Print Definitions** option of the SETUP module. The order of definitions

described herein may, however, differ from that of the sample data set, and some of the file descriptions may also be more extensive than those found in the sample data set.

The **File Definition** option of the SETUP module allows you to select and modify the predefined file definitions and add user-defined file definitions. For predefined data files, only the fields File Number, File Label and File Description can be directly changed by the user. The other fields contain either predefined constant values or are automatically updated when the SETUP or the UPDATE module is run.

For more information on how to select, modify or add file definitions, see the **File Definition** chapter of the SETUP module.

### General Animal Data

|                           |          |                         |        |
|---------------------------|----------|-------------------------|--------|
| File Name Code_____       | GEN      | Number of Key Fields__  | 1      |
| File Create Date_____     | 20/06/91 | Number of Fields_____   | 21     |
| Data Last Updated_____    | 10/06/92 | Number of Records _____ | 000152 |
| File Deactivate Date ____ | / /      | Record Length _____     | 0107   |

File Number\_\_\_\_\_ 001  
File Label\_\_\_\_\_ General Animal Data

#### File Description:

This file contains the general animal data about each animal in a LIMS data set. Each record contains the data about one animal and all animals in a data set must have unique identifications. Performance records of an animal can be added to other data files in a LIMS data set only if the animal has a general animal data record. The file is indexed by the animal identification. This data file is one of the required files in a LIMS data set.

### Environment Data

|                           |          |                         |        |
|---------------------------|----------|-------------------------|--------|
| File Name Code_____       | EN       | Number of Key Fields__  | 2      |
| File Create Date_____     | 20/06/91 | Number of Fields_____   | 08     |
| Data Last Updated_____    | 10/06/92 | Number of Records _____ | 000152 |
| File Deactivate Date ____ | / /      | Record Length_____      | 0044   |

File Number\_\_\_\_\_ 002  
File Label\_\_\_\_\_ Environment Data

#### File Description:

This file contains the history of environment changes in an animal's lifespan. Each record contains the data about a single change of environment for one

animal. The exact meaning of environment depends on the particular data set and is given by the user. Environment can be defined as village, farmer, herd, grazing plot, experiment, treatment, etc. The file is indexed by a combination of animal identification and environment change date. This data file is one of the required data files in a LIMS data set.

### Parturition Data

|                            |          |                         |        |
|----------------------------|----------|-------------------------|--------|
| File Name Code_____        | PAR      | Number of Key Fields __ | 2      |
| File Create Date_____      | 20/06/91 | Number of Fields _____  | 15     |
| Data Last Updated_____     | 10/06/92 | Number of Records _____ | 000142 |
| File Deactivate Date _____ | / /      | Record Length_____      | 0073   |

File Number\_\_\_\_\_ 003  
File Label\_\_\_\_\_ Parturition Data

### File Description:

This file contains the parturition data, each record containing the data about one parturition. Key data about a successful mating can be transferred from the mating data file to the parturition data file. Since the parturition data file stores the identifications of all offspring born in a parturition, this file contains the information needed to link animal generations. The file is indexed by a combination of dam identification and parturition date. This data file is required in every LIMS data set.

### Estrus Observation Data

|                            |     |                         |        |
|----------------------------|-----|-------------------------|--------|
| File Name Code_____        | EST | Number of Key Fields __ | 2      |
| File Create Date_____      | / / | Number of Fields _____  | 03     |
| Data Last Updated_____     | / / | Number of Records _____ | 000000 |
| File Deactivate Date _____ | / / | Record Length_____      | 0019   |

File Number\_\_\_\_\_ 009  
File Label\_\_\_\_\_ Estrus Observation

### File Description:

This file contains estrus observation data, each record containing the information about one estrus observation. Usually mating data imply estrus observations, but this file allows you to store estrus observations independent of matings. The file is indexed by a combination of animal identification and estrus observation date. It is user-selectable.

**Mating Data**

File Name Code \_\_\_\_\_ MAT      Number of Key Fields \_ 2  
 File Create Date \_\_\_\_\_ 20/06/91      Number of Fields \_\_\_\_\_ 06  
 Data Last Updated \_\_\_\_\_ 10/06/92      Number of Records \_\_\_\_\_ 000090  
 File Deactivate Date \_\_\_\_\_ / /      Record Length \_\_\_\_\_ 0031

File Number \_\_\_\_\_ 004  
 File Label \_\_\_\_\_ Mating Data

**File Description:**

This file contains information about matings, each record containing the data about one mating. Only one mating record per dam per day is allowed. In order to store multiple matings in one day in one record appropriate user-defined variables must be created. Indexing is done by a combination of dam identification and mating date. This file is user-selectable.

**Milk Data**

File Name Code \_\_\_\_\_ MLK      Number of Key Fields \_ 2  
 File Create Date \_\_\_\_\_ 04/07/91      Number of Fields \_\_\_\_\_ 11  
 Data Last Updated \_\_\_\_\_ 08/06/92      Number of Records \_\_\_\_\_ 000089  
 File Deactivate Date \_\_\_\_\_ / /      Record Length \_\_\_\_\_ 0045

File Number \_\_\_\_\_ 006  
 File Label \_\_\_\_\_ Milk Data

**File Description:**

Milk yield data is stored in this file, each record containing the daily milk yield or the milk yield of a time period. If period milk yields are recorded, the recorded milking date must be the last day of the period. The file is indexed by a combination of dam identification and milking date. This file is user-selectable.

**Weight Data**

File Name Code \_\_\_\_\_ WGT      Number of Key Fields \_ 2  
 File Create Date \_\_\_\_\_ 20/06/91      Number of Fields \_\_\_\_\_ 06  
 Data Last Updated \_\_\_\_\_ 29/11/91      Number of Records \_\_\_\_\_ 000193  
 File Deactivate Date \_\_\_\_\_ / /      Record Length \_\_\_\_\_ 0028

File Number \_\_\_\_\_ 005  
 File Label \_\_\_\_\_ Weight Data



### **File Description:**

This file contains weight data. Each record contains one recorded body weight. Key body weights such as birth weight, weaning weight or post-partum weight can also be stored in this file. They can later be extracted by linking the weight date to the birth date (General Animal Data), the weaning date (General Animal Data) or the parturition date (Parturition Data). Alternatively, such weights can be recorded in other data files (General Animal Data, Parturition Data) by creating appropriate user-defined fields in these data files. The weight data file is indexed by a combination of animal identification and weighing date. It is user-selectable.

## **7.3 Data Definition**

This chapter lists the default variable (field) definitions of the predefined variables of the supplied generic data set. Each variable definition is stored in one record of the data definition library xxxxDL.DBF (xxxx = user-defined data set name).

The variable definition information is here listed in a format similar to that produced by the **Print Definitions** option of the SETUP module. The order of definitions may, however, differ from the order produced by that option. The field descriptions may also not be as extensive in the actual sample data set. Settings that can either be turned on or off are listed as T (true/yes/on) or F (false/no/off).

Using the **Data Definition** option of the SETUP module you can select and modify the predefined variable definitions and add user-defined variable definitions. For all predefined variables the fields Field Label, Field Number and Field Description can be directly changed by the user. Depending on field category and usage may also be changed. For a full description of permitted changes to predefined variable definitions, see the chapter **Data Definition** of the SETUP module.

Predefined variables belong to two basic categories, system-defined (SD) and user-selectable (US). System-defined variables are required in every LIMS data set, whereas definitions of user-selectable variables can be deleted. Both categories of predefined variables may be user-configurable (SDUC, USUC), which means that the length (and number of decimals, if applicable) of the variable can be specified by the user.

The Date Deviation variables described in this chapter are not part of the supplied sample data set. They have, however, been included here as an optional set of variables. They are intended to accompany certain date variables in the General Animal Data, the Environment Data and the Parturition Data files. These variables allow you to accommodate

observations for which no precise date has been recorded, but rather a range of days. In such cases, Date Deviation variables can be used to indicate the precision with which a date has been recorded.

For example:

In certain cases, birth dates might not be individually observed but may still be reported. Such dates may be reported as a time interval (e.g. month, week, season, year) instead of as precise dates. However, to make full use of the special date-type variables provided in the dBASE language, an exact date value is required. For such observations, the mid-point of a reported interval can be recorded as the best estimate of the true date. The Date Deviation variable for that date can then be used to store the maximum error of the recorded date, being half the reported interval (in days). A Date Deviation value of 0 indicates an accurate date. All Date Deviation variables are user-selectable and can therefore be omitted if recorded dates are believed to be accurate.

After a data file is created with the option **Create Database Files** of the SETUP module, many of the configurable options become permanent for that data file. Only the Field Label, the Minimum and Maximum Values, the setting for Allow Missing and the Field Description can be changed once the data file has been created.

## **7.4 Code Definition**

With the exception of some sample codes in the supplied data set, no predefined codes are distributed with the LIMS system. Users must define all codes to be used in a data set, including those that are defined at the data set level. This definition is done with the **Code Definition** option of the SETUP module. Code definitions are stored in the Code Definition Library xxxxCDL.DBF (xxxx = user-defined data set name).

## **7.5 Breed Definition**

Apart from certain sample rules contained in the supplied data set, no predefined breed rules are distributed with the LIMS system. Users must therefore define all breed rules to be used in a data set. This definition is done with the **Breed Definition** option of the SETUP module. Breed rules are stored in the Breed Definition Library xxxxBDL.DBF (xxxx = user-defined data set name).

## 8 Runtime Error System

While the LIMS system is running, unrecoverable runtime errors can occur causing the system to crash and return to the operating system level. Such runtime errors may either be caused by errors in the software itself (bugs) that must therefore be corrected, or they may be caused by the operating system. Common causes of runtime errors that are due to the operating system environment are: using wrong operating system version, a too small FILES setting, too low BUFFERS setting, or omission of the command SET CLIPPER=F25. These settings are described under the heading **Installing LIMS** in the **Introduction** chapter. Another cause of error that is due to the operating system environment may be that there is not enough disk space for a certain operation. Runtime errors that are due to the same cause may not always appear in the same place and may not always be easily repeatable. If an error can easily be repeated, it is usually easy to diagnose its cause and correct the error.

Runtime errors are different from error messages displayed on the bottom line of the screen. Such error messages are caused by errors or inconsistencies in the data and the ensuing error messages are described in the corresponding chapters of this manual. Such data errors do not cause the system to crash, but only require that different data be entered or that data be modified.

Failures of the computer system hardware, such as power failure or a malfunctioning memory chip usually do not generate runtime errors that can be diagnosed by the LIMS software.

In order to keep track of runtime errors and provide a means of communicating them, the LIMS system generates the ASCII text file `xxxxSYS.ERR` (`xxxx` = data set name). This file contains a report of each runtime error that occurs and new runtime errors are appended. The content of the reports is quite complex and cannot be described here.

When reporting runtime errors to the LIMS developers and distributors please include a copy of this runtime error log file, as well as a description of the process that triggered the error (e.g. Sort & Pack Records from the Main Menu of the UPDATE module, Create Database Files from the Main Menu of the SETUP module, etc).

When the cause of an error has been identified so that the runtime error no longer occurs, the runtime error log file `xxxxSYS.ERR` (`xxxx` = data set name) can be deleted in order to conserve disk space.

The LIMS software is largely driven by the contents of the data set definition files `xxxxCFG.MEM`, `xxxxFDL.DBF`, `xxxxDDL.DBF`, `xxxxCDL.DBF` and `xxxxDBL.DBF` (`xxxx` = data set name). If changes are made either to the file content or to the file structure of these files with a

software other than LIMS, the performance of the LIMS software can change in unpredictable ways, possibly resulting in runtime errors. For the same reason the structure of the data files of a LIMS data set must not be changed.

When working on a LIMS data set with a software other than LIMS, it is advisable to make a working copy of the LIMS data set for this purpose.

## 9 Equivalent Key Sequences

All special key strokes (e.g. for cursor movement, selection etc) are documented as the single stroke keys normally available on an IBM PC keyboard. Some of the keys may serve dual purposes, as special keys and as numeric keys on the numeric key pad. For each of these keys there is an equivalent control key sequence that can be used if the numeric key pad is used exclusively for the entry of numerical values.

The equivalent key sequences are:

|                 |        |           |             |
|-----------------|--------|-----------|-------------|
| Uparrow         | Ctrl-E | Ctrl-End  | Ctrl-W      |
| Dnarrow         | Ctrl-X | Ctrl-PgUp | Ctrl-Hyphen |
| Leftarrow       | Ctrl-S | Ctrl-PgDn | Ctrl-Caret  |
| Rightarrow      | Ctrl-D | Esc       | Ctrl-[      |
| Home            | Ctrl-A | Return    | Ctrl-M      |
| End             | Ctrl-F | Enter     | Ctrl-M      |
| PgUp            | Ctrl-R | Del       | Ctrl-G      |
| PgDn            | Ctrl-C | Tab       | Ctrl-I      |
| Ctrl-Leftarrow  | Ctrl-Z | Ins       | Ctrl-V      |
| Ctrl-Rightarrow | Ctrl-B | Backspace | Ctrl-H      |
| Ctrl-Home       | Ctrl-] |           |             |

## 1 Introduction

The LIMS SETUP module is written in the CLIPPER version of the dBASE language and compiled with the CLIPPER 5.0 compiler. The data set files are compatible with the dBASE III+ standard except for the index files (.NTX), which are CLIPPER specific.

The SETUP module of the LIMS system is the only module that can modify the structure of a data set. A generic, predefined LIMS data set is distributed along with the LIMS software. This generic data set can be modified and amended to suit your individual requirements.

A LIMS data set consists of several files belonging to different categories. All data set files have the same four-character user-defined prefix (xxxx in the following table), a three-character system-defined suffix and, according to their content, one of the four file extensions: .MEM, .DBF, .DBT, or .NTX.

The following table shows the files in a LIMS data set that can be created and edited by the SETUP module:

| file category          | file name              | comment                  |
|------------------------|------------------------|--------------------------|
| data set configuration | xxxxCFG.MEM            | required, system defined |
| data set definition    | xxxxFDL.DBF + .NTX     | required, system-defined |
|                        | xxxxDDL.DBF + .NTX     | required, system-defined |
|                        | xxxxCDL.DBF + .NTX     | required, system-defined |
|                        | xxxxBDL.DBF + .NTX     | required, system-defined |
| raw data               | xxxxHIS.DBF            | required, system-defined |
|                        | xxxxGEN.DBF + .NTX     | required, system-defined |
|                        | xxxxENV.DBF + .NTX     | required, system-defined |
|                        | xxxxPAR.DBF + .NTX     | required, system-defined |
|                        | xxxxEST.DBF + .NTX     | optional, system-defined |
|                        | xxxxMAT.DBF + .NTX     | optional, system-defined |
|                        | xxxxPRG.DBF + .NTX     | optional, system-defined |
|                        | xxxxWGT.DBF + .NTX     | optional, system-defined |
|                        | xxxxMIL.DBF + .NTX     | optional, system-defined |
|                        | xxxxHLT.DBF + .NTX     | optional, system-defined |
|                        | xxxxDOC.DBF + .NTX     | optional, system-defined |
|                        | xxxxDOC.DBT            |                          |
|                        | xxxxU01.DBF + .NTX     | optional, user-defined   |
|                        | ...                    | ...                      |
| xxxxU99.DBF + .NTX     | optional, user-defined |                          |

The file with the extension .MEM contains values for global system configuration. Files with the extension .DBF contain the data set definitions or raw data in the dBASE III+ standard, depending on the file category. Files with the extension .NTX contain the index information for the .DBF file with the same name. This index information allows very fast record retrieval by binary search. It is not compatible with the dBASE III+ index. One file with the extension .DBT contains records with variable text length, referred to by a memo field in the file xxxxDOC.DBF.

The definition of a user data set is stored in the four library files, the File Definition Library (xxxxFDL.DBF), the Data Definition Library (xxxxDDL.DBF), the Code Definition Library (xxxxCDL.DBF) and the Breed Definition Library (xxxxBDL.DBF). The SETUP module mainly accesses these library files and allows you to make changes to the definitions stored in these files. On the basis of the definitions stored in the File Definition Library and the Data Definition Library, the SETUP module creates the actual database files. These files can subsequently be accessed through the UPDATE module for data entry.

The integrity of the data set definition library is of great importance for the proper functioning of all LIMS software modules. Do not attempt to access the definition libraries directly (e.g. via dBASE) to make changes in the data set definition.



## 2 Starting the SETUP Module

The filename of the executing SETUP module is LSETUP.EXE. This file should be located in the subdirectory \LIMS on a hard disk. The PATH statement in the AUTOEXEC.BAT file should include the subdirectory \LIMS so that the SETUP module can be started independent of the current drive or path. For more information on the PATH statement in the AUTOEXEC.BAT file, please refer to the DOS User's Guide.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This data set name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

If a new data set is to be created, then first create a sub-directory having the same name as the new data set. The data set name must conform to the above specified format. Change the current subdirectory to the newly created subdirectory in which the new data set will be created. Start the SETUP module with the syntax given below.

If changes are to be made in the definition of an existing data set, change the current subdirectory to the subdirectory containing that data set. Start the SETUP module using the syntax given below.

The following syntax is used to start the LIMS SETUP module:

```
LSETUP [xxxx] [/y]
```

The two items in square brackets ([ ]) are optional parameters that can be specified when calling the SETUP module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling the SETUP module, a prompt on the initial screen will request the input of the data set name.

/y This is the video mode option. By default, the SETUP module will attempt to start in colour mode.

/H sets a special monochrome display which is optimised for Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be selected.

*/M* sets monochrome mode even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor.

*/L* sets a special monochrome display which is optimised for liquid crystal and gas plasma displays. If your computer has a display of one of these types, you should select this mode.

If the data set name is given as a parameter when the SETUP module is started, the initial screen will display a number of messages which will disappear after a fixed amount of time. If no name is given, a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

### 3 Main Menu

The Main Menu displays the 8 main menu options in a window in the centre of the display area. The currently selected option is displayed in a different colour.

The following table shows the keys for use with the Main Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the DOS level                          |

The first step in creating a new data set is to complete the System Configuration process. No other option from the Main Menu can be run until the **System Configuration** has been completed. If an attempt is made to execute another option before completion of the System Configuration, the message **ERROR - SYSTEM CONFIGURATION MUST BE COMPLETED FIRST!** will be displayed.

The top line of the Main Menu screen contains information about the Setup module and the current data set. In the lefthand corner the module name is displayed, the data set code and the data set title (given during the setup process) are shown in the middle, and the current date to the right. This line is displayed throughout execution of the SETUP module.

The bottom line displays status information, which in the case of the Main Menu display is the full version of the system name and the version number of the SETUP module. The different LIMS modules can have different version numbers, but the main version numbers (the integer part) of all LIMS modules must be the same for the different modules to work together properly.

The status line is used to display information, warning and error messages, as well as for confirmation input of certain commands.

### 3.1 System Configuration

This option displays the codes and options that can be defined and set by the user. Default values are provided for convenience, but may be changed to suit particular requirements. The complete set of default values cannot be used; at least one value must be changed (e.g. the data set title). The System Configuration values stored in the file xxxxCFG.MEM determine certain aspects of the data set structure and the internal logic of all LIMS modules. The following codes and options can be changed with the **System Configuration** option.

- Data Set Title: this allows you to name the data set. The data set code and the data set title appear in the middle of the top line of the display during execution of all LIMS modules indicating which data set is presently active. The data set title can be changed at any time.
- the response codes for YES, NO and ABORT to questions such as ‘ARE ALL DETAILS CORRECT?’.
- the Mating Lookup Period, which is the length of time before parturition (in days) for which the mating lookup procedure will search the mating data file and display the matings encountered. For further information see the chapter **Mating Lookup** in the UPDATE module.
- warnings enabled (T/Y = true/yes, F/N = false/no): this determines whether or not field validation and record validation will be carried out during data entry and modification. For further information see the chapters **Field Validation** and **Record Validation** in the UPDATE module.
- confirm enabled (T/Y = true/yes, F/N = false/no), which determines whether or not an additional pressing of the RETURN key is required after the last character position of a variable has been edited.
- date display format: you can choose between several national and international date formats. The selection affects only the format of display and data entry, but not the format of internal storage. The F2 key brings up a selection list. This setting can be changed at any time, but care should be taken to avoid confusion.
- function key 6/7/8 file: this allows you to assign the function keys 6, 7 and 8 to a data file for function key editing. The F2 key brings up a selection list of data files to which the function keys can be assigned. For more information on function key editing, see the chapter **Function Key Edit** in the UPDATE module.
- ID Format: this allows you to specify a template to be used for the entry and display of animal identifications throughout the entire data set. A template is a mask to aid, define, or limit the entry and display of data. The number of

characters in the template string defines the maximum length of an animal identification. The maximum length of animal identifications is 10 characters. It is important to bear in mind possible future requirements and define the length of animal identifications so as to accommodate future potential needs.

All animal identifications are internally defined as character variables. No automatic justification to the right is done, even if only numerical values are used. Leading zeros are significant and will be maintained. To keep numerical values in the proper sorting order, justification of numerical values must be done manually during data entry, by using leading zeros.

If alphabetic characters are used as part of an animal identification, the English alphabet lower case letters will automatically be converted into their upper case equivalents. This restriction is imposed to avoid ambiguity in distinguishing animal identifications with respect to upper/lower case letters.

The following symbols are used to define a template:

- 9 - allows only digits (no blank spaces)
- A - allows only letters (no blank spaces)
- N - allows both letters and digits (but no blank spaces)

The character '-' (hyphen) can be defined as a separator. It will automatically be displayed in the position specified for it and will be part of the animal identification. Missing values for animal identifications will also contain such separators.

Examples of ID templates:

a) Template: 9999

Defines animal identifications as four digit numbers. Allows only the digits 0-9 to be entered. Smaller numbers than 1000 (i.e. having less than four digits) must be preceded by leading zeros, since no blank spaces are allowed.

b) Template: 99-NNNN

Defines animal identifications as alphanumeric strings seven characters long. The hyphen in the third place is automatically displayed and cannot be removed or edited. The first two positions are analogous with example (a), the last four positions may be any characters of the English alphabet or the digits 0-9. No blank spaces are allowed in any position and lower case characters are automatically converted to their upper case equivalent.

c) Template: AANNN

Defines animal identifications as alphanumeric strings five characters long. The first two positions allow only characters of the English alphabet. The last three positions allow characters of the English alphabet as well as the digits

0–9. No blank spaces are allowed in any position and lower case characters are automatically converted to their upper case equivalents.

Once the ID Format template has been defined and saved, it can be changed only if no database files have yet been created (using the **Create Database Files** option of the **Main Menu**). If database files have been created but no records have yet been added, the files can be detached using the option **Detach Database Files** of the **Main Menu**. The ID Format template may then be changed and the database files can be recreated, using the changed ID Format template.

Note: Once data have been entered into a data set, however, it is no longer possible to change the ID Format template without deleting all the data. Care should be taken to define the ID Format template in a way that can accommodate nonstandard identifications for unusual individuals that may be encountered in a data set, as well as possible future requirements.

- **Maximum Litter Size:** this allows you to define the maximum number of offspring born in a parturition. This feature allows structuring a data set according to the fertility of a particular animal species. The maximum number allowed is 30.

The Maximum Litter Size determines the structure of the Parturition Data File. This file contains the animal identification number of each individual offspring born in a parturition. The number of available fields per parturition for offspring identification numbers equals the Maximum Litter Size value. This value is also used for error checking during data entry, being the upper limit to the field 'Number of Animals born' in the Parturition Data file.

Once the Maximum Litter Size has been defined and saved, it can only be changed if the Parturition Data file has not yet been created (using the **Create Database Files** option of the **Main Menu**). If the Parturition Data file has been created but no records have yet been added, the file can be detached with the option **Detach Database Files** of the **Main Menu**. The Maximum Litter Size can then be changed and the Reproduction Data file recreated, on the basis of the changed Maximum Litter Size value.

Note: Once data have been entered into the Reproduction Data file, however, it is no longer possible to change the Maximum Litter Size without deleting all the data. The Maximum Litter Size should be defined large enough to accommodate all possible exceptions to normal litter sizes that may be encountered in a data set, as well as possible future requirements for larger numbers of offspring.

- Record Dead Animals (T/Y = true/yes, F/N = false/no): this allows you to specify whether or not animals that were dead at birth (aborted or stillborn) should be recorded with identification numbers of their own. If animals born dead are given individual identification numbers, they will be treated as individual animals in the data set and performance or observation records (e.g. weight at birth/abortion, sex, breed etc) that describe their individual characteristics can be recorded as well. If stillborn or aborted individuals are not given identification numbers, the only record of them will be in the Parturition Data file, in which the value of ‘Animals born’ will be larger than the value of ‘Animals born alive’.

In order to record an animal that was born dead, therefore, it must be assigned a unique identification number as any other animal recorded in the data set.

The setting of Record Dead Animals is used internally for error checking during entry of parturition data. The setting determines whether the number of animal identifications entered in a parturition record must equal the value of ‘Number of Animals born’ or ‘Number of Animals born alive’.

Note: Once data have been entered into the Reproduction Data file it is no longer possible to change the Record Dead Animals setting without deleting all the data. This should be born in mind when defining the setting for Record Dead Animals so that it is consistent with the structure of the data set and can accommodate possible future requirements.

- Epoch Year: this determines the interpretation of dates with no specified century digits. Date values in LIMS are usually entered with two year digits only and no century digits (e.g. 75, 80, rather than 1975, 1980), but they are stored with two additional century digits. If the year digits in a date form a number greater than or equal to the last two digits of the Epoch Year, the date is assumed to be within the same century as the Epoch Year. If not, the date is assumed to be in the following century.

Example: Epoch Year : 1940

Date Display Format : DD/MM/YY

| date value entered as | interpreted and stored as |
|-----------------------|---------------------------|
| 01/01/40              | 01/01/1940                |
| 25/12/54              | 25/12/1954                |
| 01/02/01              | 01/02/2001                |
| 11/11/11              | 11/11/2011                |
| 31/12/39              | 31/12/2039                |

Note: Once data have been entered into any of the data files the Epoch Year can no longer be changed without deleting all the data. The Epoch Year should therefore be set as to accommodate past and future data.

- Male, Female, Intersex: these are globally defined sex codes that can be used internally for error checking (e.g. only female animals can have parturition records). These codes have to be defined separately for the variable SEX in the General Animal Data file. For more information on how to define codes see the chapter **Code Definition**. The global definition of sex codes cannot be changed once records have been added to the General Animal Data file.
- Genetically Active, Was Genetically Active: these are globally defined codes for the field G\_ACTIVE of the General Animal Data file. The values are used internally for error checking (e.g. only genetically active animals may appear in a mating record after their disposal). Defining an animal as Genetically Active allows it to appear as parent in a mating or in a parturition record after its disposal, which would normally result in an error. This enables you to record animals as parents even after their death or disposal, if their offspring have been conceived by means of artificial insemination or embryo transfer. The codes must be defined separately for the variable G\_ACTIVE of the General Animal Data file. For more information on how to define codes see the chapter **Code Definition**. The global definition of the genetically active codes can only be changed so long as no records have been added to the General Animal Data file.
- Entry Due to Birth: this code can be globally defined for the field EREASON in the Environment Data file. Whenever an environment record is created for an animal that is born into the recording system, this code will be used. For more information see **Parturition Record** under **Add a New Record** in the UPDATE module. The code has to be defined separately for the variable EREASON in the Environment Data file. More information on how to define codes is given in the chapter **Code Definition**. The global definition of this entry reason code cannot be changed once records have been added to the Environment Data File.

### 3.2 File Definition

Definitions of certain data files are supplied along with the LIMS software. Some of these predefined files must be used with any LIMS data set, whereas others may be selected according to the needs of the user and the characteristics of the data in the data set. The definitions of both file categories can be modified by the user.



Additional files can be created by the user. The definition and content of these files are entirely user-defined, allowing you to create custom-made files to suit your personal requirements. User-defined files become part of a LIMS data set and are treated in the same way as predefined data files.

The following table lists the data files in a LIMS data set by category (xxxx = LIMS data set name):

| File category | DOS file name (SD) | LIMS file label (UC) | comment  |
|---------------|--------------------|----------------------|--|
| SD/SR         | xxxxHIS.DBF        | <not applicable>     | animal ID changes  |
| SD/UC/SR      | xxxxGEN.DBF        | General Animal Data  | animal passport data   |
|               | xxxxENV.DBF        | Environment Data     | animal environment changes                                     |
|               | xxxxPAR.DBF        | Parturition Data     |  |
| SD/US         | xxxxDOC.DBT        | <not applicable>     | free text information  |
| SD/US/UC      | xxxxEST.DBF        | Estrus Data          |  |
|               | xxxxMAT.DBF        | Mating Data          |  |
|               | xxxxWGT.DBF        | Weight Data          |  |
|               | xxxxMIL.DBF        | Milk Data            |  |
| UD/UC         | xxxxU01.DBF        |                      | user-defined files,<br>content according<br>to user definition |
|               | ...                | <user-supplied>      |  |
|               | xxxxU99.DBF        |                      |  |

SD = system-defined      SR = system-required  
 UC = user-configured      US = user-selected      UD = user-defined

The file definition information of the database files in a data set is stored in the file named xxxxFDL.DBF (xxxx = LIMS data set name). Each database file in a data set has its definition stored in one record of this file (xxxxFDL.DBF). The contents of the fields in such a File Definition record are specified either by the user or by the definition of database fields, or else the fields are updated by the UPDATE module. These fields are described in the following chapter.

### 3.2.1 Add a New File

Once the System Configuration has been completed, the definition of three system-required files, General Animal Data, Environment Data and Parturition Data, will automatically be carried over to the user data set definition from the supplied generic definition. These three files form the minimum data set and must be created using the **Create Database Files** option before any data can be entered through the UPDATE module.

The fields in these three files can be edited with the **Data Definition** option before the files are created.

The option **Add a New File** brings up a file selection list. Using this list you can carry over definitions of additional database files from the generic definition to your own data set definition. Files that already have definitions added are marked with a tick in the file selection list. The **Add a New File** option allows you to select those database files from the supplied data set definition that you need in your own data set. Additional database files can be defined and created at any time.

The last item in the file selection list is named **User-Defined File**. Up to 99 user-defined files can be defined and created in a data set. The structure of these files is wholly determined by the user. Fields in user-defined files can be defined using the **Data Definition** option. User-defined files become part of a data set and are treated in the same way as the distributed generic files. This feature allows you to create additional database files that are not contained in the distributed generic definition, e.g. files for animal traction, wool yield, nutrition and feeding, socio-economic observations, market observation, and so on.

Once a file has been selected from the file selection list, the fields **File Number**, **File Label** and **File Description** in the file definition record can be edited. The other fields that appear in the file definition record cannot be directly accessed by the user.

- File Number determines the order of files in a file selection list. This number is automatically assigned by the system as the order in which files are defined. It can be changed by the user to impose a different order on a file selection list. This may be useful during data entry through the UPDATE module, as frequently used or important files can be made to appear as the first items in a file selection list. The file number can be changed at any time using the **Edit a File** option in the **File Definition Menu**.
- File Label is the 20 characters-long internally used name of each database file in a LIMS data set. The file names that appear in file selection lists are normally File Labels and not the DOS file names. Users can change the default file labels for the supplied generic files and define new file labels for User-Defined Files.

- File Description: this is a 254 character free text field, that allows you to store a description of the file content, structure, earliest observation data, location of the raw data, frequency of observation, etc. This description enables users who are not familiar with the data set to understand the content and use of a particular database file within a data set.
- The cursor can have three different shapes in the edit window of the File Description field. A full block cursor indicates Browse Mode, in which you can switch between fields with the UpArrow and DownArrow keys. The File Definition record can be saved by pressing the Return key. Pressing any character key switches from Browse Mode to Edit Mode, which is indicated either by a half block or by an underline cursor. The half block cursor is shown when insert mode is active, and the underline cursor when overwrite mode is active.

The following keys are used to manipulate text within the active text window during **Edit Mode**.

|                      |   |
|----------------------|---|
| Uparrow/Dnarrow      | - moves the cursor up/down by one line              |
| Rightarrow/Leftarrow | - moves the cursor right/left by one column         |
| PgUp/PgDn            | - scrolls text up/down                              |
| Ctrl-PgUp/Ctrl-PgDn  | - moves the cursor to the beginning/end of the text |
| Home/End             | - moves the cursor to the beginning/end of the line |
| Return               | - inserts hard carriage return                      |
| <character>          | - enters <character>                                |
| F10                  | - saves the File Definition record                  |
| Esc                  | - abandons editing                                  |

If an attempt is made to save a File Description text that exceeds the maximum permitted length of 254 characters, the error message **ERROR - FILE DESCRIPTION LENGTH EXCEEDS LIMIT BY <n> CHARACTER(S)!** will be displayed.

- File Name Code: this is a system-defined extension code three letters long that is added to the LIMS data set name. The resulting unique name is the DOS filename of the data set. The File Name Codes for User-Defined Files are U01-U99. The File Name Code is assigned by the system and cannot be changed by the user.

- The **File Create Date** is updated to the current system date when a database file is created with the **Create Database Files** option, and is assigned a blank date if the file is detached with the **Detach Database Files** option. If the File Create Date in a File Definition record contains a valid date, the system will assume that the database file has been created. If instead this field contains a blank date, the system will assume that the database file has not yet been created or that it has been detached after creation. For further information see the chapters **Create Database Files** and **Detach Database Files**.
- Data Last Updated and **Number of Records** are automatically updated by the UPDATE module and cannot be accessed by the user. The field Number of Records is set to zero when a file is detached with the **Detach Database Files** option.
- The **File Deactivate Date** is updated to the current system date when a database file is deactivated with the **Deactivate a File** option, and is assigned a blank date when the database file is activated with the **Activate a File** option. For further information see the chapters **Deactivate a File** and **Activate a File**.
- Number of Key Fields, **Number of Fields**, and Record Length are updated when a file is created with the **Create Database Files** option. The values contained in these fields are determined by the definition of the database fields specified in the Data Definition. The field **Number of Key Fields** contains the number of fields in the database file that make up the unique index key for sorting and retrieval. The Number of Key Fields is predefined for all database files, except for the User Defined Files, and cannot be changed by the user. The field **Number of Fields** contains the number of all fields in a database file. The field **Record Length** contains the sum of the length of all fields in the database file.

### 3.2.2 Edit a File

The fields **File Number**, **File Label** and **File Description** of a File Definition record can be modified at any time. For that purpose the option Edit a File allows you to access the File Definition record of any database file that exists in the data set definition. The option Edit a File brings up a file selection list of existing file-definition records. The fields of a file definition record that can be edited are the same as those described under **Add a New File**.

### 3.2.3 Delete a File

With this option you can remove a File Definition record from a user data set definition. If a database file has been created with the **Create Database Files**

option, its file definition cannot be deleted. Attempting to do so will result in the error message **ERROR - DATA FILE EXISTS. DEFINITION RECORDS CANNOT BE DELETED!** being displayed.

In order to delete File Definition Records of created database files, the files must first be detached using the **Detach Database Files** option. Note that detaching a database file and deleting its File Definition record causes the data and the documentation of the database file to be lost. Do not, therefore, use this option if you wish to retain the file data.

### 3.2.4 Activate a File

This option activates a database file that has been deactivated with the option **Deactivate a File**. The Deactivate Date in the File Definition record will be assigned a blank date.

A file with a blank date in the Deactivate Date field of the File Definition record is assumed to be active and will be included in global sorting, packing, indexing, and validation processes.

### 3.2.5 Deactivate a File

This option deactivates a database file. The Deactivate Date in the File Definition record will be set to the current system date.

A file with a valid date in the Deactivate Date field of the File Definition record is assumed to be deactivated and thus will not be included in any of the global sorting, packing, indexing, or validation processes.

If one of the three system-required files (General Animal Data, Environment Data or Parturition Data) is deactivated, the UPDATE module cannot run, and the message **ERROR - <filename> FILE IS INACTIVE OR NOT CREATED!** will be displayed.

## 3.3 Data Definition

With this option you can change the definition of fields supplied in the predefined generic LIMS data set and define additional fields to be added to your personal database definition.

The Data Definition Menu displays the 5 Data Definition Menu options in a window in the centre of the screen. The currently selected option is shown in a different colour.

The following table shows the keys used with the Data Definition Menu:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>Main Menu</b> level             |

The data definitions of the fields in a database file are stored in a file with the name xxxxDDL.DBF (xxxx = LIMS data set name). The definition of each field in a database file is stored in one record of the file xxxxDDL.DBF. Whether the fields in such a Field Definition record can be accessed by the user depends on whether the database file has been created and whether the field is System-Defined (SD), User-Configurable (UC) or User-Defined (UD). Field Definition record fields are described in the chapter **Add a New Field**.

### 3.3.1 Select a Data File

This option displays a File Selection List with File Name Codes and File Labels of all database files that have been defined with the **File Definition** option. Selecting a database file allows further processing of fields in this file. Selected files are marked with a tick indicating that the file has been processed. The file marker is removed upon return to the Main Menu.

The keys below are used with the file selection list and have the following functions:

|         |  |
|---------|--|
| Dnarrow | - selects the following file                     |
| Uparrow | - selects the preceding file                     |
| PgDn    | - advances the file selection list by one page   |
| PgUp    | - moves the file selection list back by one page |
| Return  | - activates the selected file                    |
| Esc     | - returns to the <b>Data Definition Menu</b>     |

A file must be selected before any of the other options of the **Data Definition Menu** can be executed. If an attempt is made to run any of the other options of the Data Definition Menu without first electing a file, the message **ERROR – A FILE MUST BE SELECTED TO PROCESS FIELDS!** will be displayed.

### 3.3.2 Add a New Field

New fields can only be added to a data file definition so long as the data file itself has not been created with the **Create Database Files** option. Attempting to add a new field to the data definition of an already created data file will cause an error and the message **ERROR - DATA FILE EXISTS. DEFINITION RECORDS CANNOT BE ADDED!** will be displayed.

For data files belonging to the generic data set definition, a selection list with predefined fields will be displayed. Using this list you can add a predefined field to your own data file definition. The last field in the selection list is user-defined, allowing you to individually define additional fields. Predefined fields that have already been added to the user data file are indicated by a tick marker in the field selection list. Attempting to add a marked field to the data definition of a data file will result in an error and the message **ERROR - FIELD EXISTS. CANNOT ADD FIELD DEFINITION RECORD!** will be displayed.

When adding predefined or user-defined fields to a data file, certain fields in the Data Definition record are generated automatically by the system whereas some can be changed by the user. The following section describes the contents of the fields in a Data Definition record and how they can be accessed.

- **Data File:** The file label of the data file selected with the option **Select a Data File** under **Data Definition**.
- **Field Name:** Displays the predefined field name if a predefined field has been chosen from the Field Selection List, or is empty and can be edited if the field is user-defined. The field name can be up to 10 characters long, but for the sake of compatibility with other software it is recommended not to use more than 8 characters. If a field name is longer than 8 characters the user will be alerted by the warning message **WARNING - WIDTH OF FIELD NAME MAY NOT BE COMPATIBLE WITH OTHER SOFTWARE!** being displayed. The field name can be composed of letters of the English alphabet, digits, and underscores, but the first character of a field name must be an alphabetic character and the last character may not be an underscore. Lower case characters are automatically converted to their upper case equivalents.

The field definition of a field which is defined in another database file can be reused by typing the Field Name. The existing field definition will be

displayed and can be edited. Certain restrictions apply for changing the field definition, depending on whether the file with the original field has already been created and on whether the original field is system-defined, user-configurable or user-defined. If the Field Usage, Field Length or Field Decimals can be changed, any changes of these will be transferred to the original field definition as well.

- Field Label: Displays the predefined field label if a predefined field has been chosen from the Field Selection List or displays <Undefined> if the field is user-defined. This field can be edited at any time, regardless of whether the associated data file has been created or not. The length of the field label is restricted to 15 characters. Any characters that can be printed are allowed, including spaces. The field label is displayed during data entry and may contain additional information about the units of field values or the different field codes used, in order to facilitate data entry, e.g. 'BodyWeight (kg)' or 'Sick? (Y/N/U)'
- Field Usage: The F2 key brings up a selection list with 9 different Field Usage codes. The field usage code determines how a certain field is handled in the LIMS environment. It also determines Field Type and Field Picture and, with a few exceptions, it determines Field Length and Field Decimals. The different Field Usage codes are described in detail below.

**NF** Numeric Field. Usually selected to record values of continuous variables such as weight, milk yield, etc. For numeric fields, the Field Length and the Field Decimals can be defined by the user. The maximum Field Length of numeric fields is 16. If decimal places are defined, their number cannot exceed the Field Length - 2. Minimum, Maximum, and default Missing Values can be defined by the user.

**ID** Identification-Type Field. Such fields can be defined to store animal identification numbers. Identification-Type Fields conform to the global ID format as specified with the **System Configuration** option. Field Length and Field Picture are determined by the global ID format. User-defined Identification-Type Fields are treated in the same way as system-defined Identification-Type Fields. During data entry, the system checks whether an animal identification entered into an Identification-Type Field has a corresponding record in the General Animal Data file. The option **Change ID Numbers** of the UPDATE module checks all Identification-Type Fields, whether system-defined or user-defined, to determine whether to replace the old ID number with a new one. The Minimum and Maximum Values in a Data Definition record take on a special meaning in this context, in that they can be used to restrict the range of animals recorded in a particular Identification-Type Field to those having certain sex



codes. Thus the ID field in the Milk Data file can be restricted to permit only ID numbers belonging to female animals.

- C1** Code-Type Field Upper Case. This code is usually selected to record values of categorical variables. The Field Length of Code-Type fields can be set by the user but is limited to 10 characters. All characters that can be printed are allowed, but this kind of Code-Type Field converts lower case letters to their upper case equivalents. The option **Code Definition** allows you to define the range of codes of a Code-Type Field and to store them in the Code Definition Library. During data entry entered codes are checked to determine whether they are defined in the Code Definition Library. The Code Definition Library is used to define, document, and perform error checks on categorical variables.
- C2** Code-Type Field Mixed Case. As C1 described above, but lower case characters are not converted to upper case during data entry. Upper and lower case versions of alphabetic characters are distinguished, e.g. the code values 'ET', 'Et', 'eT' and 'et' are treated as four different codes.
- DR** Relative Date Field. The date contained in this field must be within the life-span (birth date <= relative date <= disposal date) of the animal that it refers to. The reference animal is determined by the value of the first ID-Type field in the data record where the Relative Date is defined. Example: a daily milk record consists of the fields ANIMAL ID, MILKING DATE, AM YIELD and PM YIELD. The field MILKING DATE can be defined as a Relative Date Field, since the value of MILKING DATE must be within the life-span of the animal recorded under ANIMAL ID. Checking that the date falls within the defined limits is carried out during data entry. Such a check is only done if a reference animal can be determined and if either the birth date, or the disposal date, or both dates, contain non-missing values. Minimum and Maximum Values in this context determine the age range of the animal within which the Relative Date must be. On date fields an internal check is carried out to ensure that only valid dates of the Gregorian Calendar are entered.
- DA** Absolute Date Field. The dates contained in this field are not checked for reference to an animal. Minimum and Maximum Values specify absolute date values defining the permitted range for an Absolute Date Field.
- TX** Free Form Text Field. This type of field allows you to enter unstructured text. The Field Length can be set by the user but may not exceed 254 characters. All characters that can be printed are

allowed. If the length of a Free Format Text Field exceeds 18 characters, it will be displayed during data entry as a horizontally scrolled field 18 characters wide.

- T1** Time Field. This type of field is formatted to allow the entry of time values having the format hh:mm (h = hour, m = minute). Entered values for hours and minutes may only consist of digits.
- T2** Time Field. This type of field is formatted to contain time values having the format hh:mm:ss (h = hour, m = minute, s = second). Entered values may only consist of digits.
- Field Type: The LIMS system supports four Field Types as defined by the dBASE standard. The Field Type is determined by the selection of Field Usage code. The four supported Field Types are described in detail below.
- N** Numeric Field. Numeric field type can be selected with the Field Usage code NF. This field type allows only digits, a decimal point and a leading sign to be entered. Values are justified to the decimal point or, if there is no decimal point, they are justified to the right. The decimal point and the sign (if any) are included in the Field Length count of Numeric Fields.
- C** Character Field. The Field Usage codes ID, C1, C2, TX, T1 and T2 select Character Field type. Generally Character Fields may be used to store any character that can be printed and entered via the keyboard, including blank spaces. However, certain restrictions apply depending on the Field Usage code. These restrictions are described under the descriptions of each Field Usage code.
- D** Date Field. The Field Usage codes DR and DA determine the field type as Date Field. The values entered into Date Field must be valid dates of the Gregorian Calendar, no missing or zero day, month or year values are allowed, although the entire date may be blank. Century digits are not entered or displayed, but are stored. The century of a date is determined as described under Epoch Year in the chapter System Configuration. The date display format can be changed to conform to different national and international standards with the System Configuration option.
- M** Memo Field. Memo Fields are designed to accommodate large blocks of text information. They are stored in an auxiliary file (.DBT) to the database file. The only Memo Field in the LIMS system is the system-defined field DOC\_BODY (Document Body) in the file XXXXDOC.DBF (Documentation). No additional Memo Fields can be defined for a LIMS data set.

- **Field Length:** The Field Length is either globally defined by the user, user-defined for each field, or system-defined. The following table shows the possible ranges of Field Length and the definition status depending on Field Usage code.

| Field Usage | Field Length | Field Length Definition Status |
|-------------|--------------|--------------------------------|
| NF          | 1 - 15       | user-defined, by field         |
| ID          | 1 - 10       | user-defined, globally         |
| C1          | 1 - 10       | user-defined                   |
| C2          | 1 - 10       | user-defined                   |
| DR          | 8            | system-defined                 |
| DA          | 8            | system-defined                 |
| TX          | 1 - 254      | user-defined                   |
| T1          | 5            | system-defined                 |
| T2          | 8            | system-defined                 |

If decimals are defined for a numeric field (NF), the Field Length must be greater than or equal to (the number of decimals) + 2. If the Field Length of a Free Format Text Field (TX) exceeds 18 characters, it is displayed during data entry as a horizontally scrolled field 18 characters wide.

- **Field Decimals:** Decimal places can only be defined for numeric fields (NF). If they are defined their number must be less than the Field Length -2. Changes can only be made so long as the respective data file has not been created.
- **Field Picture:** The Field Picture is determined automatically by the Field Usage code and, if applicable, by the Field Length and Field Decimal values.
- **Field Category:** The Field Category is set by the system and determines to what extent users can change a Field Definition record. There are five possible Field Category codes which are described in detail below.

The extent to which a Field Definition record can be changed also depends on whether the data file that the field belongs to is a system-defined (SD) or user-defined (UD) data file, and on whether the file has been created or not, using the option **Create Database Files**.

**SD** System-Defined Field. The definition of this type of field is distributed as part of the generic LIMS data set definition. The data definition record of a field belonging to category SD cannot be deleted from a system-defined data file.

The Field Label and the Field Description can be changed at any time, as can the Minimum Value and Maximum Value, if applicable to the field in question.

The Field Number can be changed for all fields except index fields of system-defined files, so long as the corresponding file has not been created.

The Missing Value applies only to numeric fields and can be changed so long as the corresponding file has not been created. The Allow Missing flag can be changed at any time, but only for non-index fields.

FKey Editable can only be changed for non-index fields in user-defined data files.

The Index Field can only be changed for fields belonging to user-defined data files so long as the corresponding data file has not been created. If the Index Field setting is changed to T (true/yes), Allow Missing and FKey Editable will automatically be set to F (false/no).

The Field Usage code cannot be changed for system-defined (SD) and user-selectable (US) fields.

**US** User-Selectable Field. The definition of this type of field is distributed as part of the generic LIMS data set definition. However, as opposed to a system-defined field, the data definition record of a field belonging to category US can be deleted from a system-defined data file. This allows the user to delete field definition records that are not relevant to a particular data set (e.g. WEANING DATE, if no weaning is practiced). Apart from this difference, User-Selectable Fields obey the same rules as System-Defined Fields with respect to permitted changes in the field definition record.

**SDUC** System-Defined User-Configurable Field. The definition of this type of field is distributed as part of the generic LIMS data set definition. However, in addition to the permitted changes in the field definition record of System-Defined (SD) fields, users can configure the Field Length and, if applicable, the Field Decimals, so long as the corresponding data file has not been created. This allows you to customise the length of system-defined code-type fields and the length and number of decimals of system-defined numeric fields for a particular data set (e.g. BREED, WEIGHT).

**USUC** User-Selectable User-Configurable Field. The definition of this type of field is distributed as part of the generic LIMS data set definition. It conforms to the definition of a System-Defined User Configurable (SDUC) field, except that the data definition record of a field belonging to category USUC can be deleted from a system-defined data file.

**UD** User-Defined Field. User-Defined Fields are not part of the distributed generic LIMS data set definition. They are created and fully configured by the user, within the limitations described under **Field Category**. Once the corresponding data file has been created, only the following fields in a field definition record can be changed: Field Label, Minimum Value, Maximum Value, Allow Missing, FKey Editable and Field Description.

- **Index Field:** The Index Field in a data definition record can only be changed for fields belonging to user-defined files. A value of T (true/yes) implies that the field forms part of the index key used to access data records during data entry with the UPDATE module. All Index Fields together make up the index key which must allow every data record to be individually accessed. Index fields always appear as the first fields of a data record. The Field Number can only be used to change the position within the Index Fields of a data file. A maximum of 9 index fields can be defined for a data file. The settings for Allow Missing and Fkey Editable will automatically be set to F (no/false) if the Index Field is set to T (yes/true).
- **Field Number:** The Field Number allows rearrangement of the order of fields as they appear on the data entry screen. This is useful if users want to insert user-defined fields between existing fields or rearrange existing fields so that they match the raw data sheet more closely, thus facilitating data entry. The Field Number can only be changed so long as the corresponding file has not been created. Since Index Fields always appear as the first fields in a data record, the Field Number allows rearrangement only within index fields or within non-index fields. Index fields and non-index fields cannot be mixed in a data record. The order of index fields in system-defined files cannot be changed with the Field Number setting.
- **Minimum Value / Maximum Value:** Minimum and Maximum Values specify the upper and lower limits of a field value. Entered data are checked to ensure that the entered values fall within the specified range. The value specified for Missing Value need not be included in this range. Minimum and Maximum Values are used only with fields having certain Field Usage codes and have different meanings depending on the Field Usage.

**NF** Numeric Field. For Numeric Fields, the minimum and maximum values can be entered directly and define the range within which a

data value can fall. Data are checked when entered. This check is only performed if either the Minimum Value or the Maximum Value have non-zero values and the Maximum Value is larger than the Minimum Value.

**ID** ID-Type Field. For an ID-Type Field, the Minimum Value and Maximum Value can be used to restrict the range of the animal identifications that can be entered in that field to include only those having certain sex codes. Sex codes are checked during data entry. Instead of entering the values directly into the Minimum Value or Maximum Value fields, a window allows setting restrictions on permitted sex codes by assigning a T (yes/true) or F (no/false) value for the sex codes Male, Female, Intersex and all other sex codes. These settings are translated into numerical Minimum and Maximum Values according to the table below.

| Male | Female | Minimum Value | Intersex | Other Sex Code | Maximum Value |
|------|--------|---------------|----------|----------------|---------------|
| Y    | Y      | 0             | 5        | Y              | Y             |
| Y    | N      | 1             | 5        | Y              | N             |
| N    | Y      | 2             | 5        | N              | Y             |
| N    | N      | 3             | 5        | N              | N             |

The default setting is to allow all sex codes for ID-Type fields, which is represented by Minimum Value and Maximum Value both being zero.

**DR** Relative Date Field. For Relative Date Fields, the Minimum and Maximum Values specify the age range of the reference animal. The reference animal is determined by the first ID-Type field that appears in the data record where the Relative Date is defined. Instead of entering the values directly into the Minimum Value or Maximum Value fields, the minimum and maximum age restrictions are entered through a separate window. The minimum and maximum age values are given in days and stored as such in the Minimum and Maximum Value fields. A corresponding check is carried out during data entry. This check is only performed if either the Minimum Value or the Maximum Value have a non-zero value and the Maximum Value is larger than the Minimum Value.

**DA** Absolute Date Field. For Absolute Date Fields, the Minimum Value and Maximum Value specify the absolute permitted date range of an entered date. The values are entered through a separate window as

minimum date (earliest date) and maximum date (latest date) restrictions. The minimum and maximum dates are converted into numerical values and stored in the Minimum Value and Maximum Value fields. A corresponding check is carried out during data entry. This check is only performed if both the Minimum Value and the Maximum Value have non-zero values and the Maximum Value is larger than the Minimum Value.

- Missing Value: Only numeric fields can have a missing value, and zero is normally the default missing value. This can lead to ambiguity if zero is also a valid measurement (e.g. temperature in centigrade, milk yield, parasite count). There are three possible settings for the default missing value of a numeric field:

| Missing Value Setting | Default Missing Value  |
|-----------------------|------------------------|
| 0                     | zero                   |
| +9                    | largest positive value |
| -9                    | largest negative value |

The actual values of the largest positive and largest negative values depend on the field length of the numeric field in question. Pressing the F2 function key displays the selection list with the missing value settings.

If no value has been entered into a numeric field, it is assigned the default missing value, which is displayed. If no other value is entered, the default missing value will be stored.

Definitions of default missing values can only be changed so long as the corresponding data file has not been created.

**Allow Missing:** The Allow Missing flag can be changed at any time, but only for non-index fields. A check is carried out during data validation to determine whether the field value is missing, which means having the Default Missing Value for numeric fields, containing a 'blank' for character fields, or having an empty date for date fields. If the Allow Missing flag is set to F (no/false) and the field value is missing, an error will be reported during data validation. For index fields, the Allow Missing flag is automatically set to F (no/false).

**FKey Editable:** The FKey Editable flag can be changed at any time, but only for non-index fields in user-defined data files. This flag determines whether a field can be edited during a Function Key Edit process. For more information see the chapter Function Key Edit in the UPDATE module.

**Field Description:** Field Description is a 254 character free text field, where you can enter a description of the field content, method of measurement, precision, etc. This description enables a user who is not familiar with the data set to understand the content and use of a particular data field within a data file.

The cursor can have three different shapes in the edit window of the File Description field. A full block cursor indicates Browse Mode, in which you can switch between fields using the UpArrow and DownArrow keys. The File Definition record is saved by pressing the Return key. Pressing any character key switches from Browse Mode to Edit Mode, which uses either a half block or underline cursor. The half block cursor is shown when insert mode is active, and the underline cursor is shown when overwrite mode is active.

The following keys are used with the text edit window in **Edit Mode**.

|                      |   |
|----------------------|---|
| Uparrow/Dnarrow      | - moves the cursor up/down by one line              |
| Rightarrow/Leftarrow | - moves the cursor right/left one column            |
| PgUp/PgDn            | - scrolls text up/down                              |
| Ctrl-PgUp/Ctrl-PgDn  | - moves the cursor to the beginning/end of the text |
| Home/End             | - moves the cursor to the beginning/end of a line   |
| Return               | - inserts hard carriage return                      |
| <character>          | - enters <character>                                |
| F10                  | - saves the File Definition record                  |
| Esc                  | - abandons editing                                  |

If an attempt is made to save a Field Description that is more than 254 characters long, the message ERROR - FIELD DESCRIPTION LENGTH EXCEEDS <n> LIMIT BY <n> CHARACTER(S)! will be displayed.

### 3.3.3 Edit a Field

Existing field definitions can be edited with this option. A field selection list with the Field Name and Field Label of the active file is displayed and allows you to select the field to be edited. The edit process obeys the same rules as the option **Add a New Field**, except that the item Field Name in the Field Definition record cannot be accessed.



### 3.3.4 Delete a Field

This option displays a Field Selection List of the Field Names and Field Labels of the currently selected database file. Selecting a field and confirming the prompt destines the field definition for deletion. The field still appears on the Field Selection List, but is marked with a tick. The actual deletion is performed upon return to the Data Definition Menu.

If a field that is marked for deletion is selected and the ensuing prompt confirmed the field will be undeleted. The tick mark on the Field Selection List will be removed and the field will not be deleted upon return to the Data Definition Menu.

A database file must first be selected with the **Select a Data File** option; otherwise the error message ERROR - A FILE MUST BE SELECTED TO PROCESS FIELDS! will be displayed.

Once a database file has been created with the **Create Database Files** option, Field Definition Records of that file cannot be deleted. Attempting to do so will result in an error and the message ERROR - DATA FILE EXISTS. DEFINITION RECORD CANNOT BE DELETED! will be displayed. Before the Field Definition Record of a created database file can be deleted, the database file must be detached with the **Detach Database Files** option.

Certain fields belonging to predefined database files in the generic data set definition are system-required and cannot be deleted. If an attempt is made to delete such a field, the message ERROR - SYSTEM REQUIRED FIELD. DEFINITION RECORD CANNOT BE DELETED! will be displayed.

### 3.3.5 Preview Fields

This option displays an empty record of the selected database file in the central display area, as it will appear during data entry with the UPDATE module.

If no database file has been selected with the **Select a Data File** option, the message ERROR - A FILE MUST BE SELECTED TO PROCESS FIELDS! will be displayed.

Fields are displayed in two columns, each with a maximum of 19 fields. The field labels and data entry areas are separated by filling underline characters. The index fields on top of the left column are displayed in a different colour or shade. The data entry areas display separator characters and default missing values if applicable.

The **Preview Fields** option allows you to check that index fields, the order of fields, the field labels, the size of the data entry areas and the default missing

values have been correctly defined. A careful preview of the defined fields is recommended before a database file is created with the **Create Database Files** option.

### 3.4 Code Definition

With this option you can define codes for all code-type fields in a data set. Codes can be defined and edited at any time, so long as code-type fields have been defined with the options **File Definition** or **Data Definition**.

The Code Definition Menu displays the 4 Code Definition Menu options in a window in the centre of the screen. The currently selected option is displayed in a different colour.

The following table shows the keys used with the Code Definition Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>Main Menu</b> level             |

The first step in processing a code is to Select a Code Field. No other option from the Code Definition Menu is available until a code field has been selected.

A code definition consists of the code value, a code label and a code description. The maximum length of a code value is 10, of a code label 15 and of a code description 60 characters. Code values must be defined, whereas code labels and descriptions are for display and documentation purposes and are optional.

When data are entered into a code-type field with the UPDATE module, the system checks whether the entered code has been defined. If a code has not been defined, the system will issue a warning. During the validation process

all values of code-type fields will be compared to the code definitions and undefined codes will be reported as errors.

### 3.4.1 Select a Code Field

This option displays a list of the field names and user-assigned labels of all code-type fields defined in the data set. The code-type fields are displayed in alphabetical order with respect to the field name. Selecting a code-type field allows subsequent processing of codes in the field. The current selection is shown with different background colour.

The keys below have the following functions in the field selection list:

|         |   |
|---------|---|
| Dnarrow | - selects the following field                     |
| Uparrow | - selects the preceding field                     |
| PgDn    | - advances the field selection list by one page   |
| PgUp    | - moves the field selection list back by one page |
| Return  | - activates the selected field                    |
| Esc     | - returns to the <b>Code Definition Menu</b>      |

Up to 10 field names and labels can be displayed on one page of the field selection list. If there are more than 10 code-type fields in a data set, a set of up and down arrows will be displayed on the line below the last field label. The field selection list will scroll when the selection is moved past the last selection displayed.

### 3.4.2 Add a New Code

This option allows you to define a new code value, label and description for the selected code-type field.

If no code-type field has been selected the error message **ERROR - A CODE TYPE FIELD MUST BE SELECTED TO PROCESS CODES!** will be displayed.

An attempt to enter a duplicate code value will cause the error message **ERROR - DUPLICATE CODE VALUE!** to be displayed.

Empty code values are not permitted.

### 3.4.3 Edit a Code

This option allows editing an existing code value, label and description for the selected code-type field. A code selection list displays all the codes defined for the selected code-type variable. The current selection is shown with different background colour.

The following keys are used with the code selection list:

|         |   |
|---------|---|
| Dnarrow | - selects the following code                      |
| Uparrow | - selects the preceding code                      |
| PgDn    | - advances the code selection list by one page    |
| PgUp    | - moves the field selection list back by one page |
| Return  | - activates the selected code                     |
| Esc     | - returns to the <b>Code Definition Menu</b>      |

If no code-type field has been selected the error message ERROR - A CODE TYPE FIELD MUST BE SELECTED TO PROCESS CODES! will be displayed.

Entering a duplicate code value will cause the error message ERROR - DUPLICATE CODE VALUE! to be displayed. Empty code values are not permitted.

### 3.4.4 Delete a Code

This option allows you to delete an existing code value, label and description for the selected code-type field. A code selection list displays all the codes defined for the selected code-type variable. The current selection is shown with different background colour.

The following keys are used with the code selection list:

|         |   |
|---------|---|
| Dnarrow | - selects the following code                      |
| Uparrow | - selects the preceding code                      |
| PgDn    | - advances the code selection list by one page    |
| PgUp    | - moves the field selection list back by one page |
| Return  | - activates the selected code                     |
| Esc     | - returns to the <b>Code Definition Menu</b>      |

If no code-type field has been selected the error message ERROR - A CODE TYPE FIELD MUST BE SELECTED TO PROCESS CODES! will be displayed.

Before a code definition is deleted a prompt will ask you to confirm the deletion.

### 3.5 Breed Definition

With this option you can define how paternal and maternal breed codes should be combined to yield offspring breed codes. All breed codes used in a data set can be defined in terms of combinations of their parental breed codes. During data entry with the UPDATE module, the system will propose a breed code for a newborn offspring if the dam and the sire are known, and the manner in which their breed codes should be combined has been defined as a rule. During validation the breed codes of offspring are checked to see whether they conform to the rules given in the breed definition.

The Breed Definition Menu displays the two menu options in a window in the centre of the screen. The currently selected option is displayed in a different colour.

The following table shows the keys used with the Breed Definition Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>Main Menu</b> level             |

Before a breed rule can be defined, all breed codes involved must be defined with the **Code Definition** option.

### 3.5.1 Add a New Breed Rule

This option allows you to specify a new combination of sire and dam breeds and the resulting progeny breed. Only the fields SIRE BREED, DAM BREED and PROGENY BREED can be edited. New breed values can be entered directly or carried over from a breed code selection list that is displayed by pressing the F2 key. The currently selected option is displayed in a different colour.

The following keys are used with the breed code selection list:

|         |  |
|---------|--|
| Dnarrow | - selects the following breed code                     |
| Uparrow | - selects the preceding breed code                     |
| PgDn    | - advances the breed code selection list by one page   |
| PgUp    | - moves the breed code selection list back by one page |
| Return  | - activates and transfers the selected breed code      |
| Esc     | - returns to the add screen                            |

Whether a breed code is entered directly or is selected from the breed code selection list, its code value, corresponding label and description will be displayed. If a breed code value is entered directly that has not been defined with the **Code Definition** option, the message ERROR - THIS BREED CODE IS UNDEFINED! will be displayed.

If a new breed rule is entered that has the same SIRE BREED and DAM BREED values as an existing breed rule, the message ERROR - DUPLICATE BREED RULE! will be displayed.

After editing the PROGENY BREED or pressing the F10 function key, a prompt will request confirmation before the new breed rule is saved.

### 3.5.2 Delete a Breed Rule

This option allows you to select an existing breed rule for deletion. The breed rule can be selected from a list of breed rules. The currently selected option is displayed in a different colour. The first column in the breed rule selection list displays the sire breed codes, the second column the dam breed codes and the last column the offspring breed codes.

The following keys are used with the breed rule selection list:

|         |  |
|---------|--|
| Dnarrow | - selects the following breed rule                     |
| Uparrow | - selects the preceding breed rule                     |
| PgDn    | - advances the breed rule selection list by one page   |
| PgUp    | - moves the breed rule selection list back by one page |
| Return  | - activates the selected breed rule                    |
| Esc     | - returns to the <b>Breed Definition Menu</b>          |

After selecting a breed rule from the breed rule selection list a prompt will request confirmation before the rule is deleted.

### 3.6 Create Database Files

With this option you can create database files based on the definitions given with the **File Definition** and **Data Definition** options. Data can only be entered (using the UPDATE module) into files that have been created with this option.

When a database file is created, the **File Create Date** will be set to the DOS system date.

The file to be created can be selected from a displayed file selection list. The current selection is shown with different background colour.

The following keys are used with the file selection list:

|         |  |
|---------|--|
| Dnarrow | - selects the following file                     |
| Uparrow | - selects the preceding file                     |
| PgDn    | - advances the file selection list by one page   |
| PgUp    | - moves the file selection list back by one page |
| Return  | - creates the selected file                      |
| Esc     | - returns to the <b>Main Menu</b>                |

Database files that have been created are marked with a tick in the File Selection List. Attempting to recreate a file will result in an error and the message ERROR - THIS FILE HAS BEEN CREATED! will be displayed. When the **Create Database Files** option is called, files that have been created no longer appear in the File Selection List.

If any one of the three system-required files, General Animal Data, Environment Data, or Parturition Data, is not created, the UPDATE module will not run. The error message ERROR - <filename> FILE IS INACTIVE OR NOT CREATED! will be displayed.

When the General Animal Data file is created, an additional file xxxxHIS.DBF (xxxx = user-defined data set name) is also created. This file contains the changes of animal identifications.

### 3.7 Detach Database Files

This option allows you to detach a database file from the data set. Detached database files have a hyphen added in front of the file name and can no longer be used in the system. Although a detached database file is removed from the data set, its definition stays intact and can still be accessed. Detached database files can be recreated with the Create Database Files option.

The purpose of this option is to allow the removal of a database file from a data set even if the database file contains data. The definition of the database file can thereafter be changed by adding, editing or deleting fields with the **Data Definition** option whereupon the database file can be recreated with the **Create Database Files** option. This process is useful when the structure of a data set is being developed.

When a database file is detached, the **File Create Date**, **Data Last Updated Date**, **File Deactivate Date** and **Number of Records** are set to blank or zero.

The file to be detached can be selected from a displayed file selection list. Only files that have been created with the **Create Database Files** option appear on the file selection list. The current selection is shown with different background colour.

The following keys are used with the file selection list:

|         |  |
|---------|--|
| Dnarrow | - selects the following file                     |
| Uparrow | - selects the preceding file                     |
| PgDn    | - advances the file selection list by one page   |
| PgUp    | - moves the file selection list back by one page |
| Return  | - detaches the selected file                     |
| Esc     | - returns to the <b>Main Menu</b>                |



Before a file is detached a warning is shown and a prompt requests confirmation. If the detached file already exists, a further warning is issued and a confirmation is required to erase the existing detached file.

When the General Animal Data file is detached, the file xxxxHIS.DBF (xxxx = user- defined data set name) is also detached. This file contains the changes of animal identifications.

WARNING: Detaching a database file will generally lead to the loss of the data in that database file.

### **3.8 Print Definitions**

This option allows you to make a printout of the entire data set definition. The printout can be sent directly to the printer or to a file for subsequent editing and printing.

#### **3.8.1 Print to Printer**

This option prints the data set documentation on a printer.

#### **3.8.2 Print to File**

This option requests a filename as input and prints the data set definition to the specified file. The filename may include a drive designator and a path. If no file extension is specified, the default extension .PRN will be given. The default filename for the data set definition file is xxxxDEF.PRN (xxxx = user-defined data set name). The output file is in ASCII format and can be used with a word processor for further editing and printing.

## 1 Introduction

The LIMS UPDATE module is written in the CLIPPER version of the dBASE language and compiled with the CLIPPER 5.0 compiler. The data set files are compatible with the dBASE III+ standard, except for the index files (.NTX), which are CLIPPER specific.

The UPDATE module is the only part of the LIMS system that can modify (input and edit) raw data. Data entry and modification comprises adding new records to a file, changing field values in existing records and deleting records in a file.

A LIMS data set consists of several files of different categories. All data set files have the same four-character user-defined prefix (xxxx in the following table), a three character system defined suffix, and according to their content, one of the four file extensions: .MEM, .DBF, .DBT, or .NTX.

The following table shows the files in a LIMS data set used by the UPDATE module:

| file category          | file name              | comment                  |
|------------------------|------------------------|--------------------------|
| data set configuration | xxxxCFG.MEM            | required, system defined |
| data set definition    | xxxxFDL.DBF            | required, system defined |
|                        | xxxxDDL.DBF + .NTX     | required, system defined |
|                        | xxxxCDL.DBF + .NTX     | required, system defined |
|                        | xxxxBDL.DBF + .NTX     | required, system defined |
| raw data               | xxxxHIS.DBF            | required, system defined |
|                        | xxxxGEN.DBF + .NTX     | required, system defined |
|                        | xxxxENV.DBF + .NTX     | required, system defined |
|                        | xxxxPAR.DBF + .NTX     | required, system defined |
|                        | xxxxEST.DBF + .NTX     | optional, system defined |
|                        | xxxxMAT.DBF + .NTX     | optional, system defined |
|                        | xxxxPRG.DBF + .NTX     | optional, system defined |
|                        | xxxxWGT.DBF + .NTX     | optional, system defined |
|                        | xxxxMIL.DBF + .NTX     | optional, system defined |
|                        | xxxxHLT.DBF + .NTX     | optional, system defined |
|                        | xxxxDOC.DBF + .NTX     | optional, system defined |
|                        | xxxxDOC.DBT            |                          |
|                        | xxxxU01.DBF + .NTX     | optional, user defined   |
|                        | • • •                  | • • •                    |
| xxxxU99.DBF + .NTX     | optional, user defined |                          |

The file with the extension .MEM contains values for global system configuration. Files with the extension .DBF contain the data set definition or raw data in the dBASE III+ standard, according to the file category. Files with the extension .NTX contain the index information for the .DBF file with the same name. This index information allows very fast record retrieval by binary search. The index is not compatible with the dBASE III+ index. There is one file with the extension .DBT, which contains records with text of variable length, referred to by a memo field in the file xxxxDOC.DBF.

For more information on the files in a LIMS data set and their function and customisation, see the SETUP manual of the LIMS system.

## 2 Starting the UPDATE Module

The filename of the executing UPDATE module is LUPDATE.EXE. This file should be located in the subdirectory \LIMS on a hard disk. The PATH statement in the AUTOEXEC.BAT file should include the subdirectory \LIMS so that the UPDATE module can be started independent of the current drive or path. For more information on the PATH statement in the AUTOEXEC.BAT file, please refer to the DOS User's Guide.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This data set name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

Before calling the UPDATE module, change the current subdirectory to the subdirectory containing the data set that you want to work with.

Use the following syntax to start the LIMS UPDATE module:

```
LUPDATE [xxxx] [/y]
```

The two items in square brackets ([ ]) are optional parameters that can be specified when calling the UPDATE module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling the UPDATE module, a prompt on the initial screen will request the input of the data set name.

/Y This is the video mode option. By default, the UPDATE module will attempt to start in colour mode.

/H sets a special monochrome display which is optimised for Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be selected.

/M sets monochrome mode, even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor.

/L sets a special monochrome display which is optimised for liquid crystal and gas plasma displays. If your computer has a display of one of these types, you should select this mode.

If the data set name is given as a parameter when the UPDATE module is started, the initial screen will display a number of messages, which will disappear after a fixed amount of time. If no name is given, a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

If the initial screen displays the following error message on the status line at the bottom of the screen:

The 'XXXX' data set was not found. Press any key to continue . . .

- then check:
- whether you made a typing mistake; if so, re-enter the correct data set name on the initial screen, or as a parameter when you start the UPDATE module from the operating system level.
  - whether \xxxx (which stands for the user-defined data set name) is the current subdirectory.
  - whether the following file, having the prefix of the current subdirectory, exists in the current subdirectory:  
xxxxCFG.MEM

### 3 Main Menu

The Main Menu displays the 6 main menu options in a window in the centre of the display area. The currently selected option is displayed in a different colour. The following table shows the keys for use with the Main Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>DOS</b> level                   |

The top line of the Main Menu screen contains information about the Update module and the current data set. In the lefthand corner the module name is displayed, the data set code and the data set title (given during the setup process) are shown in the middle, and the current date to the right. This line is displayed throughout execution of the UPDATE module.

The bottom line displays status information, which in the case of the Main Menu display is the full version of the system name and the version number of the UPDATE module. The different LIMS modules can have different version numbers, but the main version numbers (the integer part) of all LIMS modules must be the same for the different modules to work together properly.

The status line is used to display information, warning and error messages, as well as for confirmation input of certain commands.

#### 3.1 Update a Data File

This option displays a list which allows selection of one of the active data files in the data set for subsequent work.

For further information on data file selection, see the **File Selection List**.

## 3.2 Change ID Numbers

With this option you can change an animal's identification throughout the entire data set. For each change, a record with the old identification, the new identification, the date of the change and the reason for the change is added to the identification history file (xxxxHIS.DBF).

A list of old and new identifications and the respective dates of and reasons for the change can be entered in a window in the central display area. The default value for the date of the change is the system date. The field for the reason for the change is fixed to a length of 10 characters. If there are more changes than lines in the window, the list will scroll vertically. The maximum size of the list is 4096 changes, or the size of the available memory, whichever is reached first. It is recommended that not more than 100 changes be made at one time. The reason for this is that all changes in one list are processed together, and processing time increases with the number of changes in the list. Processing time increases even more with the number of files and records within files that contain animal identification fields.

The identification change list can be completed in two different modes, the **EDIT** and the **BROWSE** modes. During **EDIT** mode, a cursor is visible, and values can be entered into the identification, date and reason fields. During **BROWSE** mode, no cursor is visible, and another field can be activated. After an identification field has been edited, the system validates that identification according to the restrictions given for identification changes. Only after the field value passes the validation will the system change to **BROWSE** mode. If the value does not pass the validation, an appropriate error message will be displayed and the previous field value will be restored. When the whole identification change list is submitted for processing, the system will check for incomplete change records (i.e. records with one valid identification and one missing identification). If a whole identification change record is to be deleted, both identification values must be deleted and the record will disappear. The last record in the change list may have both identification values missing. Such a record will be ignored when the change list is processed.

Below is a list of the keys and their corresponding functions for use with the **BROWSE** mode of the Change **ID** Numbers process.

|             |   |
|-------------|---|
| Uparrow     | - activates the field above (+scroll)   |
| Dnarrow     | - activates the field below (+scroll); if used on the last record, it creates a new record and activates the first field of that record |
| Rightarrow  | - activates the field to the right  |
| Leftarrow   | - activates the field to the left   |
| PgUp        | - scrolls the list up by one page; if used on the first page, it estimates  |
| PgDn        | - scrolls the list down by one page; if used on the last page, it activates the current column field of the last record                 |
| Ctrl-PgUp   | - displays the first page and activates the current column field of the first record  |
| Spacebar    | - activates the first field of the next record (+scroll); if used on the last record, it creates a new record                           |
| Home        | - activates the first field in a record   |
| End         | - activates the last field in a record  |
| Return      | - enters <b>EDIT</b> mode for the active field  |
| <Character> | - enters <b>EDIT</b> mode for the active field and types <character> (if the character is permissible in that field and position)       |
| F10         | - processes the list of ID changes  |
| Esc         | - abandons the list of ID changes   |



The following keys, with corresponding functions, are used with the **EDIT** mode of the Change ID Numbers process:

|                 |  |
|-----------------|--|
| Leftarrow       | - moves the cursor one character to the left   |
| Rightarrow      | - moves the cursor one character to the right  |
| Ctrl-Leftarrow  | - moves the cursor one word to the left  |
| Ctrl-Rightarrow | - moves the cursor one word to the right   |
| Home            | - moves the cursor to the beginning of a field   |
| End             | - moves the cursor to the end of a word/field  |
| Ctrl-Y          | - deletes from the cursor to the end of a field  |
| Ctrl-T          | - deletes a word   |
| Ctrl-U          | - restores the previous field value  |
| Del             | - deletes the character above the cursor   |
| Backspace       | - deletes the character before the cursor  |
| <Character>     | - types <character> (if permissible in that field and position)  |
| Return          | - leaves <b>EDIT</b> mode and activates the field to the right in <b>BROWSE</b> mode; if used on the last field, it activates the first field of the next/new record in <b>BROWSE</b> mode |
| PgUp            | - leaves <b>EDIT</b> mode and activates the current field in <b>BROWSE</b> mode  |
| PgDn            | - leaves <b>EDIT</b> mode and activates the current field in <b>BROWSE</b> mode  |
| Ins             | - toggles between Insert and Overwrite mode  |
| F10             | - leaves <b>EDIT</b> mode and processes the ID change list   |
| Esc             | - restores the previous field value and abandons ID change   |

The F10 key starts the processing of all identification changes in the list. All changes in the list are processed together on a file-by-file basis and within each file on a record-by-record basis. The program examines each identification-type field within each record, within each data file, to see

whether it matches any of the old identification numbers in the change list. If there is a match, the old identification in the field is replaced by the new identification, according to the record in the change list. This sequential process is time consuming, but ensures a consistent change of animal identifications throughout the data set. User-defined identification-type fields will be changed in the same manner as system-defined identification-type fields.

The following restrictions apply when changing animal identifications. The corresponding error messages will appear on the status line at the bottom of the screen.

- A General Animal Record with the OLD ID must exist.  
THE OLD ANIMAL ID WAS NOT FOUND!
- A General Animal Record with the NEW ID must not exist.  
THE NEW ANIMAL ID EXISTS!
- The data of the animal with the OLD ID must not be archived.  
THE OLD ANIMAL ID IS ARCHIVED!
- The OLD ID must not appear elsewhere in the current change list.  
DUPLICATE ANIMAL ID!
- The NEW ID must not appear elsewhere in the current change list.  
DUPLICATE ANIMAL ID!
- The NEW ID must not be missing, even though the OLD ID is valid.  
THIS ID CHANGE RECORD IS INCOMPLETE!
- The OLD ID must not be missing, even though the NEW ID is valid.  
THIS ID CHANGE RECORD IS INCOMPLETE!

After pressing the F10 key, a prompt appears on the status line requesting confirmation that all entered details are correct. ABORT (A) returns to the main menu without processing or saving the data in the change list, YES (Y) starts the change process, while NO (N) allows re-editing of the change list. During execution of the change process, the status line at the bottom of the screen displays the name of the data file currently being processed.

When the F10 key is pressed, starting the actual change process, a system flag is set to indicate whether the change process completes successfully or not. If the change process is interrupted (e.g. due to power failure), all LIMS modules that start to work with this data set will alert the user that an identification change process has been interrupted. When the UPDATE module is started anew, the interrupted identification change process is automatically restarted and completed, thus ensuring the integrity of the animal identifications in the data set.

### 3.3 Sort & Pack Records

Sorting is the physical rearrangement of records according to their index values. This is not a necessary operation, but may speed up data processing.

Packing is the removal of deleted records from a data file. Whenever a record is deleted, it is not removed from the data file but all the field values are set either to blank, zero or their default missing values, according to the data set definition. Unlike in dBASE, it is therefore not possible to recall deleted records in the UPDATE module.

Deleted ('blank') records are re-used whenever new records are added to a database file. If no 'blank' records exist in a database file, a new record is added at the end of the database file.

This system of record deletion has advantages when working with large database files that grow continuously. Deleted ('blank') records are used quickly.

When the **Sort & Pack Records** option is selected, all active database files of the raw data category will be sorted by their respective index fields, and at the same time deleted ('blank') records will be removed. This process can take a considerable amount of time, especially if the data files are large. During sorting and packing the name of the file currently being processed is displayed on the status line.

Sorting will most likely change the original data entry order. This original data entry order may be very important for data validation, especially for correcting typing mistakes involving the index fields. Such errors can often only be traced by comparing the data in their original data entry order with the raw data lists.

To preserve the original data entry order and because of the time requirements, use the **Sort & Pack Records** option sparingly.

Deleted ('blank') records can be removed without changing the original order of data entry, using the **Pack Records Only** option.

Data files can also be individually packed and sorted. See **Sort & Pack Records** and **Pack Records Only** under the **Data Processing Menu**.

### 3.4 Pack Records Only

This option removes deleted ('blank') records from all active database files of the raw data category. Files will only be packed if they contain deleted records.

The use of this option ensures that subsequently added records are stored in the original order of data entry.

The reason for this is that if deleted records exist, they will be re-used whenever new records are added, no matter where the deleted records are located within the data file. Therefore newly added records, stored in the locations of deleted records, are unlikely to be added in the original order of data entry.

The importance of retaining the original order of data entry is discussed in the chapter **Sort & Pack Records** under the **Main Menu**.

Since this process packs only those files that contain deleted records, the time requirements may vary widely depending on the number and size of files containing deleted records. During packing the name of the file currently being packed is displayed on the status line.

Data files can also be individually packed. See **Pack Records Only** under the **Data Processing Menu**.

### 3.5 Re-Index All Files

Index files (.NTX) contain information that allows very fast retrieval of records based on their index values. The UPDATE module uses these index files for practically every record retrieval operation. Under normal operating conditions the data files (.DBF) and the index files (.NTX) are maintained and updated together. However, under certain conditions (e.g. power failure while editing the data files with dBASE, etc) the index files may become corrupted or they may no longer match the data files. Whenever such a situation is expected, the index files should be recreated with this option.

When the UPDATE module is started, all index files of the data set definition category are automatically recreated. If index files of the raw data category are missing they are also recreated, but no check is made of existing index files in this category.

The option **Re-Index Data Files** recreates all index files corresponding to the active database files of the raw data category. This process may require some time, especially if the data files are large.

Data files can also be individually re-indexed. See **Re-Index Data File** under the **Data Processing Menu**.

### 3.6 System Configuration

This option displays codes and options that have been defined and set from the SETUP module. You can change the following codes and options for the duration of the updating session using the System Configuration option:

- the response codes for YES, NO and ABORT to questions such as ‘ARE ALL DETAILS CORRECT?’.
- the Mating Lookup Period, which determines the period before parturition (in days) for which the mating lookup procedure will search the mating data file and display the respective matings.
- warnings enabled (T/Y=true/yes, F/N=false/no), which determines whether or not field validation and record validation are carried out during data entry and modification. For further information see the chapters **Field Validation** and **Record Validation**.
- confirm enabled (T/Y=true/yes, F/N=false/no), which determines whether or not an additional pressing of the RETURN key is required after the last character position of a variable value has been edited.
- date display format, which allows you to choose between several different display formats. The F2 key brings up a selection list.
- function key 6/7/8 file, which allows you to assign the function keys 6, 7 and 8 to a data file for function key editing. The F2 key brings up a selection list of available data files to be assigned. Further instructions on function key editing are given in the chapter **Function Key Edit**.

For more information on the other displayed settings, see the SETUP module.

## 4 File Selection List

This list displays the user-assigned labels of the data files created with the SETUP module. At this stage a data file can be selected for subsequent data input and editing. The current selection is shown with a different background colour. Pressing the RETURN key will open the selected data file and activate the **Data Processing Menu**.

The following keys, with corresponding functions, are used with the file selection list:

|             |   |
|-------------|---|
| Spacebar    | - selects the following file                            |
| Dnarrow     | - selects the following file                            |
| Uparrow     | - selects the preceding file                            |
| <character> | - selects the following file beginning with <character> |
| PgDn        | - advances the file selection list by one page          |
| PgUp        | - moves the file selection list back by one page        |
| End         | - selects the last file                                 |
| Home        | - selects the first file                                |
| Return      | - activates the selected option                         |
| Esc         | - returns to the <b>Main Menu</b>                       |

Up to 8 file labels are displayed on one page of the file selection list. If there are more than 8 data files in a data set, a set of up and down arrows will appear on the line below the last file label. This indicates that the file selection list will scroll when the selection is moved past the end of the list.

## 5 Data Processing Menu

After a file has been chosen from the **File Selection List** and the Return key has been pressed, the **Data Processing Menu** becomes active. There are 7 numbered menu options available. Pressing the Esc key will return to the **File Selection List**.

The centre of the screen contains the Data Processing Menu in the foreground and an empty record of the currently selected file in the background. The file label is displayed above the empty record.

A column on the lefthand side of the central display area shows a record of the currently selected data file, one field per row, in all up to 19 fields. If there are more than 19 displayed fields in a data file, the remaining fields will be shown in a column to the right of the screen centre.

Each field is displayed with the corresponding field label and the data entry area. The space between the field label and the data entry area is taken up by the 'underline' character. The data entry area is normally empty for character variables while for numeric fields it contains the default missing value. Some fields may contain characters that are part of the field value but do not have to be typed. The slashes in the date fields are an example.

The 7 numbered options of the Data Processing Menu are displayed in a window on the righthand side of the central display area. The currently selected option is shown in a different colour. After selecting an option, activate it by pressing the Return key. The options can be selected using the following keys:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - moves to the last option                          |
| Home     | - moves to the first option                         |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>File Selection List</b>         |

While the Data Processing Menu is active, the status line on the bottom displays the DOS filename of the currently selected file in the lefthand corner.

## 5.1 Add a New Record

This option from the Data Processing Menu enables you to add a new record to the currently selected data file. This chapter describes the adding process in general as it applies to all data files that can be selected from the File Selection List. The adding process for records of the Parturition Data file and the Documentation Data file requires additional documentation, which is given in separate chapters.

After the option **Add a New Record** has been chosen from the **Data Processing Menu**, the centre of the screen will display an empty data record of the selected file. The variable values can now be edited and the following **Data Processing Sub-Systems** are available:

- Assist System
- Validation (according to SETUP/System Configuration)
- Function Key Edit (according to SETUP/System Configuration)

The status line at the bottom of the screen displays the record pointer as 'RECDpx/y' in the righthand corner. The value x is the current record number, based on the physical position of the record in the data file, and y is the total number of records in the data file. If there are any deleted ('blank') records in the data file they are included in the count of total records. To remove deleted records, see **Pack Records Only** under the **Data Processing Menu**.

More detailed descriptions of these systems are given in the corresponding chapters under the heading **Data Processing Sub-Systems**.



The following table shows the keys used for editing the variable values:

|              |  |
|--------------|--|
| Leftarrow    | - moves the cursor one character to the left                         |
| Rightarrow   | - moves the cursor one character to the right                        |
| Uparrow      | - leaves the current field, activates the previous field for editing |
| Dnarrow      | - leaves the current field, activates the next field for editing     |
| Home         | - moves the cursor to the beginning of a field                       |
| End          | - moves the cursor to the end of a word/field                        |
| Ctrl-Y       | - deletes from the cursor to the end of a field                      |
| Ctrl-U       | - restores the previous field value                                  |
| Del          | - deletes the character above the cursor                             |
| Backspace    | - deletes the character before the cursor                            |
| <character>  | - types <character> (if permissible in that field and position)      |
| Return       | - leaves the present field, activates the next field for editing     |
| Ins          | - toggles between insert and overwrite mode                          |
| F2           | - invokes the <b>Assist System</b>                                   |
| F3/4/5/6/7/8 | - invokes the <b>Function Key Edit</b>                               |
| F9           | - jumps to the next marked field                                     |
| F10          | - saves edited record  |
| Esc          | - aborts edited record, returns to the <b>Data Processing Menu</b>   |

Once a record has been completed, a prompt on the status line at the bottom of the screen requests confirmation that all details displayed on the screen are correct. YES (Y) adds the record to the selected data file, NO (N) allows re-editing the record, and ABORT (A) aborts the record adding process, discards the data and displays an empty record.

When the record has been confirmed as being correct, it will be submitted to an error check to see whether the data in the record are consistent with the data in the current file. If an error is found, the record cannot be added to the

respective data file. The error system activates the field where the error was found, and the error must be corrected before the record can be saved.

The following errors, with corresponding error messages, can occur:

#### **All data files**

- an index field must not contain a missing value  
RECORD INDEX FIELDS CANNOT CONTAIN MISSING VALUES!
- an index ID must exist in the General Animal Data file  
ANIMAL ID (id) DOES NOT EXIST IN THE GENERAL DATA FILE!

#### **General Animal Data, Parturition Data, Environment Data**

- duplicate records may not be created  
DUPLICATE RECORD. THE RECORD INDEX EXISTS IN THIS FILE!

Further error conditions are documented in the chapter **Edit a Record**.

Once the record has passed the error check it will be added to the selected data file and a new empty data record of the selected file will appear on the screen.

### **5.1.1 Parturition Record**

When the Dam Identification Number and the Parturition Date have been edited, the parity number for this parturition record will be calculated and displayed. The calculation of the parturition number follows the rules described in **Parity Lookup** under the **Assist System**.

Apart from the more general Assist functions, the Mating Lookup function is available when the Sire Identification field is active and the Parity Lookup function is available when the Parity Number field is active.

After completion of the Parturition record, it is validated and warning messages are displayed if appropriate. For more information on validation of the parturition record, see the chapter **Record Validation**.

Validation is followed by confirmation and error checking. The following errors, with corresponding error messages, can occur:

#### **Within the Parturition Record**

- index fields of the Parturition Record must not contain missing values  
RECORD INDEX FIELDS CANNOT CONTAIN MISSING VALUES!
- there must not be a duplicate parturition record  
DUPLICATE RECORD. THE RECORD INDEX EXISTS IN THIS FILE!
- a General Animal Record for the dam must exist  
ANIMAL ID (id) DOES NOT EXIST IN THE GENERAL DATA FILE!

- the General Animal Record for the dam must not be archived  
ANIMAL ID (id) IS ARCHIVED!
- the General Animal Record for the dam must have sex code female  
GENDER OF ANIMAL (id) IS INCOMPATIBLE WITH THIS RECORD!
- the parturition date must be within the life-span of the dam  
INVALID DATE (date). <PART\_DT> IS OUT OF GENETICALLY ACTIVE PERIOD!
- a General Animal Record for the sire must exist  
ANIMAL ID (id) DOES NOT EXIST IN THE GENERAL DATA FILE!
- the General Animal Record for the sire must have sex code male  
GENDER OF ANIMAL (id) IS INCOMPATIBLE WITH THIS RECORD!
- the mating date must be within the genetically active period of the sire  
INVALID DATE (date). <MATE\_DT> IS OUT OF GENETICALLY ACTIVE PERIOD!
- no. animals born <= no. animals born alive <= maximum litter size  
INVALID NUMBER (num). <NO\_LIVE> IS OUT OF RANGE!
- no. animals born alive = no. offspring ID's (live born animals get ID)  
INVALID NUMBER (num). <NO\_LIVE> IS OUT OF RANGE!
- no. animals born = no. offspring ID's (all born animals get ID)  
INVALID NUMBER (num). <NO\_BORN> IS OUT OF RANGE!
- dam ID, sire ID and offspring ID's must all be unique  
ANIMAL ID (id) IS DUPLICATE IN THIS RECORD!
- a General Animal Record for all non-offspring ID's must exist  
ANIMAL ID (id) DOES NOT EXIST IN THE GENERAL DATA FILE!

#### Correspondence between Offspring General Animal and Parturition Records

- offspring General Animal record must not be archived  
ANIMAL ID (id) IS ARCHIVED!
- correspondence of GEN->DAM\_ID with PAR->DAM\_ID  
OFFSPRING RECORD (id) CONTAINS CONTRADICTING DAM ID!
- correspondence of GEN->BIRTH\_DT with PAR->PART\_DT  
OFFSPRING RECORD (id) CONTAINS CONTRADICTING BIRTH DATE!
- correspondence of GEN->PARITY with PAR->PARITY  
OFFSPRING RECORD (id) CONTAINS CONTRADICTING PARITY!

- correspondence of GEN->BIRTH\_TY with PAR->NO\_BORN OFFSPRING RECORD (id) CONTAINS CONTRADICTING BIRTH TYPE!
- correspondence of GEN->BIRTH\_DF with PAR->BIRTH\_DF OFFSPRING RECORD (id) CONTAINS CONTRADICTING BIRTH DIFFICULTY!
- correspondence of general-> SIRE\_ID with PAR-> SIRE\_ID OFFSPRING RECORD (id) CONTAINS CONTRADICTING SIRE ID!
- correspondence of GEN->BIRTH\_DV with PAR->PART\_DV OFFSPRING RECORD (id) CONTAINS CONTRADICTING BIRTH DEVIATION!

If an error is found in the parturition record, the record cannot be added to the Parturition Data file. The error system activates the variable where the error was found, and the error must be corrected before the record can be saved.

After the error check has been passed, the message 'PLEASE NOTE - OFFSPRING RECORDS WILL BE MODIFIED! CONTINUE (Y/N)' will be displayed on the status line. YES (Y) will add the parturition record to the Parturition data file and proceed with updating the offspring General Animal Records and the offspring Environment records. NO (N) allows re-editing the parturition record.

For each offspring that has an identification number in the parturition record, a window with the General Animal Data and the Environment Data is opened for editing. Only fields with a triangular marker in front can be edited. Certain fields will have values that have been carried over from the parturition record or which have been derived from global settings or rules.

Within the editing windows, the Assist and the Validation sub-systems, but not the Function Key Edit system, are available.

If General Animal Records of offspring specified in the Parturition Record exist, they will be checked for contradicting information (see error check above) and will subsequently be attached to the Parturition Record by transfer of certain information (see below). If General Animal Records of offspring do not exist they will be created after editing the General Animal Data window.

For the General Animal Record of each offspring, the following field values are generated or carried over from the parturition record:

| General Animal Record | Source  |
|-----------------------|---|
| Animal Identification | <- Offspring Identification   |
| Dam Identification    | <- Dam Identification   |
| Sire Identification   | <- Sire Identification  |
| Birth Date            | <- Parturition Date   |
| Birth Date Deviation  | <- Parturition Date Deviation   |
| Birth Type            | <- Number of Offspring Born   |
| Parity                | <- Parity   |
| Breed                 | <- given by the Breed Definition Library according to breed of Dam and Sire |

For the Environment Record of each offspring, the following field values are generated or carried over from the parturition record:

| Environment Record         | Source  |
|----------------------------|---|
| Animal Identification      | <- Offspring Identification                           |
| Environment Date           | <- Parturition Date                                   |
| Environment Date Deviation | <- Parturition Date Deviation                         |
| Environment 1/2/3 Codes    | <- given by the Dam's current environment record      |
| Environment Entry Reason   | <- globally defined code for Environment Entry Reason |

Once all General Animal Records and Environment Records of the offspring have been edited and appended, a new empty parturition record will appear in the centre of the screen.

### 5.1.2 Documentation Record

The Documentation Data file is a special file in that one of its fields can contain text information of variable length. You can use this file to store data that cannot be stored in any of the other data files. Such information could be

research protocols, progress reports, management or work procedures, detailed post-mortem reports, etc.

Unlike the other data files, it is not possible to add user-defined fields to the Documentation Data file. The index of the Documentation Data file consists of three fields, the documentation date, the documentation type and the documentation number. An additional non-index field can contain an animal identification number. Text can be entered when the memo field is active. Activate the text edit window by pressing the spacebar or a character key. When the text edit window is active, the memo field displays a line and column counter. Text is edited by a simple ASCII text editor, with features such as automatic wrapping within the predefined margins and the possibility of importing and exporting ASCII text files.

Below is a list of the keys and their corresponding functions for use with the active text edit window.

|                      |   |
|----------------------|---|
| Uparrow/Dnarrow      | - moves the cursor up/down one line               |
| Rightarrow/Leftarrow | - moves the cursor right/left one column          |
| PgUp/PgDn            | - scrolls text up/down                            |
| Ctrl-PgUp/Ctrl-PgDn  | - moves the cursor to the beginning/end of a text |
| Home/End             | - moves the cursor to the beginning/end of a line |
| Return               | - inserts hard carriage return                    |
| <character>          | - enters <character>                              |
| Ctrl-Y/Ctrl-T        | - deletes a line/word                             |
| Alt-W/Alt-R          | - writes to file (PRN=printer)/reads from file    |
| Alt-F/Alt-A          | - finds text/finds next occurrence                |
| Alt-X                | - saves text and exits text edit window           |
| F10                  | - saves Documentation Data record                 |
| Esc                  | - abandons memo editing                           |

Text which is entered into the memo field is stored in a separate file under the name xxxxDOC.DBT. All other data of the Documentation Records are stored in the file xxxxDOC.DBF.

## 5.2 Edit a Record

This option from the Data Processing Menu allows you to edit an existing record in the currently selected data file. This chapter describes the edit process in general as it applies to all data files that can be selected from the File Selection List. The edit process for records of the General Animal Data file and the Parturition Data file requires additional documentation, which is given in separate chapters.

After the option **Edit a Record** has been activated from the **Data Processing Menu**, the **Record Retrieval System** becomes active and an empty data record is displayed in the central display area. For more information, see **Record Retrieval System** under **Data Processing Sub-Systems**.

Once a record has been found and displayed using the Record Retrieval System, the Edit process can be started. Variable values can now be edited and the following **Data Processing Sub-Systems** are available:

- Assist System
- Validation (according to SETUP/System Configuration)
- Function Key Edit (according to SETUP/System Configuration)

For more information on these systems, see the appropriate chapters under **Data Processing Sub-Systems**.

The keys used for editing are documented in the chapter **Add a New Record**.

### 5.2.1 General Animal Record

As opposed to other data files, the index field (ID) of the General Animal Data file cannot be edited. The General Animal Data file is the reference file for all other animal related data files. It is assumed that changes of ID numbers should apply throughout the entire data set. To change an animal identification number, follow the procedure described in the **Change ID Numbers** chapter.

### 5.2.2 Parturition Record

During the edit process of a parturition record, the automatic calculation of the parity number as described in **Parturition Record** under **Add a Record** is not available. However, the **Parity Lookup** of the **Assist System** is available.

During the edit process of a Parturition Record, the General Animal Records of the offspring become **detached**. All information that was transferred from the Parturition Record to the General Animal Records of the offspring during the Add a New Record process will be assigned blank values in the offspring's

General Animal Records. If the breed value resulted from an application of the breed rules it will also be assigned a blank value; if not it will be retained. Such General Animal Records are called detached records, because the linkage to a specific Parturition Record has been removed. During the detaching process the Environment Records of the offspring that have as environment entry date the parturition/birth date are deleted.

When the editing of a Parturition Record is completed, the General Animal Records of the offspring recorded in the Parturition Record become **attached** again. If General Animal Records of offspring do not exist they are created. For all offspring an Environment Record with the parturition/birth date as environment entry date is created.

The underlying rationale for the use of this approach of **detaching** and **attaching** is that the offspring which have been identified in a parturition record are assumed to have existed at one point in time and that individual performances or observations on the offspring may have been recorded as well (e.g. sex, body weight, mortality, etc). This is the reason why General Animal Records are detached but not deleted when editing a Parturition Record. The offspring may not have been born in the particular parturition that is to be edited and therefore the linkage between the Parturition Record and the offspring's General Animal Record can be removed by deleting the offspring's identification number in the Parturition Record. The offspring's detached General Animal Record can be attached to some other Parturition Record by adding the detached offspring's identification number to the list of offspring identification numbers in that particular Parturition Record.

Apart from the above points, the edit process of a Parturition Record proceeds as described in the chapter **Parturition Record** under **Add a New Record**.

### 5.3 Delete a Record

This option from the Data Processing Menu permits deletion of a record in the currently selected data file. This chapter describes the delete process in general as it applies to all data files that can be selected from the File Selection List. The delete process for records of the General Animal Data file and the Parturition Data file requires additional documentation, which is given in separate chapters.

After the option **Delete a Record** has been activated from the **Data Processing Menu**, the **Record Retrieval System** becomes active and an empty data record is displayed in the central display area. For more information see **Record Retrieval System** under **Data Processing Sub-Systems**.



Once a record has been found and displayed using the Record Retrieval System, the delete process can be initiated. A prompt on the status line at the bottom of the screen requests confirmation that the record is to be deleted. YES (Y) starts the record deletion process, while NO (N) aborts record deletion and activates the Record Retrieval System. Pressing the Esc key aborts the record deletion process and returns to the Data Processing Menu.

Deleting a record in the LIMS system is a different process from deleting a record in dBASE. In LIMS all fields of a record are set to blank/zero/missing when the record is deleted. The 'blank' record remains in the file and can be re-used. It is not possible to recall a record once it has been deleted. For further information on deleted records see **Sort & Pack Records** and **Pack Records Only** under **Main Menu**.

### 5.3.1 General Animal Record

A General Animal Record can only be deleted if it is not archived, the animal ID is not referenced by the index key of any other data file and the animal does not appear as an offspring in a corresponding parturition record. In other words, all performance records of the animal, including the Environment Records as well as the Parturition Record in which the animal appears as an offspring, must be deleted, before the animal's General Animal Record can be deleted. Apart from these restrictions the deletion process follows the general description given under **Delete a Record**.

### 5.3.2 Parturition Record

When a Parturition Record is added, a General Animal Record is created/attached for each offspring identification recorded in the parturition. The Parturition Record and the corresponding General Animal Record contain duplicate data (e.g. dam ID, sire ID, parturition date/birth date, parity, animals born alive/birth type). Furthermore, the General Animal Record contains the breed code for the offspring, which is generated from the breed codes of the dam and the sire using a rule defined in the Breed Definition Library. When a Parturition Record is deleted, the General Animal Records of the offspring remain, but they become detached from the parturition record, e.g. all information in the General Animal Record that relates to the parturition is deleted.

The underlying assumption for this approach is that the offspring which have been identified in a parturition record existed at one point in time and that individual performances or observations on the offspring may have been recorded as well (e.g. sex, body weight, mortality, etc). However, the offspring may not have been born in the particular parturition that is to be deleted.

If the offspring identified in the parturition record to be deleted never existed, their General Animal Records have to be deleted in separate steps, after deleting the parturition record in which they were identified.

Detached General Animal Records can be **attached** to an existing or a new Parturition Record. For more information on attaching General Animal Records to a Parturition Record, see **Add a New Record** and **Edit a Record**.

The following steps take place when a parturition record is deleted:

for the **General Animal Record** of each offspring:

- Dam ID, sire ID, birth date, birth date deviation (if any), parity, birth type are set to blank/missing.
- If the breed code of the offspring conforms to a rule of the Breed Definition Library, based on the breed codes of the dam and sire, it is set to blank, otherwise the offspring breed code is retained. The assumption is that the offspring breed code has been created automatically if a rule from the Breed Definition Library applies; if no rule applies, it is assumed that the offspring breed code has been given after an individual assessment of the animal.

for the **Environment Record** of each offspring:

- The Environment Record that has the parturition (birth) date as the environment entry date is deleted. The assumption is that only the Environment Record that refers to the parturition (birth) date has been created as part of the parturition data; other Environment Records of the offspring are based on individual observation or performance and are therefore not deleted.

for the **Parturition Record**:

- The Parturition Record is deleted, and all fields in the record are set to blank/zero/missing.

The following example should illustrate the usefulness of the detaching/attaching approach:

Two lambs have been recorded as having been born from two different ewes, both lambs born as singles. Later it is discovered that in fact one of the ewes had given birth to both lambs, whereas the other ewe did not give birth at all. It is now possible to delete the parturition record of the ewe that did not give birth, thereby detaching the General Animal Record of the lamb from this Parturition Record. Subsequently the detached lamb can be attached to the parturition record of the ewe that gave birth, through Edit a Record, without disturbing any of the data referring to individual observations and performances of the lamb.

## 5.4 Mark Data Fields

This option allows you to assign markers to fields in the currently selected data file. There are two different markers, one for carry and one for jump. A field marked carry will carry over its value from the previous record to the current record during the **Add a New Record** process. This feature is useful when entering a series of records having the same value for one or more fields (e.g. weight records of different animals from one weighing date). A field marked jump can be jumped to during the **Add a New Record** or **Edit a Record** process, without having to pass through preceding fields. This feature is useful when completing existing records or adding values for some selected fields of new records (e.g. entering a list of disposal dates and reasons into existing General Animal Records).

For more information on how to edit fields, see **Add a New Record** and **Edit a Record**.

When the option Mark Data Fields is activated, the Data Processing Menu disappears and the cursor is placed in front of the first field label. A carry marker can be set or removed by pressing the F8 key, while pressing the F9 key sets or removes a jump marker. A carry marker is displayed as a small square whereas a jump marker appears as a triangle in front of the field label. If both markers are set for one field, a diamond will be displayed in front of the field label.

The following keys and key combinations access the fields, set and remove markers and save the markers and return to the Data Processing Menu:

|          |   |
|----------|---|
| F8       | - sets/removes carry marker [toggle]                        |
| F9       | - sets/removes jump marker [toggle]                         |
| Spacebar | - removes carry and jump marker                             |
| Return   | - moves the cursor to the following field                   |
| Uparrow  | - moves the cursor to the preceding field                   |
| Dnarrow  | - moves the cursor to the following field                   |
| Home     | - moves the cursor to the first field                       |
| End      | - moves the cursor to the last field                        |
| Shift-F8 | - removes all carry markers                                 |
| Shift-F9 | - removes all jump markers                                  |
| F10      | - saves markers, returns to the <b>Data Processing Menu</b> |
| Esc      | - saves markers, returns to the <b>Data Processing Menu</b> |

The markers are displayed during the execution of all options of the Data Processing Menu. Only the jump marker is effective during **Edit a Record**, whereas both markers are effective during **Add a New Record**. Pressing the Esc key from the active Data Processing Menu returns to the File Selection List and removes all markers. Markers cannot be saved for subsequent use with the UPDATE module.

When the Mark Data Fields option is activated, the status line displays the message 'F8=CARRY/F9=JUMP' in the righthand corner to indicate which function keys are used to set and remove the different markers.

## **5.5 Sort & Pack Records**

This option sorts records according to their index value and removes deleted ('blank') records from the database file selected in the **File Selection Menu**.

The procedures for sorting and packing files are described further in the chapter **Sort & Pack Records** under the **Main Menu**.

## **5.6 Pack Records Only**

This option removes deleted ('blank') records from the database file selected in the **File Selection Menu**.

More information on packing files is given in the chapter **Pack Records Only** under the **Main Menu**.

## **5.7 Re-Index Data File**

This option allows recreating the index file of a corresponding database file selected in the **File Selection Menu**.

The chapter **Re-Index Data Files** under the **Main Menu** describes indexing in more detail.

## 6 Data Processing Sub-Systems

From some of the options of the Data Processing Menu, certain sub-systems are available. Since the sub-systems are not an essential part of the various data processing options and may be available from more than one option, they are documented separately from the data processing options in the following chapters.

### 6.1 Record Retrieval System

The Record Retrieval System retrieves an individual record by its unique index key value from the currently selected data file for editing or for deletion.

The options **Edit a Record** and **Delete a Record** from the **Data Processing Menu** start the Record Retrieval System.

While the Record Retrieval System is running, the fields making up the index key value are displayed in a different colour than the rest of the fields in the record.

When the Record Retrieval System has been activated, index key values can be entered and the cursor moved between the index fields using the up- and down-arrow keys. Within a field the Home and End keys will move the cursor to the beginning or the end of the field, respectively. Pressing the Return key will enter the edited value into the index field and move the cursor to the beginning of the next index field, or will initiate the retrieval process, after the value for the last index field has been entered.

If a record is found that matches the entered index key value, the Record Retrieval System will display the record and activate either the edit or the delete mode, according to the option selected from the Data Processing Menu. When the record is activated the display of the index key fields returns to normal.

If the index key value is not found in the currently selected data file, the record with the next higher index key value will be displayed and the Record Retrieval System remains active. The PageUp and the PageDown keys can be used to display the previous and subsequent records, based on the index key value. When the first record (lowest index value) is displayed and the PageUp key is pressed, a beeping sound will be emitted and the last record (highest index value) in the file will come up for display. Similarly, when the last record is displayed and the PageDown key is pressed, the speaker will emit a beep and the first record will be shown. A new retrieval process can be initiated by entering new index key values while the Record Retrieval System is still active, or the edit or delete mode can be entered for the displayed record by pressing the Return key with the cursor on the last index field.

Pressing the Esc key while the Record Retrieval System is active returns to the Data Processing Menu, while keeping the currently selected record displayed.

The following table shows the keys for use with the Record Retrieval System and their respective functions:

|             |   |
|-------------|---|
| <character> | - types <character> into the index field  |
| Uparrow     | - selects the preceding index field for editing   |
| Dnarrow     | - selects the following index field for editing   |
| PgUp        | - displays the preceding record   |
| PgDn        | - displays the following record   |
| Return      | - enters a value into the index field, activates the next index field; starts retrieve process with the cursor on the last index field; if index is found, it initiates the Edit/Delete process; if index is not found, the record with next higher index value is displayed and retrieval mode is retained |
| F10         | - enters a value into the index field and starts retrieve process; if index is found, it initiates the Edit/Delete process; if index is not found, the record with next higher index value is displayed and retrieval mode is retained  |
| Esc         | - returns to the <b>Data Processing Menu</b>  |

The Record Retrieval System will switch either to the edit or to the delete mode, according to the option selected from the Data Processing Menu, whenever a record is found that matches the index key values entered into the index key fields. If the index key cannot be found the Record Retrieval System will remain active, permitting further searches of records.

While the Record Retrieval System is running, the status line at the bottom of the screen displays the record pointer as 'RECD x/y' in the right corner. The value x is the current record number, based on the physical position of the record in the data file, and y is the total number of records in the data file. If there are any deleted ('blank') records in the data file they will be included in the count of total records. To remove deleted records, see **Pack Records Only** under the **Data Processing Menu**.

## 6.2 Assist System [F2]

The **Assist System** is available from the options **Add a New Record** and **Edit a Record** from the **Data Processing Menu**. When pressing the F2 key with the cursor on an active data field for which assistance is available, a lookup list will appear in the centre of the display area. If assistance is not available for an active field, the status line will display the message 'ASSISTANCE IS NOT AVAILABLE FOR THIS FIELD'.

Within the lookup list a selection can be made, and by pressing the Return key the selected value will be carried over to the active data field. If there are more items than can be displayed on one page of the lookup list, the list can be scrolled.

The following keys, with corresponding functions, are used with a lookup list:

|             |  |
|-------------|--|
| Spacebar    | - selects the following item                             |
| Dnarrow     | - selects the following item                             |
| Uparrow     | - selects the preceding item                             |
| <character> | - selects the following item beginning with <character>  |
| PgDn        | - advances the selection list by one page                |
| PgUp        | - moves the selection list back by one page              |
| End         | - selects the last item                                  |
| Home        | - selects the first item                                 |
| Return      | - carries over a selected value to the active field      |
| Esc         | - returns to the active field without changing its value |

There are six different lookup lists available, each of which acts upon specific fields only.

### 6.2.1 Breed Lookup

Assistance for breed is only available when the animal breed field of the General Animal Data file is active. The lookup list contains the breed codes and their labels as defined in the code definition library. A breed code can be selected from the Breed Lookup List and carried over to the animal breed field.

The system checks whether both sire and dam of the animal exist in the system and whether their particular breed code combination and the resulting offspring breed code are defined in the Breed Definition Library. If that is the case, the appropriate breed code for the animal is marked with an arrow and will automatically be selected when the Breed Lookup list is displayed. If breed code for a progeny is entered in reverse order from the recommended one it will be printed out as an error during the VALIDATION process.

This feature allows you to check whether the breed code of a particular animal agrees with the breed code combinations defined in the Breed Definition Library, or selecting and carrying over one of the breed codes into the active breed field. The Esc key aborts the lookup and returns to the active field in the record without changing its value.

For more information on the functions and contents of the Code Definition Library and the Breed Definition Library, see the documentation of the SETUP module.

While the Breed Lookup list is displayed, the status line at the bottom of the screen displays the animal identification number to which the breed lookup refers in the lefthand corner, and to the right the appropriate breed code for the animal according to the Breed Definition Library. If no appropriate breed code can be found in the Breed Definition Library, the message 'N/A' (not applicable) will be displayed.

## **6.2.2 Code Lookup**

Assistance for codes is only available for an active code-type field. The lookup list contains the code values and their labels as defined in the Code Definition Library. A code value can be selected from the Code Lookup list and carried over to the active code-type field in the record.

For more information on the function and content of the Code Definition Library, see the documentation of the SETUP module.

The Code Lookup allows comparing an existing code value in the active field with a list of all code values defined for that field in the Code Definition Library, or selecting and carrying over one of the defined code values into the active code type field. The Esc key aborts the lookup and returns to the active field in the record without changing its value.

If the Assist System is invoked on a code type field for which no codes are defined in the Code Definition Library, the message 'CODE WAS NOT FOUND FOR THIS FIELD' will be displayed on the status line.



### **6.2.3 Value Lookup**

Assistance for values is only available for an active numeric-type field. The lookup list contains the default missing value, the minimum and maximum values, according to the user's definition in the Data Definition Library.

The function and content of the Data Definition Library are described further in the SETUP module.

If the Assist System is invoked on a numeric-type field for which no minimum, maximum and default missing values are defined in the Data Definition Library, the message 'A MINIMUM AND MAXIMUM VALUE WAS NOT GIVEN FOR THIS FIELD' will be displayed on the status line.

### **6.2.4 Date Lookup**

Assistance for dates is only available for date-type fields, if the active record contains an animal identification field. The Date Lookup list contains all date-type fields from the general animal record of the key animal. The key animal is determined by the first identification-type field in the active record.

This feature allows comparing a specific date value with all date values in the animal's general record, or selecting and carrying over one of these date values into the active date field. The Esc key aborts the lookup and returns to the active date field in the record without changing its value.

While the Date Lookup list is displayed on the screen, the status line displays the identification number of the key animal to which the date lookup refers in the lefthand corner, and to the right the name of the identification field that determines the key animal.

### **6.2.5 Mating Lookup**

Assistance for mating is only available when the sire ID field of a parturition data record is active. The lookup list contains partial mating records from a fixed period preceding the parturition date. These records contain the sire ID, the mating date and the calculated difference in days between the mating date and the parturition date (gestation length). The length of the period from which matings are displayed can be set by the user as a permanent default using the SETUP module or as a temporary setting for the duration of the update session using the System Configuration option of the UPDATE module.

This feature allows listing of all recorded matings for the dam ID of the active parturition record from a defined period preceding the parturition, selecting the successful mating based on the gestation length and carrying over the

mating date and the sire ID to the parturition record. The Esc key aborts the lookup and returns to the active sire ID field in the record without changing the values of the mating date field and the sire ID field.

The sire ID number is again carried over from the parturition record to the general animal records that are automatically created for each offspring, once the parturition record is saved. Lookup of the appropriate mating does not need to be performed when the parturition data are first entered. If no sire is recorded in the parturition record, the sire and breed fields of the offspring's general animal records also remain empty. In a subsequent editing of the parturition record, the connection between mating and parturition can be made with the assistance of the Mating Lookup list, and does not necessarily need to be done by the person who originally entered the data. If a mating date and a sire ID are completed in the parturition record, the offspring's general animal records will also be updated with the sire ID and the offspring's breed. The offspring's breed can only be updated automatically if the combination of sire breed and dam breed, with resulting breed of offspring, is defined in the Breed Definition Library.

More information on the function and content of the Breed Definition Library is given in the SETUP module.

While the Mating Lookup list is displayed on the screen, the left side of the status line displays the dam's identification number, which is used for lookup in the mating data file, and the length of the period preceding the parturition date for which the lookup is done is shown in the righthand corner. If no mating is found for the dam in the defined period preceding the parturition, the message 'MATING RECORD WAS NOT FOUND FOR THIS FIELD' will be displayed on the status line.

### **6.2.6 Parity Lookup**

Assistance for parity is only available when the parity field of the Parturition Data file or General Animal Data file is active. The lookup contains a calculated parity number. The following rules apply for the calculation of the parity number:

- If a previous parturition record exists, the parity number of that record + 1 will be proposed as the parity number of the present record.
- If no previous parturition record exists and the first environment record of the dam carries the birth date of the dam, a parity number of 1 will be proposed. It is presumed that the dam was in the recording system from birth onwards and since there is no previous parturition record, the current record is assumed to be the first parturition.

- If no previous parturition record exists and the first environment record of the dam carries a date later than the birth date of the dam, no parity number will be proposed. The assumption is that the dam spent part of her life outside the recording system and may have reproduced before she was included into the recording system.

### **6.3 Validation**

When variable values are modified, the system checks these modified values and generates a warning message if they do not pass the validation. For the General Animal Data and the Parturition Data the entire record is validated before it is processed further.

The validation can be toggled on and off either by setting a permanent default using the SETUP module or by switching the validation on or off for the duration of the UPDATE session using the System Configuration option of the UPDATE module.

#### **6.3.1 Field Validation**

Once the last position of a variable value has been edited, the value is validated. Depending on the type of variable, the following validation procedures, yielding the corresponding warning messages, are carried out:

- Check whether a missing value is allowed or not.  
A MISSING VALUE IS NOT ALLOWED FOR THIS FIELD!
- Check for existence of code for code-type variables.  
CODE WAS NOT FOUND IN THE CODE DEFINITION LIBRARY!
- Check sex for ID-type variable.  
SEX OF ANIMAL IS INCOMPATIBLE WITH THIS FIELD!
- Relative/absolute range check on date-type variables.  
DATE IS NOT WITHIN THE LIFESPAN OF ANIMAL <id>!  
ANIMAL <id> BIRTH DATE IS EMPTY. DATE CANNOT BE CHECKED!  
DATE IS GREATER THAN THE MAXIMUM AGE ALLOWED - ANIMAL <id>!  
DATE IS LESS THAN THE MINIMUM AGE ALLOWED-ANIMAL <id>!  
DATE ENTERED IS OUT OF RANGE <minimum date> - <aximum date>!

- Range check for numeric-type variables.  
VALUE ENTERED IS OUT OF RANGE <maximum value> - <aximum value>!

### 6.3.2 Record Validation

Records from certain data files are validated before they are processed further. The validation process generates a warning message if necessary, but will not stop further processing of the record. Warning messages should not be confused with error messages; the latter will stop further processing of the record until the error is rectified. Error messages are described under **Add a New Record** and **Edit a Record**.

In the **General Animal Data** file the validation consists of the following checks, with corresponding warning messages:

- Animals of female sex may not have a castration date.  
CONTRADICTION BETWEEN SEX AND CASTRATION DATE!
- Animals of male sex may not have a first estrus date.  
CONTRADICTION BETWEEN SEX AND FIRST ESTRUS DATE!
- Relative dates must be within the animal's life-span.  
<date field> DATE IS NOT WITHIN THE LIFESPAN OF ANIMAL  
<id>!

In the **Parturition Data** file the validation consists of the following checks with their respective warning messages:

- The parturition date may not be before the mating date.  
CONTRADICTION BETWEEN MATING AND PARTURITION DATES!
- The lactation end date may not be before the parturition date.  
CONTRADICTION BETWEEN LACTATION END AND PARTURITION DATES!

If an attempt is made to generate a duplicate data record (duplicate with respect to the record index), an error message will be generated for the General Animal Data, the Parturition Data and the Environment Data, which will stop further processing of the record until the error is corrected. For other data files a warning message will be created if an attempt is made to enter a duplicate record. For such files the duplicate record may be processed further.

- A duplicate record may not be created.  
DUPLICATE RECORD. THE RECORD INDEX EXISTS IN THIS FILE!

## 6.4 Function Key Edit

The Function Key Edit system is available during the Add a New Record and Edit a Record processes, while a data record is displayed and one of the data fields is active.

The Function Key Edit systems allows viewing and editing a record of a data file different from the one active, by pressing one of the assigned function keys.

The following function keys are used with the Function Key Edit system:

|    |   |                        |
|----|---|------------------------|
| F3 | - | General Animal Data    |
| F4 | - | Environment Data file  |
| F5 | - | Parturition Data file  |
| F6 | - | to be assigned by user |
| F7 | - | to be assigned by user |
| F8 | - | to be assigned by user |

The function keys F3, F4 and F5 are permanently assigned to the respective files shown in the above table. The function keys F6, F7 and F8 can be assigned to any other data file in the system except the Documentation Data file. The assignment can be made a permanent default using the SETUP module or a temporary setting for the duration of the update session using the System Configuration option of the UPDATE module.

Within the active Function Key Edit window, the cursor movement and edit keys described under Add a New Record can be used, except for the function keys of the Function Key Edit system itself. The Assist System (F2) is also available.

For the sake of data integrity, not all fields in the active Function Key Edit window may be edited. Fields that can be edited are preceded by a triangular marker.

Fields in the Function Key Edit window can be active in two modes, Browse mode and Edit mode. Fields that cannot be edited can only be active in Browse mode, while fields that can be edited may be active in both modes.

From the Browse mode, the RETURN key switches an active editable field to Edit mode, where the current field value can be changed. From the Edit mode, the RETURN key saves the edited value, leaves the current field and activates the next field which can be edited in Browse mode. The F9 key (jump key)

activates the next field that can be edited in Browse mode.

In order to display a record in the Function Key Edit system, an index key must be available that matches the index key of the target file. The index key is created from field values of the displayed record.

If the target file from which a record is to be displayed in the Function Key Edit system only has an Identification number as index key (e.g. F3 = General Animal Data), the value of the first ID-type field in the record is used, unless the currently active field is of ID-type, in which case the field value of the active field will be used for lookup in the target file.

The same rule applies if the target file has an index key that is a combination of an ID number and a date. The values of the first ID-type field and the first date-type field in the record are used to construct the search key for the target file, unless the currently active field is an ID-type field. In this case the search key is constructed from the value of the active ID-type field and the first date-type field in the record.

A record of the target file will generally only be displayed if the search key is found in the target file. If the search key is not found, the message 'A CORRESPONDING (data file name) RECORD WAS NOT FOUND' will be displayed on the status line.

There are two exceptions to this rule, namely for the Environment Data file (F4) and the Parturition Data file (F5). If a corresponding record for these two target files is not found, the preceding record for the same animal will be displayed.

Example:

The milk yield on the dry-off date of a cow is entered into the Milk Data file. The dry-off date should now be entered into the Parturition Data file (F5), while the milk yield record remains on the screen. The search key is constructed from the COW\_ID and the MILK\_DATE. This search key is not found in the target file (Parturition Data), and the Function Key Edit system searches for a parturition record of COW\_ID that immediately precedes MILK\_DATE. If such a record exists, it will be displayed in the Function Key Edit window, and the dry-off date can be entered into the appropriate field.

## **1 Introduction**

The LIMS VALIDATION module is written in the CLIPPER version of the dBASE language and compiled with the CLIPPER 5.0 compiler. The data set files are compatible with the dBASE III+ standard except for the index files (.NTX), which are CLIPPER specific.

The VALIDATION module is the part of the LIMS system that performs checks on the data. Any errors or inconsistencies encountered are compiled in the form of a printed list. This manual gives a detailed description of the error checks performed and their corresponding error messages.

The VALIDATION module uses the user supplied data set definitions given during the SETUP procedure. In addition, it allows you to specify certain defined constants, which are species and system dependent. These constants (e.g. minimum gestation length) are then used for data validation when the module is run.

## 2 Starting the VALIDATION Module

The filename of the executing LIMS VALIDATION module is LVALID.EXE. This file should be located in the subdirectory \LIMS on a hard disk. The PATH statement in the AUTOEXEC.BAT file should include the subdirectory \LIMS so that the VALIDATION module can be started independent of the current drive or path. For more information on the PATH statement in the AUTOEXEC.BAT file, please refer to the DOS User's Guide.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This data set name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

Use the following syntax to start the LIMS VALIDATION module:

```
LVALID [xxxx] [/y]
```

The two items in square brackets ([]) are optional parameters that can be specified when calling the VALIDATION module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling the VALIDATION module, a prompt on the initial screen will request the input of the data set name.

/y This is the video mode option. By default, the VALIDATION module will attempt to start in colour mode.

/H sets a special monochrome display which is optimized for Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be set.

/M sets monochrome mode even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor.

/L sets a special monochrome display which is optimized for liquid crystal and as plasma displays. If your computer display is of one of these types, then you should select this mode.

If the data set name is given as a parameter when the VALIDATION module is started, the initial screen will display a number of messages



which will disappear after a fixed amount of time. If no name is given a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

### 3 Main Menu

The Main Menu displays the 3 Main Menu options in a window in the centre of the display area. The currently selected option is displayed in a different colour.

The following table shows the keys for use with the Main Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the DOS level                          |

The top line of the Main Menu display contains information about the VALIDATION module and the current data set. In the lefthand corner the module name is displayed, the data set code and the data set title (given during the setup process) are shown in the middle, and the current date to the right. This line is displayed throughout execution of the VALIDATION module.

The bottom line displays status information, which in the case of the Main Menu display is the full version of the system name and the version number of the VALIDATION module. The different LIMS modules can have different version numbers, but the main version numbers (the integer part) of all LIMS modules must be the same for the different modules to work together properly.

The status line is used to display information, warning and error messages, as well as for confirmation input of certain commands.

#### 3.1 Validation Configuration

This option displays codes and defined constants which can be set by the user which will be used by the VALIDATION module during the validation process. Only constants that have non-zero values can be used for error checking. The default missing values given for the defined constants are all zero, which implies that error checks that require the defined constants are not

carried out. The following codes and constant values can be changed using the **System Configuration** option.

- Validation Process Level (1-2). This allows setting two different levels of validation. If all data have been entered into the data set with the LIMS UPDATE module, level 1 should be selected. Validation at level 2 is required if data have been imported or converted from some other system into a LIMS data set. The validation level 2 increases validation run time and should only be used when indicated. Once a converted or imported data set has been checked at level 2 and errors have been corrected, only level 1 is required if subsequent data entry is done through the UPDATE module.
- Validate with Prior Indexing. The validation process depends very much on the degree to which the index files are correct. Using this option all index files can be recreated before the validation is performed. The indexing may however require a considerable amount of time for large data sets. Indexing can therefore be omitted by setting the option code to 'F', if there is no indication that the index files might be corrupted. If a validation run is done without prior indexing and indicates many errors that cannot be found in the actual data, re-run the validation after indexing.
- Minimum Gestation Length (days). This constant defines the minimum gestation length for a normal gestation, with live offspring born.
- Maximum Gestation Length (days). This constant defines the maximum gestation length for a normal gestation, with live offspring born.
- Minimum Estrus Cycle Length (days). Defines the minimum estrus cycle length for a normal estrus cycle.
- Maximum Estrus Cycle Length (days). This constant defines the maximum estrus cycle length for a normal estrus cycle.
- Maximum Daily Weight Gain. This constant defines the maximum daily weight gain considered normal, given the same unit as the body weight recorded in the Weight Data File. This maximum must include the weight gain due to pregnancy.
- Maximum Daily Weight Loss. Defines the maximum daily weight loss considered normal, given the same unit as the body weight recorded in the Weight Data File. This maximum does not include weight loss due to parturition.
- Minimum Maturity Age for Males (days). Defines the minimum age at maturity for male animals. Maturity for a male animal in this context is defined as the ability to mate with a successful outcome.

- Minimum Maturity Age for Females (days). This constant defines the minimum age at maturity for female animals. Maturity for a female animal in this context is defined as the ability to conceive.

The codes and constants described above can be changed and the changes saved at any time. However, changes made here are only saved after a successful validation run. The default values are stored in the file `xxxxVAL.MEM` (`xxxx` = data set name). If no file with this name exists, it is created after the first validation run for the specific data set has been completed. For an initial validation it may be advisable to be rather liberal with the defined constants in order to avoid too many reported errors. The values for the defined constants can later be changed for subsequent validation runs to provide a stricter validation.

### 3.2 Validate Data Files

This option will initiate the validation process after confirmation. The screen displays the following information:

- **Previous Validation Date.** The date when a validation was last successfully performed on the present data set. When a validation is run for the first time on a data set, a blank date is displayed.
- **Previous Validation Run Time.** This is the time in hours:minutes:seconds that the previous successful validation run needed to validate the data set.
- **Days Since Last Validation.** The difference in days between the current date and the Previous Validation Date.
- **Previous Number of Records Validated.** This is the number of all records in all data files of the data set that were validated in the last successful validation run.
- **Current Number of Records to be Validated.** The sum of all records in all data files of the data set that are included in the present validation run.
- **Elapsed Time.** This shows the elapsed time in hours:minutes:seconds since the start of the validation run.

The information displayed enables the user to see the current validation status of the data set and to decide whether a new validation run is needed.

The information on validation run time and number of records involved in the validation together with the display of the elapsed time allows more efficient planning of computer time use. For large data sets the validation run time may be of considerable length and the validation might best be done during off-hours.

Data files can be excluded from validation by deactivating them, as described in the chapter Edit a File under the File Definition option of the SETUP module. However, the three main data files, General Animal Data, Environment Data and Parturition Data must be created and active for the VALIDATION module to run.

The information on the previous validation run is only updated after the validation has completed successfully (e.g. provided that no power failure occurs, etc).

After completing the validation, the system displays a message and returns to the Main Menu.

### 3.3 Print Data Validation

Using this option you can print the records of the previous validation run. The printout can be sent directly to the printer or to a file for subsequent editing and printing.

The printout consists of a header with information related to system and validation configuration and the previous validation run. Below the header follows a list of errors or inconsistencies found in the data set.

Each error listing consists of following three parts:

- The **header line**, which contains a running number, the error type and number.
- One or more **file and record key lines**, which contain the file label(s) and record key(s) of the file(s) and record(s) involved in the error or inconsistency.
- The **error message line**, which contains the specific error message.

#### 3.3.1 Print to Printer

This option prints the error listing on a printer.

#### 3.3.2 Print to File

This option requests a filename as input and prints the error listing to the specified file. The filename may include a drive designator and a path. If no file extension is specified, the default extension .PRN is given. The default filename for error listing is xxxxVAL.PRN (xxxx = data set code). The output file is in ASCII format and can be used with a word processor for further editing and printing.

## 4 Data Validation

The VALIDATION module performs all checks that are done during the UPDATE process as well as a number of additional checks. The scope of the validation depends largely on the degree to which the data definition given at SETUP is correct and complete, and on the constants defined within the VALIDATION module.

The validation process is performed in separate stages:

- All fields, whether system-defined or user-defined, in system-defined files or user-defined files, are checked in a process called Basic Field Check. This error check only uses data set definition information provided by the user during SETUP.
- Additional checks of system-defined data files are performed according to logic rules using the constants defined with the System Configuration option of VALIDATE.

The data file(s) and record(s) containing an error are identified by file label(s) and record key(s) in the error printout. In addition a specific error message indicates the field(s) involved and the nature of the error. The errors are referenced by their error number.

### 4.1 Basic Field Check

The following standard checks are performed on the value(s) of any field(s) in an existing and active data file.

BFC01: Duplicate record

**Message:** DUPLICATE RECORD

**Description:** Checks whether the combination of index field values (the record key) appears more than once within a data file. In such a case the records with the same record key are considered duplicate, irrespective of the other field values.

BFC02: Missing value allowed

**Message:** MISSING VALUE NOT ALLOWED: <field label> <field value>

**Description:** Checks whether the field value is the default missing value and the Allow Missing flag for that field is set to 'F' (false).

BFC03: Code defined

**Message:** CODE NOT DEFINED: <field label> <field value>

**Description:** Checks whether for code-type fields (categorical variables) with Field Usage code 'C1' or 'C2', the code values that appear in the data set are defined in the code definition library.

BFC04: Animal defined

**Message:** GENERAL ANIMAL RECORD NOT FOUND: <field value> <field label>

**Description:** Checks whether for ID-type fields with Field Usage code 'ID', 'IP' or 'IM', there is a General Animal Record corresponding to the ID-value. Each animal referenced in any ID-type field must be defined in the data set through its own General Animal Record.

BFC05: Sex of animal

**Message:** SEX OF ANIMAL NOT ALLOWED: <field label> <field value>

**Description:** Checks whether an animal referenced in an ID-type field has a permitted sex code. Permitted sex codes are defined in the Minimum and Maximum fields of the Data Definition Record. This setting has to be done by the user even for ID-type fields such as 'Sire Id' in the General Animal Data file, otherwise the system would allow animals of any sex to be referenced in such fields. The label and value of the ID-type field appear as part of the error message.

BFC06: Relative date within animal life-span

**Message:** DATE IS BEFORE BIRTH OF ANIMAL <id>: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DR' (relative date field) have a date that is before the birth date of the reference animal. The reference animal is determined by the value of the first ID-type field in the record. The Animal ID and the label and value of the relative date field are included in the error message.

BFC07: Relative date within animal life-span

**Message:** DATE IS AFTER DISPOSAL OF ANIMAL <id>: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DR' (relative date field) have a date that is after the disposal date of the reference animal. The reference animal is determined by

the value of the first ID-type field in the record. This check is only done for animals that are not or have not been genetically active. The Animal ID and the label and value of the relative date field appear in the error message.

BFC08: Relative date before defined minimum age of animal

**Message:** DATE IS BEFORE MIN AGE ALLOWED <id>: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DR' (relative date field) have a date that is before the defined minimum age of the reference animal. The reference animal is determined by the value of the first ID-type field in the record. The Animal ID and the label and value of the relative date field are included in the error message.

BFC09: Relative date after defined maximum age of animal

**Message:** DATE IS AFTER MAX AGE ALLOWED: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DR' (relative date field) have a date that is after the defined maximum age of the reference animal. The reference animal is determined by the value of the first ID-type field in the record. The Animal ID and the label and value of the relative date field appear in the error message.

BFC10: Absolute date before defined minimum date

**Message:** DATE IS BEFORE MIN DATE ALLOWED: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DA' (absolute date field) have a date that is before the defined minimum date.

BFC11: Absolute date after defined maximum date

**Message:** DATE IS AFTER MAX DATE ALLOWED: <field label> <field value>

**Description:** Checks whether date fields with Field Usage code 'DA' (absolute date field) have a date that is after the defined maximum date.

BFC12: Numeric value smaller than defined minimum



**Message:** VALUE IS SMALLER THAN MIN ALLOWED: <field label> <field value>

**Description:** Checks whether numeric fields with Field Usage code 'NF' have a value that is smaller than the defined minimum value.

BFC13: Numeric value larger than defined maximum

**Message:** VALUE IS LARGER THAN MAX ALLOWED: <field label> <field value>

**Description:** Checks whether numeric fields with Field Usage code 'NF' have a value that is larger than the defined maximum value.

## 4.2 General Animal Data Check

For records of the General Animal Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

GEN01: Contradiction between sex and castration date

**Message:** CONTRADICTION: <sex label> <label code> / <castration date label> <castration date>

**Description:** Checks whether animals of female sex have a non-missing castration date.

GEN02: Contradiction between sex and first estrus date

**Message:** CONTRADICTION: <sex label> <sex code> / <first estrus date label> <first estrus date>

**Description:** Checks whether animals of male sex have a non-missing first estrus date.

GEN03: Contradiction between breed and breed rule

**Message:** CONTRADICTION: <breed label> <breed code> / BREED RULE <breed rule code>

**Description:** Checks whether the actual breed code of the animal referenced in the field 'ID number' (ID) is in accordance with the breed code derived by the use of a breed rule. This check can only be performed if the sire and dam of the animal referenced under the field 'ID number' are not missing, have non-missing breed codes and a breed rule containing their specific breed codes exists.

GEN04: Missing breed rule

**Message:**

BREED NOT DEFINED BY BREED RULE: <breed label>  
<breed code>

**Description:** Checks whether the actual breed code of the animal referenced in the field 'ID number' (ID) is defined by a breed rule. This check can only be performed if the sire and dam of the animal referenced under the field 'ID number' are not missing, have non-missing breed codes and the breed definition library contains breed rules.

GEN05: Birth date and birth deviation

**Message:** CONTRADICTION: <birth date label> <birth date> / <birth deviation label> <birth deviation value>

**Description:** Checks whether the birth date (BIRTH\_DT) and the birth date deviation (BIRTH\_DV) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

GEN06: Castration date and castration date deviation

**Message:** CONTRADICTION: <castration date label> <castration date> / <castration deviation label> <castration deviation value>

**Description:** Checks whether the castration date (CAST\_DT) and the castration date deviation (CAST\_DV) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

GEN07: Weaning date and weaning date deviation

**Message:** CONTRADICTION: <weaning date label> <weaning date /weaning deviation label> <weaning deviation value>

**Description:** Checks whether the weaning date (WEAN\_DT) and the weaning date deviation (WEAN\_DV) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

GEN08: First estrus date and first estrus date deviation

**Message:** CONTRADICTION: <first estrus date label> <first estrus date > / <first estrus deviation label> <first estrus deviation value>

**Description:** Checks whether the first estrus date (OEST1\_DT) and the first estrus date deviation (OEST1\_DV) have contradictory values, i.e. that one has a missing and the other a non-

missing value, or vice versa.

GEN09: Disposal date and disposal date deviation

**Message:** CONTRADICTION: <disposal date label> <disposal date> / <disposal deviation label> <disposal deviation value>

**Description:** Checks whether the disposal date (DISP\_DT) and the disposal date deviation (DISP\_DV) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

GEN10: Disposal date and disposal reason

**Message:** CONTRADICTION: <disposal date label> <disposal date> / <disposal reason label> <disposal reason code>

**Description:** Checks whether the disposal date (DISP\_DT) and the disposal reason (DREASON) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

GEN11: Missing environment record

**Message:** NO ENVIRONMENT RECORD FOUND: <id label> <id number>

**Description:** Checks whether there exists a record in the Environment Data file for the animal referenced in the field 'ID number' (ID).

GEN12: Minimum age of dam at birth of offspring

**Message:** AGE AT BIRTH OF OFFSPRING BELOW MIN DEFINED: <dam id label> <dam id number> <age>

**Description:** Checks the age of the dam referenced in the field 'Dam ID number' (DAM\_ID) at the birth of the animal referenced in the field 'ID number' (ID). The difference between the birth dates of offspring and dam must be larger than the sum of the defined constants Minimum Maturity Age for Females and Minimum Gestation Length. This check is only done if the defined constants have non-zero values. If a corresponding record exists in the Parturition Data file, this check will only be performed when running a Parturition Data Check in order to avoid duplicate error messages.

GEN13: Minimum age of sire at birth of offspring

**Message:** AGE AT BIRTH OF OFFSPRING BELOW MIN DEFINED: <sire id label> <sire id number> <age>

**Description:** Checks the age of the sire referenced in the field 'Sire ID number' (SIRE\_ID) at the birth of the animal referenced in the field 'ID number' (ID). The difference between the birth dates of offspring and sire must be larger than the sum of the defined constants Minimum Maturity Age for Males and Minimum Gestation Length. This check is only done if the defined constants have non-zero values. If a corresponding record exists in the Parturition Data file, this check will only be performed when running a Parturition Data Check in order to avoid duplicate error messages.

GEN14: Unique parent/offspring identifications

**Message:** DUPLICATE PARENT/OFFSPRING ID: <id number>

**Description:** Checks that the identifications of the animals referenced in the fields 'ID Number' (ID), 'Sire ID Number' (SIRE\_ID) and 'Dam ID Number' (DAM\_ID) are unique with respect to each other.

GEN15: Corresponding parturition record

**Message:** GENERAL ID REC NOT LINKED TO PARTURITION REC: <id label> <id number>

**Description:** If the animal referenced in the field 'ID Number' (ID) has a corresponding parturition record (by Dam ID and Birth Date/Parturition Date), this checks whether it also appears as an offspring in that parturition record (=is linked).

GEN16: Corresponding parturition record

**Message:** NO CORRESPONDENCE: <parity label> <parity value > / <parity label> <parity value>

**Description:** If a corresponding and linked parturition record exists (see above paragraph), this checks whether the corresponding fields 'Parity' (PARITY) contain the same values in both records.

GEN17: Corresponding parturition record

**Message:** NO CORRESPONDENCE: <birth type label> <birth type code> / <number born label> <number born value>

**Description:** If a corresponding and linked parturition record exists (see above, GEN15), this checks whether the corresponding fields 'Birth Type' (BIRTH\_TY) and 'Offspring Born'

(NO\_BORN) contain the same values.

GEN18: Corresponding parturition record

**Message:** NO CORRESPONDENCE: <sire id label >< sire id number>  
/ <sire id label> <sire id number>

**Description:** If a corresponding and linked parturition record exists (see above, GEN15), this checks whether the corresponding fields ‘Sire ID Number’ (SIRE\_ID) contain the same values.

GEN19: Corresponding parturition record

**Message:** NO CORRESPONDENCE: <birth deviation label> < sire deviation value> / <parturition deviation label> <parturition deviation value>

**Description:** If a corresponding and linked parturition record exists (see above, GEN15), this checks whether the corresponding fields ‘Birth Dev.’ (BIRTH\_DV) and ‘Parturition Dev.’ (PART\_DV) contain the same values in both records.

GEN20: Genetically active flag

**Message:** GENETICALLY ACTIVE CODE NOT ALLOWED BEFORE DISPOSAL OR CASTRATION: <genetically active code>

**Description:** Checks whether a non-missing genetically active code is set for an animal that has neither a non-missing disposal date nor a non-missing castration date. A genetically active flag can only be set after an animal has been either disposed or castrated.

GEN21: Identification format

**Message:** CONTRADICTION: GLOBAL ID FORMAT <global id format> / <animal id>

**Description:** Checks whether the value in the field ‘ID’ (Animal ID number) corresponds to the globally defined identification format.

### 4.3 Environment Data Check

For records of the Environment Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

ENV01: Birth environment record

**Message:**

CONTRADICTION: <entry date label> <entry date> / <birthdate label> <birth date>

**Description:** Checks environment records with the 'Entry Reason' (EREASON) equal to the globally defined code Entry Due to Birth. For such environment records the 'Env. Entry Date' (ENVIR\_DT) must be the 'Birth Date' (BIRTH\_DT) given in the General Animal Record.

ENV02: Consecutive environment records

**Message:** DUPLICATE ENVIRONMENT CODES

**Description:** Checks consecutive environment records. These must not have equal environment codes (ENVIRON1, ENVIRON2, ENVIRON3).

#### 4.4 Parturition Data Check

For records of the Parturition Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

PAR01: Contradiction between mating and parturition date

**Message:** CONTRADICTION: <mating date label> <mating date> / <parturition date label> <parturition date>

**Description:** Checks whether the mating date is before the parturition date.

PAR02: Minimum age of dam at mating

**Message:** AGE AT MATING BELOW MIN DEFINED: <dam id label> <dam id number> <age>

**Description:** Checks the age of the dam referenced in the field 'Dam ID Number' (DAM\_ID) at the date of mating. The difference between the mating date and the birth date of the dam must be larger than the defined constant Minimum Maturity Age for Females. This check is only done if the defined constant has a non-zero value.

PAR03: Minimum age of sire at mating

**Message:** AGE AT MATING BELOW MIN DEFINED: <sire id label> <sire id number> <age>

**Description:** Checks the age of the sire referenced in the field 'Sire ID Number' (SIRE\_ID) at the date of mating. The difference between the mating date and the birth date of the sire must be larger than the defined constant Minimum Maturity Age for

Males. This check is only done if the defined constant has a non-zero value.

PAR04: Gestation length

**Message:** GESTATION LENGTH SHORTER THAN MIN DEFINED:  
<gestation length>

**Description:** Checks whether the difference between the parturition date and the mating date (=gestation length) is less than the defined constant Minimum Gestation Length. This check is only done if there are live born offspring in this parturition (NO\_ALIVE 0) and if the defined constant has a non-zero value.

PAR05: Gestation length

**Message:** GESTATION LENGTH LONGER THAN MAX DEFINED:  
gestation length

**Description:** Checks whether the difference between the parturition date and the mating date (=gestation length) is greater than the defined constant Maximum Gestation Length. This check is only done if the defined constant has a non-zero value.

PAR06: Minimum age of dam at parturition

**Message:** AGE AT PARTURITION BELOW MIN DEFINED: <dam id label> <dam id number> <age>

**Description:** Checks the age of the dam referenced in the field 'Dam ID Number' (DAM\_ID) at the parturition date (PART\_DT). The difference between the parturition date and the birth date of the dam must be larger than the sum of the defined constants Minimum Maturity Age for Females and Minimum Gestation Length. This check is only done if the defined constants have non-zero values.

PAR07: Minimum age of sire at parturition

**Message:** AGE AT PARTURITION BELOW MIN DEFINED: <sire id label> <sire id number> <age>

**Description:** Checks the age of the sire referenced in the field 'Sire ID Number' (SIRE\_ID) at the parturition date (PART\_DT). The difference between the parturition date and the birth date of the sire must be larger than the sum of the defined constants Minimum Maturity Age for Males and Minimum Gestation Length. This check is only done if the defined constants have non-zero values.

PAR08: Lactation end date

**Message:** CONTRADICTION: <lactation end date label> <lactation end date> / <parturition date label> <parturition date>

**Description:** Checks whether the lactation end date (LEND\_DT) is before the parturition date (PART\_DT) of the same parturition record.

PAR09: Lactation end date and lactation end type

**Message:** CONTRADICTION: <lactation end date label> <lactation end date> / <lactation end type label> <lactation end type>

**Description:** Checks whether the lactation end date (LEND\_DT) and the lactation end type (LEND\_TY) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

PAR10: Lactation end date with existing disposal date

**Message:** MISSING DATE NOT ALLOWED: <lactation end date label> > FOR DISPOSED ANIMAL

**Description:** Checks whether the lactation end date (LEND\_DT) has a missing value and the animal referenced under 'Dam ID Number' (DAM\_ID) has a disposal date in its General Animal Data record.

PAR11: Lactation end type with existing disposal date

**Message:** MISSING CODE NOT ALLOWED: <lactation end type label> FOR DISPOSED ANIMAL

**Description:** Checks whether the lactation end type (LEND\_TY) has a missing value and the animal referenced under 'Dam ID Number' (DAM\_ID) has a disposal date in its General Animal Data record.

PAR12: Parturition before first estrus date

**Message:** CONTRADICTION: <parturition date label> <parturition date> / <first estrus date label> <first estrus date>

**Description:** Checks whether the date of a parturition record is before the 'First Estrus' date (OEST1\_DT) defined in the General Animal Data Record.

PAR13: Consecutive parturition records

**Message:** PARITY OUT OF SEQUENCE: <parity label> <parity value> / <subsequent parity value>



**Description:** Checks whether the parity fields (PARITY) of consecutive parturition records of one and the same dam (DAM\_ID) contain values that are in sequence.

PAR14: Consecutive parturition records

**Message:** MISSING DATE IN PRECEDING RECORD NOT ALLOWED: <lactation end date label>

**Description:** Checks whether the lactation end date (LEND\_DT) of a preceding parturition record of one and the same dam (DAM\_ID) has a missing value.

PAR15: Consecutive parturition records

**Message:** MISSING CODE IN PRECEDING RECORD NOT ALLOWED: <lactation end type label>

**Description:** Checks whether the lactation end type field (LEND\_TY) of a preceding parturition record of one and the same dam (DAM\_ID) contains a missing value.

PAR16: Consecutive parturition records

**Message:** PRECEDING: <lactation end date label> <lactation end date>  
AFTER <parturition date label> <parturition date>

**Description:** Checks whether the lactation end date (LEND\_DT) of a preceding parturition record of one and the same dam (DAM\_ID) is after the parturition date (PART\_DT) of the following record.

PAR17: Consecutive parturition records

**Message:** PARTURITION INTERVAL BELOW MIN DEFINED:  
<parturition interval>

**Description:** Checks the difference between the parturition dates (PART\_DT) of consecutive parturition records (=parturition interval) of one and the same dam (DAM\_ID), if any offspring were born alive (NO\_ALIVE > 0) in the previous parturition record. The parturition interval must be larger than the sum of the defined constants Minimum Gestation Length and Minimum Estrus CycleLength. The check is only performed if both defined constants have non-zero values.

PAR18: Sire genetically active at mating

**Message:** NOT GENETICALLY ACTIVE: <sire id label> <sire id number> <mating date label> <mating date>

**Description:** Checks whether the animal referenced under 'Sire ID Number' (SIRE\_ID) is genetically active at mating (MATE\_DT). If the sire is disposed (DISP\_DT) or castrated (CAST\_DT) at the time of the mating, the field 'Genetically Active' (G\_ACTIVE) in its General Animal Data record must contain the globally defined code either for 'is genetically active' or for 'was genetically active'.

PAR19: Number of animals born and maximum litter size

**Message:** OUT OF RANGE: <number of animals born label> <number of animals born> <1-maximum litter size>

**Description:** Checks whether the field 'Offspring Born' (NO\_BORN) contains a non-missing value that is smaller than 1 or larger than the globally defined Maximum Litter Size.

PAR20: Number of animals born alive and maximum litter size

**Message:** OUT OF RANGE: <number of animals born alive label> <number of animals born alive> <0-maximum litter size>

**Description:** Checks whether the field 'No. Born Alive' (NO\_ALIVE) contains a non-missing value that is smaller than 0 or larger than the globally defined Maximum Litter Size.

PAR21: Number of animals born and number of animals born alive

**Message:** CONTRADICTION: <number of animals born alive label> <number of animals born alive> <number of animals born label> <number of animals born>

**Description:** Checks whether the field 'No. Born Alive' (NO\_ALIVE) contains a value that is larger than the value contained in the field 'Offspring Born' (NO\_BORN).

PAR22: Number of animals to be identified and offspring ID count

**Message:** CONTRADICTION:<number of animals born (alive) label> <number of animals born (alive)> PROGENY ID COUNT <progeny id count>

**Description:** Checks whether the number of offspring to be identified in a parturition record corresponds with the number of non-missing offspring identifications (PRG\_IDnn). The number of offspring to be identified depends on the setting of the global flag Record Dead Animals. If this flag is set to 'F' (false), the number of offspring to be identified equals 'No. Born Alive' (NO\_ALIVE), otherwise it equals 'Offspring Born' (NO\_BORN).

PAR23: Unique parent/offspring identifications

**Message:** DUPLICATE PARENT/OFFSPRING ID: < id number>

**Description:** Checks that the identifications of the animals referenced in the fields ‘Dam ID Number’ (DAM\_ID), ‘Sire ID Number’ (SIRE\_ID) and the offspring identification fields ‘Offspring ID’ (PRG\_IDnn) are unique with respect to each other.

PAR24: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <dam id label > <dam id number> / <dam id label> <dam id number>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same ‘Dam ID Number’ (DAM\_ID) as the field ‘Dam ID Number’ (DAM\_ID) in the parturition record.

PAR25: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <parturition date label> <parturition date> / <birth date label> <birth date>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same value in the field ‘Birth Date’ (BIRTH\_DT) as the field ‘Parturition Dt.’ (PART\_DT) in the parturition record.

PAR26: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <parity label> <parity value> / <parity label> <parity value>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same value in the field ‘Parity’ (PARITY) as the field ‘Parity’ (PARITY) in the parturition record.

PAR27: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <number of born label> <number of born> / <birth type label> <birth type>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same value in the field ‘Birth Type’ (BIRTH\_TY) as the field ‘Offspring Born’ (NO\_BORN) in the parturition record.

PAR28: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <parturition date deviation label> <parturition date deviation value> / <birth date deviation label> <birth date deviation value>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same value in the field 'Birth Dev.' (BIRTH\_DV) as the field 'Parturition Dev.' (PART\_DV) in the parturition record.

PAR29: Corresponding General Animal Data record of offspring

**Message:** NO CORRESPONDENCE: <sire id label> <sire id number> / <sire id label> <sire id number>

**Description:** Checks whether the corresponding General Animal Data record for an offspring contains the same 'Sire ID Number' (SIRE\_ID) as the field 'Sire ID Number' (SIRE\_ID) in the parturition record.

PAR30: Corresponding Environment Data record of offspring

**Message:** INVALID CODE FOR BIRTH ENVIRONMENT RECORD: <environment entry reason label> <environment entry reason code>

**Description:** Checks whether the corresponding (by offspring ID and parturition date) Environment Data record for an offspring contains the 'Entry Reason' (EREASON) code as globally defined for 'Entry Due to Birth'.

PAR31: Corresponding Environment Data record of offspring

**Message:** CONTRADICTION: <parturition deviation label> <parturition deviation value> <environment deviation label> <environment deviation value>

**Description:** Checks whether the corresponding (by offspring ID and parturition date) Environment Data record for an offspring contains the same 'Env. Entry Dev.' (ENVIR\_DV) value as the field 'Parturition Dev.' (PART\_DV) in the parturition record.

PAR32: Corresponding Environment Data record of offspring

**Message:** BIRTH ENVIRONMENT RECORD NOT FOUND: <offspring id label> <offspring id number>

**Description:** Checks whether the corresponding (by offspring ID and parturition date) Environment Data record for an offspring exists.

## 4.5 Estrus Data Check

For records of the Estrus Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

EST01: Estrus during the gestation period

**Message:** ESTRUS AT PARTURITION OR DURING GESTATION PERIOD

**Description:** Checks whether the date of an estrus observation record coincides with a parturition date or a gestation period. If a mating date is not defined or is missing in a parturition record, the gestation period is calculated from the parturition date and the defined constant Minimum Gestation Length.

EST02: Estrus before first estrus date

**Message:** CONTRADICTION: <estrus date label> <estrus date> <first estrus date label> <first estrus date>

**Description:** Checks whether the date of an estrus observation record is before the 'First Estrus' date (OEST1\_DT) defined in the General Animal Data Record.

EST03: Interval between consecutive estrus observations

**Message:** INTERVAL BETWEEN CONSECUTIVE ESTRUS OUT OF RANGE: <estrus interval>

**Description:** Checks the interval between consecutive estrus observations. The interval between consecutive estrus observations may be 1 day, or equal to, or longer than the defined constant Minimum Estrus Cycle Length.

## 4.6 Mating Data Check

For records of the Mating Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

MAT01: Unique identifications

**Message:** DUPLICATE ID: <dam id label> <dam id number> <sire id label> <sire id number>

**Description:** Checks that the identifications of the animals referenced in the fields 'Dam ID Number' (DAM\_ID) and 'Sire ID Number' (SIRE\_ID) are not equal.

MAT02: Mating during the gestation period

**Message:** MATING AT PARTURITION OR DURING GESTATION PERIOD

**Description:** Checks whether the date of a mating record coincides with a parturition date or a gestation period. If a mating date is not defined or is missing in a parturition record, the gestation period is calculated from the parturition date and the defined constant Minimum Gestation Length.

MAT03: Sire genetically active at mating

**Message:** NOT GENETICALLY ACTIVE: <sire id label> <sire id number> <mating date label> <mating date>

**Description:** Checks whether the animal referenced under 'Sire ID Number' (SIRE\_ID) is genetically active at mating (MATE\_DT). If the sire is disposed (DISP\_DT) or castrated (CAST\_DT) at the date of mating, the field 'Genetically Active' (G\_ACTIVE) in its General AnimalData record must contain the globally defined code either for 'is genetically active' or for 'was genetically active'.

MAT04: Mating before first estrus date

**Message:** CONTRADICTION: <mating date label> <mating date> <first estrus date label> <first estrus date>

**Description:** Checks whether the date of a mating record is before the 'First Estrus' date (OEST1\_DT) defined in the General Animal Data Record.

MAT05: Interval between consecutive matings

**Message:** INTERVAL BETWEEN CONSECUTIVE MATINGS OUT OF RANGE: <mating interval>

**Description:** Checks the interval between consecutive matings. The interval between consecutive matings may be 1 day, or equal to, or longer than the defined constant Minimum Estrus Cycle Length.

## 4.7 Pregnancy Diagnosis Data Check

For records of the Pregnancy Diagnosis Data file, the following error check is performed in addition to the checks described under Basic Field Check.

PRD01: Pregnancy diagnosis check before first estrus date

**Message:** CONTRADICTION: <pregnancy diagnosis date label>  
<pregnancy diagnosis date> <frist estrus date label> <first estrus date >

**Description:** Checks whether the date of a pregnancy diagnosis record is before the 'First Estrus' date (OEST1\_DT) defined in the General Animal Data Record.

#### 4.8 Milk Data Check

For records of the Milk Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

MLK01: Period yield and period length

**Message:** CONTRADICTION: <period yield label > <period yield> / <period length label> <period length>

**Description:** Checks whether the period yield (YIELD\_P) and the period length (LENGTH\_P) have contradictory values, i.e. that one has a missing and the other a non-missing value, or vice versa.

MLK02: Milk yield before first estrus date

**Message:** CONTRADICTION:<milk date label> <milk date> <first estrus datelabel> <first estrus date >

**Description:** Checks whether the date of a milk record is before the 'First Estrus' date (OEST1\_DT) defined in the General Animal Data Record.

MLK03: Milk record outside lactation period

**Message:** MILK RECORD NOT WITHIN A LACTATION PERIOD

**Description:** Checks whether the date of a milk record falls within a lactation period defined by the corresponding fields 'Parturition Dt.' (PART\_DT) and 'Lact. End Date' (LEND\_DT) in the parturition records.

MLK04: Overlapping milk periods

**Message:** OVERLAPPING PERIODS IN CONSECUTIVE RECORDS: <period length label>

**Description:** Checks whether the milk periods in consecutive milk records are overlapping.

#### 4.9 Weight Data Check

For records of the Weight Data file, the following error checks are performed in addition to the checks described under Basic Field Check.

WGT01: Daily weight gain

**Message:** DAILY WEIGHT GAIN EXCEEDS MAX DEFINED:  
<daily weight gain>

**Description:** Checks the daily weight gain between two consecutive non-missing weight records. The daily weight gain may not exceed the defined constant Maximum Daily Weight Gain.

WGT02: Daily weight loss

**Message:** DAILY WEIGHT LOSS EXCEEDS MAX DEFINED:  
<daily weight loss>

**Description:** Checks the daily weight loss between two consecutive non-missing weight records. The daily weight loss may not exceed the defined constant Maximum Daily Weight Loss. If female animals have a parturition between two consecutive weight records, these weight records are not checked for weight loss. If the date of a weight record coincides with a parturition date, it is assumed to be a postpartum weight.



# 1 Introduction

The LIMS REPORT module is written in the CLIPPER version of the dBASE language and compiled with the CLIPPER 5.0 compiler. The data set files are compatible with the dBASE III+ standard except for the index files (.NTX), which are CLIPPER specific.

The REPORT module can compile reports on the animals and the data in a LIMS data set. There are several categories of reports based on the type of data presented in the reports. Within each category you can define individual reports according to your specific needs and requirements.

The following table shows the LIMS files (xxxx = data set name) used to store the report definitions and the report data:

| file name   | file content  |
|-------------|---|
| xxxxRPT.DBF | identifying information on all user-defined reports |
| xxxxR01.MEM | user-defined report definitions                     |
| :           |   |
| :           |   |
| xxxxR99.MEM |   |
| xxxxR01.DBF | report data   |
| :           |   |
| :           |   |
| xxxxR99.DBF |   |

A maximum of 99 user-defined report definitions can be created, with multiple reports within each report category. Report definitions can be edited and deleted.

## 2 Starting the REPORT Module

The filename of the executing REPORT module is LREPORT.EXE. This file should be located in the subdirectory \LIMS on a hard disk. The PATH statement in the AUTOEXEC.BAT file should include the subdirectory \LIMS so that the REPORT module can be started independent of the current drive or path. For more information on the PATH statement in the AUTOEXEC.BAT file, please refer to the DOS User's Guide.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This data set name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

Use the following syntax to start the LIMS REPORT module:

```
LREPORT [xxxx] [/y]
```

The two items in square brackets ( [ ] ) are optional parameters that can be specified when calling the REPORT module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling the REPORT module, a prompt on the initial screen will request the input of the data set name.

/y This is the video mode option. By default, the REPORT module will attempt to start in colour mode.

/H sets a special monochrome display which is optimised for Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be set.

/M sets monochrome mode even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor.

/L sets a special monochrome display which is optimised for liquid crystal and gas plasma displays. If your computer display is of one of these types, then you should select this mode.

If the data set name is given as a parameter when the REPORT module is started, the initial screen will display a number of messages which will disappear after a fixed amount of time. If no name is given a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

### 3 Main Menu

The Main Menu displays the 3 Main Menu options in a window in the centre of the display area. The currently selected option is displayed in a different colour.

The following table shows the keys for use with the Main Menu and their functions:

|          |   |
|----------|---|
| Spacebar | - selects the following option                      |
| Dnarrow  | - selects the following option                      |
| Uparrow  | - selects the preceding option                      |
| <number> | - selects and activates the option with that number |
| End      | - selects the last option                           |
| Home     | - selects the first option                          |
| Return   | - activates the selected option                     |
| Esc      | - returns to the <b>DOS</b> level                   |

The top line of the Main Menu display contains information about the REPORT module and the current data set. In the lefthand corner the module name is displayed, the data set code and the data set title (given during the setup process) are shown in the middle, and the current date to the right. This line is displayed throughout execution of the REPORT module.

The bottom line displays status information, which in the case of the Main Menu display is the full version of the system name and the version number of the REPORT module. The different LIMS modules can have different version numbers, but the main version numbers (the integer part) of all LIMS modules must be the same for the different modules to work together properly.

The status line is used to display information, warning and error messages, as well as for confirmation input of certain commands.

#### 3.1 Report Definition

Selecting this option brings up the three options of the Report Definition Menu: add a new report, edit an already defined report and delete a defined report. The different types or categories of reports that can be compiled by the LIMS REPORT module are discussed in separate chapters under the heading **Report Categories**.

### 3.1.1 Add a New Report

This option displays a list of report categories. A report category can be selected by moving the highlight bar with the up- and down-arrow keys. When a category is selected and the Return key pressed, an empty report configuration display appears on the screen. The appearance of the display will differ according to the report category selected. A new report can be defined by supplying the details requested in the configuration display. The procedure for making a report definition within each category is described further in the respective chapters under the heading **Report Categories**.

### 3.1.2 Modify a Report

This option displays a list of report titles and the dates they were last run. Individual reports can be selected by moving the highlight bar with the up- and down-arrow keys. Pressing the Return key brings up the report configuration display for the selected report for modification. For details on how to modify reports belonging to a certain category, see the respective chapter under **Report Categories**.

### 3.1.3 Delete a Report

This option displays a list of report titles and the dates they were last run. Individual reports can be selected by moving the highlight bar with the up- and down-arrow keys. Pressing the Return key brings up the report configuration screen for the selected report. A prompt on the status line will request confirmation before the selected report is deleted.

## 3.2 Report Generation

This option displays a list of report titles and the dates they were last run. Individual reports can be activated by moving the highlight bar with the up- and down-arrow keys. The following report selections can be made using the key combinations below:

|          |   |                                   |
|----------|---|-----------------------------------|
| ALT-T    | - | select all reports                |
| ALT-U    | - | deselect all reports              |
| Spacebar | - | select/deselect the active report |
| CTRL-T   | - | select the active report          |
| CTRL-U   | - | deselect the active report        |

Selected reports are marked with a triangular marker. Pressing the Return key brings up the report configuration display for the active report. The F10 key saves the report markers and initiates the report generation process after confirmation.

During report generation a status window displays the title of the current report being generated and a counter indicating the number of reports generated as well as the total number of reports to be generated. It also displays a percentage counter which shows the proportion that has been completed of the report currently being generated. The finished reports are saved in dBASE data file format for subsequent printing.

The dBASE output files can also be used directly or they can be converted to ASCII format using the **File Conversion** option of the EXTRACT module.

### 3.3 Report Printing

With this option you can make a printout of the reports that have been selected and created using the option **Report Selection**. The printout can be sent directly to the printer or to a file for subsequent editing and printing.

The printout consists of separate reports according to the specifications made. The printed reports usually contain a header with the report title, the report run date and other details of the report configuration.

#### 3.3.1 Print to Printer

This option prints the reports on a printer. Depending on the user-defined report configuration, printing width of the reports may exceed 80 characters.

#### 3.3.2 Print to File

This option displays a prompt requesting a filename and will print the reports to that file. The filename may include a drive designator and a path. If no file extension is specified, the default extension .PRN is given. The default filename for reports is xxxxRPT.PRN (xxxx = data set name). The output file is in ASCII format and can be used with a word processor for further editing and printing.

## 4 Report Types

### 4.1 Population Structure

Reports belonging to this category contain a breakdown of a population on a specified date, by age and a combination of attributes described by user-defined classification fields. Only animals alive on the user-defined target date are counted. Animals are included in the report on the basis of data contained in the General Animal Data file in the fields BIRTH\_DT (Birth Date) and DISP\_DT (Disposal Date). Before the report definition can be saved (by pressing the F10 key), it will be checked for errors and inconsistencies and any errors found will be indicated by error messages displayed on the status line.

The columns of this type of report correspond to user-defined age classes, while the rows represent categories defined by a combination of classification fields selected by the user. Each cell contains a count of the number of animals alive on the target date that fall into each category. Row and column totals are included in the printed report.

The following settings and selections define a report belonging to this category:

- **Report Title:** this is a user-defined title of up to 40 characters that identifies each individual report. If no title is given, the following error message will be displayed: ERROR - A REPORT TITLE IS REQUIRED!
- **File:** displays the file name of the dBASE data file (.DBF) and memory variable file (.MEM) where the report data and the report configuration are stored. The file name is automatically assigned by the REPORT module and cannot be edited by the user.
- **Target Date:** this is a user-defined date which is the reference date for the report. Only those animals that are alive on the target date (birth date before or coinciding with the target date and the target date before disposal date) are included in the count. Animals with missing birth date are excluded and animals with missing disposal date are assumed to be alive at any date at or after their birth date. If no target date is given, the following error message will be displayed: ERROR - TARGET DATE IS REQUIRED!

Age Classification (days)

- **From Age - To Age:** this pair of values allow you to define age classes. Animals are classified into the user-defined age classes according to their age on the Target Date. Age classes are displayed as columns in a population structure report. The limits of the age classes are given in days

and are inclusive. Animals with a missing birth date cannot be classified into age classes and are therefore excluded from the report. In order to define a complete set of age classes, the From Age value in the lowest age class must be 0, age classes must be consecutive without gaps and the To Age value of the highest age class should be a value high enough to include the oldest possible animal (e.g. 9999). Age classes with a From Age and a To Age value of 0 are considered undefined and are not used when compiling a report. If no age class is defined, the following error message will be displayed: ERROR - AGE CLASSIFICATION IS REQUIRED! If the To Age value of an age class is smaller than or equal to the From Age value, the following error message will be displayed: ERROR - THE 'TO' AGE MUST BE GREATER THAN THE 'FROM' AGE! If consecutive age classes are not in ascending order or if consecutive age classes overlap, the following error message will be displayed: ERROR - THE 'FROM' AGE MUST BE GREATER THAN THE PREVIOUS 'TO' AGE! An undefined age class located between defined age classes is automatically removed from the age class list.

#### Group Classification (code fields)

- **Field Name:** this is a code-type field selected by the user. A field name can either be typed or selected from the FIELD LIST selection list which can be activated by pressing the F2 function key. A field name is transferred from the FIELD LIST by moving the highlight bar with the up- and down-arrow keys and pressing the Return key with the cursor on the selected field. The order in which field names are entered defines the hierarchical order in which the fields are combined into categories in the report. If no field name is given, the following error message will be displayed: ERROR - A GROUP FIELD NAME IS REQUIRED! Codes can be selected for a currently active field from the CODE LIST which is activated by pressing the F3 function key. Individual codes are activated by moving the highlight bar with the up- and down-arrow keys. The following key combinations are used to make a code selection:

|          |                                   |
|----------|-----------------------------------|
| ALT-T    | - select all codes                |
| ALT-U    | - unselect all codes              |
| Spacebar | - select/unselect the active code |
| CTRL-T   | - select the active code          |
| CTRL-U   | - unselect the active code        |

Selected codes are marked with a triangular marker. Pressing the Return key when the CODE LIST is active saves the selection of code values. In addition

to the selected code values an additional code with the value <Oth/Miss> is created for each selected field. This additional code comprises the non-selected and missing code values. The rows of a population structure report correspond to all the possible code combinations of the selected fields.

- **Label:** this is the user-defined field label that corresponds to a selected Field Name. The label is automatically displayed when a valid Field Name is defined.
- **Codes Selected:** displays the number of selected codes out of the number of defined codes for the corresponding selected field. The display of the number of defined codes is updated when a Field Name has been selected, the display of the number of selected codes is updated when the codes have been selected from the CODE LIST.

## 4.2 Population Dynamics: Entry

Reports belonging to this category contain a breakdown of the number of animals moving into/within a population during a specified time period, by age and a combination of attributes described by user-defined classification fields. Only animals moving into/within the population during the user-defined time period are counted. Animals are included in the report on the basis of data contained in the Environment Data file in the field ENVIR\_DT (Environment Entry Date). Before the report definition can be saved (by pressing the F10 key), it will be checked for errors and inconsistencies and any errors found will be indicated by error messages displayed on the status line.

The columns of this type of report correspond to user-defined age classes while the rows represent categories defined by a combination of classification fields selected by the user. Each cell contains a count of the number of animals moving into/within the population during the specified time period that fall into each category. Row and column totals are included in the printed report.

The following settings and selections define a report belonging to this category:

- **Report Title:** this is a user-defined title of up to 40 characters that identifies each individual report. If no title is given, the following error message will be displayed: ERROR - A REPORT TITLE IS REQUIRED!
- **File:** displays the file name of the dBASE data file (.DBF) and memory variable file (.MEM) where the report data and the report configuration are stored. The file name is automatically assigned by the REPORT module and cannot be edited by the user.



- **Start Date, End Date, Period:** these determine the user-defined period. The Period is calculated from the given Start and End Dates. If either the Start or the End Date is missing, it will be calculated by specifying a Period. If the report is subsequently modified, with all three values non-missing, the period length will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if either the Start Date or the End Date is deleted. Only animals that moved into/within the population during the user-defined period (Start Date before or coinciding with Environment Entry Date and Environment Entry Date before or coinciding with the End Date) are counted. Animals with missing birth date or missing Environment Record are excluded from the report. If no Start Date is given, the following error message will be displayed: ERROR - A START DATE IS REQUIRED! Likewise, failure to specify an End Date will result in the following error message: ERROR - AN END DATE IS REQUIRED! If the End Date is not later than the Start Date, the following error message will be displayed: ERROR - INVALID RANGE BETWEEN START AND END DATES!

#### Age Classification (days)

- **From Age - To Age:** this pair of values allow you to define age classes. Animals are classified into the user-defined age classes according to their age at entry. Age classes are displayed as columns in a population dynamics report. The limits of the age classes are given in days and are inclusive. Animals with a missing birth date cannot be classified into age classes and are therefore excluded from the report. In order to define a complete set of age classes, the From Age value in the lowest age class must be 0, age classes must be consecutive without gaps and the To Age value of the highest age class should be a value high enough to include the oldest possible animal (e.g. 9999). Age classes with a From Age and a To Age value of 0 are considered undefined and are not used when compiling a report. If no age class is defined, the following error message will be displayed: ERROR - AGE CLASSIFICATION IS REQUIRED! If the To Age value of an age class is smaller than or equal to the From Age value, the following error message will be displayed: ERROR - THE 'TO' AGE MUST BE GREATER THAN THE 'FROM' AGE! If consecutive age classes are not in ascending order or if consecutive age classes overlap, the following error message will be displayed: ERROR - THE 'FROM' AGE MUST BE GREATER THAN THE PREVIOUS 'TO' AGE! An undefined age class located between defined age classes is automatically removed from the age class list.

## Group Classification (code fields)

- **Field Name:** this is a code-type field selected by the user. A field name can either be typed or selected from the FIELD LIST selection list which can be activated by pressing the F2 function key. A field name is transferred from the FIELD LIST by moving the highlight bar with the up- and down-arrow keys and pressing the Return key with the cursor on the selected field. The order in which field names are entered defines the hierarchical order in which the fields are combined into categories in the report. If no field name is given, the following error message will be displayed: ERROR - A GROUP FIELD NAME IS REQUIRED! Codes can be selected for a currently active field from the CODE LIST which is activated by pressing the F3 function key. Individual codes are activated by moving the highlight bar with the up- and down-arrow keys. The following key combinations are used to make a code selection:

|          |   |                                 |
|----------|---|---------------------------------|
| ALT-T    | - | select all codes                |
| ALT-U    | - | unselect all codes              |
| Spacebar | - | select/unselect the active code |
| CTRL-T   | - | select the active code          |
| CTRL-U   | - | unselect the active code        |

Selected codes are marked with a triangular marker. Pressing the Return key when the CODE LIST is active saves the selection of code values. In addition to the selected code values an additional code with the value <Oth/Miss> is created for each selected field. This additional code comprises the non-selected and missing code values. The rows of a population dynamics report correspond to all the possible code combinations of the selected fields.

- **Label:** this is the user-defined field label that corresponds to a selected Field Name. The label is automatically displayed when a valid Field Name is defined.
- **Codes Selected:** displays the number of selected codes out of the number of defined codes for the corresponding selected field. The display of the number of defined codes is updated when a Field Name has been selected, the display of the number of selected codes is updated when the codes have been selected from the CODE LIST.
- **Report First Environment Entry Only:** (Y = yes, N = no) determines whether only new entries into the population will be considered, or whether movements between environments within the population will be

considered as well. The first record of each animal in the Environment Data file contains the information about the animal's entry into the population. The subsequent record contains information of the animal's movement between different environments within the population. If the report is compiled on basis of the first environment record only, each animal can appear only once in the count. Otherwise an animal can appear more than once. If, for example, environment is defined as the farmer to which an animal belongs, an animal can move twice between farmers during the reporting period with each move being counted.

### 4.3 Population Dynamics: Disposal

Reports belonging to this category contain a breakdown of the number of animals leaving a population during a specified time period, by age and a combination of attributes described by user-defined classification fields. Only animals leaving the population during the user-defined time period are counted. Animals are included in the report on the basis of data contained in the General Animal Data file in the field DISP\_DT (Disposal Date). Before the report definition can be saved (by pressing the F10 key), it will be checked for errors and inconsistencies and any errors found will be indicated by error messages displayed on the status line.

The columns of this type of report correspond to user-defined age classes while the rows represent categories defined by a combination of classification fields selected by the user. Each cell contains a count of the number of animals leaving the population during the specified time period that fall into each category. Row and column totals are included in the printed report.

The following settings and selections define a report belonging to this category:

- **Report Title:** this is a user-defined title of up to 40 characters that identifies each individual report. If no title is given, the following error message will be displayed: ERROR - A REPORT TITLE IS REQUIRED!
- **File:** displays the file name of the dBASE data file (.DBF) and memory variable file (.MEM) where the report data and the report configuration are stored. The file name is automatically assigned by the REPORT module and cannot be edited by the user.
- **Start Date, End Date, Period:** these determine the user-defined period. The Period is calculated from the given Start and End Dates. If either the Start or the End Date is missing, it will be calculated by specifying a Period. If the report is subsequently modified, with all three values non-missing, the period length will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if

either the Start Date or the End Date is deleted. Only animals that left the population during the user-defined period (Start Date before or coinciding with Disposal Date and Disposal Date before or coinciding with the End Date) are counted. Animals with missing birth date are excluded from the report. If no Start Date is given, the following error message will be displayed: **ERROR - A START DATE IS REQUIRED!** Likewise, failure to specify an End Date will result in the following error message: **ERROR - AN END DATE IS REQUIRED!** If the End Date is not later than the Start Date, the following error message will be displayed: **ERROR - INVALID RANGE BETWEEN START AND END DATES!**

#### Age Classification (days)

- **From Age - To Age:** this pair of values allow you to define age classes. Animals are classified into the user-defined age classes according to their age at disposal. Age classes are displayed as columns in a population dynamics report. The limits of the age classes are given in days and are inclusive. Animals with a missing birth date cannot be classified into age classes and are therefore excluded from the report. In order to define a complete set of age classes, the From Age value in the lowest age class must be 0, age classes must be consecutive without gaps and the To Age value of the highest age class should be a value high enough to include the oldest possible animal (e.g. 9999). Age classes with a From Age and a To Age value of 0 are considered undefined and are not used when compiling a report. If no age class is defined, the following error message will be displayed: **ERROR - AGE CLASSIFICATION IS REQUIRED!** If the To Age value of an age class is smaller than or equal to the From Age value, the following error message will be displayed: **ERROR - THE 'TO' AGE MUST BE GREATER THAN THE 'FROM' AGE!** If consecutive age classes are not in ascending order or if consecutive age classes overlap, the following error message will be displayed: **ERROR - THE 'FROM' AGE MUST BE GREATER THAN THE PREVIOUS 'TO' AGE!** An undefined age class located between defined age classes is automatically removed from the age class list.

#### Group Classification (code fields)

- **Field Name:** this is a code-type field selected by the user. A field name can either be typed or selected from the FIELD LIST selection list which can be activated by pressing the F2 function key. A field name is transferred from the FIELD LIST by moving the highlight bar with the up- and down-arrow keys and pressing the Return key with the cursor on the selected field. The order in which field names are entered defines the hierarchical order in which the fields are combined into categories in the report. If no field name

is given, the following error message will be displayed: ERROR - A GROUP FIELD NAME IS REQUIRED! Codes can be selected for a currently active field from the CODE LIST which is activated by pressing the F3 function key. Individual codes are activated by moving the highlight bar with the up- and down-arrow keys. The following key combinations are used to make a code selection:

|          |   |                                 |
|----------|---|---------------------------------|
| ALT-T    | - | select all codes                |
| ALT-U    | - | unselect all codes              |
| Spacebar | - | select/unselect the active code |
| CTRL-T   | - | select the active code          |
| CTRL-U   | - | unselect the active code        |

Selected codes are marked with a triangular marker. Pressing the Return key when the CODE LIST is active saves the selection of code values. In addition to the selected code values an additional code with the value <Oth/Miss> is created for each selected field. This additional code comprises the non-selected and missing code values. The rows of a population dynamics report correspond to all the possible code combinations of the selected fields.

- **Label:** this is the user-defined field label that corresponds to a selected Field Name. The label is automatically displayed when a valid Field Name is defined.
- **Codes Selected:** displays the number of selected codes out of the number of defined codes for the corresponding selected field. The display of the number of defined codes is updated when a Field Name has been selected, the display of the number of selected codes is updated when the codes have been selected from the CODE LIST.

#### 4.4 Action List: Age

Reports belonging to this category contain a listing of animals that reach a specified age within a specified future period. Such listings can be used to facilitate certain treatments of animals that are carried out at a specific age, such as weaning, castration, etc. The data used to compile these listings are also written onto a dBASE output file that can either be used as such or be converted into an ASCII file using the File Conversion option of the EXTRACT module.

The columns in this report correspond to selected fields from the General Animal Data file plus certain system-generated fields. The dBASE data file

which is created to make the report contains all fields of the General Animal Data file and up to six system-generated additional fields.

The following settings and selections define a report belonging to this category:

- **Report Title:** this is a user-defined title of up to 40 characters that identifies each individual report. If no title is given, the following error message will be displayed: ERROR - A REPORT TITLE IS REQUIRED!
- **File:** displays the file name of the dBASE data file (.DBF) and memory variable file (.MEM) where the report data and the report configuration are stored. The file name is automatically assigned by the REPORT module and cannot be edited by the user.
- **Start Date, End Date, Period:** these determine the user-defined period. The Period is calculated from the given Start and End Dates. If either the Start or the End Date is missing, it will be calculated by specifying a Period. If the report is subsequently modified, with all three values non-missing, the period length will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if either the Start Date or the End Date is deleted. Only animals that are alive and reach the specified age within the specified period are counted. Animals with missing birth date are excluded from the report. If no Start Date is given, the following error message will be displayed: ERROR - A START DATE IS REQUIRED! Likewise, failure to specify an End Date will result in the following error message: ERROR - AN END DATE IS REQUIRED! If the End Date is not later than the Start Date, the following error message will be displayed: ERROR - INVALID RANGE BETWEEN START AND END DATES!
- **Target Age:** this is the user-defined age in days that is used to select animals from the data set. Animals selected for this report have to reach the user-defined target age within the specified period (Start Date - End Date). If a negative value is entered, all animals that are alive at the beginning (Begin Date) of the specified period are included in the report.
- **Print Weaning Date?:** allows you to specify whether the field WEAN\_DT from the General Animal Data file should be included in the report.
- **Print Castration Date?:** allows you to specify whether the field CAST\_DT from the General Animal Data file should be included in the report.
- **Print First Estrus Date?:** allows you to include or exclude the field OEST1\_DT from the General Animal Data file in the report.

- **Print Target Date?:** allows you to specify whether the system-generated field T\_DATE\_ should be included in the report. The Target Date is defined as an animal's birth date (BIRTH\_DT) plus the user-defined Target Age. If no Target Age is specified, by entering a negative value into the field Target Age from the report definition display, the user-defined Start Date will be the Target Date.
- **Print Last Weight?:** allows you to include or exclude the system generated fields T\_WGT\_ and T\_WDY\_ in the report. The target weight (T\_WGT\_) is the last weight before or on the target date, and the field T\_WDY\_ contains the difference in days between the date of the target weight (T\_WGT\_) and the target date.
- **Print Environment?:** allows you to specify whether the system generated fields T\_EN1\_, T\_EN2\_ and T\_EN3\_ should be included in the report. These fields contain code values from the fields ENVIRON1, ENVIRON2 and ENVIRON3 from the Environment Data file. These values describe the environment of the animal on the target date. The fields T\_EN2\_ and T\_EN3\_ are generated only if the user-selectable fields ENVIRON2 and ENVIRON3 exist in the Environment Data file.

# 1 Introduction

The LIMS EXTRACT module is written in the CLIPPER version of the dBASE language and compiled with the CLIPPER 5.0 compiler. The data set files are compatible with the dBASE III+ standard except for the index files (.NTX), which are CLIPPER specific.

The EXTRACT module produces output files containing raw data and calculated performance traits extracted from a LIMS data set on an animal and a parturition basis. Three types of extracts can be made and for each type users can configure extract definitions to suit their specific requirements.

The quality and accuracy of the extracted data obviously depend on the quality and accuracy of the raw data. The EXTRACT module does not carry out any data validation, but assumes that data have previously been validated and corrected with the LIMS VALIDATION module.

The following table shows the LIMS files (xxxx = data set name) used to store the extract definitions and the extracted data:

| file name   | file content                              |
|-------------|---|
| xxxxX01.MEM | configuration, animal performance by age  |
| xxxxX02.MEM | configuration, animal performance by date |
| xxxxX03.MEM | configuration, parturition performance    |
| xxxxX01.DBF | dBASE data, animal performance by age     |
| xxxxX02.DBF | dBASE data, animal performance by date    |
| xxxxX03.DBF | dBASE data, parturition performance       |
| xxxxX01.TXT | ASCII data, animal performance by age     |
| xxxxX02.TXT | ASCII data, animal performance by date    |
| xxxxX03.TXT | ASCII data, parturition performance       |

You can create configurations for up to three extract definitions. These definitions can later be edited.



## 2 Starting the EXTRACT Module

The filename of the executing EXTRACT module is LEXTRACT.EXE. This file should be located in the subdirectory \LIMS on a hard disk. The PATH statement in the AUTOEXEC.BAT file should include the subdirectory \LIMS so that the EXTRACT module can be started independent of the current drive or path. For more information on the PATH statement in the AUTOEXEC.BAT file, please refer to the DOS User's Guide.

Each data set has a name which is given during the data set creation procedure in the SETUP module of LIMS. This data set name must consist of exactly 4 characters, which must be letters or digits. Each data set should be stored in a subdirectory with the same name as the data set.

Use the following syntax to start the LIMS EXTRACT module:

```
LEXTRACT [xxxx] [/y]
```

The two items in square brackets ( [ ] ) are optional parameters that can be specified when calling the EXTRACT module. If you want to specify optional parameters, do not include the square brackets, only the information within the brackets. The command and the options can be typed in either upper or lower case characters. If both parameters are specified, they must be entered in the above order.

xxxx This is the data set name and should be the same as the current subdirectory name. Only letters and digits are allowed and all four characters must be used. These four characters also make up the prefix of all data set files. If this parameter is not specified when calling the EXTRACT module, a prompt displayed on the initial screen will request the input of the data set name.

/y This is the video mode option. By default, the EXTRACT module will attempt to start in colour mode.

/H sets a special monochrome display which is optimised for Hercules graphics cards. If your computer is equipped with a Hercules graphics card this mode should be selected.

/M sets monochrome mode, even if a colour board is installed. This mode is to be used if your computer has a CGA, EGA or VGA graphics board and a monochrome monitor.

/L sets a special monochrome display which is optimised for liquid crystal and gas plasma displays. If your computer has a display of one of these kinds, you should select this mode.

If the data set name is given as a parameter when the EXTRACT module is started, the initial screen will display a number of messages, which will disappear after a fixed amount of time. If no name is given, a prompt on the bottom line of the initial screen will request the data set name. At this point the escape key [Esc] will return to the operating system (DOS).

### 3 Main Menu

The Main Menu displays the 3 main menu options in a window in the centre of the display area. The currently selected option is displayed in a different colour. The following table shows the keys for use with the Main Menu and their functions:

|          |   |   |
|----------|---|---|
| Spacebar | - | selects the following option                      |
| Dnarrow  | - | selects the following option                      |
| Uparrow  | - | selects the preceding option                      |
| <number> | - | selects and activates the option with that number |
| End      | - | selects the last option                           |
| Home     | - | selects the first option                          |
| Return   | - | activates the selected option                     |
| Esc      | - | returns to the <b>DOS</b> level                   |

The top line of the Main Menu screen contains information about the Update module and the current data set. In the lefthand corner the module name is displayed, the data set code and the data set title (given during the setup process) are shown in the middle, and the current date to the right. This line is displayed throughout execution of the EXTRACT module.

The bottom line displays status information, which in the case of the Main Menu display is the full version of the system name and the version number of the EXTRACT module. The different LIMS modules can have different version numbers, but the main version numbers (the integer part) of all LIMS modules must be the same for the different modules to work together properly.

The status line is used to display information, warning and error messages, as well as for user confirmation input of certain commands.

#### 3.1 Extract Definition

This option displays a list of extraction types. An extraction type can be selected by moving the highlight bar with the up- and down-arrow keys. Pressing the Return key brings up the extract configuration display for the selected extraction category.

The option **Extract Definition** can be used both to make an initial extract definition and to subsequently modify it. The different extraction types that

can be created by the LIMS EXTRACT module are discussed in separate chapters under **Extraction Types**.

The settings of the extract definitions are saved in the extract configuration files with the extension .MEM.

### 3.2 Extract Generation

This option displays a list of extraction types and the dates they were last run. Extraction types can be activated by moving the highlight bar with the up- and down-arrow keys. The following extraction selections can be made using the key combinations below:

|          |   |                                       |
|----------|---|---------------------------------------|
| ALT-T    | - | select all extractions                |
| ALT-U    | - | deselect all extractions              |
| Spacebar | - | select/deselect the active extraction |
| CTRL-T   | - | select the active extractions         |
| CTRL-U   | - | deselect the active extractions       |

Selected extractions are marked with a triangular marker. Pressing the Return key brings up the extract configuration display for the active extraction type. The F10 key saves the extraction type markers and initiates the extract generation process after confirmation.

During extract generation a status window displays the type of the current extract being generated and a counter indicating the number of extracts generated, as well as the total number of extracts to be generated. It also displays a percentage counter which shows how much has been completed of the extract currently being generated.

The finished extracts are saved in dBASE data file format and can be converted into ASCII files of different types using the option **File Conversion** from the Main Menu.

### 3.3 File Conversion

With this option you can make an ASCII copy of the extracted data stored (by default) in dBASE data files. This option allows you to convert raw data files and output data files into ASCII format for use with other software, such as statistical packages, spreadsheet programs, word processing programs, etc. Converted data files have the same name as the original dBASE file, but with the extension .TXT.

Extraction files, report files, raw data files and the ID Change History file (which otherwise is not directly accessible by users) can be selected from a displayed list. Individual data files can be activated by moving the highlight bar with the up- and down-arrow keys. The following key combinations are used to make a data file selection:

|          |   |                                      |
|----------|---|--------------------------------------|
| ALT-T    | - | select all data files                |
| ALT-U    | - | deselect all data files              |
| Spacebar | - | select/deselect the active data file |
| CTRL-T   | - | select the active data file          |
| CTRL-U   | - | deselect the active data file        |

Selected data files are marked with a triangular marker. Pressing the F10 key saves the data file markers and brings up the Conversion Type Menu.

### **3.4 Conversion Type Menu**

The Conversion Type Menu allows you to select one of the three currently available ASCII conversion formats for the marked data files from the data file selection list.

Each converted file consists of a header followed by the body of converted data. The header contains the data set name, the file label, the DOS file name of the dBASE file, the date of last update of the dBASE file, the file structure and the conversion format.

The file structure information consists of four columns which contain the dBASE field name, the field type, the field length and the number of decimal places. Field types can be character (C), date (D), numeric (N) and memo (M).

Date values are formatted as numeric values with 8 digits having the structure CCYYMMDD (C=century, Y=year, M=month, D=day).

#### **3.4.1 SDF Format**

This ASCII conversion format is called System Data Format (SDF). Data are copied character by character beginning from the left. Records have fixed length and are terminated by carriage return/line feed.

Converted data are listed in columns in the output file, but there are no delimiters or spaces between variables.

### **3.4.2 Delimited Format**

In the delimited conversion format data are copied character by character beginning from the left. Records are of variable length and end with carriage return/line feed.

Converted data are not placed in columns in the output file. A comma separates variable values and double quotation marks enclose character values.

### **3.4.3 Report Format**

In this format data are copied character by character beginning from the left. Records have fixed length and end with carriage return/line feed.

Converted data are listed in columns in the output file, with two blank spaces inserted between variables.

## 4 Extraction Types

### 4.1 Individual Animal Performance by Age

This extraction creates an output file with one record per animal. The extraction file contains all fields of the General Animal Data file, the environment code fields of the Environment Data file, and several derived fields containing calculated weights and survival codes. These fields and the assumptions and algorithms used to derive and calculate their values are described below.

The following settings and selections define an extraction of this type:

- **Start Date, End Date, Period:** these determine a user-defined period. The Period is calculated from the given Start and End Dates. If either one of the Start Date or End Date values is missing, it will be calculated by specifying a Period. If the extract is subsequently modified, with all three values non-missing, the length of the period will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if either the Start Date or the End Date is deleted. Only animals with a birth date within the user- defined period (Start Date before or coinciding with Birth Date and Birth Date before or coinciding with End Date) are included in the extract. If both Start Date and End Date are left blank, all animals will be included in the extract, irrespective of their birth dates. If only an End Date is specified, the following error message will be displayed: ERROR - A START DATE IS REQUIRED IF END DATE IS GIVEN! Likewise, if only a Start Date is specified, the following error message will appear: ERROR - AN END DATE IS REQUIRED IF START DATE IS GIVEN! If the End Date is not later than the Start date, the following error message will be displayed: ERROR - INVALID RANGE BETWEEN START AND END DATES!
- **Last Update:** this is the latest date for which the data set is complete with respect to disposal information. If the field is left blank, the data set is assumed to be complete up to the computer's current system date. This information is necessary for determining which animals are currently alive in the population. If disposal information is missing, animals are assumed to be alive in the population up to the Last Update date. After the Last Update date, their survival status in the population is unknown and will be assigned a missing value in the extraction output file.
- **Include Environment:** this determines whether or not environment information should be included in the extraction output file. The response codes are Y (yes) or N (no). If the response Y is given, the animal's corresponding environment codes will be added to the extraction output file

for each event (birth, weaning, disposal, etc) and age for which data are extracted.

- **File:** this is the name of the extraction output file. By default the extracted data are sent to a dBASE file with the extension .DBF. If this file is later converted to ASCII format using the File Conversion option of the EXTRACT module, the converted ASCII file will retain the name of the original dBASE file, but with the extension .TXT.
- **Weight at Event:** six events can be specified for which weight data will be extracted. The first five events refer to the animal itself, the sixth refers to the animal's dam. For each event users can specify a range (in days) within which a weight record must fall in order to be extracted. The first weight at or after each event is extracted. If no weight record is found during the period defined by the specified range of days (counted from the date of the event), the last weight record before the event, within the given range, will be extracted. If a negative range value is entered, the following error message will be displayed: ERROR - INVALID RANGE FOR WEIGHT AT EVENT! Weight data for weaning, castration and first estrus can only be extracted if the corresponding date fields exist in the General Animal Data file. For each event, an 'accuracy field' is created in the output file. This field contains a value which is the number of days that the weight record deviates from the event date. A negative value indicates that the extracted weight record is before the event date, a positive value that it is after the event date. A zero value, for a non-missing weight record, means that the weight record was taken exactly on the date of the event.
- **Weight at Age:** up to six ages can be specified for which weights will be interpolated. An offset should be specified for each age, determining the range of the interpolation in days before and after the corresponding age. The last weight record before or at the age and the first weight record after the age are used to calculate the weight. Both weight records must be within the period of Age  $\pm$  Offset. If a weight record is found that has been made at exactly the specified age, this weight record will be extracted. If there is no such record, a linear interpolation will be performed between the two weights before and after each age. For each specified age, the survival of the animal in the population is determined and extracted. If two equal, non-zero age values are entered, an error message will be displayed: ERROR - DUPLICATE AGE FOR WEIGHT AT AGE! If a negative age is entered, the following error message will be displayed: ERROR - INVALID AGE FOR WEIGHT AT AGE! Similarly, entering a negative offset value will cause the following error message to appear: ERROR - INVALID OFFSET FOR WEIGHT AT AGE!

The extracted data are sent to the output file xxxxX02.DBF. The unit of observation in this output file is one animal. One record per animal is

extracted. The animal identification is contained in the variable with the name ID. The output file contains two types of fields, the first type being fields defined in the raw data set, the second fields that are defined by the extraction procedure. For the fields in the output file that are also defined in the raw data set, values are carried over from the raw data set without change. For fields defined by the extraction procedure, values are normally calculated from raw data values. Below is a description of the system-generated fields for each individual animal in the output file xxxxX01.DBF (xxxx = data set code):

**BIRTH\_WT\_:** this field contains the first non-missing weight found in the Weight Data file (xxxxWGT.DBF) after the birth date (BIRTH\_DT). The weighing date must be within the range specified in the extract configuration. The size of the field depends on the size of the field WEIGHT in the Weight Data file (xxxxWGT.DBF). If no weight record is found a maximum negative value will be assigned to the field to indicate a missing value.

**BIRTH\_WD\_:** the value in this field is the number of days between the weighing date of the extracted BIRTH\_WT\_ and the birth date. This measure is useful if not all animals have been weighed on the day of birth, requiring that the first weight after birth be taken as an approximation for the true birth weight. This variable should have a value between zero and the birth range value specified in the extract configuration.

**BIRTH\_E1\_:** this field contains the same code as the variable ENVIRON1 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. This field has the same format as ENVIRON1.

**BIRTH\_E2\_:** this field contains the same code as the variable ENVIRON2 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. The format of this field is the same as that of ENVIRON2. This field is only generated if the user-selectable variable ENVIRON2 exists in the Environment Data file.

**BIRTH\_E3\_:** this field contains the same code as the variable ENVIRON3 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. The format is the same as that of ENVIRON3. This field is only generated if the user-selectable variable ENVIRON3 exists in the Environment Data file.

**DAMPP\_AG\_:** this is the age of the dam (DAM\_ID) in days at birth (BIRTH\_DT) of the animal (ID).



**DAMPP\_WT\_**: this field contains the weight of the dam (DAM\_ID) at the birth (BIRTH\_DT) of the animal (ID). The first weight at or after the birth date of the animal that is within the user-specified range is extracted. If no such weight is found, the last weight before the birth of the animal that is within the user-specified range will be extracted.

**DAMPP\_WD\_**: the value in this field is the difference in days between the weighing date of the extracted dam weight and the birth date of the animal.

**CAST\_AG\_**, **CAST\_WT\_**, **CAST\_WD\_**, **CAST\_E1\_**, **CAST\_E2\_**, **CAST\_E3\_** : these fields contain castration data for the animal (ID). They are only created if the user-selectable field CAST\_DT exists in the General Animal Data file. Data will only be extracted if the field CAST\_DT does not contain a missing value. The value in the field CAST\_AG\_ is the age in days at castration of the animal. The fields CAST\_WT\_ and CAST\_WD\_ contain the weight of the animal at castration and the difference in days between the weighing date and the castration date. Castration weight is derived in the same manner as described for DAMPP\_WT\_ (see above). The fields CAST\_E1\_, CAST\_E2\_ and CAST\_E3\_ contain the environment codes of the environment in which the animal is castrated. The description given above for the corresponding fields BIRTH\_E1\_, BIRTH\_E2\_ and BIRTH\_E3\_ also applies to these fields, except that the environment records refer to castration rather than to birth.

**WEAN\_AG\_**, **WEAN\_WT\_**, **WEAN\_WD\_**, **WEAN\_E1\_**, **WEAN\_E2\_**, **WEAN\_E3\_**: these fields contain information about weaning of the animal (ID). They are only created if the user-selectable field WEAN\_DT exists in the General Animal Data file. Data will only be extracted if the field WEAN\_DT does not contain a missing value. These variables are analogous to the variables referring to castration. Thus the description given above for castration fields above applies to weaning fields also, the recorded event being weaning instead of castration.

**OEST1\_AG\_**, **OEST1\_WT\_**, **OEST1\_WD\_**, **OEST1\_E1\_**, **OEST1\_E2\_**, **OEST1\_E3\_** : these fields contain data about the first estrus of the animal (ID). They are only created if the user-selectable field OEST1\_DT exists in the General Animal Data file. Data will only be extracted if the field OEST1\_DT does not contain a missing value. The description above of the analogous variables referring to castration also applies to estrus fields.

**DISP\_AG\_**, **DISP\_WT\_**, **DISP\_WD\_**, **DISP\_E1\_**, **DISP\_E2\_**, **DISP\_E3\_** : these fields contain information about disposal of the animal (ID). Data will only be

extracted if the field DISP\_DT in the General Animal Data file does not contain a missing value. The description above of the analogous variables referring to castration also applies to disposal variables.

**SRV\_nnnn\_**, **WGT\_nnnn\_**, **EN1\_nnnn\_**, **EN2\_nnnn\_**, **EN3\_nnnn\_**: these fields contain information about the animal (ID) at a specified age nnnn. The field SRV\_nnnn\_ specifies whether the animal was alive in the population at the age nnnn. A value of 1 means the animal was alive in the population at age nnnn, a value of 0 means the animal was disposed from the population before that age, and a value of -9 means that no conclusive information about the animal's survival in the population could be derived. This will be the case if the age nnnn of an animal is reached after the Last Update date as defined in the extract configuration. The field WGT\_nnnn\_ contains an interpolated weight of the animal at the age nnnn. Two weight records are needed to make an interpolation, one before or at the age nnnn and one after that age. Both weight records must be within the offset value specified in the extract configuration. The variables EN1\_nnnn\_, EN2\_nnnn\_ and EN3\_nnnn\_ contain the codes for the animal's environment at the age nnnn. If the animal died before reaching the age nnnn, the field SRV\_nnnn\_ will contain a value of 0, WGT\_nnnn\_ will have a value of 0 and the environment fields will contain the codes of the animal's last environment.

## 4.2 Individual Animal Performance by Date

This extraction creates an output file with one record per animal. The extraction file contains all fields of the General Animal Data file, the environment code fields of the Environment Data file, and several derived fields containing calculated weights, ages and survival codes. These fields and the assumptions and algorithms used to derive and calculate their values are discussed below.

The following settings and selections define an extraction of this type:

- **Start Date, End Date, Period**: these determine a user-defined period. The Period is calculated from the given Start and End Dates. If either one of the Start Date or End Date values is missing, it will be calculated by specifying a Period. If the extract is subsequently modified, with all three values non-missing, the length of the period will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if either the Start Date or the End Date is deleted. Only animals with a birth date within the user-defined period (Start Date before or coinciding with Birth Date and Birth Date before or coinciding with End Date) are included in the extract. If both Start

Date and End Date are left blank, all animals will be included in the extract, irrespective of their birth dates. If only an End Date is specified, the following error message will be displayed: ERROR - A START DATE IS REQUIRED IF END DATE IS GIVEN! Likewise, if only a Start Date is specified, the following error message will appear: ERROR - AN END DATE IS REQUIRED IF START DATE IS GIVEN! If the End Date is not later than the Start date, the following error message will be displayed: ERROR - INVALID RANGE BETWEEN START AND END DATES!

- **Last Update:** this is the latest date for which the data set is complete with respect to disposal information. If the field is left blank, the data set is assumed to be complete up to the computer's current system date. This information is necessary for determining which animals are currently alive in the population. If disposal information is missing, animals are assumed to be alive in the population up to the Last Update date. After the Last Update date, their survival status in the population is unknown and will be assigned a missing value in the extraction output file.
- **Include Environment:** this determines whether or not environment information should be included in the extraction output file. The response codes are Y (yes) or N (no). If the response Y is given, the animal's corresponding environment codes will be added to the extraction output file for each event (birth, weaning, disposal, etc) and age for which data are extracted.
- **File:** this is the name of the extraction output file. By default the extracted data are sent to a dBASE file with the extension .DBF. If this file is later converted to ASCII format using the File Conversion option of the EXTRACT module, the converted ASCII file will retain the name of the original dBASE file, but with the extension .TXT.
- **Weight at Event:** six events can be specified for which weight data will be extracted. The first five events refer to the animal itself, the sixth refers to the animal's dam. For each event users can specify a range (in days) within which a weight record must fall in order to be extracted. The first weight at or after each event is extracted. If no weight record is found during the period defined by the specified range of days (counted from the date of the event), the last weight record before the event, within the given range, will be extracted. If a negative range value is entered, the following error message will be displayed: ERROR - INVALID RANGE FOR WEIGHT AT EVENT! Weight data for weaning, castration and first estrus can only be extracted if the corresponding date fields exist in the General Animal Data file. For each event, an 'accuracy field' is created in the output file. This field contains a value which is the number of days that the weight record deviates from the event date. A negative value indicates that the extracted weight record is before the event date, a positive value that it is

after the event date. A zero value, for a non-missing weight record, means that the weight record was taken exactly on the date of the event.

- **Weight at Date:** up to six dates can be specified for which weights will be interpolated. An offset should be specified for each date, determining the range of the interpolation in days before and after the corresponding date. The last weight record before or on the specified date and the first weight record after the date are used to calculate the weight. Both weight records must be within the period of  $\text{Date} \pm \text{Offset}$ . If a weight record is found that has been made on exactly the specified date, this weight record will be extracted. If there is no such record, a linear interpolation will be performed between the two weights before and after each date. For each specified date, the survival of the animal in the population is determined and extracted. If two equal dates are entered, the following error message will be displayed: ERROR - DUPLICATE DATE FOR WEIGHT AT DATE! Entering a negative offset value will cause the following error message to appear: ERROR - INVALID OFFSET FOR WEIGHT AT DATE!

The extracted data are sent to the output file xxxxX02.DBF. The unit of observation in this output file is one animal. One record per animal is extracted. The animal identification is contained in the variable with the name ID. The output file contains two types of fields, the first type being fields defined in the raw data set, the second fields that are defined by the extraction procedure. For the fields in the output file that are also defined in the raw data set, values are carried over from the raw data set without change. For fields defined by the extraction procedure, values are normally calculated from raw data values. Below is a description of the system-generated fields for each individual animal in the output file xxxxX02.DBF (xxxx = data set code):

**BIRTH\_WT\_:** this field contains the first non-missing weight found in the Weight Data file (xxxxWGT.DBF) after the birth date (BIRTH\_DT). The weighing date must be within the range specified in the extract configuration. The size of the field depends on the size of the field WEIGHT in the Weight Data file (xxxxWGT.DBF). If no weight record is found a maximum negative value will be assigned to the field to indicate a missing value.

**BIRTH\_WD\_:** the value in this field is the number of days between the weighing date of the extracted BIRTH\_WT\_ and the birth date. This measure is useful if not all animals have been weighed on the day of birth, requiring that the first weight after birth be taken as an approximation for the true birth weight. This variable should have a value between zero and the birth range value specified in the extract configuration.

**BIRTH\_E1\_**: this field contains the same code as the variable ENVIRON1 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. This field has the same format as ENVIRON1.

**BIRTH\_E2\_**: this field contains the same code as the variable ENVIRON2 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. The format of this field is the same as that of ENVIRON2. This field is only generated if the user-selectable variable ENVIRON2 exists in the Environment Data file.

– **BIRTH\_E3\_**: this field contains the same code as the variable ENVIRON3 in the Environment Data file. The code denotes the environment in which the animal was born. If there is no environment record at birth, this field is blank. The format is the same as that of ENVIRON3. This field is only generated if the user-selectable variable ENVIRON3 exists in the Environment Data file.

– **DAMPP\_AG\_**: this is the age of the dam (DAM\_ID) in days at birth (BIRTH\_DT) of the animal (ID).

**DAMPP\_WT\_**: this field contains the weight of the dam (DAM\_ID) at the birth (BIRTH\_DT) of the animal (ID). The first weight at or after the birth date of the animal that is within the user-specified range is extracted. If no such weight is found, the last weight before the birth of the animal that is within the user-specified range will be extracted.

**DAMPP\_WD\_**: the value in this field is the difference in days between the weighing date of the extracted dam weight and the birth date of the animal.

**CAST\_AG\_, CAST\_WT\_, CAST\_WD\_, CAST\_E1\_, CAST\_E2\_, CAST\_E3\_** : these fields contain castration data for the animal (ID). They are only created if the user-selectable field CAST\_DT exists in the General Animal Data file. Data will only be extracted if the field CAST\_DT does not contain a missing value. The value in the field CAST\_AG\_ is the age in days at castration of the animal. The fields CAST\_WT\_ and CAST\_WD\_ contain the weight of the animal at castration and the difference in days between the weighing date and the castration date. Castration weight is derived in the same manner as described for DAMPP\_WT\_ (see above). The fields CAST\_E1\_, CAST\_E2\_ and CAST\_E3\_ contain the environment codes of the environment in which the animal is castrated. The description given above for the corresponding fields BIRTH\_E1\_, BIRTH\_E2\_ and BIRTH\_E3\_ also applies to these fields, except that the environment records refer to castration rather than to birth.

**WEAN\_AG\_, WEAN\_WT\_, WEAN\_WD\_, WEAN\_E1\_, WEAN\_E2\_,**

**WEAN\_E3\_:** these fields contain information about weaning of the animal (ID). They are only created if the user-selectable field WEAN\_DT exists in the General Animal Data file. Data will only be extracted if the field WEAN\_DT does not contain a missing value. These variables are analogous to the variables referring to castration. Thus the description given above for castration fields above applies to weaning fields also, the recorded event being weaning instead of castration.

**OEST1\_AG\_, OEST1\_WT\_, OEST1\_WD\_, OEST1\_E1\_, OEST1\_E2\_,**

**OEST1\_E3\_:** these fields contain data about the first estrus of the animal (ID). They are only created if the user-selectable field OEST1\_DT exists in the General Animal Data file. Data will only be extracted if the field OEST1\_DT does not contain a missing value. The description of the analogous variables referring to castration also applies to estrus fields.

**DISP\_AG\_, DISP\_WT\_, DISP\_WD\_, DISP\_E1\_, DISP\_E2\_,**

**DISP\_E3\_:** these fields contain information about disposal of the animal (ID). Data will only be extracted if the field DISP\_DT in the General Animal Data file does not contain a missing value. The description above of the analogous variables referring to castration also applies to disposal variables.

**S\_yymmdd\_, A\_yymmdd\_, W\_yymmdd\_, E1yymmdd\_, E2yymmdd\_,**

**E3yymmdd\_:** these fields contain information about the animal (ID) on the user-specified date yymmdd (y=year, m=month, d=day). The value contained in the field S\_yymmdd\_ denotes whether the animal was alive in the population on the date yymmdd. A value of 1 means the animal was alive in the population on the date yymmdd, a value of 0 means the animal was disposed from the population before that date, and a value of -9 means that no conclusive information about the animal's survival in the population could be derived. This will be the case if the date yymmdd is after the Last Update date as defined in the extract configuration. The field A\_yymmdd\_ contains the age in days of the animal on the date yymmdd. The field W\_yymmdd\_ contains an interpolated weight of the animal on the date yymmdd. Two weight records are needed for interpolation, one before or on the date yymmdd and one after the date yymmdd. Both weight records must be within the offset value specified in the extract configuration. The variables E1yymmdd\_, E2yymmdd\_ and E3yymmdd\_ contain the codes for the animal's environment on the date yymmdd. If the animal died before the date yymmdd, the field S\_yymmdd\_ will contain a value of 0, A\_yymmdd\_ will contain the hypothetical age of the animal on the date yymmdd, W\_yymmdd\_ will have a value of 0 and the environment

fields will contain the codes of the animal's last environment. If the date yymmdd is before the birth date of an animal, the fields S\_yymmdd\_, A\_yymmdd\_ and W\_yymmdd\_ will contain the largest negative values (i.e. will be set to missing) that fit into these fields and the environment fields will be blank.

### 4.3 Parturition and Lactation Performance

This extraction creates an output file with one record per parturition. The extraction file contains all fields of the Parturition Data file, most fields of the dam's General Animal Data record, the environment code fields of the Environment Data file, and several derived fields containing calculated weights, milk yields, ages and survival codes. These fields and the assumptions and algorithms used to derive and calculate their values are described below.

During the extraction, the dam's General Animal Data records are merged with the corresponding parturition records. Certain fields in the General Animal Data records are treated differently from other fields:

|         |                      |
|---------|----------------------|
| CAST_DT | is not extracted     |
| CAST_DV | is not extracted     |
| DAM_ID  | is renamed DDAM_ID_  |
| SIRE_ID | is renamed DSIRE_ID_ |
| PARITY  | is renamed DPARITY_  |
| ID      | is renamed DAM_ID    |

After renaming, any field in the General Animal Data records that has the same name as a field in the Parturition Data record will be dropped from the General Animal Data record and will not be included in the extract.

The extraction file contains the identification numbers of three generations of animals. Below is a list of the fields containing animal identification numbers and a description of their relation to one another.

**PRG\_ID01, PRG\_ID02, ... PRG\_IDnn:** these fields contain the identification numbers of the offspring generation, which is the most recent generation in the extracted data.

**SIRE\_ID, DAM\_ID:** these are the identification numbers of the parents of the offspring generation (PRG\_IDnn).

**DSIRE\_ID\_, DDAM\_ID\_:** these are the identification numbers of the parents of the dam (DAM\_ID), which are also the maternal grandparents of the offspring generation (PRG\_IDnn).

**Calculating the lactation yield from daily milk records:**

- The lactation beginning date is the day after the parturition date.
- The daily milk yield of the first milk record is extrapolated as the daily yield between the lactation beginning date and the first milk record.
- Daily milk yields between subsequent milk records are calculated by linear interpolation.
- The daily milk yield of the last milk record is extrapolated as the daily yield between the last milk record and the lactation end date.
- If the lactation end date has not been specified explicitly (by LEND\_DT), it will be calculated as the date of the last milk yield plus the user-defined Lactation End Offset. If this calculated lactation end date is after the disposal date or the next parturition date, extrapolation will only be made to the nearest of these dates.

The unit of observation in the extracted file is one parturition and all fields that are carried over from the raw data refer to the dam (DAM\_ID).

The following settings and selections define an extraction of this type:

- **Start Date, End Date, Period:** these determine a user-defined period. The Period is calculated from the given Start and End Dates. If either one of the Start Date or End Date values is missing, it will be calculated by specifying a Period. If the extract is subsequently modified, with all three values non-missing, the length of the period will be considered fixed and a change of either Start Date or End Date will result in an adjustment of the other date. The period length can then only be accessed if either the Start Date or the End Date is deleted. Only parturition records with a date within the user-defined period (Start Date before or coinciding with Parturition Date and Parturition Date before or coinciding with End Date) are included in the extract. If both Start Date and End Date are left blank, all parturition records will be included in the extract, irrespective of their parturition dates. If only an End Date is specified, the following error message will be displayed: **ERROR - A START DATE IS REQUIRED IF END DATE IS GIVEN!** Likewise, if only a Start Date is specified, the following error message will appear: **ERROR - AN END DATE IS REQUIRED IF START DATE IS GIVEN!** If the End Date is not later than the Start date, the following error message will be displayed: **ERROR - INVALID RANGE BETWEEN START AND END DATES!**
- **Last Update:** this is the latest date for which the data set is complete with respect to disposal information. If the field is left blank, the data set is assumed to be complete up to the computer's current system date. This information is necessary for determining which animals are currently alive



in the population. If disposal information is missing, animals are assumed to be alive in the population up to the Last Update date. After the Last Update date, their survival status in the population is unknown and will be assigned a missing value in the extraction output file.

- **Include Environment:** this determines whether or not environment information should be included in the extraction output file. The response codes are Y (yes) or N (no). If the response Y is given, the animal's corresponding environment codes will be added to the extraction output file for the parturition date, the lactation end date, and each user-defined point in time after parturition.
- **File:** this is the name of the extraction output file. By default the extracted data are sent to a dBASE file with the extension .DBF. If this file is later converted to ASCII format using the File Conversion option of the EXTRACT module, the converted ASCII file will retain the name of the original dBASE file, but with the extension .TXT.
- **Lactation Begin Offset:** this allows you to deduct a specified initial period (in days) from the lactation period used to calculate the total lactation yield, the reference lactation yield and the total lactation length. If the value in this field is zero, lactation yields and lactation length will be computed from the first day after parturition. By using this setting the colostrum period can be excluded, by the subtraction of a fixed number of days at the beginning of the lactation, from all extracted lactation records.
- **Lactation End Offset:** this allows you to specify a period (in days) after the last milk yield for which the lactation length will be extrapolated. This offset is useful only if no explicit lactation end date has been specified (by LEND\_DT). If an implicit lactation end date can be derived, e.g. by disposal date (DISP\_DT) or by subsequent parturition date (PART\_DT), extrapolation will be made to the nearest date, whether the implicit lactation end date, or the last milking date plus the Lactation End Offset. Lactations that lack both an explicit and an implicit lactation end date are assumed to be incomplete and an extrapolation after the last milk record will be done for the number of days specified by the Lactation End Offset. This setting is useful if no explicit lactation end date (LEND\_DT) has been recorded and the milk recording intervals are regular. In this case the Lactation End Offset value may be set to half the length of the recording interval, thereby increasing the average lactation length by the same amount of time.
- **Reference Lactation Length:** this setting specifies a standard breakpoint in the lactation up to which date the lactation yield will be extrapolated or interpolated from the available milk records. By the use of this setting it is possible to calculate a reference milk yield for each lactation, defined as the

milk yield for a standard length of time after the parturition date.

- **Litter Birth Weight Range:** the birth weights of every offspring in a litter are summed to arrive at a litter birth weight. Using this setting you can specify a range, in days after parturition, within which an offspring weight record must fall in order to be considered a birth weight. The first weight on or after the parturition date is extracted. If no offspring weight record is found within the specified range of days after the parturition date, the litter birth weight will be assigned a missing value. An ‘accuracy field’ is created in the output file, containing the rounded average number of days that the offspring weight records deviate from the parturition date. For a non-missing extracted litter birth weight, a zero value in the accuracy field indicates that the weight records for all offspring have been made on exactly the parturition date.
- **Dam Post-Partum Weight Range:** the post-partum weight of the dam is extracted from the weight records for each parturition. Using this setting, you can specify a range (in days) within which a weight record must fall in order to be extracted. The first weight record at or after the parturition date is extracted as the dam’s post-partum weight. If no weight record is found within the specified range of days after the parturition date, the last weight record before the parturition date that is within range will be extracted. An ‘accuracy field’ is created in the output file, containing the rounded average number of days that the extracted record deviates from the parturition date. A negative value indicates that the extracted weight is before the parturition date, a positive value that it is after the parturition date. For a non-missing extracted weight, a zero value indicates that the extracted weight record was made on exactly the parturition date.
- **Litter Weight Offset Range:** this is a value (in days) which determines the maximum interpolation period for offspring weights at specified points in time after the parturition date. The interpolation is made using the last weight record before or on the specified date and the first weight record after the specified date. Both weight records must be within the period of user-defined date  $\pm$  Offset. If a weight record has been made on exactly the specified date, this weight record will be used, otherwise a linear interpolation will be made between the two closest weight records before and after the given date. The interpolated weights of all offspring of one parturition are summed to yield a litter weight. If a date is specified that is after the disposal of an offspring, the offspring’s weight will be considered zero. If a weight cannot be interpolated for an offspring because no weight records within the maximum interpolation period exist, the entire litter weight will be assigned a missing value.
- **Dam Weight Offset Range:** this is a value (in days) which determines the maximum interpolation period for dam weights at specified points in time

after the parturition date. The interpolation is made using the last weight record before or on the specified date and the first weight record after the specified date. Both weight records must be within the period of user-defined date  $\pm$  Offset. If a weight record has been made on exactly the specified date, this weight record will be used, otherwise a linear interpolation will be made between the two closest weight records before and after the given date. If a date is specified that is after the disposal of the dam, the dam's weight will be considered zero. If a weight cannot be interpolated for the dam because no weight records within the maximum interpolation period exist, the dam's extracted weight will be assigned a missing value.

- **Parturition/Lactation Performance:** this allows users to specify up to six points in time (in days) after the parturition date for which litter size, litter weight, dam weight, a daily milk yield and the dam's current environment are extracted.

The extracted data are sent to the output file xxxxX03.DBF. The unit of observation in this output file is one parturition. One record per parturition is extracted. The dam identification is contained in the field DAM\_ID. The output file contains two types of fields, the first type being fields defined in the raw data set, the second fields that are defined by the extraction procedure. For the first type of fields values are carried over from the raw data set without change. For fields defined by the extraction procedure, values are normally calculated from raw data values. Below is a description of the system-generated fields for each individual parturition in the output file xxxxX03.DBF (xxxx = data set code):

**PART\_E1\_, PART\_E2\_, PART\_E3\_:** these fields contain the codes given by the variables ENVIRON1, ENVIRON2 and ENVIRON3 in the Environment Data file. These codes denote the environment of the dam (DAM\_ID) at parturition. The fields PART\_E2\_ and PART\_E3\_ are generated only if the user-selectable variables ENVIRON2 and ENVIRON3 exist in the Environment Data file.

**PPAR\_INT\_:** this is the interval in days between the current and the previous parturition. If there is no previous parturition record, this variable has the value -999, indicating missing information.

**SPAR\_INT\_:** this is the interval in days between the current and the subsequent parturition. If there is no subsequent parturition record, this variable has the value -999, indicating missing information.

**DAMPP\_AG\_:** this is the age in days of the dam at parturition. If the dam's birth date (BIRTH\_DT) is missing, this variable has the value -9999, indicating missing information.

**DAMPP\_WT\_**: this field contains the weight of the dam at parturition. If no weight record can be found within the user-defined range, this variable will be assigned a maximum negative value, indicating missing information.

**DAMPP\_WD\_**: this is the difference in days between the parturition date and the weighing date for DAMPP\_WT\_.

**LBPTH\_WT\_**: this is the weight of the litter at birth. If no weight record can be found within the user-defined range, this variable will be assigned a maximum negative value, indicating missing information.

**LBPTH\_WD\_**: this is the difference in days between the parturition date and the weighing date for LBPTH\_WT\_.

**MYLD\_nnn\_**: this is the reference milk yield for the reference period nnn. The reference period starts after the Lactation Begin Offset and ends on day nnn of the lactation. If no milk records are available to calculate the reference yield, this variable will be assigned a maximum negative value, indicating missing information.

**MSMP\_nnn\_**: this field contains a numeric value indicating the number of daily milk records within the reference period. If no milk records are available, this variable will be assigned the value -999, indicating missing information.

**MYLD\_TOT\_**: this is the total milk yield for the entire lactation period. The period used to calculate the total milk yield begins after the Lactation Begin Offset and ends either on an explicit lactation end date (LEND\_DT) or on an implicit lactation end date. If no milk records are available to calculate the total yield, this variable will be assigned a maximum negative value, indicating missing information.

**MSMP\_TOT\_**: this field contains a numeric value indicating the number of daily milk records within the period used to calculate the total milk yield. If no milk records are available to calculate the total yield, this variable will be assigned the value -999, indicating missing information.

**LACT\_LEN\_**: this is the lactation length. The lactation length is calculated from the day after the Lactation Begin Offset either to an explicit lactation end date (LEND\_DT) or to an implicit lactation end date. If no milk records and no explicit lactation end date (LEND\_DT) are available, this variable will be assigned the value -999, indicating missing information.

**LEND\_EN1\_**, **LEND\_EN2\_**, **LEND\_EN3\_**: these fields contain the codes contained in the fields ENVIRON1, ENVIRON2 and ENVIRON3 in the

Environment Data file. These codes denote the environment of the dam (DAM\_ID) on the lactation end date. The fields LEND\_EN2\_ and LEND\_EN3\_ are generated only if the user-selectable variables ENVIRON2 and ENVIRON3 exist in the Environment Data file.

**LSZ\_nnn\_, LWT\_nnn\_, DWT\_nnn\_, MLK\_nnn\_,**

**EN1\_nnn\_ EN2\_nnn\_, EN3\_nnn\_:** these fields contain dam and offspring information for day nnn of the lactation or after parturition. LSZ\_nnn\_ is the number of offspring surviving in the population at day nnn after parturition. LWT\_nnn\_ contains the litter weight of the offspring at day nnn after parturition. If no weight records are available to derive the litter weight, this variable will be assigned a maximum negative value, indicating missing information. DWT\_nnn\_ is the dam weight at day nnn after parturition. If no weight records for derivation of the dam weight exist, this variable will be assigned a maximum negative value, indicating that information is missing. MLK\_nnn\_ is the milk yield of the dam at day nnn after parturition. If no milk records are available, this variable will be assigned a maximum negative value, indicating missing information. The fields EN1\_nnn\_, EN2\_nnn\_ and EN3\_nnn\_ contain the codes contained in the fields ENVIRON1, ENVIRON2 and ENVIRON3 from the Environment Data file. These codes denote the environment of the dam (DAM\_ID) on day nnn after parturition. The fields EN2\_nnn\_ and EN3\_nnn\_ are generated only if the user-selectable variables ENVIRON2 and ENVIRON3 exist in the Environment Data file.