



Fingerprinting II **Informatix™** Software

User Guide

Version 3.0

For Technical Service Call your local Bio-Rad Office or
in the U.S. Call **1-800-4BIO-RAD (1-800-424-6723)**

BIO-RAD

NOTES

SUPPORT BY BIO-RAD LABORATORIES

The Fingerprinting II Informatix™ software and this guide are released by Bio-Rad Laboratories. While the best efforts have been made in preparing this manuscript, no liability is assumed by the authors with respect to the use of the information provided.

Bio-Rad Laboratories will provide support to research laboratories in developing new and highly specialized applications, as well as to diagnostic laboratories where speed, efficiency and continuity are of primary importance. Our software thanks its current status for a part to the response of many customers worldwide. Please contact us if you have any problems or questions concerning the use of Fingerprinting II Informatix, or suggestions for improvement, refinement or extension of the software to your specific applications:

Bio-Rad Laboratories
2000 Alfred Nobel Drive
Hercules
CA 94547
U.S.A

PHONE: +1 510-741-1000
FAX: +1 510-741-5811
E-MAIL: lsg.techserv.us@bio-rad.com
URL: <http://www.bio-rad.com>

LIMITATIONS ON USE

The Fingerprinting II Informatix software and this accompanying guide are subject to the terms and conditions outlined in the License Agreement. The support, entitlement to upgrades and the right to use the software automatically terminate if the user fails to comply with any of the statements of the License Agreement.

No part of this guide may be reproduced by any means without prior written permission of the authors.

Copyright (C) 1998, 2002, Applied Maths BVBA. All rights reserved.

Fingerprinting II Informatix is a trademark of Bio-Rad Laboratories.
All other product names or trademarks are the property of their respective owners.
Fingerprinting II Informatix includes software developed by Applied Maths BVBA (<http://www.applied-maths.com>) and the Apache Software Foundation (<http://www.apache.org/>).

TABLE OF CONTENTS

1.	<u>THE CONCEPTS OF FINGERPRINTING II INFORMATIX</u>	9
1.1	THE PROGRAMS	9
1.2	THE DATABASE AND THE EXPERIMENTS	9
1.3	MULTI-DATABASE SETUP	11
2.	<u>ABOUT THIS GUIDE</u>	15
2.1	CONVENTIONS	15
2.2	FLOATING MENUS	16
3.	<u>INSTALLING THE SOFTWARE</u>	17
3.1	THE SETUP PROGRAM	17
3.2	EXAMPLE DATABASE	17
4.	<u>INSTALLING THE FINGERPRINTING II INFORMATIX NETWORK SOFTWARE</u>	19
4.1	INTRODUCTION	19
4.2	SETUP	19
4.3	ADVANCED FEATURES OF THE NETKEY SERVER PROGRAM	23
4.4	FEATURES OF THE CLIENT PROGRAM (FINGERPRINTING II INFORMATIX)	24
5.	<u>STARTING AND SETTING UP FINGERPRINTING II INFORMATIX</u>	25
5.1	THE FINGERPRINTING II INFORMATIX STARTUP PROGRAM	25
5.2	CREATING A DATABASE	26
5.3	SETTINGS OF A DATABASE	27
5.4	DATABASE PROTECTION TOOLS	28
5.5	LOG FILES	28
6.	<u>DATABASE FUNCTIONS</u>	31
6.1	THE FINGERPRINTING II INFORMATIX MAIN WINDOW	31
6.2	ADDING ENTRIES TO THE DATABASE	32
6.3	CREATING DATABASE INFORMATION FIELDS	33
6.4	ENTERING INFORMATION FIELDS	34
6.5	CONFIGURING THE DATABASE LAYOUT	35
7.	<u>SETTING UP EXPERIMENTS</u>	37
7.1	DEFINING A NEW FINGERPRINT TYPE	37
7.2	PROCESSING GELS	38
7.3	ASSIGNING BANDS AND QUANTIFICATION	53
7.4	ADDING THE GEL LANES TO THE DATABASE	64
7.5	SUPERIMPOSED NORMALIZATION BASED ON INTERNAL REFERENCE PATTERNS	65

7.6	IMPORT OF MOLECULAR SIZE TABLES AS FINGERPRINT TYPE	69
7.7	CONVERSION OF GEL PATTERNS FROM GELCOMPAR VERSIONS 4.1 AND 4.2 AND FROM MOLECULAR ANALYST FINGERPRINTING	73
7.8	DEFINING A NEW CHARACTER TYPE	73
7.9	INPUT OF CHARACTER DATA	77
7.10	IMPORT OF CHARACTER DATA BY QUANTIFICATION OF IMAGES SCANNED AS TIFF FILES	80
7.11	DEFINING A NEW SEQUENCE TYPE	92
7.12	INPUT OF SEQUENCES USING THE CONTIG ASSEMBLER PROGRAM GENEBuilder	95
7.13	DEFINING A NEW MATRIX TYPE	101
8.	EXPERIMENT DISPLAY AND EDIT FUNCTIONS	103
8.2	THE EXPERIMENT CARD	103
8.3	ENTERING EXPERIMENT DATA VIA THE EXPERIMENT CARD	105
8.4	ENTERING EXPERIMENT DATA VIA THE EXPERIMENT FILE	106
9.	COMPARISON FUNCTIONS	107
9.1	DEFINITION	107
9.2	MANUAL SELECTION FUNCTIONS	107
9.3	AUTOMATIC SEARCH AND SELECT FUNCTIONS	108
9.4	THE ADVANCED QUERY TOOL	109
9.5	SUBSETS	115
9.6	PAIRWISE COMPARISON BETWEEN TWO ENTRIES	117
9.7	THE COMPARISON WINDOW	118
10.	BAND MATCHING AND POLYMORPHISM ANALYSIS	123
10.1	COMPOSITE DATA SETS	125
10.2	CREATING A BAND MATCHING	126
10.3	CREATING A BAND MATCHING TABLE FOR POLYMORPHISM ANALYSIS	133
10.4	TOOLS TO DISPLAY SELECTIVE BAND CLASSES	136
10.5	FINDING DISCRIMINATIVE BANDS BETWEEN ENTRIES	137
11.	CLUSTER ANALYSIS	139
11.1	PRINCIPLES	139
11.2	COMPOSITE DATA SETS	140
11.3	CALCULATING A DENDROGRAM FOR A SINGLE EXPERIMENT	143
11.4	CALCULATION PRIORITY SETTINGS	146
11.5	GENERAL EDIT FUNCTIONS	147
11.6	ADDING AND DELETING ENTRIES	147
11.7	DENDROGRAM DISPLAY FUNCTIONS	148
11.8	CLUSTER SIGNIFICANCE TOOLS	150
11.9	MATRIX DISPLAY FUNCTIONS	152
11.10	GROUP STATISTICS	153
11.11	PRINTING A CLUSTER ANALYSIS	157
11.12	ANALYSIS OF THE CONCORDANCE BETWEEN TECHNIQUES	159
11.13	CALCULATING A DENDROGRAM FROM A COMPOSITE DATA SET	161
12.	CLUSTER ANALYSIS OF FINGERPRINTS	165

12.1	DEFINING 'ACTIVE ZONES' ON FINGERPRINTS	165
12.2	CALCULATION OF OPTIMAL POSITION TOLERANCE OPTIMIZATION AND SETTINGS	166
13.	CLUSTER ANALYSIS OF CHARACTERS	169
13.1	COEFFICIENTS FOR CHARACTER DATA AND CONVERSION TO BINARY	169
13.2	ADVANCED ANALYSIS OF MASSIVE CHARACTER SETS USING GENE MATHS	170
14.	CLUSTER ANALYSIS OF SEQUENCES	171
14.1	CALCULATING A CLUSTER ANALYSIS BASED ON PAIRWISE ALIGNMENT (STEPS 1 AND 2)	172
14.2	CALCULATING A MULTIPLE ALIGNMENT (STEPS 3 AND 4)	174
14.3	CALCULATING A CLUSTERING BASED ON THE MULTIPLE ALIGNMENT (STEPS 5 AND 6)	180
14.4	ADDING ENTRIES TO AND DELETING ENTRIES FROM AN EXISTING GLOBAL ALIGNMENT	181
14.5	EXCLUDING REGIONS FROM THE SEQUENCE COMPARISONS	182
14.6	WRITING COMMENTS IN THE ALIGNMENT	184
15.	PHYLOGENETIC CLUSTERING METHODS	187
15.1	MAXIMUM PARSIMONY CLUSTERING	187
15.2	MAXIMUM LIKELIHOOD CLUSTERING	189
16.	DIMENSIONING TECHNIQUES (PCA, MDS AND SOM)	191
16.1	CALCULATING AN MDS	191
16.2	EDITING AN MDS	192
16.3	CALCULATING A PCA	193
16.4	CALCULATING A DISCRIMINANT ANALYSIS	198
16.5	SELF ORGANIZING MAPS	199
16.6	MULTIVARIATE ANALYSIS OF VARIANCE (MANOVA) AND DISCRIMINANT ANALYSIS	201
17.	IDENTIFICATION WITH DATABASE ENTRIES	205
17.1	CREATING LISTS FOR IDENTIFICATION	205
17.2	IDENTIFYING UNKNOWN ENTRIES	205
18.	IDENTIFICATION USING LIBRARIES	207
18.1	CREATING A LIBRARY	207
18.2	IDENTIFYING ENTRIES AGAINST A LIBRARY	209
18.3	DETAILED IDENTIFICATION REPORTS	210
18.4	CREATING A NEURAL NETWORK	211
19.	DATABASE EXCHANGE TOOLS	215
19.1	CREATING A NEW BUNDLE	215
19.2	OPENING AN EXISTING BUNDLE	216
20.	THE FINGERPRINTING II INFORMATIX CLIENT FUNCTIONS	219

20.1	INTRODUCTION	219
20.2	CONNECTING TO THE SERVER DATATASE	221
20.3	SEARCHING AND DOWNLOADING ENTRIES ON THE SERVER DATABASE	222
20.4	UPLOADING BUNDLES AND GELS TO THE SERVER	224
20.5	PERFORMING IDENTIFICATIONS ON THE SERVER	225
21.	IMPORT OF DATA FROM EXTERNAL DATABASES	229
21.1	SETTING UP THE ODBC LINK	229
21.2	IMPORT OF DATABASE FIELDS USING ODBC	230
21.3	IMPORT OF CHARACTER DATA USING ODBC	231
21.4	IMPORT OF SEQUENCE DATA USING ODBC	231
22.	CONNECTED DATABASES	233
22.1	SETTING UP A NEW CONNECTED DATABASE	234
22.2	CONFIGURING THE CONNECTED DATABASE LINK IN FINGERPRINTING II INFORMATIX	235
22.3	WORKING IN A CONNECTED DATABASE	238
22.4	LINKING TO AN EXISTING DATABASE WITH STANDARD FINGERPRINTING II INFORMATIX TABLE STRUCTURE	239
22.5	LINKING TO AN EXISTING DATABASE WITH TABLE STRUCTURE NOT IN FINGERPRINTING II INFORMATIX FORMAT	239
22.6	CONVERTING A LOCAL DATABASE TO A CONNECTED DATABASE	241
22.7	OPENING AND CLOSING DATABASE CONNECTIONS	242
22.8	RESTRICTING QUERIES	244
23.	PRESERVING THE FINGERPRINTING II INFORMATIX DATABASE INTEGRITY	245
23.1	TAKING BACKUPS OF A DATABASE	245
23.2	DETECTING AND CORRECTING FAULTS IN A DATABASE	245
23.3	MISSING DIRECTORIES	246
23.4	CORRUPTED FILES	247
23.5	EMPTY FILES	247
23.6	DATABASE ENTRIES WITH IDENTICAL KEYS	247
23.7	DATABASE ENTRIES WITHOUT KEYS	247
23.8	EXPERIMENTS WITH IDENTICAL KEYS	247
23.9	EXPERIMENT KEYS WITHOUT DATABASE ENTRIES	248
24.	READING DATA FILES USING MACROS	249
24.1	IMPORTING EXPERIMENT DATA FROM TEXT FILES	249
24.2	RUNNING THE MACRO AUTOMATICALLY	250
24.3	RUNNING THE MACRO STEP BY STEP (DEBUGGING)	251
24.4	SELECTION OF DATA IMPORT FILES	251
24.5	CHANGING THE NAMES OF THE DATABASE FIELDS	251
24.6	THE MACRO LANGUAGE	252
25.	APPENDIX	265
25.1	CONNECTED DATABASE TABLE STRUCTURE	265
25.2	REGULAR EXPRESSIONS	270

1. The concepts of Fingerprinting II Informatix

1.1 The programs

The core of the Fingerprinting II Informatix software is composed of two executable units: a **Startup program** that creates and manages the *databases* and associated directories and that starts the **Analyze program** with a selected database. Most import functions and all analysis functions are done in the Analyze program. This includes processing of gel files starting from TIFF images, including lane finding and normalization. Independent conversion programs allow the import of character data in different file formats. These programs can also be loaded from the Fingerprinting II Informatix Startup program.

1.2 The database and the experiments

The logical flow of processing raw experiment files is represented in Fig. 1. The basis of Fingerprinting II Informatix is a powerful relational *database* (4) consisting of *entries*. The entries correspond to the individual organisms under study: animals, plants, fungi, bacterial strains. Each database entry is characterized by a unique *key*, assigned either automatically by the software or manually, and by a number of user-defined information fields. Each entry may be characterized by one or more *experiments* that can be linked easily to the entry (3). What we call experiments in Fingerprinting II Informatix are in fact the experimental data that are the numerical results of the biological experiments or assays performed to estimate the relationship between the organisms. The Fingerprinting II Informatix software is able to process and analyze three major types of experiments (1):

- *Fingerprint types*: Any densitometric record seen as a profile of peaks or bands can be considered as a fingerprint type. Examples are of course gel and capillary electrophoresis patterns, but also gas chromatography or HPLC profiles, spectrophotometric curves, etc.
- *Character types*: Any array of named characters, binary or continuous, with fixed or undefined length can be classified within the character types. The main difference between character types and electrophoresis types is that in the character types, each character has a well-determined name, whereas in the electrophoresis types, the bands, peaks or densitometric values are unnamed (a molecular size is NOT a well-determined name!).
- *Sequence types*: Within the sequence types, the user can enter nucleic acid (DNA and RNA) sequences and amino acid (protein) sequences.

A fourth type, *Matrix matrices*, is not a native experiment type, but the result of a comparison between database entries, expressed as similarity values between certain database entries. An example of a matrix type is a matrix of DNA homology values. DNA homology between organisms can only be expressed as pairwise similarity, not as native character data.

Essentially, adding a single organism (entry) with its associated experiments to the database constitutes three steps (steps 1 to 4 on the figure):

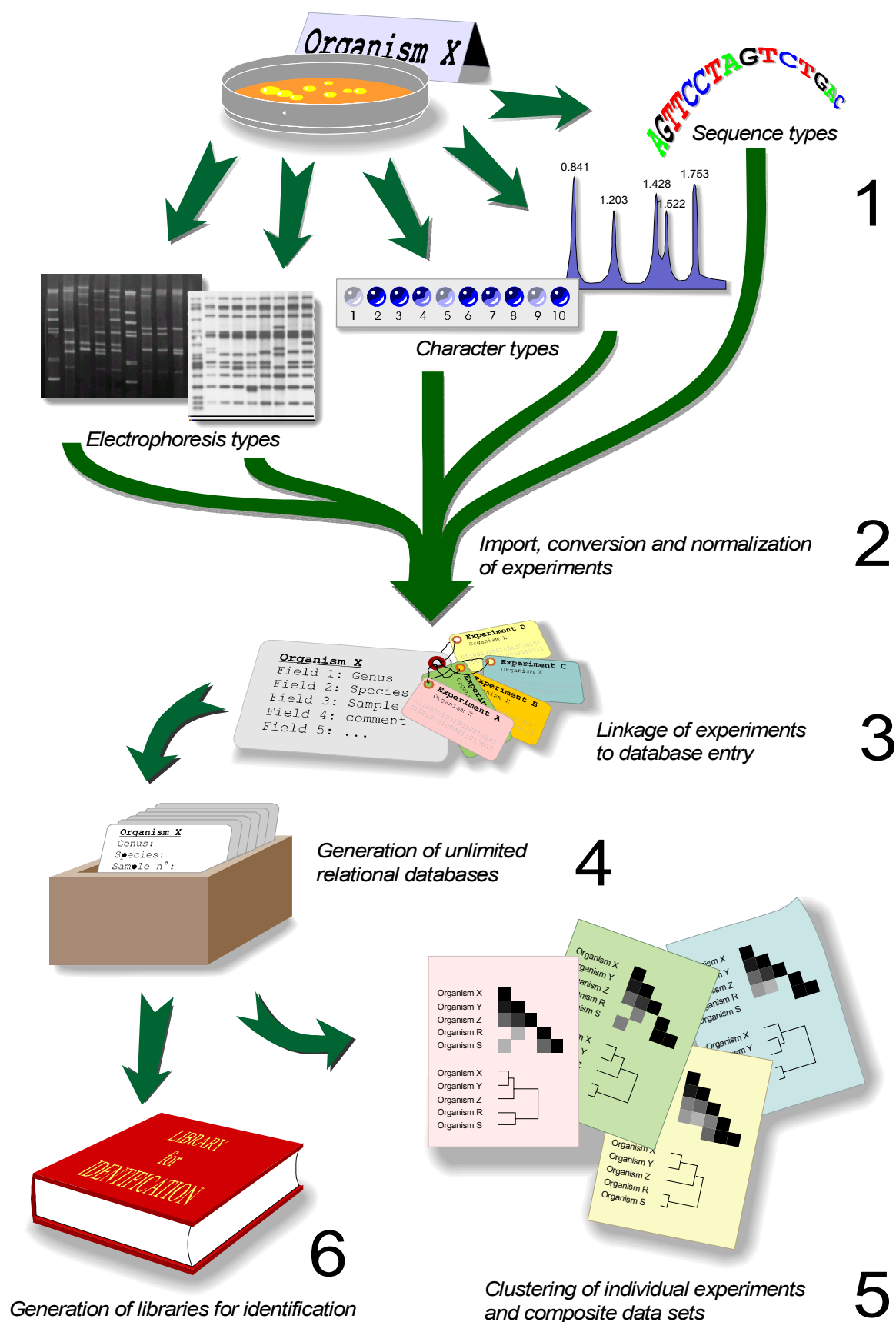


Figure 1. flow chart of conversion and import of TIFF and densitometric files.

1. **Generating experimental data.** This involves carrying out the experiments on the organism, and creating a data file of the experiment in any readable (text) format.
2. **Import, conversion and normalization of experiments.** Import of *character type* data is done by a macro language available in the Fingerprinting II Informatix software. This macro tool allows the user to define reading protocols for virtually any character file in readable text format. Most other import and conversion functions are done by the same macro tool. These include processing of gel images starting from densitometric curve files, sequences, etc. Special formats such as ABI four channel chromatogram files are imported using a separate program. Fingerprinting II Informatix offers the state-of-the-art technology for processing electrophoresis fingerprints, including lane-finding, normalization, band searching and quantification tools. *Normalization* means the correction of shifts of whatever kind or origin, within and between gels, so that bands or peaks of the same size have the same physical position after normalization, even if they occur on different gels.
3. See under 4.
4. **Creation of database entries and linkage of experiments.** There are three ways to create entries (note: an *entry* is the record for an individual organism) in the Fingerprinting II Informatix database: (1) an entry or a batch of entries can be generated directly in the database. Afterwards, experiments can be linked to these entries; (2) new database entries can be created automatically from a file of experiments; (3) Fingerprinting II Informatix can search automatically for entries with corresponding database information fields in the database and in an experiment file. This means that new database entries will be created for the experiments if the information fields do not correspond with one of the existing database entries. If on the other hand, the information fields of an experiment correspond to an existing entry, this experiment will be linked to the entry.
5. **Cluster analysis.** The grouping analysis functions involves the calculation of dendrograms to reveal groups of related organisms, principal component analysis, estimation of significance of groups, etc.
6. **Generation of libraries and identification.** The user can create highly specific libraries for identification. New and unknown organisms, available as database entries, can be identified against libraries of known entries, using individual experiments, or combinations of experiments, which results in a consensus-identification.

1.3 Multi-database setup

Fingerprinting II Informatix is a multi-database software, which supports the setup of different users in Windows NT. It is very important to understand the hierarchical structure of the user, database, and experiment setup in order to make optimal use of these features.

Windows NT (Windows 2000, Windows XP): The Fingerprinting II Informatix users are associated with the Windows NT login users. Each Windows NT user can specify his/her Fingerprinting II Informatix databases directory, and Fingerprinting II Informatix saves this information in the user's system registry. For example, suppose that a user X logs in on a Windows NT machine with Fingerprinting II Informatix installed. This user can create a directory, e.g. XXX, and specify this directory as the Home directory in the Startup program. Fingerprinting II Informatix will save this information in this user's system registry, so that each time the user logs in, Fingerprinting II Informatix will automatically consider the same directory as the home directory. In this way, each Windows NT user can define his/her own Fingerprinting II Informatix home directory, without interfering with other users. Within this home directory, the user can specify as many *Databases* as desired. Protection of the Fingerprinting II Informatix databases depends on the protection of the specified directory by the Windows NT user. If a user protects the directory containing the Fingerprinting II Informatix databases, other users will not be able to change or to read the databases, depending on whether the directory is write or read protected.

Figure 2 is an example of the database structure of one Windows NT user. The figure illustrates that one user can have many databases, and within each database, can define various experiment types. If it is the intention to compare database entries across different databases, the experiment types that the entries of different databases share should have the same name and definition.

Windows NT USER 'X' (Home directory XXX)

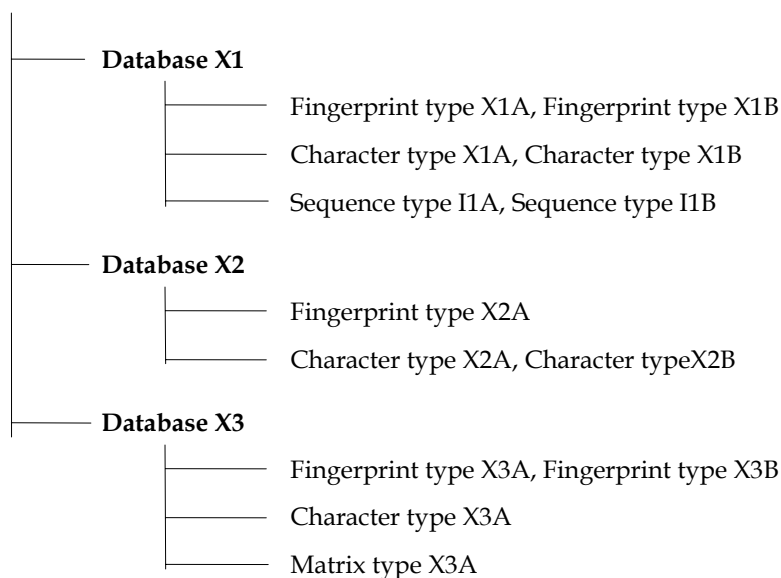


Figure 2. Structure of databases and experiment types within one user.

Within the specified home directory, the Fingerprinting II Informatix Startup program automatically creates the necessary subdirectories for each database. The following directory structure (Figure 3) corresponds to the above database structure of user X. The files shown are the configuration files for the experiment types defined under this user. If you want to create exactly the same experiment types under another database or another user, you can copy these .CNF files to the corresponding directories.

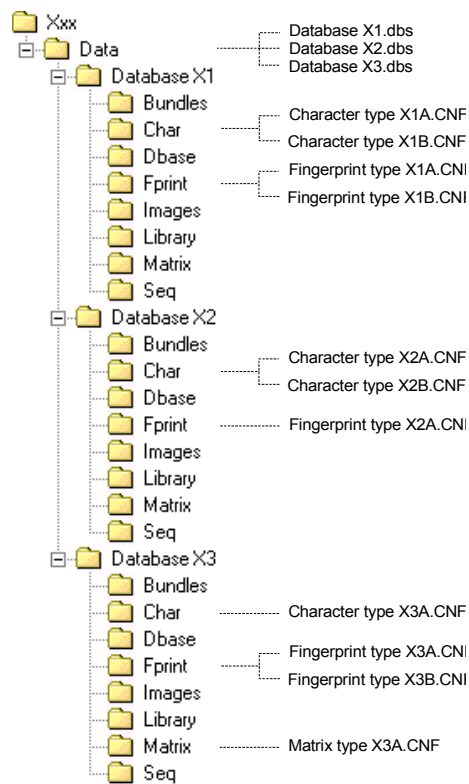


Figure 3. Structure of databases and experiment types within one user.

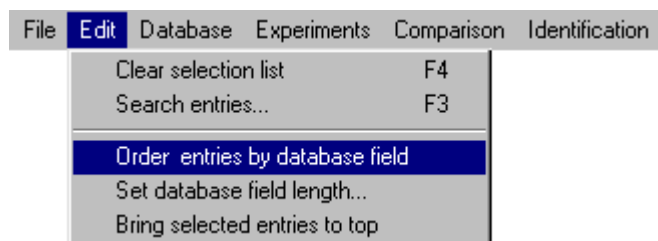
2. About this guide

2.1 Conventions

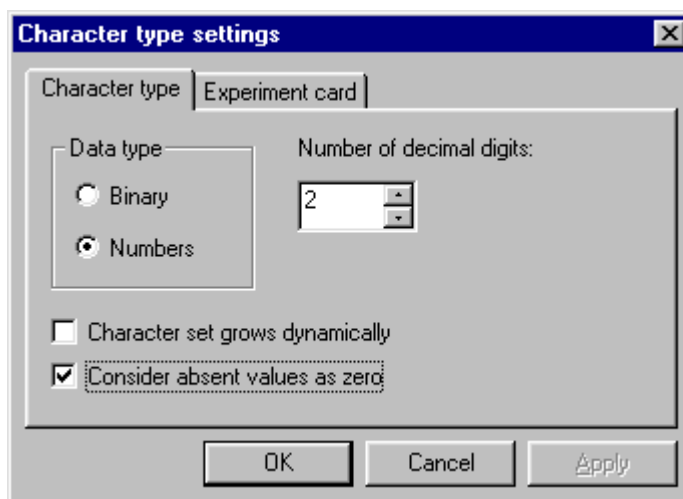
In the sections that follow, all menu commands and button text is typed in ***bold-italic***. Submenus are separated from parent menus by vertical slashes. Button text is always given between < and > characters.

Examples:

The following menu command will be indicated as *Edit | Order entries by database field*.



The following buttons are indicated as *Consider absent values as zero* (check box), <OK>, <Cancel>, <Apply> (disabled).



Each window and dialog box described in the guide will be given a name. This name is shown in *italic*, and usually corresponds to the name in the caption of the window or dialog box. For example, the dialog box above will be called the *Character type settings* dialog box.

2.2 Floating menus

In almost every window in the Fingerprinting II Informatix Analyze program, the use of place-specific “floating menus” is supported. For example, if you *right-click* (clicking the right mouse button) on a database entry, a floating menu is popped up, showing you all the possible menu commands that apply to the selected entry (see Figure 4). Clicking right in the Comparison panel shows a menu with all commands related to comparisons: *Create new comparison*, *Load comparison*, and *Delete comparison*.



Figure 4. Floating menu appearing after clicking right on a database entry.

The floating menus make the use of Fingerprinting II Informatix easier and more intuitive for beginners, and much faster for experienced users. In describing menu commands in this guide, we will not usually mention the corresponding floating menu command. It is up to the user to try right-clicking in all window panels in order to find out which is more convenient in every specific case: calling the command from the window’s menu or toolbar button or from the place-specific floating menu.

3. Installing the software

3.1 The Setup program

The Fingerprinting II Informatix software is delivered on CD-ROM. If you insert the CD-ROM in the drive, the Installation program will automatically load.

***NOTE:** The Installation program will automatically load if the Auto insert notification of the CD-ROM drive is enabled. You can change this setting in the **Control Panel, System, Device Manager, CD-ROM, Properties, Settings**, where you check the **Auto insert notification** option.*

On the installation screen, click **Install Fingerprinting II Informatix**. If this is the first installation of Fingerprinting II Informatix, you should allow the program to install the demo database. The software is installed in a subdirectory **Fingerprinting II Informatix** of the **Program files** directory. Upon completion, the installation program creates a shortcut to the Fingerprinting II Informatix Startup program on the desktop.

Insert the protection key (dongle) in the parallel port of the computer. Fingerprinting II Informatix is now ready for use.

Chapter 4 deals with the installation and features of the Fingerprinting II Informatix network software.

3.2 Example database

One database, **DemoBase**, is preinstalled with the software, and this database will serve as a tutorial and as an example in this guide. This database contains experimental data on some fictitious bacterial genera. The database contains the following experiment types:

Fingerprint types:

RFLP: Two different RFLP techniques, called RFLP1 and RFLP2, resulting in two patterns for each bacterial strain.

Character types:

FAME: Fatty Acid Methyl Esters (FAME) profiles obtained on a Hewlett Packard 5890A gas-liquid chromatography instrument. This is a typical example of an *open* data set: the number of fatty acids found depends on the group of entries analyzed. If more entries are added, more fatty acids will probably be found. Furthermore, FAME profiles are an example of a *continuous* character type: the percentage occurrence of a fatty acid in a bacterium can have any real value between zero and 100%.

PhenoTest: This is a fictitious phenotypic test assay that reveals the metabolic activity or enzyme activities of bacteria on 19 different compounds. The first cup of the test is a blank control. This is an example of a *closed* data set: the 20 characters are well-defined, and regardless of the number of entries examined, the number of characters in the experiment will always remain 20. Real examples of such types of assays are Biolog microplates, API test galleries, Vitek, etc. They can be interpreted in two ways. One can read the reactions by eye and score them as positive or negative; in this case the

character type is *binary*. If the microplates are read automatically using a microplate reader, the reactions in the cups may have any real value between an OD of zero and 2.5 to 3.0, which again is a *continuous* character type. In the example database, the reactions are scored as *continuous* characters.

In addition to the binary and continuous character types, one can also distinguish the *semi-quantitative* character types. These are tests that can have a number discrete values, e.g. 0, 1, 2, 3, 4, or 5. In practice, a number of continuous character types are interpreted as multistate characters for convenience. Examples are API galleries (BioMérieux SA, La Balme-les-Grottes, France) that can be read using a color scale ranging from 0 to 5.

Sequence types:

16S rDNA: For all of the strains, and a number of additional strains, the nearly complete 16S ribosomal RNA gene has been sequenced. The sequences are approximately 1500 bases long, but not all of them are sequenced completely.

Matrix types:

A partial homology matrix based upon hybridization of total genomic DNA has been generated for the genera.

4. Installing the Fingerprinting II Informatix Network Software

4.1 Introduction

The Fingerprinting II Informatix network software is compatible with any TCP/IP supporting network in combination with Windows 2000, Windows NT 4.0 and Windows XP. The communication is based on TCP/IP sockets provided by Windows. The *server* may be any Windows 2000, XP, or NT 4.0 computer in the network, and the *clients*, running Fingerprinting II Informatix, are all computers with the same *Domain Name*, including the server computer. The network software even allows licenses to be granted to physically distant locations via Dial-up connections, provided that the domain name for such distant clients is the same.

The system consists of three components: the *security driver program*, the *security key* (dongle) and the *client software*.

The **security key** is a hardware device (dongle). It attaches to the parallel port (or USB port) of a computer that is part of the network. This computer will be the *License server*.

The **security driver** is a program, **NETKEY.EXE**, that is available on the server computer. This program manages multiple licensing over the network. It is permanently running as a *Windows Service* on the server computer in the network, i.e. where the security key is attached.

The **client software** is a Fingerprinting II Informatix software version that contains the routines needed to register with the security server. This can be installed on *any* computer connected to the network, but only a restricted number of computers, the *license limit*, can run the software at the same time.

NOTE: In a TCP/IP network with Internet access, each computer has its own name in addition to its IP address. These computer names must be valid and registered names for all client computers, since the Fingerprinting II Informatix network software uses these names to recognize the client computers. If a Name Server is used, the names of the client computers must be validly registered in the Name Server of the network, otherwise, license granting will not be possible!

4.2 Setup

The License server computer has the *security key* inserted in the LPT1 port and runs the **Netkey server program**, which manages the network licenses. All computers connected to the network can have the Fingerprinting II Informatix *application software* installed, but only the number allowed by the *license limit* is able to run the software simultaneously. If the license limit is reached, a new license becomes free whenever the application is closed on one computer.

First, identify a suitable License server computer. The server should be a stable computer in terms of hardware and software configuration, that is permanently working and available over the network to other computers. A computer running Windows 2000/NT or XP Professional is to be preferred, only for reasons of stability of the operating system.

Once the License server is located, install Fingerprinting II Informatix on the License server and on one or more client computers that will run Fingerprinting II Informatix. When installing Fingerprinting II Informatix on the server, you should check the option *Install Netkey server program* in the startup wizard. Next, start perhaps with installing on just one client computer and take note of all steps needed to configure the network software. Do not forget to plug the network security key into the parallel port of the License server computer!

After installation on the server, the following programs are installed under *Program files\ Fingerprinting II Informatix* :

- Fingerprinting II Informatix
- Netkey

The **Netkey Server** program manages the network licenses. This program runs as a Service that is automatically is started on the License server, and should never be halted as long as licenses are in use.

Before the network software can be put into use successfully, there are some settings that will need to be made or changed. **If changes to the network settings of the computers are needed, we recommend to have these changes made by the system administrator or computer expert of your department or institution!**

TCP/IP

Each computer that will be used in the Fingerprinting II Informatix network configuration needs the *TCP/IP protocol* installed on the network. The TCP/IP protocol is provided with the installation package of Windows.

IP address and DNS host name

Furthermore, each computer in the Fingerprinting II Informatix network configuration needs a valid and unique *IP address*, to be specified in the TCP/IP properties. The IP address may be a permanent address assigned to the computer, or an IP address assigned by the DHCP server (Dynamic Host Configuration Protocol). It also must have a valid and unique *DNS host name*, which should not include spaces or periods. The DNS host name can be found by opening *Network* in the Control Panel, selecting *TCP/IP* and clicking *Properties*, under *IP address* and *DNS Configuration*. If permanent, the IP address can be found in the same window. They can also be seen in the Fingerprinting II Informatix Startup program by clicking the  button right from the *Homedir* button. In the *Netkey settings* box that appears, click *Info* to show the computer name, domain name and the IP address. Note down the DNS host names of the client computers and the server computer, and if permanent, also note down the IP addresses.

Initial settings

On each computer, including the server computer, Fingerprinting II Informatix has created a settings file *NETKEY.INI*, which needs to be completed for the network. Run the Fingerprinting II Informatix Startup program on the server, and click the small *Network settings* button  right from the *Homedir* button. Under *Server computer name*, fill in the DNS host name without the domain name. For example, if a computer is known as **computer.dept.univ.ext**, you should fill in **computer** without the domain name **dept.univ.ext**. You should not change the *Port number* unless there is a conflict with other software that uses the same port number.

You can also edit *NETKEY.INI* in Notepad by double clicking on the file name in the Windows Explorer. When opened in Notepad, the contents of the file look as follows:

SERVERNAME=

SERVERPORT=2350

After **SERVERNAME=** , enter the DNS host name of the server computer, and save the file. This change must be made on each client computer and on the server computer, in order to allow Fingerprinting II Informatix to find the server in the network.

Start the **Netkey** configuration program on the server computer. The following window appears (Figure 5).

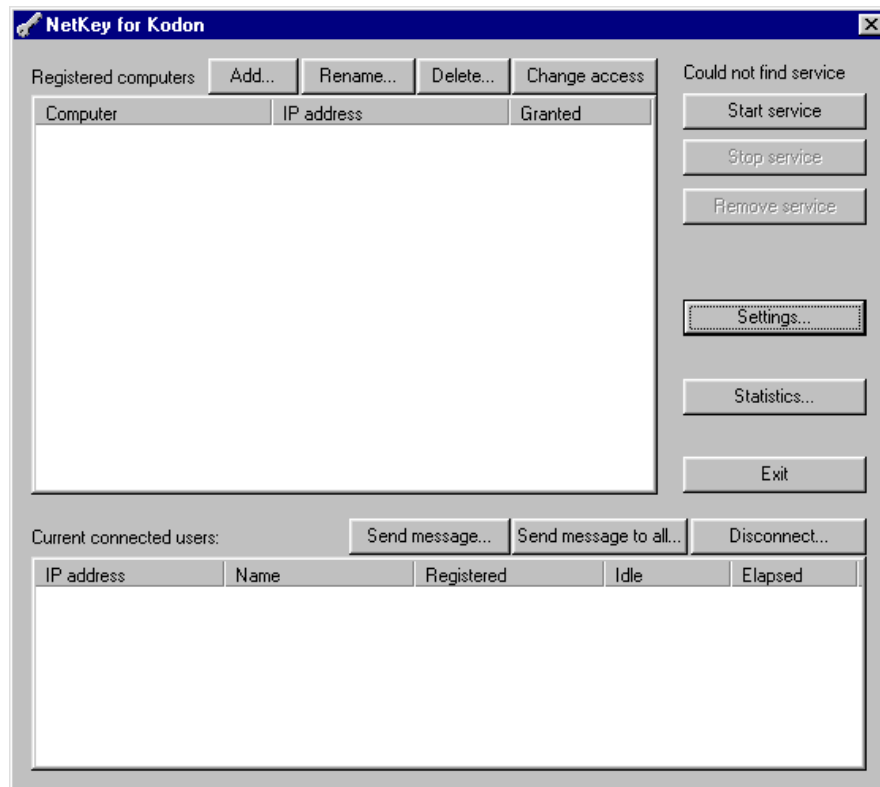


Figure 5. The netkey configuration program, initial view.

Initially, both panels 'Registered computers' and 'Current connected users' are empty. The upper panel lists the computers that are granted access to the Fingerprinting II Informatix network, and the lower panel lists the computers where the software is currently in use. **Every computer that can get access to the Fingerprinting II Informatix network must be specified in the server program, by means of its IP address and DNS host name.** In large institutions, this feature allows perfect control over which computers/users that are allowed to use the Fingerprinting II Informatix software.

Start the Netkey service by clicking the button **<Start service>**. Since the service is not installed yet, you will be asked to confirm to install it. When this is finished, a message "The NetKey service has been successfully installed" appears.

After clicking **<OK>** to this message, another message tells that the "Service has been started". The Netkey service is now ready to distribute licenses.

Configuring a client

On each client computer, configure the file NETKEY.INI in the same way as described above.

Defining a client

On the server computer, add the client computer to the list of Fingerprinting II Informatix clients as follows: Click **<Add>**. Enter the DNS host name of the client computer in the dialog box. In non-DHCP configurations (i.e. in case of permanent IP addresses), also enter the IP address. Press **<OK>**. The client is now shown in the upper panel, with its name only (DHCP) or with its name and IP address (permanent). From this point on, the client has access to the Fingerprinting II Informatix network software.

NOTE: If you do not wish to define specific computers to have permission to obtain a license for Fingerprinting II Informatix, you can enter an asterisk () in place of the computer name, without specifying an IP address. When doing so, every computer in the LAN will be able to obtain a Fingerprinting II Informatix license.*

Running Fingerprinting II Informatix

On the client computer, start Fingerprinting II Informatix and press the **Analyze** button. The program should load if the network is configured correctly and if the server name, the IP addresses, and domain host names are filled in correctly.

On the server computer, the client that uses Fingerprinting II Informatix is now listed in the lower panel, showing its IP address, DNS host name, total usage time (elapsed) and idle time (4.3).

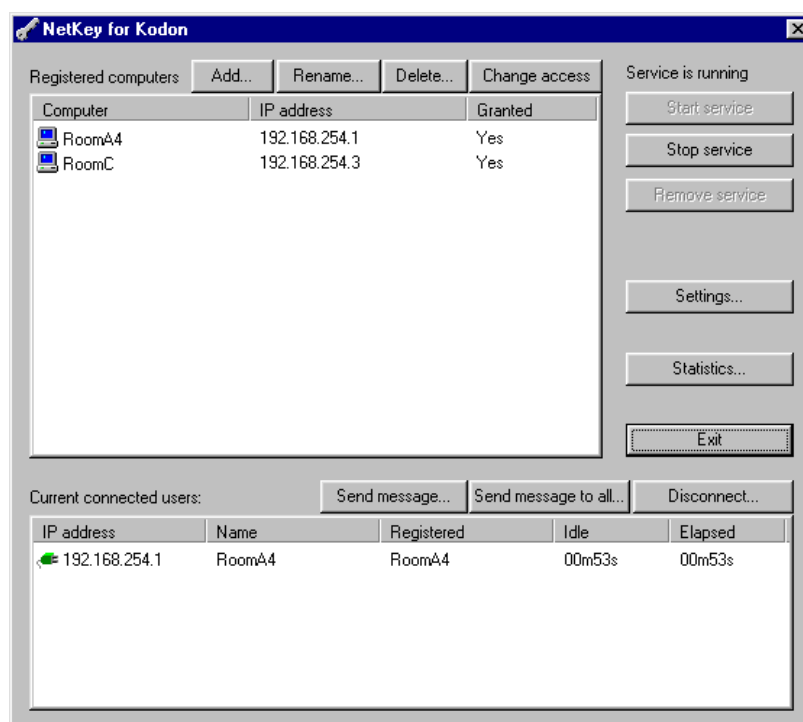


Figure 6. The Netkey server program, listing all computers that are granted access (top) and licenses in use (bottom).

More client computers can be added to the network by simply adding the IP address and the computer name as described in the previous paragraph.

4.3 Advanced features of the Netkey server program

The Netkey server program is a Windows NT/2000 Service. As such, it can be seen in the list of installed *Services*. The startup settings, i.e. Manual, Automatic or Disabled, can be specified from the Windows **Services** administration tool (*Control Panel > Administrative tools > Services*) If you close the Netkey configuration program, the service will not be halted. Even when the current user logs off, the service remains running in the background. To effectively shut down the service, click the **<Stop service>** button in the Netkey configuration window. If licenses are still in use, the program will produce a warning message, asking you to continue or not.

License granting

Each computer in the network can be granted or refused access to the application software by the server program. To refuse access to a particular computer, select it in the upper panel, and refuse its access with **<Change access>**. The blue screen icon changes into a red screen. To grant the access again, click **<Change access>** a second time. To permanently remove a computer from the users list, select the computer in the upper panel, and click **<Delete>**.

Disconnect users

The server can disconnect a client if needed. Select a user in the lower panel, and disconnect it (withdraw its license) with **<Disconnect>**.

Time-out

The *idle time* of each user is recorded by the Netkey server program. A time-out for inactive licenses can be specified: in case there is a waiting list, a client for whom the idle time exceeds the time-out value will lose his license in favor of the first in the waiting list. Specify a maximum idle time with **<Settings>**, and enter the minutes of idle time. **Note that a user who has exceeded the idle time limit will not be disconnected by the server as long as there is no waiting list.**

Maximum usage limit

The *usage time* by each client is recorded by the Netkey server program; it is the total connection time of the current session. A maximum usage time can be specified: in case there is a waiting list, a client for which the usage time exceeds the maximum usage time will lose his license in favor of the first in the waiting list. Specify a maximum usage limit with **<Settings>**, and enter the minutes of usage time. **Note that a user who has exceeded the maximum usage time limit will not be disconnected by the License server as long as there is no waiting list.**

Messaging

The License server can send messages to any or all connected clients, for example in case the server computer will be shut down or if a client will be disconnected. Send a message to one user by selecting the user in the lower panel, and **<Send message>**. Enter a message string and press **<OK>**. The user will receive the message in a dialog box. Send a message to all users with **<Send message to all users>**. Enter a message string and press **<OK>**. All active users will receive the message in a dialog box.

Usage statistics

The Netkey server program records every usage of each client. Graphical statistics can be displayed about the history of the usage over longer periods, and the relative usage of each client computer can be shown for any time interval. To view the usage history of the Fingerprinting II Informatix network version, click **<Statistics>**. The panel shows a detailed view of the number of computers that have used the software on a time scale divided in hours. You can scroll in this panel to view back in the past. The license limit is shown as a red line; computers in a waiting list are shown in red. The relative usage of each client computer can be shown by clicking the **<Relative usage>** tab. Enter the time

period (from-to) in Days / Months / Years. The result is a circle diagram with the percentage usage time for each computer shown.

4.4 Features of the Client program (Fingerprinting II Informatix)

Waiting lists

In case the maximum license number is exceeded, the server program manages a waiting list. The client receives a message with its number in the waiting queue, and the Fingerprinting II Informatix software pops up as soon as the client's license becomes available.

The user can request an overview of the computers currently using a Fingerprinting II Informatix license by click the *Network settings* button  in the Fingerprinting II Informatix Startup program, and then clicking <Status>. It shows for each connected computer the IP address, the computer name, the total usage time and the idle time.

Disconnection by server or license loss

If the client is disconnected by the server or loses its license, e.g. due to idle time or maximum usage limit, a warning box flashes that you should save any unsaved data and quit the program immediately. Fingerprinting II Informatix tries four times again to negotiate its license with intervals of 15 seconds. After the fourth time (1' 15" in total), the program halts automatically.

5. Starting and setting up Fingerprinting II Informatix

5.1 The Fingerprinting II Informatix Startup program

- 5.1.1 Double-click the “Fingerprinting II Informatix” icon on the desktop to run the **Startup program**. This program (see Figure 7) shows the *Intro screen* with version number and license site information (1). It allows you to run the Fingerprinting II Informatix server application *Analysis*, to *Inspect* the databases with the *Diagnostics* program (4 and 5 respectively), to create *New* databases (6), to run import or *Filter* programs (7), and customize various *Settings* (8) such as colors, directories, order of information fields and experiments, for the selected database (2). The *Home directory* for the databases can be specified (2).
- 5.1.2 Use the <*Exit*> button (9) when you are finished running the Fingerprinting II Informatix applications.

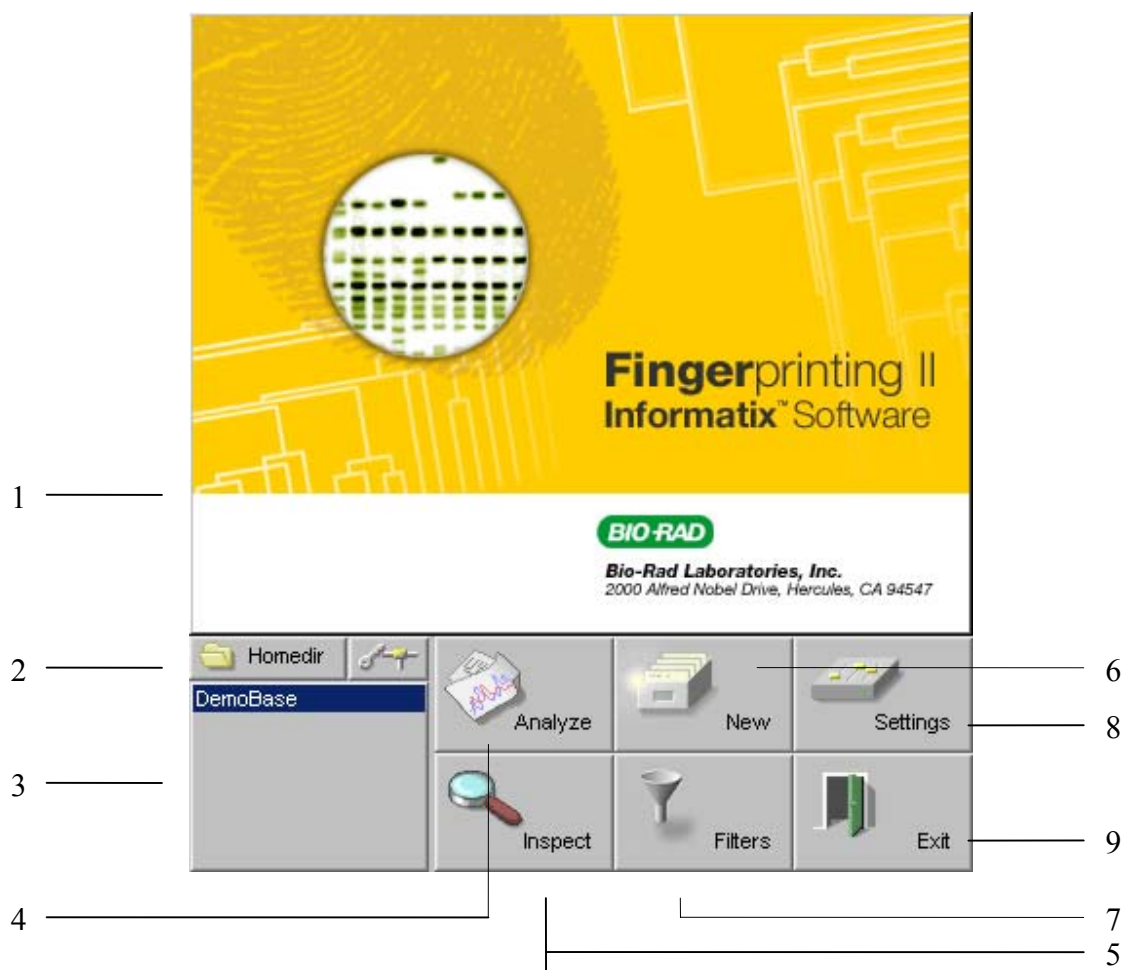


Figure 7. The Fingerprinting II Informatix Startup program.

The **Diagnostics program** analyzes system features of your computer and detects and reports any errors in the selected database. It checks the Fingerprinting II Informatix drives and directories of the selected database and inspects the validity of the files contained therein.

5.2 Creating a database

In order to facilitate the use of Fingerprinting II Informatix in different research projects, it is possible to set up *Databases*. The principles of a database are explained in 1.3. The Fingerprinting II Informatix Startup program will look for all databases in one *Home directory*, specified by the user. Note that, in Windows NT, Windows 2000, and Windows XP, each Windows user may specify a different home directory. The Fingerprinting II Informatix home directory is saved with the system registry of the user.

If you want to change the current home directory follow the steps below:

- 5.2.1 Press the button **<Homedir>** on the startup screen, and select the desired directory. You can also specify a directory on a network drive, on condition that this drive is permanently available.
- 5.2.2 Press **<OK>** to select the new home directory. The program updates the list of available databases in the new directory.

Fingerprinting II Informatix offers two alternative database solutions to store its databases: the program's own built-in database (=local database) or an external ODBC compatible database engine. The latter solution is called a *Connected Database*. Connected databases are particularly useful for environments where data is already stored in a relational database, or where vast amounts of data are generated. It also allows more professional database setups to be achieved, for example with different access/permission settings for different users. In addition, when the connected database support multi-user access, Fingerprinting II Informatix becomes a true multi-user application. Currently supported are Microsoft Access, SQL Server, and Oracle. Note that Connected Databases are only available with the Database Sharing Tools module.

However, for most purposes and single-user access, the local Fingerprinting II Informatix database is quick, powerful, and easy to set up and maintain. We will describe the use of local databases in this chapter. The use of connected databases is discussed in detail in chapter 22.

Create a new database as follows:

- 5.2.3 Press the **<New>** button to enter the *New database* wizard.
- 5.2.4 Enter a name for the database, e.g. **Example**, and press **<Next>**.

Notes:

- *The program automatically creates a subdirectory of the current home directory for the database directories. If you want to change this (not recommended), press **<Change>**. This option allows you to select any directory from any permanent drive. It is recommended to create a new empty directory before you choose it as database directory.*
- *If you do not want the program to automatically create subdirectories, click **No** to this question (not recommended). In that case, you will have to create the subdirectories manually (see Figure 3).*

- 5.2.5 Press **<Next>** again.

5.2.6 You are now asked whether or not you want to create log files. If you enable Fingerprinting II Informatix to create log files, every change made to a database component (entry, experiment etc.) is recorded to the log file with indication of the kind, the date, and the time of change.

5.2.7 Press **<Finish>** to complete the setup of the new database.

The Fingerprinting II Informatix database is now created. A new dialog box pops up, asking if you want to store the data as a **Local database** or as a **Connected database**.

5.2.8 Select **Local database** and press **<OK>**.

5.3 Settings of a database

5.3.1 In the Startup program, select the new database and press the **<Settings>** button. The *Database settings* dialog box appears (Figure 8). Here you can change the **Database directory**, you can **Enable the use of log files** (5.2.6), you can remove the current database with **Delete database**, and you can modify the window background colors for the database using red, green and blue slide bars. With **<ID code>** you can install an ID code to protect all important settings in the database.

5.3.2 The database tab allows the order of the database fields to be changed, and four experiment types tabs allow the order of the experiments to be changed. In the new database, these tabs are initially empty since new database fields or experiments are not defined yet.

5.3.3 Press **<OK>** or **<Cancel>** to exit the database settings dialog box.

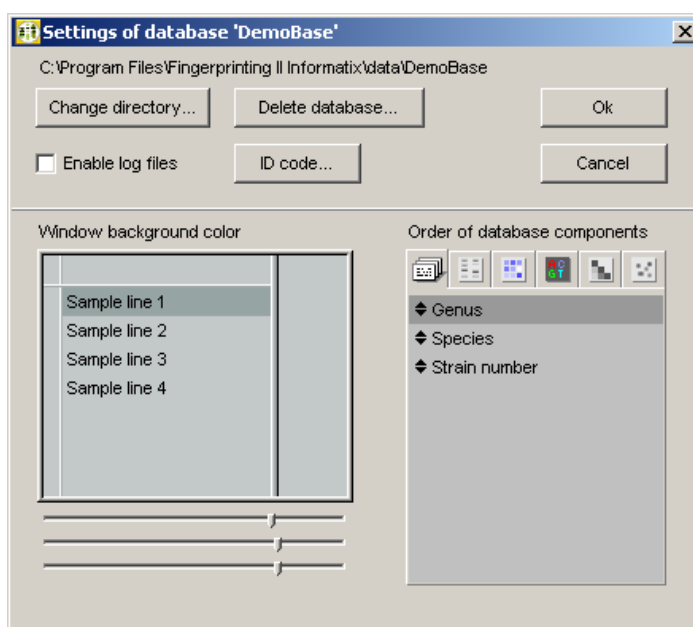


Figure 8. The *Database settings* dialog box in the Startup program.

5.4 Database protection tools

In order to protect a database against incidental data loss, it is possible to lock the important settings and data files in the database. The following files can be locked:

- The settings files of the experiment types: as long as this file is locked, the settings for the experiment type cannot be changed.
- The data files for the experiment types: data of existing entries cannot be changed, however, new entries can be added in a new experiment data file.
- Libraries for identification: nothing can be changed in a locked library, but it still can be used for identification.

Each file can be locked and unlocked separately, so that it is possible to lock and protect "final" files and leave other files open for additional input.

5.4.1 The setting files, data files, and libraries in the database can be locked using the **File|Lock** command in the file's edit window. Once the settings are locked they cannot be changed anymore, until they are unlocked again by executing the command **File|Lock**.

A locked file is shown with a small key icon left from the filename in the **Files** panel of the Fingerprinting II Informatix main window, and a key icon also appears in the left upper corner of the file's edit window.

To protect a database against modification by others or misuse, Fingerprinting II Informatix allows an *ID code* to be set for a database. Once an ID code is set, the database settings can only be changed after entering the ID code. In addition, locked files can only be unlocked or vice versa after entering the ID code.

5.4.2 In order to set an ID code for the database, run the startup program and press **<Settings>**.

5.4.3 In the database settings (Figure 8), press **<ID code>** and enter the ID code. Any string of characters is allowed. The program will ask you to confirm this by entering the ID code a second time.

5.4.4 If you want to remove an ID code, press **<ID code>** in the database settings and leave the input box empty.

WARNING: if you forget the ID code, you will need a special bypass code from Bio-Rad Laboratories.

5.5 Log files

In certified environments and laboratories where conscientious recording of manipulations is important, the *log files* in Fingerprinting II Informatix are a very useful tool. For every Fingerprinting II Informatix session, the log files show the Windows user who has last made the changes together with the kind of changes and the date and hour.

Log files are recorded for the following files:

- The database: the log file lists any changes in names of database fields, any entries that are added or deleted, and keys of entries that are changed. It also reports if new experiment types are created, if experiment types have been renamed or removed.

- The settings files of the experiment types: for every change made, the kind of change are indicated. All settings are recorded in the log file, so that the user may restore the previous settings based upon the log file, if enabled.
- The data files for the experiment types: if data for entries are changed, the log file lists from which entries. It also mentions the creation of new experiments, and the deletion of experiments.
- Libraries for identification: the log file keeps record of any changes in library units and records the addition and deletion of library units.

5.5.1 In order to enable the creation of log files for the database, run the startup program and press **<Settings>**.

5.5.2 In the database settings (Figure 8), check the *Enable log files* checkbox.

5.5.3 In the data file windows, experiment file windows, the main window, or the library window, select **File|View log file** or  to display the log file. In a local database (as opposed to a Connected Database; see chapter 22), Fingerprinting II Informatix creates a temporary file, DBASE.LOG or <EXPERIMENTNAME>.LG. For data files, it creates a log file <DATAFILE>.LOG. The log files are loaded in Fingerprinting II Informatix' Event log viewer (Figure 9).

5.5.4 You can clear a log file with the command **<Clear>** or copy it to the clipboard with **<Copy to clipboard>**.

From the clipboard it can be pasted in other applications with a **Paste** command. The text is formatted as RTF (Rich Text Format) which enables the formatting to be kept in other software that supports RTF.

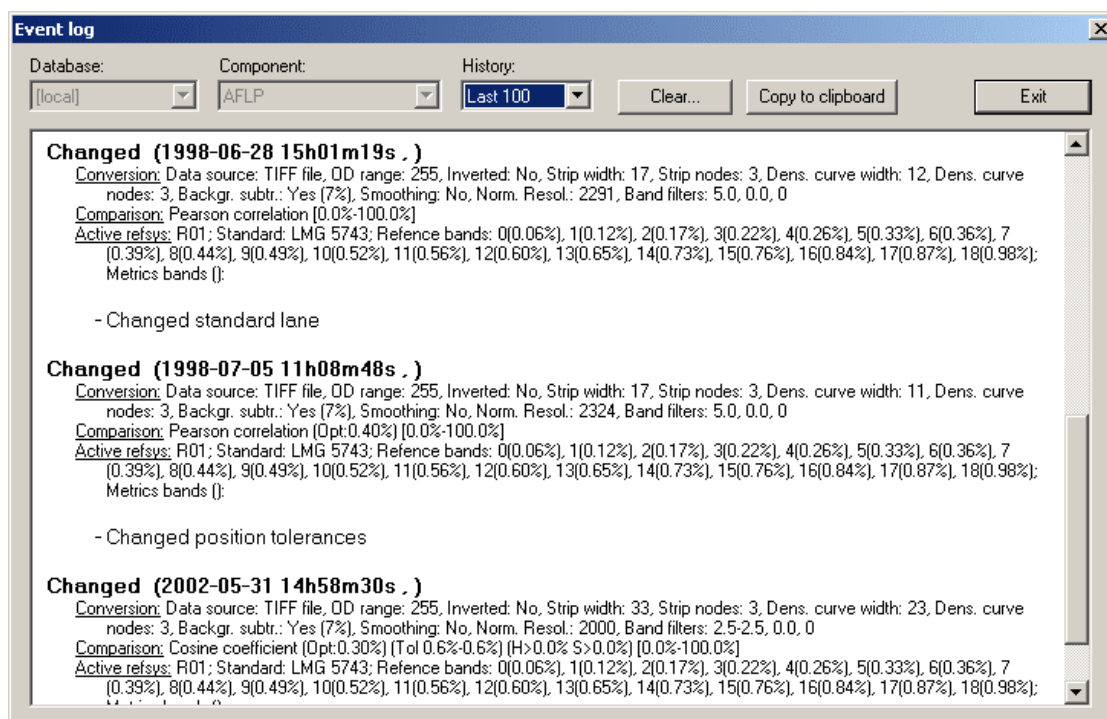


Figure 9. Event log viewer for a database component.

The items *Database* and *Component* are only applicable to Connected databases (see chapter 22).

6. Database functions

6.1 The Fingerprinting II Informatix Main window

6.1.1 In the Startup program, select the user Demobase, and click on the <Analyze> button.

The *Main* window (Figure 10) consists of a menu, a toolbar for quick access to the most important functions, a status bar, and the following five panels:

- The *Database* panel, listing all the available entries in the database, with their information fields and their unique keys (see 1.2).
- The *Experiment types* panel, showing the different experiment types, and the experiments that are defined under each type.
- The *Experiment presence* panel, which for each database entry shows whether an experiment is available (green dot) or not. Clicking on a green dot causes the *Experiment card* for that experiment to be popped up (see 8.2).
- The *Experiment files* panel, showing the available data files for the selected experiment.
- The *Comparisons* panel, listing all comparisons that are saved.
- The *Libraries* panel, which shows the available identification libraries.

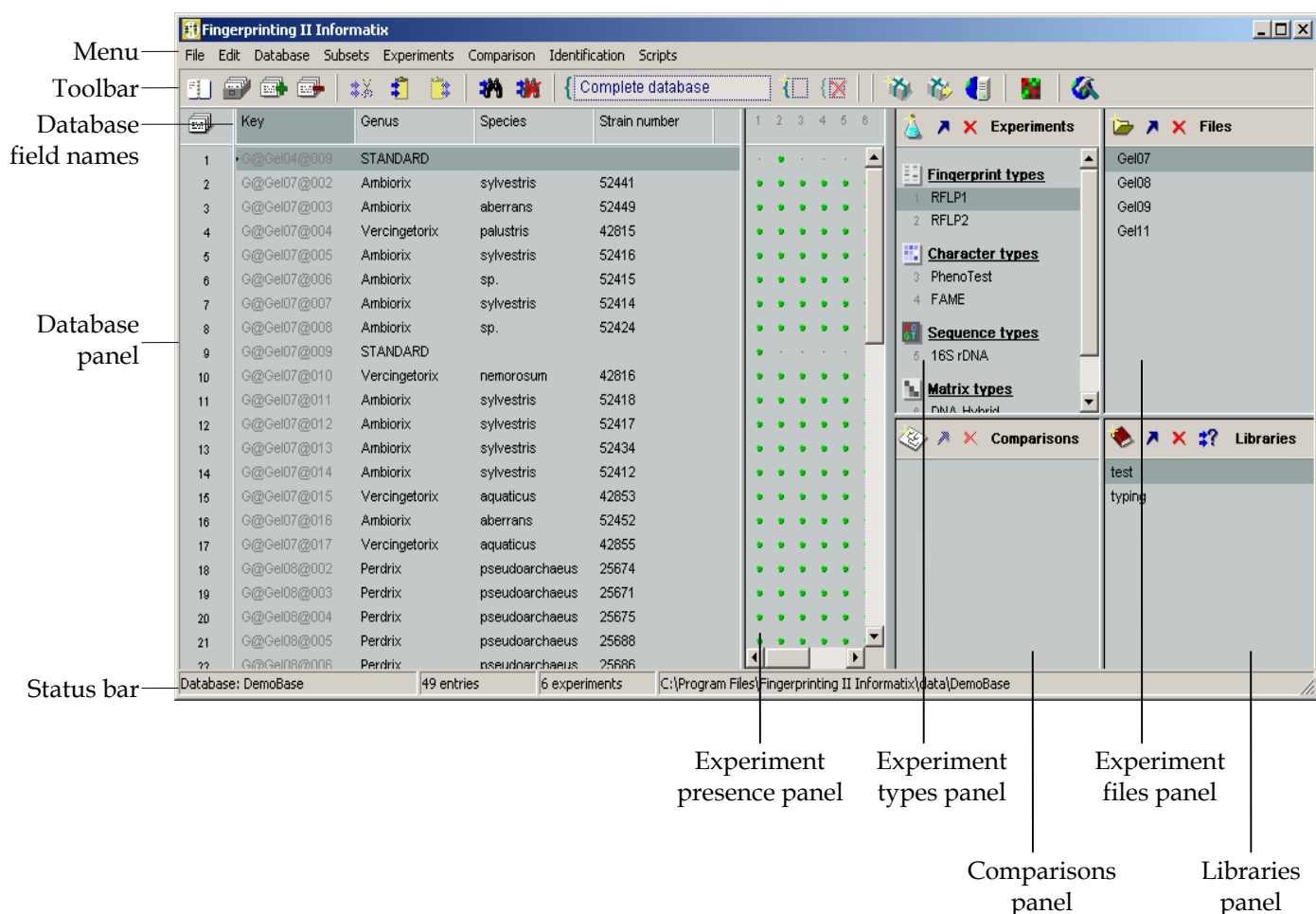


Figure 10. The main window of the Fingerprinting II Informatix Analyze program.

6.2 Adding entries to the database

In the database **Demobase**, there are already entries defined. In most further exercises in this guide, we will work on our own database **Example**. Therefore we will start Fingerprinting II Informatix again with this new database:

6.2.1 Select **File | Exit** to quit the Analyze program.

6.2.2 Back in the Startup program, select the database **Example**.

6.2.3 Press **<Analyze>** to start the Analyze program with the empty **Example** database.

Adding entries to the database can happen in two ways:

- You can add one or more entries directly to the database. Initially, these entries will be empty and no experiments will be linked to them. When you import experiment data later, you can link the data to the entries.
- When you import a file of experiments, the program will ask you whether you want it to automatically create a corresponding database entry for each experiment.

We will now create a few database entries without importing experiments.

6.2.4 Select **Database | Add new entries** or  in the toolbar.

A dialog box appears, asking for the number of new entries to create, and the database where they should be created. When there are Connected Databases associated with the database (see chapter 22), there is a possibility to add the new entries either in the local database or in the Connected Database.

The input field in the bottom of the window allows a key to be entered by the user. This input field is only accessible when one single entry is added. As soon as the number of entries is specified to be more than one, the field is disabled.

6.2.5 Enter the number of entries you want to create, e.g. 3, and press **<OK>**.

The database now lists three entries with a unique key automatically assigned by the software. Usually, one will not want to change this entry key, but in special cases, it may be useful to change or correct the key manually. This can be done as follows.

6.2.6 Select the entry and **Database | Change entry key**.

6.2.7 Change the entry key in the input box, e.g. *Entry 1*, and press **<OK>**.

The key is a critical identifier of the database entries, and if you already have unique labels that identify your organisms under study, you can use these labels as keys in Fingerprinting II Informatix. The key can then be considered as an additional database field. As we will explain later, the key is also an important component in automatically linking experiments to existing database entries.

*NOTE: Remember the use of floating menus as described in 2.2: right-clicking in the database panel of the main window would directly pop up the menu **Add new entries** and **Change entry key** (if you click on an entry).*

6.2.8 To remove an entry from the database, select one of the entries, e.g. the third one, and **Database | Remove entry** or  in the toolbar. The program asks to confirm this action.

To remove all entries that have no experiment linked to them, you can select **Database | Remove unlinked entries**. In the case of our example database this would result in removal of all entries, since none has an experiment linked yet.

6.3 Creating database information fields

Besides the key, which is the primary information field for each database entry, the user can define a number of database information fields. An information field may contain up to 80 characters, and a maximum of 60 fields can be defined for a database.

6.3.1 Select **Database | Add new information field**.

6.3.2 Enter the name of the database information field, for example *Genus*, and press **<OK>**.

6.3.3 Select **Database | Add new information field** again to define the second field, *Species*.

6.3.4 Then, select **Database | Add new information field** again to define a third field, *Subspecies*.

6.3.5 Finally, select **Database | Add new information field** again to define a field *Strain no*.

The menu functions *Database|Rename information field* and *Database|Remove information field* can be used to rename and remove an information field, respectively.

6.4 Entering information fields

6.4.1 By double-clicking, or pressing Enter on a database entry, the *Entry edit window* appears (Figure 11). Right-clicking on the entry, and selecting *Open entry* also works.

The left panel shows the information fields and the right panel shows the available experiments for the entry. This window can be rescaled to see more and/or longer information fields.

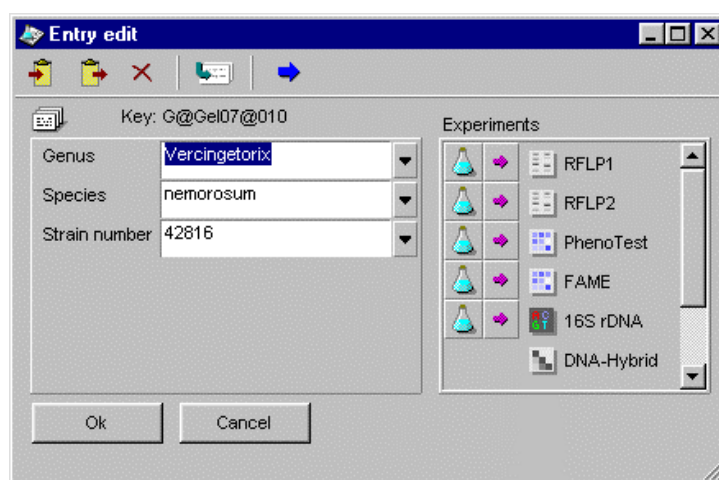


Figure 11. The *Entry edit window*.

6.4.2 Enter some information in each of the fields (see Figure 11).

6.4.3 If a number of entries have mostly the same fields, you can copy the complete entry information to the clipboard using the F7 key or .

6.4.4 To clear the complete information of the entry, press .

6.4.5 To paste the information from the clipboard, press the F8 key or .

If some of the information fields are the same as entered for previous entries (for example genus and species name), you can drop down a history list for each information field. The history lists can contain up to 10 previously entered strings for the information field. Using the history lists is recommended (i) to save time and work and (ii) to avoid typographical errors.

6.4.6 Drop down a history list by clicking the  button right from the information field. A floating menu appears from which you can select an information string.

The  button is related to ODBC communication with an external database (see chapter 20.4.6).

6.4.7 Using , you can select or unselect the opened entry in the database (see 9.2), for the construction of comparisons. When the entry is selected, this button shows as .

6.4.8 Press the Enter key or <OK> to close the entry edit window and store the information, or press the Escape key or <Cancel> to close the window without changing any information.

In order to quickly enter the same information for many entries, the use of the keyboard is recommended: use the Arrow Up and Down keys to move through the entries in the database, use the Enter key to edit an entry, use the F7 and F8 keys to copy and paste information, and use the Enter key again to close the entry edit window.

6.5 Configuring the database layout

Entries in the database can be ordered alphabetically by any of the information fields.

6.5.1 Click on one of the database field names in the database panel header (see Figure 10).

6.5.2 Select *Edit | Arrange entries by database field*.

When two or more entries have identical strings in a field used to rearrange the order, the existing order of the entries is preserved. As such it is possible to categorize entries according to fields that contain information of different hierarchical rank, for example *genus* and *species*. In this case, first arrange the entries based upon the field with the lowest hierarchical rank, i.e. *species*, and then upon the higher rank, i.e. *genus*.

When a field contains numerical values, which you want to sort according to increasing number, use *Edit | Arrange entries by database field (numerical)*. In case numbers are combined numerical and alphabetical, for example entry numbers [213, 126c, 126a, 126c], you can first arrange the entries alphabetically (*Edit | Arrange entries by database field*), and then numerically using *Edit | Arrange entries by database field (numerical)*. The result will be [126a, 126b, 126c, 213].

The number of characters to be displayed for a database field can be specified as follows.

6.5.3 Click on one of the database info fields in the database panel header (see Figure 10).

6.5.4 Select *Edit | Set database field length*.

6.5.5 Enter a number between 0 and 80, and press <OK>.

6.5.6 The width of each database field can be adjusted by dragging the separator lines between the database field names to the left or to the right.

For example, if the key is not informative for your database, you can drag the separator line between the key and the first information field as much to the left as possible. If the genus name for the organisms is known or mostly the same, you can abbreviate it to one character and drag the separator between Genus and Species as much to the left as possible to show one character.

6.5.7 The width of the database panel as a whole can be changed by dragging the separator line between the database panel and the experiment presence panel to the left or to the right.

Settings 6.5.4 to 6.5.7 as well as all window sizes and positions are stored when you exit the software and are specific for each database.

7. Setting up experiments

In Fingerprinting II Informatix, experiments are divided in four classes: *fingerprint types*, *character types*, *sequence types*, and *matrix types*.

The **fingerprint types** include any densitometric record seen as a profile of peaks or bands. Examples are electrophoresis patterns, gas chromatography or HPLC profiles, spectrophotometric curves, etc. For example, within the fingerprint types, you can create a Pulsed Field Gel Electrophoresis (PFGE) experiment type with specific settings such as reference marker, MW regression, stain, band matching tolerance, similarity coefficient, clustering method, etc.

With the **character types**, it is possible to define any array of named characters, binary or continuous, with fixed or undefined length. Further adjustable features include the range of the characters, the number of digits, the color scale, etc. Examples of character types are antibiotics resistance profiles, fatty acid profiles (if the fatty acids are known), metabolic assimilation or enzyme activity test panels such as API, Biolog, and Vitek, etc. Single characters such as Gram stain, length, etc. also fall within this category. A number of existing microbial identification systems are present as predefined character types.

Within the **sequence types**, the user can enter sequences of nucleic acids (DNA and RNA) and amino acids. Fingerprinting II Informatix recognizes widely used sequence file formats such as EBML, GenBank, and Fasta with import of user-selected header tags as information fields, and optional storage of headers. Other sequence formats can be imported easily.

With the **matrix types**, it is possible to import external similarity matrices, providing similarity between entries revealed directly by the technique, or by other software. These matrices can be linked to the database entries in Fingerprinting II Informatix and they are used together with other information to obtain classifications and identifications. An example of a matrix type is a matrix of DNA homology values. DNA homology between organisms can only be expressed as pairwise similarity, not as character data.

7.1 Defining a new fingerprint type

Before we create a new fingerprint type, we will copy an example TIFF file from the CD-ROM. The CD-ROM contains a directory **EXAMPLES**, which contains a gel **GEL_01.TIF**. After a standard installation of Fingerprinting II Informatix, the default path to copy gel files is

C:\Program Files\Fingerprinting II Informatix\Data\Example\Images.

The recommended way of working is to save new gel files immediately in this directory. However, if TIFF files are already present in another directory, you can select the fingerprint type in the **Experiments** panel, simply drag the files from the Windows explorer and drop it into the **Files** panel. Alternatively, you can select **File|Import experiment file** in the Fingerprinting II Informatix main menu, and select the TIFF files you want to process in this fingerprint type. In both cases, Fingerprinting II Informatix makes a copy of the original TIFF file in the Images subdirectory.

7.1.1 In Fingerprinting II Informatix, select **Experiments|Create new fingerprint type** from the main menu, or press  and **New fingerprint type**.

- 7.1.2 The *New fingerprint type* wizard prompts you to enter a name for the new type. Enter a name, for example "RFLP".
- 7.1.3 Press **<Next>** and check the type of the fingerprint data files. The default settings correspond to the most common case, i.e. two-dimensional TIFF files with 8-bit OD depth (256 gray values).
- 7.1.4 After pressing **<Next>** again, the wizard asks whether the fingerprints have inverted densitometric values. This is the case when you are using EtBr stained gels, photographed under UV light (such as the example provided). The bands then appear as fluorescently lighting on a black background. Since Fingerprinting II Informatix recognizes the darkness as the intensity of a band, you should answer *Yes*, to allow the program to automatically invert the values when converting the images to densitometric curves. Furthermore, the wizard allows you to adjust the color of the background and the bands to match the reality. The red, green and blue components can be adjusted individually for both the background color and the band color. Usually, you will leave the colors unaltered.
- 7.1.5 In the next step, you are prompted to allow a **Background subtraction**, and to enter the size of the disk, as a percentage of the track length. The default disk size of 10% will suit for most fingerprint types. For high resolution fingerprints (e.g. AFLP and sequencer-generated patterns) you can try a smaller disk size. Later, we will see how we can have the program propose the optimal background subtraction settings automatically. At this time, we leave the background subtraction disabled.
- 7.1.6 Press **Finish** to complete the creation of the new fingerprint type.

NOTE: You will be able to adjust any of these parameters later on.

The *experiment types* panel (Figure 10) now lists "RFLP" under **Fingerprint types**.

- 7.1.7 Run the Windows Explorer, and select file **GEL_01.TIF** in the directory **EXAMPLES** on the CD-ROM.
- 7.1.8 Drag and drop the file in the **Files** panel in the Fingerprinting II Informatix main window.

One gel becomes available in the *experiment files* panel (Figure 10), **Gel_01**. The file is marked with **N**, which means that it has not been edited yet. Any other gel TIFF file you want to process can be imported in the same way in the current database (see Figure 3). The program will list these TIFF files in the experiment files panel.

*NOTE: Experiment files added to the Experiment files panel can also be deleted by selecting the file and choosing **File|Delete experiment file** from the main menu. Deleted experiment files are struck through (red line) but are not actually deleted until you exit the program. So long, you can undo the deletion of the file by selecting **File|Delete experiment file** again.*

7.2 Processing gels

An experiment file is edited in two steps: in a first step, the data are entered or edited, and in a second step, the data is assigned to the database entries.

- 7.2.1 Click on **Gel_01** in the experiment file panel, and then select **File|Open experiment file (data)** in the main menu.

Since the gel is new (unprocessed), Fingerprinting II Informatix doesn't know what fingerprint type it belongs to. Therefore, a list box is first shown, listing all available fingerprint types, and allowing you to select one of them, or to create a new fingerprint type with <Create new>. In this case, there is only one fingerprint type available, RFLP.

7.2.2 Select RFLP and press <OK>.

The gel file is being loaded, which may take some time, depending on the size of the image. The Fingerprint data editor window appears (Figure 12), showing the image of the gel.

7.2.3 If the gel needs to be mirrored, use the tools *File|Tools|Vertical mirror of TIFF image* or *File|Tools|Horizontal mirror of TIFF image*. Using two times the same command restores the original TIFF file.

The whole process of lane finding, normalization, band finding and band quantification is contained in a wizard, allowing the user to move back and forth through the process and make changes easily in whichever step of the process. The  and  buttons in the toolbar are to move back and forth, respectively. The process involves the following steps, shown in the toolbar of the window: **1. Strips** (defining lanes), **2. Curves** (defining densitometric curves), **3. Normalization**, and **4. Bands** (defining bands and quantification).

Within each of these four steps, there is an undo/redo function. To undo one or more actions, you can use the undo button , or *Edit|Undo* (CTRL+ Z) from the menu. To redo one or more actions, use the redo button , or *Edit|Redo* (CTRL+Y) from the menu. Once you have moved from one step to another, the undo/redo function within that step is lost.

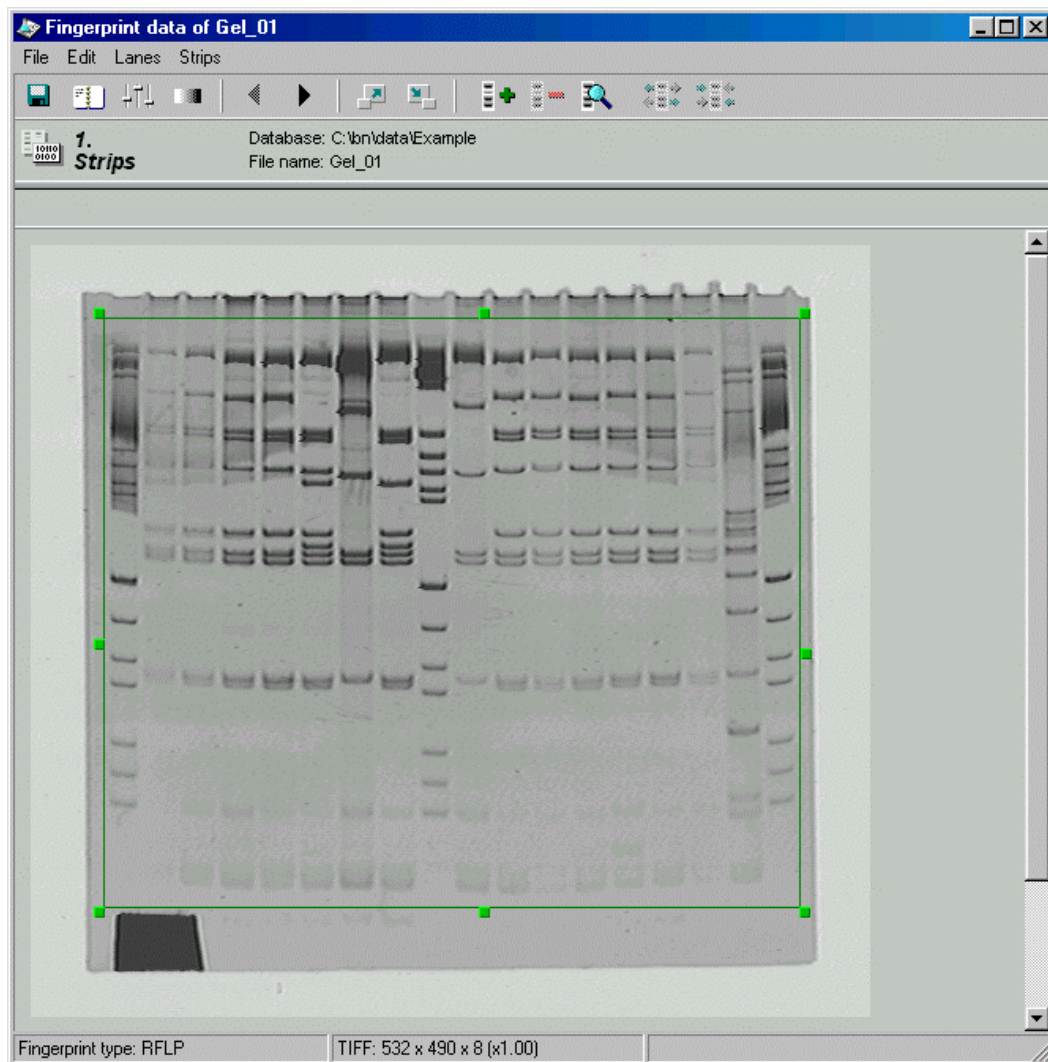


Figure 12. The *Fingerprint data editor* window. Step 1: defining pattern strips.

- 7.2.4 At the start, the image is shown in original size (x 1.00, see status bar of the window). You can zoom in and zoom out with *Edit|Zoom in* and *Edit|Zoom out*, or using the  and  buttons, or the + and - keys, respectively.
- 7.2.5 When a large image is loaded, a *Navigator window* can be popped up to focus on a region of the image. To call the navigator, double-click on the image, or right-click and select *Navigator* from the floating menu.
- 7.2.6 You can change the brightness and contrast of the image with *Edit|Change brightness & contrast* or with . This pops up the Image brightness & contrast dialog box (Figure 13).
- 7.2.7 In the brightness and contrast dialog box, click *Dynamical preview* to have the image directly updated with changes you make.
- 7.2.8 Use the *Minimum value* slide bar to reduce background if the background of the whole image is too high.

7.2.9 Use the *Maximum value* slide bar to darken the image if the darkest bands are too weak.

The option *Rainbow palette* can be used to reveal even more visual information in areas of poor contrast (weak and oversaturated areas) by using a palette that exists of multiple color transitions.

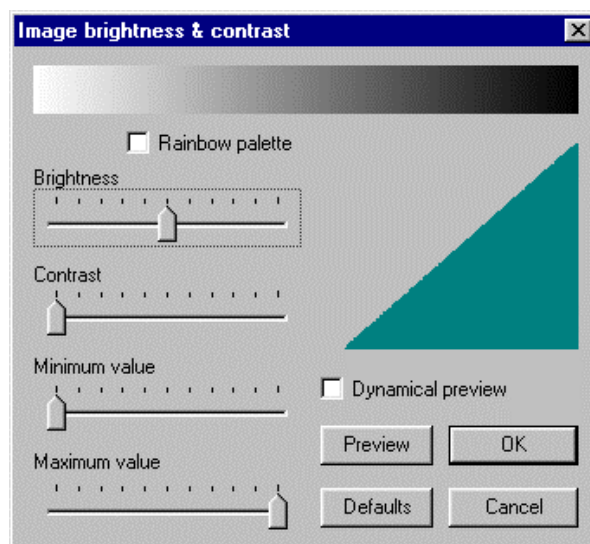


Figure 13. Image brightness & contrast dialog box.

7.2.10 If you press <OK>, the changes made to the image appearance are saved along with the fingerprint type.

NOTE: The brightness and contrast settings are saved along with the fingerprint type, but are not specific for a particular gel. The Tone curve editor, as explained further, is a more powerful image enhancement tool for which the settings are saved for each particular gel.

7.2.11 To save the work done at any stage of the process, you can select *File|Save*, press the F2 key or the  button. In case you work with complex gels, it is advisable to save the work at regular times.

When you save the gel file with *File|Save*, the program may prompt you with the following question: “*The resolution of this gel differs considerably from the normalized track resolution. Do you wish to update the normalized track resolution?*”. The resolution is explained further (see 7.3.27). If the question appears (not for the example gel), answer <Yes>.

The green rectangle delimits the region of interest of the gel: tracks and gelstrips will be extracted within the borders of this rectangle. Adjust the rectangle by dragging the green squares as necessary, and use the SHIFT key to distort the rectangle in each square separately.

7.2.12 Select *Lanes|Auto search lanes* or  to let the program find the patterns automatically. A dialog box asks you to enter the approximate number of tracks on the gel.

Each found lane on the image is represented by a *strip*: a small image that is extracted from the complete file to represent a particular pattern. The borders of these strips are represented as blue lines, or red for the selected lane (see Figure 15). By default, the strip thickness is 31 points, which is too wide in this example.

- 7.2.13 Call the *Fingerprint conversion settings* dialog box with *Edit|Settings* or . This dialog box consists of four tabs, of which the tab corresponding to the current stage of the processing is automatically selected. Since we are now in the first step (defining strips), the *Raw data* tab is selected (see Figure 14).

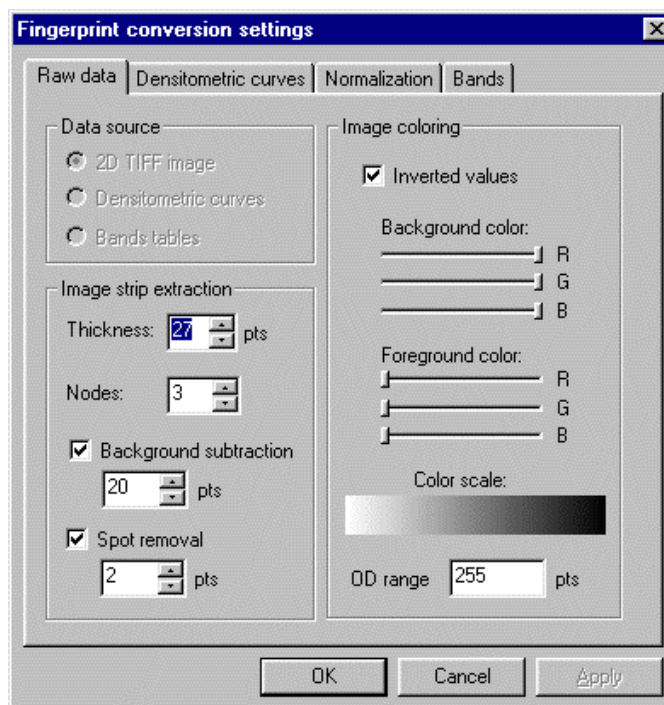


Figure 14. The *Fingerprint conversion settings* dialog box. *Raw data* tab.

- 7.2.14 Adjust the *Thickness* of the image strips so that the blue lines enclose the complete patterns (blue lines of neighboring patterns should nearly touch each other). See Figure 15 for an optimally adjusted example.

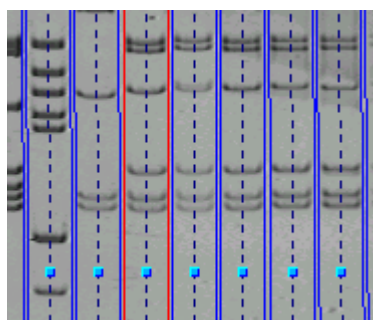


Figure 15. Optimal strip thickness settings, detail.

- 7.2.15 If necessary, increase the number of distortion nodes. These nodes allow you to bend the strips locally. Usually, three nodes should be fine.

Two more options, *Background subtraction* and *Spot removal* allow gel scanings with irregular background and spots or artifacts to be cleaned up to a certain extent. It should be emphasized that the options *Background subtraction* and *Spot removal* have an influence on *gelstrips* in all further processes of the program: gelstrips will always be shown with background subtracted and with spots

removed. In addition, when two-dimensional quantification is done, the gelstrips with background subtracted and spots removed are used. Hence, we recommend NOT to use these options unless (1) the image has a strong irregular background, for example by non-homogeneous illumination of the gel, so that the gelstrips would not look appropriate for presentation or publication; (2) the gel contains numerous spots that would influence the densitometric curves extracted from the gelstrips (spots on the image are seen as peaks on a densitometric curve, and hence have a strong impact on correlation coefficients, band searching etc.).

The **Background subtraction** is based on the “rolling ball” principle, and the size of the ball in pixels of the image can be entered. The larger the size of the ball, the less background will be subtracted.

The **Spot removal** is a similar mechanism as the rolling ball, but an ellipse is used instead, in order to separate bands from spots. The size of the ellipse can be entered in pixels. Unlike the background subtraction, the size of the ellipse should be kept as small as possible in order not to erase bands.

NOTES:

1. *The spot removal mechanism inevitably causes some distortion on the patterns. The smaller the size of the background removal, the less the distortion.*
2. *If background subtraction on the gelstrips is applied, it is not necessary anymore to perform background on the densitometric curves, since this is doing exactly the same but on one-dimensional patterns.*

The effect of background subtraction and spot removal on gelstrips is only seen in the next step, when the gelstrips are shown. Since the example gels do not require these features, we will not further discuss them.

7.2.16 Click **<OK>** to validate the changes.

7.2.17 Adjust the position of each spline as necessary by grabbing the nodes using the mouse. Use the SHIFT key to bend a spline locally in one node.

7.2.18 Add lanes with **Lanes | Add new lane** or the ENTER key or . A new lane is placed right from the selected one.

7.2.19 Remove a selected lane with **Lanes | Delete selected lane** or DEL or  if necessary.

7.2.20 If one lane is more distorted than the number of nodes can follow, you can increase the number of nodes in that lane by selecting it and **Strips | Increase number of nodes**.

7.2.21 If the lanes are not equally thick, you can increase or decrease the thickness of each individual strip with **Strips | Make larger** and **Strips | Make smaller** (F7 and F8, or  and ), respectively.

Once the lanes are defined on the gel, a more powerful tool to edit the appearance of the image is the *Tone curve editor*. While the *Image brightness and contrast* settings act at the screen (monitor) level, i.e. after the TIFF grayscale information is converted into 8-bit grayscale, the *Tone curve editor* acts at the original TIFF information level. This means that, in case a gel image is scanned as 16-bit TIFF file, the tone curve settings are applied to the full 16-bit (65000) grayscale information which allows much more information to be magnified in particular areas of darkness. The advantages are:

- Weak bands are much better enhanced resulting in a smoother and more reliable picture.

- The tone curve acts at a level below the brightness and contrast settings and can be saved along with a particular gel. In all further imaging tools of the program, the tone curve for the particular gel is applied. Brightness and contrast settings are not specific to a particular gel.
- The user can fine-tune the tone curve to obtain optimal results. This will be explained below.

7.2.22 First select the brightness and contrast window with *Edit|Change brightness & contrast* or with , and press <Defaults> to restore the defaults.

7.2.23 In the gel window menu, select *Edit|Edit tone curve*. The *tone curve editor* appears as in Figure 16.

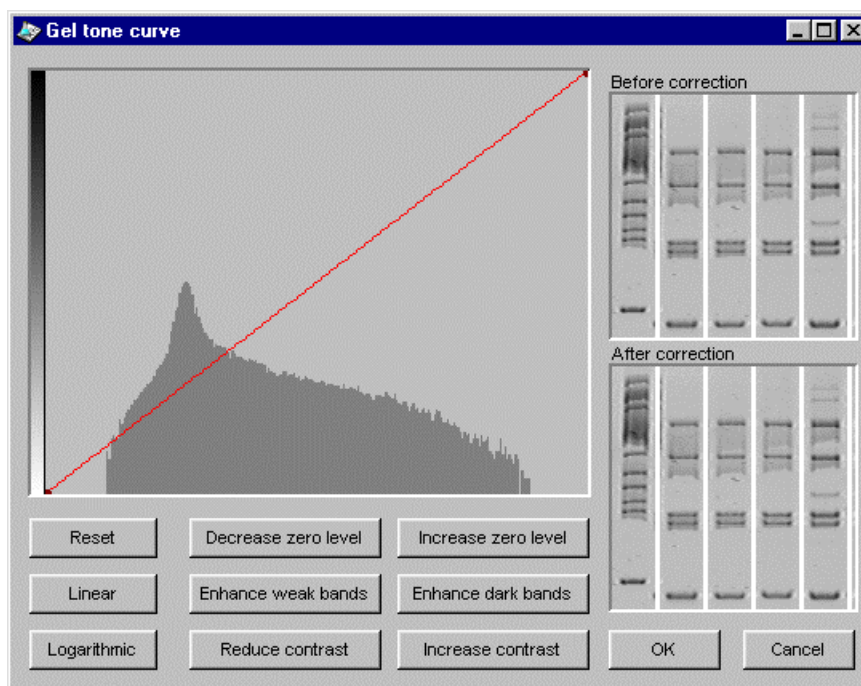


Figure 16. Gel image tone curve editor.

The upper panel is a distribution plot of the densitometric values in the TIFF file over the available range. The right two windows are a part of the image *Before correction* and *After correction*, respectively.

7.2.24 You can scroll through the preview images by left-clicking and moving the mouse while keeping the mouse button pressed.

7.2.25 Select a part of the preview images which contains both very weak and dark bands.

Left, there are two buttons, <Linear> and <Logarithmic>. Both functions introduce a number of distortion points on the tone curve, and reposition the tone curve so that it begins at the grayscale level where the first densitometric values are found, and ends at its maximum where the darkest densitometric values are found. This is a simple optimization function that rescales the used grayscale interval optimally within the available display range. The difference between linear and logarithmic is whether a linear or a logarithmic curve is used.

7.2.26 In case of 8-bit gels, a linear curve is the best starting point, so press <Linear>. The interval is now optimized between minimum and maximum available values, and the preview *After correction* looks a little bit brighter.

There are six other buttons that are more or less self-explaining: **<Decrease zero level>** and **<Increase zero level>** are to decrease and increase the starting point of the curve, respectively.

<Enhance weak bands> and **<Enhance dark bands>** are also complementary to each other, the first making the curve more logarithmic so that more contrast is revealed in the left part of the curve (bright area), and the second making the curve more exponential so that more contrast is revealed in the right part of the curve (dark area).

<Reduce contrast> and **<Increase contrast>** make the curve more sigmoid so that the total contrast of the image is reduced or enhanced, respectively.

7.2.27 For the image loaded, pressing three times **<Enhance weak bands>** and subsequently 10 times **<Increase zero level>** provides a clear, sharp and contrastful picture.

7.2.28 Press **<OK>** to save these tone curve settings.

*NOTE: It is also possible to edit the tone curve manually: nodes can be added by double-clicking on the curve, or can be deleted by selecting them and pressing the DEL key. The curve can be edited in each node by left-clicking on the node and moving it. There is a **<Reset>** button to restore the original linear zero-to-100% curve.*

7.2.29 Press  to go to the next step: defining densitometric curves.

In this step, the window is divided in two panels (Figure 17): the left panel shows the strips extracted from the image file and the right panel shows the densitometric curve of the selected pattern, extracted from the image file.

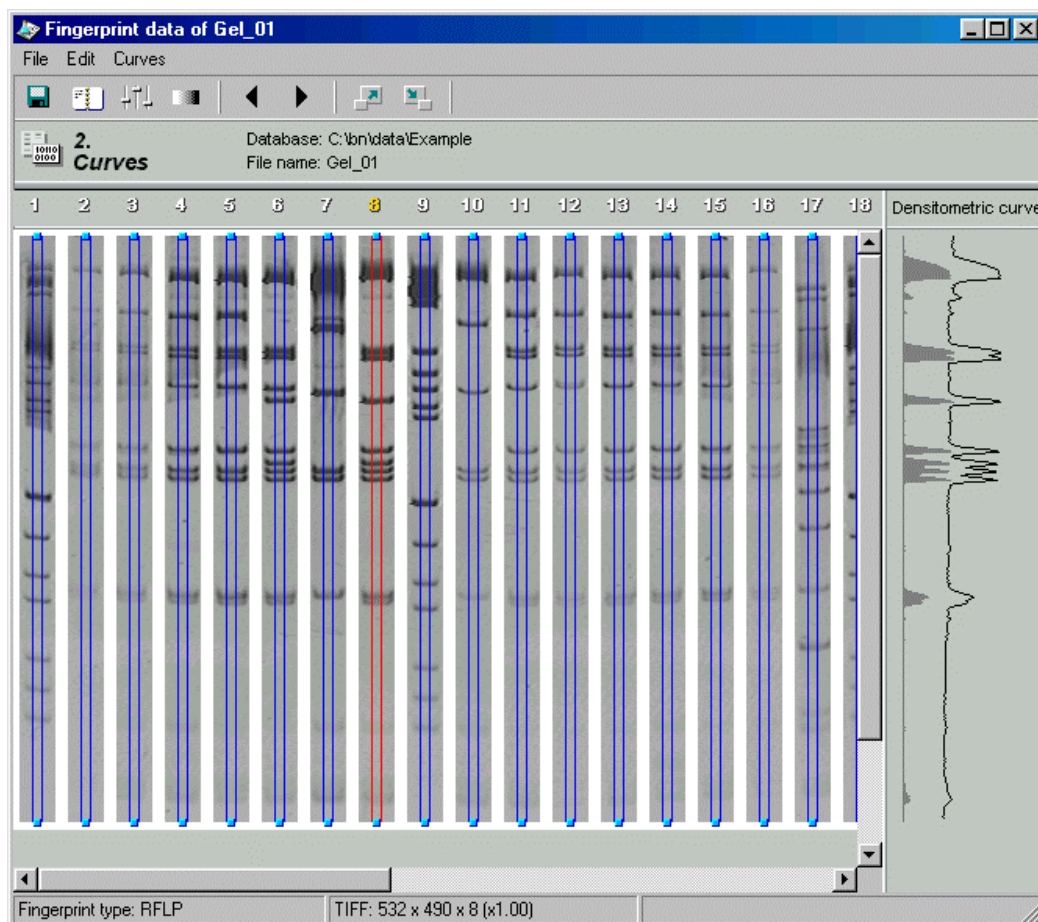


Figure 17. The *Fingerprint data editor* window. Step 2: defining densitometric curves.

7.2.30 You can move the separator between both panels to the left or to the right to allow more space for the strips or for the curves.

The program has automatically defined the densitometric curves using the information of the lane strips you entered in the previous step. Normally, you will not have to change the positions of the densitometric curves anymore, except when you want to avoid a distorted region within a pattern, e.g. due to an air bubble within the gel.

7.2.31 If necessary, adjust the position of a spline by grabbing the nodes using the mouse. Use the SHIFT key to bend the spline locally in one node.

The blue lines represent the width of the area within which the curve will be averaged. The default value is 7 points. In most cases, you will have to optimize this value for a given type of gel images.

7.2.32 Call the *Fingerprint conversion settings* dialog box with *Edit|Settings*. This time, the *Densitometric curves* tab is displayed (Figure 18).

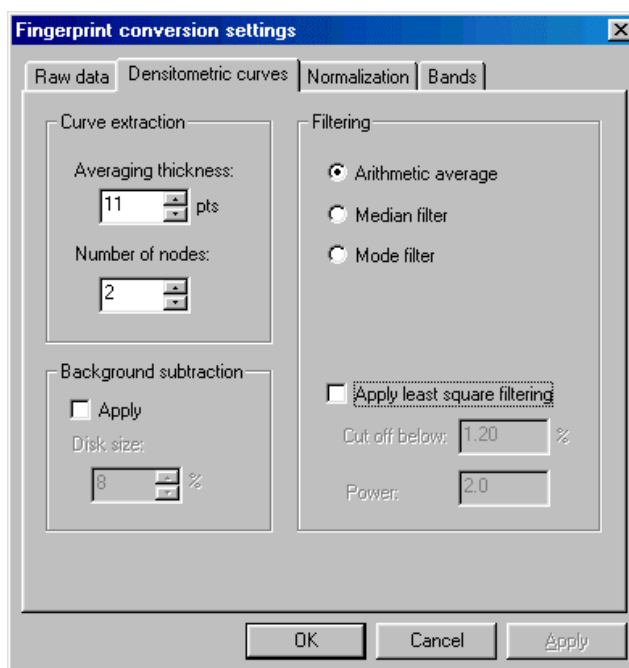


Figure 18. The *Fingerprint conversion settings* dialog box. *Densitometric curves* tab.

7.2.33 Change the *Averaging thickness* for curve extraction. For the example, enter 11. Ideally, the thickness should be chosen as broad as possible. However, smiling and distortion at the edges of the bands should be excluded (see Figure 19).

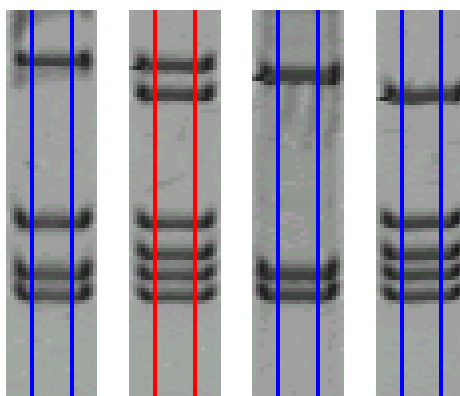


Figure 19. Optimal settings for curve averaging thickness.

7.2.34 Select *Edit | Settings* again to specify other settings.

The curve extraction settings include other important parameters which apply to the background removal and smoothing.

When we defined the fingerprint type, we left the *Background subtraction* disabled (see 7.1.5), because we will see how we can have the program propose the optimal settings.

The *Filtering* is a method to make an average of the values within the specified thickness. Simple averaging is obtained with *Arithmetic average*, whereas *Median filter* and *Mode filter* are more sophisticated methods to reduce peak-like artifacts caused by spots on the patterns. Figure 20 illustrates the effect of the Median filter on a small spot. These filters, however, reduce less noise on

the curves (particularly the Mode filter). Only in case your gels contain hampering spots, you should use the Mode filter.

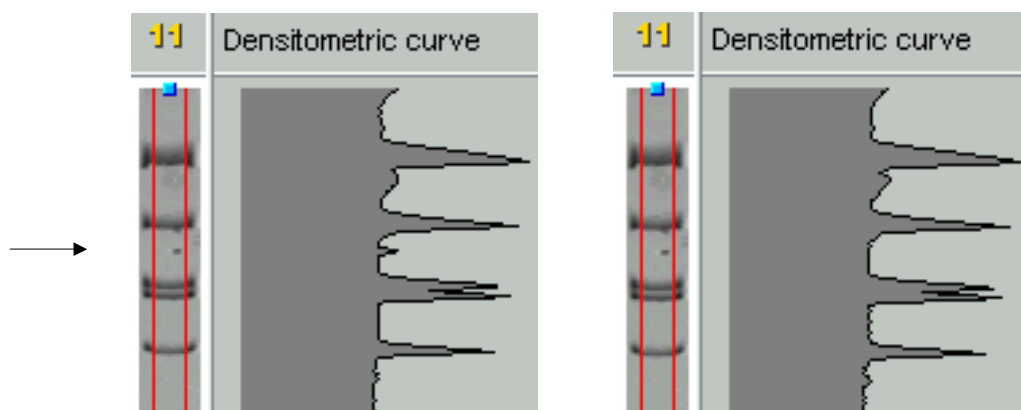


Figure 20. Result of Arithmetic average filtering (left) and Median filtering (right).

7.2.35 Select Median filter.

The *Least square filtering* applies to the smoothing of the profiles. This filter will remove background noise, seen as small irregular peaks, from the profile of real (broader) peaks. Like for background subtraction, the program can predict the optimal settings for least square filtering, if at all necessary. For now, we leave this parameter disabled.

7.2.36 Press <OK> to save the settings.

We will now determine the optimal settings for background and filtering settings using *spectral (Fourier) analysis*.

7.2.37 Select *Curves | Spectral analysis*. This shows the *Spectral analysis* window (Figure 21).

The black line is the spectral analysis of the curves in function of the frequency in number of points (logarithmic scale). Ideally, the curve should show a flat background line at the right hand side, and then slowly raise further to the left. This indicates that the scanning resolution is high enough. Another parameter which indicates the quality is the *Signal/noise ratio*, which should be above 50 if possible. The example gel is only of moderate resolution.

The *Wiener cut-off scale* determines the optimal setting for the least square filtering. Figure 21 shows an optimal setting of 0.65%.

The Background scale is an estimation of the disk size for background subtraction. The figure shows a setting of 8%.

7.2.38 Call *Edit | Settings* again and specify the background subtraction and the least square filtering.

7.2.39 If you want to have a better look at the curves (right panel) you can rescale them with *Edit | Rescale curves*. This will rescale the gray processed curves (background subtracted and filtering applied) to fit within the available window space. The raw curves (lines) may then fall beyond the window.

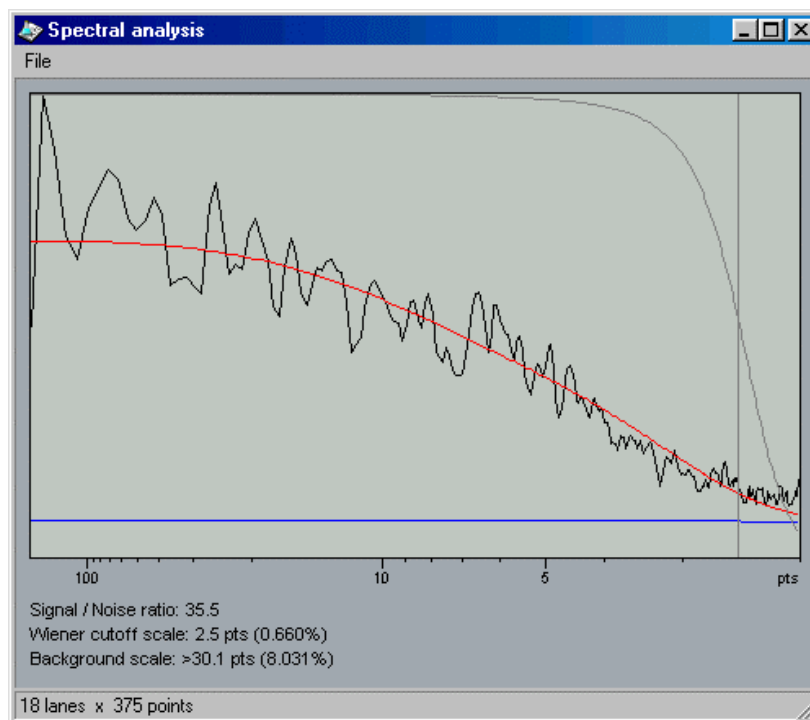


Figure 21. Spectral analysis of the patterns of a gel.

7.2.40 With the command *File | Print report* or *File | Export report*, you can generate a printed or text report of the non-normalized densitometric curves, respectively.

7.2.41 Press  to enter the next phase: normalization of the patterns.

The window consists of three panels (Figure 22): left the *Reference system* panel, which will show the *reference positions*, and the *standard pattern*; the center panel shows the pattern strips; and the right panel shows the densitometric curve of the selected pattern.

When setting up a new database, the normalization process of the first gel involves the following steps. The underlined steps are the ones that will be followed for all subsequent gels.

- Marking the reference patterns (reference patterns are identical samples loaded at different positions on the gel for normalization purposes);
- Showing the gel in normalized view;
- Identifying a suitable reference pattern on which we will define bands as *reference positions*. Reference positions are bands that will be used to align the corresponding bands on all reference patterns from the same and from other gels.
- Defining the *reference positions*;
- Assigning the bands on the reference patterns to the corresponding reference positions;
- Updating the normalization.
- Defining a standard (optional).



Figure 22. The *Fingerprint data editor* window. Step 3: normalization.

We proceed as follows:

7.2.42 Select the first reference pattern (lane 1 on the example) and **Reference** | *Use as reference lane* or . Repeat this action for all other reference lanes (lanes 9 and 18 on the example).

7.2.43 Select **Normalization** | *Show normalized view* or .

7.2.44 Choose the most suitable reference pattern to serve as standard: lane 9.

7.2.45 Select a suitable band on the destined standard pattern and **References** | *Add external reference position*.

You are prompted to enter a name for the band. You can enter any name, or if possible, the molecular weight of the band. In the latter case, the program will be able to determine the molecular weight regression from the sizes entered at this stage.

7.2.46 Use the following scheme to enter all reference positions on the example gel (Figure 23).

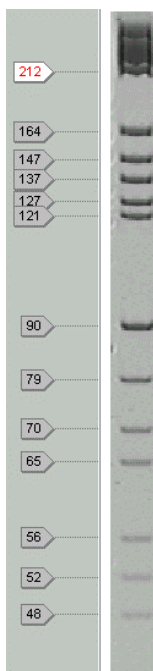


Figure 23. Band sizes of the reference positions on the example gel.

Within a fingerprint type, the set of reference positions as defined, and their names, together form a *reference system*. Once a gel is normalized using the defined reference positions and saved, the reference system is saved as well. As soon as you change anything in the reference system, a position or a name, a new reference system will automatically be created in addition to the original reference system. Once a reference system has been used in one or more gels however, the program will produce a warning if you want to change anything to the reference positions.

If more than one reference system exists, one of them is the *active reference system*, i.e. the reference system to which all new gels will be normalized. Without intervention of the user, the first created reference system will always remain the default. Later, we will see how we can set the active reference system and delete unused reference systems.

NOTE: Our current gel shows “No active reference system defined” in the left panel. This message is displayed because we are processing the first gel of this fingerprint type. We already have created the reference system, but it is not saved to disk yet. Once a second gel is normalized, this message will not be displayed anymore.

The normalization is done in two steps: first are the reference bands assigned to the corresponding reference positions, and then is the display updated according to the assignments made. The last step is optional, but is useful to facilitate the correctness of the alignments made.

Assign bands manually as follows:

- 7.2.47 Click on a label of a reference position, or wherever on the gel at the height of the reference position.
- 7.2.48 Then, hold the CTRL key and click on the reference band you want to assign to that reference position.
- 7.2.49 Repeat this action for all other reference bands you want to assign to the same reference position.

7.2.50 Repeat actions 7.2.47 to 7.2.49 until all reference bands are assigned to their corresponding reference positions.

NOTE: the cursor automatically jumps to the closest peak; to avoid this, hold down the TAB key while clicking on a band.

7.2.51 With **Normalization | Show normalized view**, or the  button, the gel will be shown in *normalized view*, i.e. the gelstrips will be stretched or shrunk so that assigned bands on the reference patterns match with their corresponding reference positions.

To show how the automated assignment works, we will undo the manual normalization:

7.2.52 Show the gel back in original view by pressing the  button again.

7.2.53 Remove all the manual assignments by **Normalization | Delete all assignments**.

There are three methods to search for the assignments automatically:

Normalization | Auto assign (bands) or : The program searches for bands on the reference patterns and tries to match them optimally with the defined reference positions. This method is always applicable, even for the very first gel, when no standard is defined.

Normalization | Auto assign using standard (bands): This method uses the defined standard pattern to match its bands with the bands of the references. It may find the correct assignments better than the first algorithm, but requires a standard to be defined (see further) AND the bands on the standard to be defined.

Normalization | Auto assign using standard (curves): This method uses the defined standard pattern to match its curve with the curves of the reference patterns. This method employs the best pattern recognition algorithm for complex banding patterns, but is less suitable for simple restriction analysis patterns. It also requires a standard to be defined.

7.2.54 Execute **Normalization | Auto assign (bands)**. Carefully inspect the assignments made, and if some are incorrect, correct them manually, as explained in 7.2.47 to 7.2.49.

*NOTE: if the program has difficulties in assigning the bands correctly, you can first make a few assignments manually (for example, the first and the last band of the reference patterns), then display the normalized view with **Normalization | Show normalized view**, or the  button and then have the program find the assignments automatically. The program then asks “**Do you want to preserve the existing alignments?**”. Answer <Yes> to this question. Successive manual and automatic assignments can be combined without any restriction.*

7.2.55 Finally, when all assignments are made correctly, select **Normalization | Show normalized view**, or .

*NOTE: In case most or all of the patterns on a gel contain one or more identical bands, such bands can be used for internal alignment of the gel. The software therefore creates an internal reference position which is saved with the gel but is not part of the reference system. An internal reference position can be created with **References | Add internal reference position**, or right-clicking on the band and **Add internal reference position**. The program then asks “Do you want to automatically search for this reference band?”. If you answer <Yes>, it will try to find all the correct assignments, but you can change or delete assignments afterwards.*

When the gel is in normalized view, a very reliable way to reveal remaining mismatches is by showing the distortion bars: these bars indicate local deviations with respect to the general shift of a reference pattern compared to the reference positions. A too strong shift is seen as a zone ranging from yellow over red to black, whereas a too weak shift is indicated by a zone ranging from bright blue over dark blue to black.

7.2.56 Show the distortion bars with *Normalization | Show distortion bars*.

Slight transitions from bright yellow to bright blue are normal, as long as the color doesn't change abruptly. In the latter case, a wrong assignment was made. You can correct the misalignment by assigning the correct band manually and *Normalization | Update normalization* or . Alternatively, you can show back the original view (7.2.52), assign the correct band manually, and show the normalized view again (7.2.55).

7.2.57 Save the normalized gel with *File | Save* (F2) or .

7.2.58 It is possible to generate a text file or a printout of the complete alignment of the gel, by selecting the command *File | Print report* or *File | Export report*, respectively.

The file lists all the reference bands defined in the reference system with their relative positions, and the corresponding bands on each reference pattern, with the absolute occurrence on the pattern in distance from the start.

If you are going to use band-matching coefficients to compare the patterns, you should read the next paragraph (7.3), corresponding to the fourth phase in the processing of a gel (see page 39). If you are going to use a curve-based coefficient, you can skip paragraph 7.3 and continue with 7.4.

7.3 Assigning bands and quantification

In step **3. Normalization**, press , which brings you in step **4. Bands**. This is the last step in processing a gel, which involves defining bands and quantifying band areas and/or volumes (see Figure 26). Bands were already found on the references since the program needed these bands for finding the assignments with the reference positions.

7.3.1 Call the *Fingerprint conversion settings* dialog box with *Edit | Settings* or . The fourth tab, *Bands* is shown, which allows you to enter the *Band search filters* and the *Quantification units* (Figure 24).

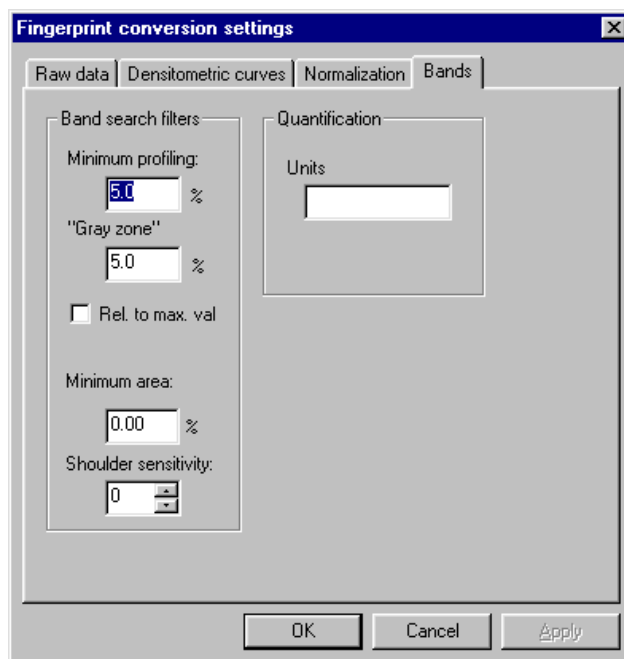


Figure 24. The *Fingerprint conversion settings* dialog box. *Bands* tab.

The *band search filters* involve a *Minimum profiling* which is the elevation of the band with respect to the surrounding background, also as percentage. The minimal profiling is dependent on the OD range you specified under **Raw data** (same dialog box, first tab). However, when *Rel. to max. val* is checked, the minimal profiling, i.e. the minimal height of the bands will be taken relative to the highest band on that pattern. When patterns of uneven intensity are analyzed, it is recommended to enable this option. Along with the minimum profiling, it is possible to specify a *"Gray zone"*, also as a height percentage. This gray zone specifies bands that will be marked *uncertain*. In comparing two patterns, the software will ignore all the positions in which one of the patterns has an uncertain band. The percentage value for the gray zone is added to the minimum profiling value. To take the example of Figure 24, all bands with a profiling of less than 5% are excluded; bands with a profiling between 5% and 10% are marked uncertain, and all bands with a profiling of more than 10% are selected (see Figure 25).

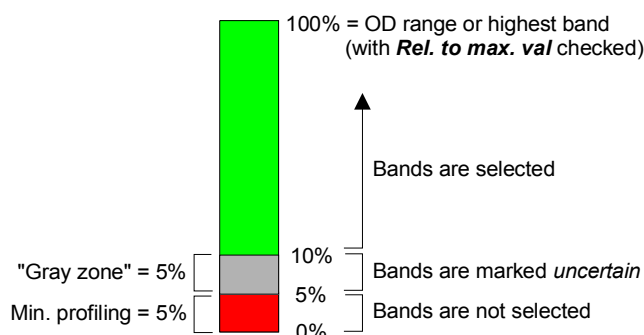


Figure 25. Understanding the meaning of the "gray zone" of uncertain bands in relation to the minimum profiling.

A *Minimum area* can also be specified, as percentage of the total area of the pattern.

A more advanced tool based on deconvolution algorithms, *Shoulder sensitivity*, allows shoulders without a local maximum as well as doublets of bands with one maximum to be found. If you want to use the shoulder sensitivity feature, we recommend to start with a sensitivity of 5, but optimal parameters may depend on the type of gels analyzed.

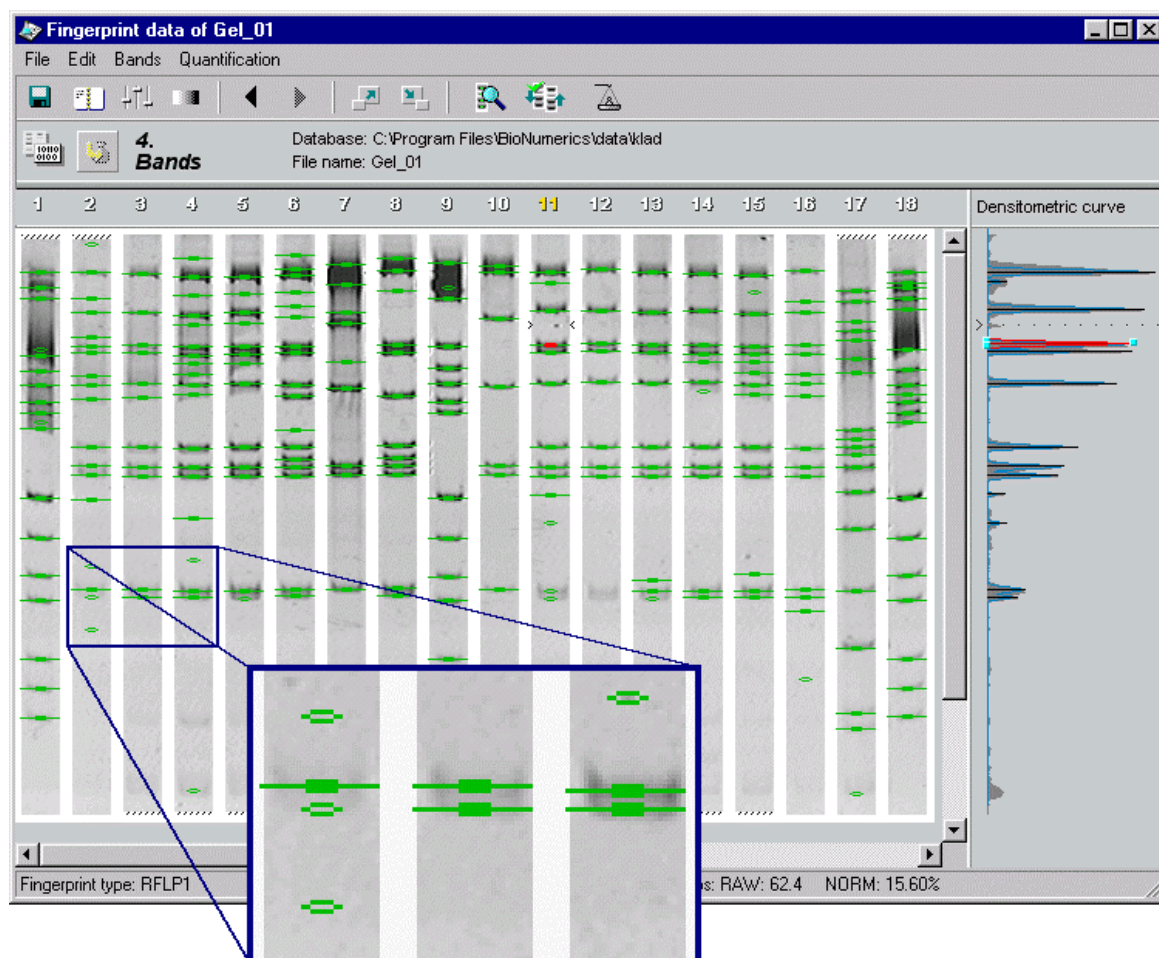


Figure 26. The *Fingerprint data editor* window. Step 4. Bands.

7.3.2 Change *Minimal profiling* to adjust the minimal peak height (in % of the heighest peak of the pattern), and/or *Minimal area* to adjust the minimal area, in % of the total area of the pattern. Usually, setting 5% minimal profiling will be convenient, whereas the minimal area can be left zero in most cases. The present example however, requires a higher minimal profiling (e.g. 10%). Optionally, you can enter a percentage for uncertain bands (gray zone). As an example to see what happens, enter 5%. Click *Relative to max. value of lane*. Specify a *Shoulder sensitivity* only if you want to allow the program to find band doublets and bands on shoulders (sensitivity of 5 should be fine for most gels).

7.3.3 Press <OK> to accept the settings.

7.3.4 Select **Bands | Auto search bands** or  to find bands on all the patterns.

Before actually defining the bands on the patterns, the software displays a preview window (Figure 27). This preview shows the first pattern on the gel with its curve and gelstrip. Press the <Preview> button to see what bands the program finds using the current settings. A pink mask shows the

threshold level based upon both the minimal profiling and the minimal area (if set). Only bands that exceed the threshold will be selected. If inappropriate, the settings can be changed in this preview window. The sensitivity of this search depends on the *band search settings*: If too many (false) peaks are found, or if real bands are undetected, you can change the search sensitivity using the band search filters as described above.

In addition, a blue mask shows the threshold level for bands that will be found as uncertain (gray zone). All bands exceeding the pink mask but not exceeding the blue mask will become uncertain bands.

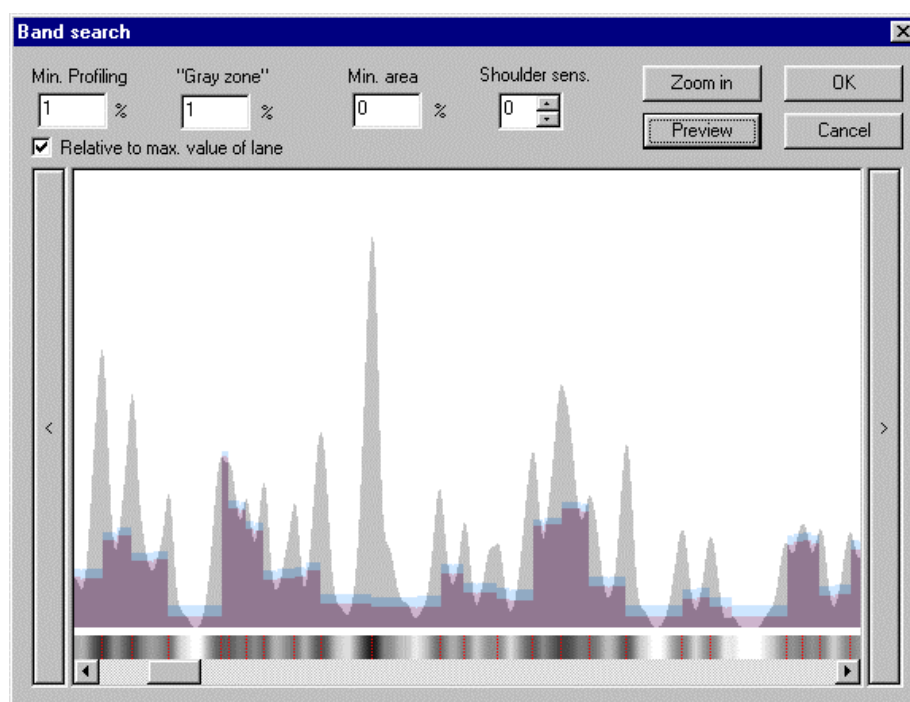


Figure 27. Band search preview window.

In the band search preview window, the currently selected pattern is shown. To scroll through other patterns in the preview, press the < or > button (left and right from the curve).

7.3.5 Press <OK> to start the search.

Bands that were found are marked with a green horizontal line, whereas uncertain bands are marked with a small green ellipse (see magnification in Figure 26).

7.3.6 Add a band with **Bands | Add new band**, the ENTER key, or CTRL + left-click.

NOTES:

(1) The cursor automatically jumps to the closest peak; to avoid this, hold down the TAB key while clicking on a band.

(2) When there is evidence of a double band at a certain position, you can add a band over an existing one (7.3.6). Double bands (or multiplets) are indicated by outwards pointing arrows on the band marker: . Double uncertain bands are marked with a filled ellipse instead of an open ellipse. The clustering and identification functions using band based similarity coefficients (11.3) support the existence of double overlapping bands. For example, two patterns, having a single band and a double band, respectively, at the same position will be treated as having one

matching and one unmatched band. Two patterns, each having a double band th the same position will be treated as having two matching bands.

7.3.7 Hold the SHIFT key and drag the mouse pointer holding the left mouse button to select a group of bands.

7.3.8 Press the DEL key or **Bands | Delete selected band(s)** to delete all selected bands.

7.3.9 Select a band and **Bands | Mark band(s) as uncertain** (or press F5).

7.3.10 With **Bands | Mark band(s) as certain** (or press F6), the band is marked again as certain.

The right panel of the window shows the densitometric curve of the selected pattern. For each found band, the program automatically calculates a best-fitting *Gaussian* curve, which makes more reliable quantification possible.

7.3.11 Select a band on a pattern.

7.3.12 Show rescaled curves with **Edit | Rescale curves**.

7.3.13 Zoom in on the band by pressing  repeatedly. Figure 28 shows a strongly zoomed band with its densitometric representation and the Gaussian fit (red). The blue points are dragging nodes where you can change the position and the shape of the Gaussian fit for each band separately.

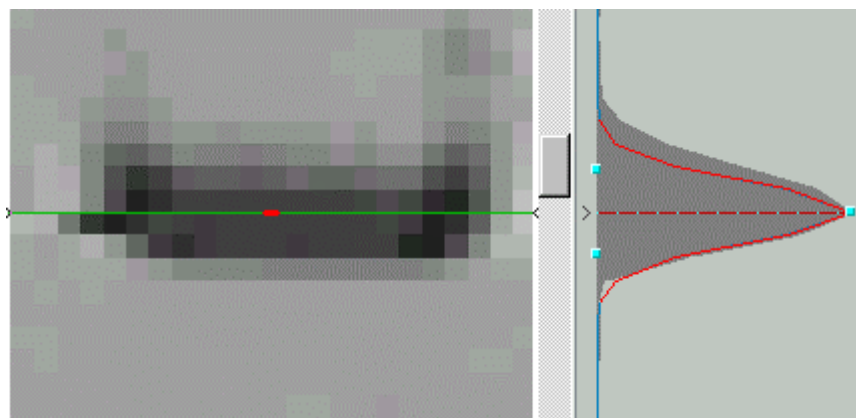


Figure 28. Zoomed band with its densitometric curve and best-fitting Gaussian approach.

7.3.14 Save the gel with **File | Save** (F2) or .

7.3.15 It is possible to generate a text file or a printout of the complete band information of the gel, by selecting the command **File | Print report** or **File | Export report**, respectively.

The file lists all the bands defined for each pattern with their normalized relative positions, the metrics (e.g. molecular weight), the height, and relative one-dimensional surface, as calculated by Gaussian fit.

Once bands are defined, two-dimensional quantification is done as follows.

7.3.16 Bring the window in *Quantification mode* with  or *Quantification | Band quantification*. The quantification button now shows as  and two more quantification buttons are shown.

7.3.17 To find the surfaces (contours) of the bands, use *Quantification | Search all surfaces* or .

If you have added a band later, you can search the surface of that band alone with *Quantification | Search surface of band*.

When the contours are found, the program shows for each selected band its *volume* in the status bar: the sum of the densitometric values within the contour.

7.3.18 To change the contour of a band manually, first select the band and zoom in heavily (7.3.11 and 7.3.13).

7.3.19 Hold the CTRL key and drag the mouse (holding left button) to correct the upper and lower contours.

7.3.20 For known reference bands, you can enter a concentration value by selecting the band and *Quantification | Assign value*. (or floating menu by right-clicking, or double click). Known reference bands are marked with .

7.3.21 Once multiple reference bands are assigned their concentrations, a regression to determine each unknown band concentration is calculated by selecting *Quantification | Calculate concentrations*.

The band quantification window (Figure 29) shows the real concentration in function of the band volumes, using cubic spline regression functions.

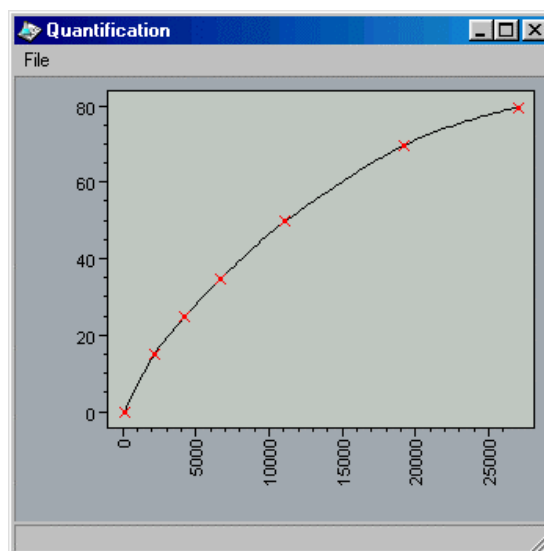


Figure 29. Band quantification window: concentration in function of known band volumes.

7.3.22 Save the gel with *File | Save* (F2) or  in order to store the quantification data.

- 7.3.23 It is possible to generate a text file or a printout of the complete two-dimensional band information of the gel, by selecting the command *File|Print report* or *File|Export report*, respectively.

The file lists all the bands defined for each pattern with their normalized relative positions, the absolute volume, and if regression is done, the relative volume as determined by the calibration bands.

We are now at a point that we can discuss the functioning of the *reference system*. We will explain how to calculate molecular weights for the fingerprint type and how to link a *standard pattern* to the fingerprint type.

- 7.3.24 Exit the fingerprint data window: *File|Exit*.

The program asks “*Settings have been changed. Do you want to use the current settings as new defaults?*”. This question is asked when changes have been made to the fingerprint type-related settings, for example the gelstrip thickness, the rolling disk size, etc. If you answer *<Yes>*, the settings used for this gel will be saved in the fingerprint type’s settings, and all new gels will be processed using the same settings.

- 7.3.25 Answer *<Yes>* to save the changes made into the fingerprint type settings.

*NOTE: Answering <Yes> to the above question has the same effect as the menu function **Edit|Save as default settings** in the fingerprint data window. Conversely, the current default settings can be copied to the current gel with **Edit|Load default settings**.*

To show that the reference system is now defined for our gel type RFLP, we will open the *Fingerprint type* window.

- 7.3.26 In the main window, select RFLP under **Fingerprint types** in the experiment types panel (see Figure 10). Double-click on RFLP, or select *Experiments|Edit experiment type* in the main menu. This opens the experiment type window (Figure 30).

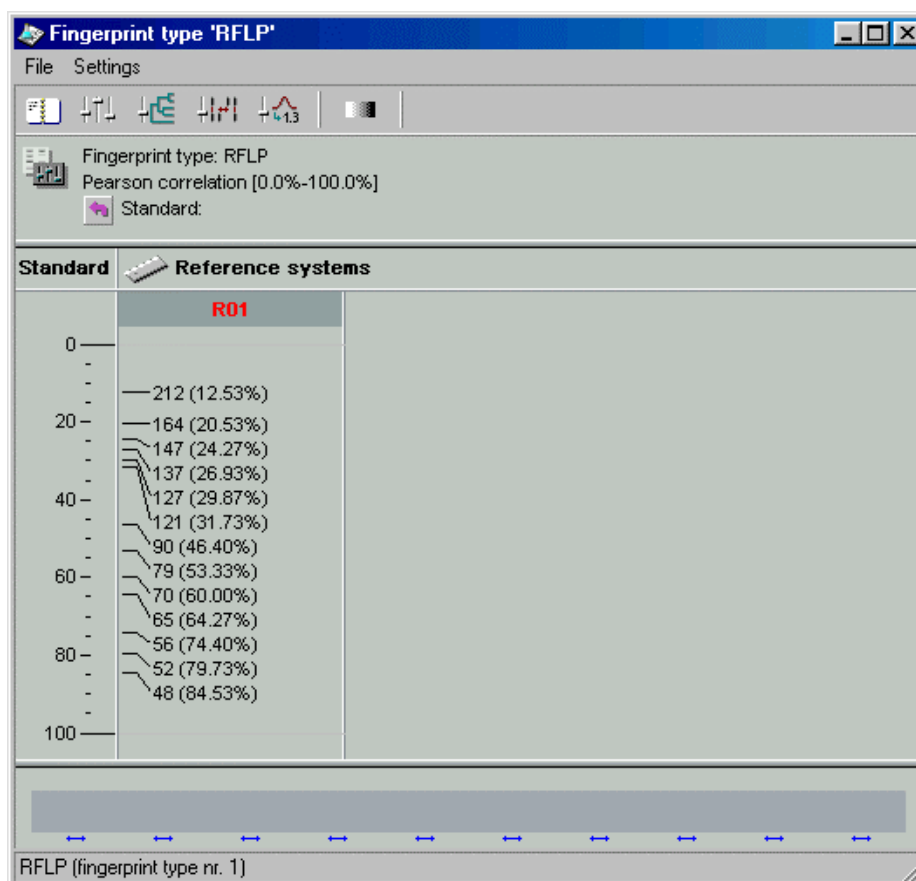


Figure 30. The Experiment type window; Standard is not yet defined.

7.3.27 The experiment type window allows you to change all settings which we have defined when creating the fingerprint type, and when processing the first gel with *Settings | General settings* or .

One setting which we have not discussed during the normalization of the example gel is the *Normalization* tab. This tab shows the *Resolution of normalized tracks* as only setting. In reality, the program always stores the real length of the raw patterns. For display purposes however, the program converts the tracks to the same length at real-time, so that the gel strips are properly aligned to each other. For comparison of patterns by means of the *Pearson* product-moment correlation also, the densitometric curves need to be of the same length. Thus, the resolution value only influences two features: the length of the patterns shown on the screen, and the length (resolution, number of points) of the densitometric curves to be compared by the Pearson product-moment correlation coefficient. By default, the program uses 600 as resolution, but when you normalize the first gel, the program automatically uses the *average track length* for that gel as the new resolution value. Whenever you save the gel, and the value differs more than 50% from the default value, Fingerprinting II Informatix will ask you to copy the resolution of the current gel to the default for the fingerprint type (see 7.2.11). Another option is *Bypass normalization*. You can use this option to have the program process the densitometric curves of the tracks *without any change*. This option is only useful to import patterns in Fingerprinting II Informatix that are already normalized, and for which you want the values of the densitometric curves to remain exactly the same after the normalization process.

The default brightness and contrast setting can be changed with *Layout | Brightness & contrast* or , and the quantification settings with *Settings | Comparative quantification* or .

settings include the comparison settings, and the position tolerance settings, which will be discussed later.

The experiment type window shows the defined reference positions in relation to the distance on the pattern (in percentage), and calls this reference system R01. Other reference systems (if created) will be called R02, R03 etc. Currently, R01 is shown in red because it is the *active reference system*.

In this window, the panel for the **Standard** is still blank: the fingerprint type still misses a standard pattern. The standard pattern actually has no essential contribution to the normalization; it is only intended to show a normalized reference pattern next to the reference positions, in order to make visual assignment of bands to the reference positions easier. Another feature for which the standard is required is the automated normalization by pattern recognition. This algorithm requires a curve of a normalized reference pattern to be present in order to be able to align other reference patterns to it.

Now, link a standard to the fingerprint type as follows:

7.3.28 Close the experiment type window for now (**File|Exit**).

7.3.29 Select the gel file in the experiment files panel (Figure 10) and choose **File|Open experiment file (entries)** from the main menu.

This opens the *Fingerprint entry file* window, listing the lanes defined for the example gel (Figure 31).

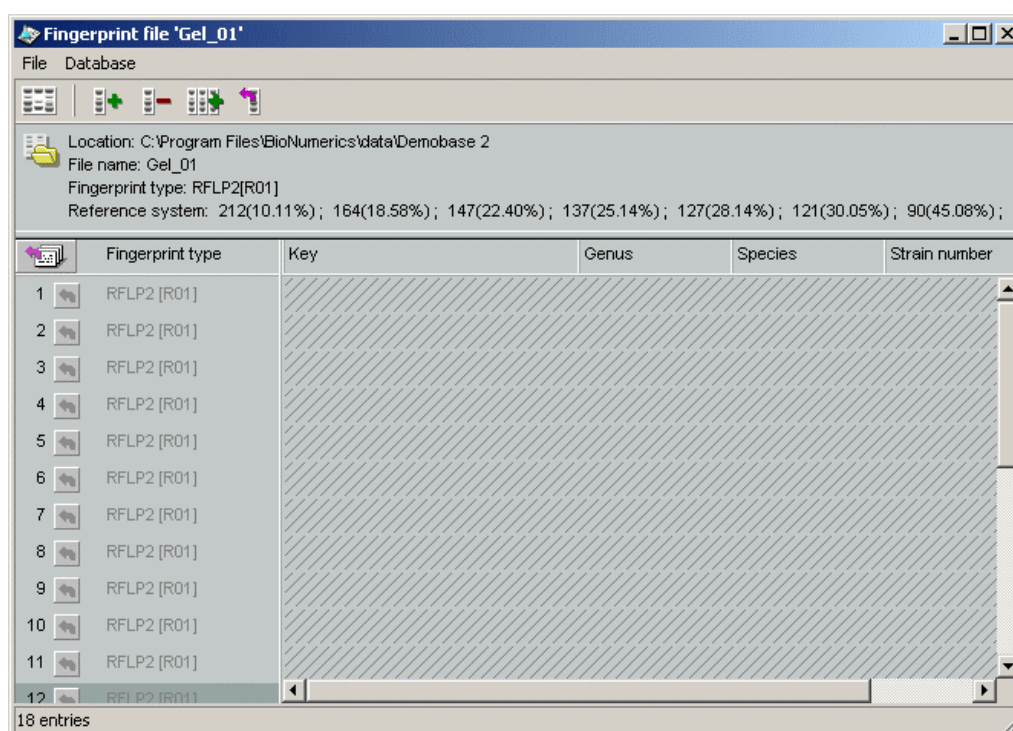


Figure 31. The Fingerprint entry file.

These lanes are not linked to database entries yet. A *link arrow*  for each lane allows you to link a lane to a database entry, by clicking on the arrow and dragging it onto a database entry, and then releasing the mouse button. When the experiment is linked, its link arrow is purple: . The window also shows the *fingerprint type* of the gel, the *reference system* according to which the gel is normalized, and the *reference positions* of this reference system.

- 7.3.30 In the main window, add a new database entry with **Database | Add new entries** (see 6.2.4 to 6.2.5).
- 7.3.31 Edit the new entry's information fields (see 6.4.1 to 6.4.2) and enter STANDARD as genus name.
- 7.3.32 Drag the link arrow of **lane 9** to the new database entry 'STANDARD': pattern 9 is now linked to this database entry.
- 7.3.33 Close the fingerprint entry file window with **File | Exit**.
- 7.3.34 In the main window, open the fingerprint types window again for RFLP (7.3.26).
- 7.3.35 Link a reference lane (for example lane 9) to the fingerprint type by dragging the  button to the database entry STANDARD.

The standard pattern is now displayed in the *standard* panel next to the reference positions, and the database entry key of the standard is indicated next to the link arrow (Figure 32). From this point on, all further gels that are normalized will display the standard pattern left from the gel panel in the normalization step. This makes manual association of peaks easier and allows automated alignment using curve matching.

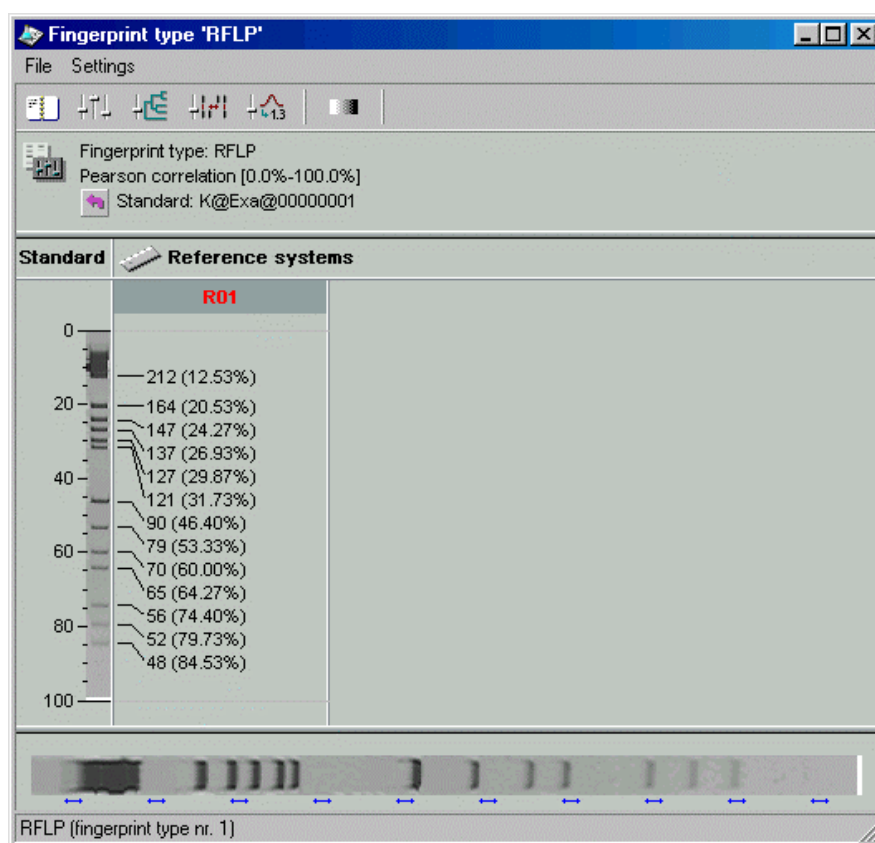


Figure 32. The Experiment type window; Standard is defined.

NOTE: The choice of a standard has no influence on the normalization process, since it is only used as a visual aid. One can change the standard pattern at any time later on, e.g. if another reference pattern appears to be more suitable for this purpose.

The *molecular sizes* of the bands are not calculated within a particular gel file, but for a whole reference system. This means that, once you have created a reference system and normalized one gel, you can define the molecular size regression for all further gels that will be normalized using the same reference system.

- 7.3.36 In the experiment type window for RFLP, call **Settings | Edit reference system** (or double-click in the **R01** panel). This pops up the *Reference system* window for fingerprint type RFLP (Figure 33).

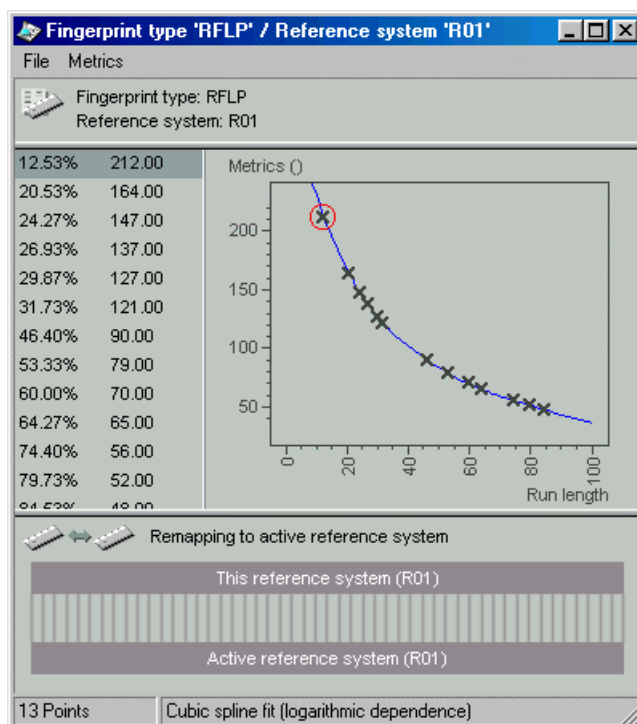


Figure 33. The reference system window, showing molecular weight regression and remapping function to the active reference system (if different).

Initially, the regression cannot be calculated, since the program doesn't know where to take the marker points from. The message "Could not calculate calibration curve. Not enough markers" is displayed.

- 7.3.37 You can add the markers manually (**Metrics | Add marker**), but if you have entered the molecular weights as names for the reference positions (see 7.2.45 and 7.2.46), the obvious solution is to copy these molecular weights: **Metrics | Copy markers from reference system**.

The result is a regression curve, shown in Figure 33. As regression function, you can choose between a first degree, third degree, cubic spline, and pole fit, and each of these functions can be combined with a logarithmic dependence.

- 7.3.38 For this example, choose **Metrics | Cubic spline fit** with **Logarithmic Dependence**.

- 7.3.39 Choose a unit with **Metric | Assign unit**, and enter **bp** (base pairs).

- 7.3.40 Close the reference system window, and close the fingerprint type window.

7.4 Adding the gel lanes to the database

In Paragraph 6.2, we have seen how entries are added to the database. Once these entries are defined in the database, it is easy to link the experiments, which are gel lanes in this case, to the corresponding entries. We have done so with the STANDARD lane, explained in the previous paragraph. In summary, adding lanes to the database and linking experiments to them works as follows:

7.4.1 Select **Database | Add new entries** or  in the toolbar.

7.4.2 Enter the number of entries you want to create, e.g. 1, and press <OK>.

The database now lists one more entry with a unique key automatically assigned by the software.

7.4.3 Select the gel file in the experiment files panel (Figure 10) and choose **File | Open experiment file (entries)** from the main menu.

This opens the *Fingerprint entry file* window, listing the lanes defined for the example gel (Figure 31).

These lanes are not linked to database entries yet. A *link arrow*  for each lane allows you to link a lane to a database entry, by clicking on the arrow and dragging it onto a database entry, and then releasing the mouse button. When the experiment is linked, its link arrow is purple: .

7.4.4 Drag the link arrow of **lane 2** (lane one is a reference) to the new database entry: as soon as you pass over a database entry, the cursor shape changes into .

7.4.5 Release the mouse button on the above created database entry; pattern 2 is now linked to this database entry, and its arrow in the fingerprint entry window has become purple  instead of gray .

The corresponding menu command is **Database | Link lane** (). The program asks you to enter the *key* of the database entry to which the experiment is to be linked.

NOTES:

(1) If you try to link a lane to an entry which already has a lane of the same experiment type linked to it, the program will ask whether you want to create a **duplicate key** for this entry. This feature is very useful in case you want to define experiments that are run in duplicate for one or more organisms. Rather than overwriting the first entry or disregarding duplicate entries, Fingerprinting II Informatix automatically considers them as duplicates and assigns an extension /#x to such duplicates. In case for a given entry a duplicate already exists (after import of another experiment), Fingerprinting II Informatix will automatically fill such existing duplicates that are still empty for the experiment type that is being imported. Database fields are automatically taken over from the "master" entry, i.e. the entry without extension. If the database fields from the "master" entry are changed, the /#x duplicates are automatically changed accordingly.

(2) If you enter an entry key which does not already exist, the program asks whether you want to create an entry with that key.

As soon as an experiment is linked to a database entry, the experiment presence panel (see Figure 10) shows a green dot for the experiment of this entry.

7.4.6 You can click on a green dot, which pops up the *Experiment card* for that experiment (see 8.2).

7.4.7 You can edit the information fields for this entry in two places: directly in the database (see 6.4.1 to 6.4.2), or in the fingerprint entry file window, by double clicking on the entry.

If no database entries are defined for the current gel lanes, you can have the program create new entries and link the gel lanes automatically in a very simple way:

7.4.8 In the fingerprint entry file window, select **Database|Add all lanes to database** (). All lanes that were not linked yet, will be added as new entries to the database, with the gel lanes linked.

*NOTE: In some cases, a gel can be composed of patterns belonging to different fingerprint types. For example, if you are running digests by three different restriction enzymes for the same set of organisms, for some remaining entries, you may want to run all three RE digests on the same gel. In this case, you should process the gel according to one of the fingerprint types, and then, in the Fingerprint entry file window, select a lane that belongs to another fingerprint type and **Database|Change fingerprint type of lane**. A condition for this feature to work is that both fingerprint types are based upon the same reference system (the same set of reference markers, defined consistently using the same names). If the reference system for both fingerprint types is not the same, the software can still use the molecular weight calibration curves as a basis for conversion, if these are defined.*

If you do not wish to add all lanes to the database, you can select individual lanes, and use the menu command **Database|Add lane to database** ().

You can unlink a gel lane from the database using **Database|Remove link** (). All entries from the gel are unlinked at once using **Database|Remove all links**.

7.5 Superimposed normalization based on internal reference patterns

This paragraph describes how to normalize patterns based upon “inline” reference patterns, i.e. reference patterns that are loaded in each lane, but that are revealed using a different color dye or hybridization probe. Examples within this category are (1) the four channel ABI 310/377 sequencer chromatograms and (2) RFLP gels that contain internal reference patterns which are visualized using a different color dye or hybridization probe.

In case (1), a special import program, **ABICONV** is required to convert the ABI sample chromatogram files into the Fingerprinting II Informatix curve format. **ABICONV** splits the multichannel sample files into separate gel files for each available channel (color). The gel files are automatically saved in the **IMAGES** directory of the selected database. Logically, the separate gels all contain the same lanes at the same position. One of the gels contains the internal reference patterns, whereas the other gel (or gels) contain the real data samples, to be normalized according to the reference patterns. The aim is to normalize the obtained reference gel, and to superimpose the normalization on the other gel(s). The only difference with TIFF files is that there are no two-dimensional gelstrips available for the ABI sequencer patterns. Fingerprinting II Informatix creates reconstructed gelstrips instead. A non-reference gel (i.e. a real data gel) is normalized by first normalizing the reference gel (i.e. the gel containing the internal reference patterns), and then copying the normalization of the reference gel to the data gel. This can be done easily by simply linking the data gel(s) to the corresponding reference gel: each data gel is automatically updated when anything in the conversion and normalization of the reference gel is changed.

In case (2), the conversion from the ABI sample files is not needed, but on the other hand, an initial alignment between the images of the reference gel and the data gel is needed, since both images are not usually scanned in exactly the same position. The further steps are the same as for the ABI gels: A

non-reference gel (i.e. a real data gel) is normalized by first normalizing the reference gel (i.e. the gel containing the internal reference patterns), and then copying the normalization of the reference gel to the data gel, or linking the data gel to the reference gel.

A. Multichannel ABI 310/377 gels

- 7.5.1 In the Fingerprinting II Informatix startup program, press the **<Filters>** button and select ABICONV.
- 7.5.2 Drag and drop ABI sample chromatogram files from the Windows Explorer to the ABICONV window. A set of example files together composing one gel can be found on the CD-ROM, in directory **EXAMPLES\ABI AFLP**. If you are using this example, select all files.
- 7.5.3 If necessary, you can change the order of the lanes with **Lane|Move up** and **Lane|Move down**, or using the + and - keys.
- 7.5.4 You can remove lanes if necessary with **Lane|Remove**.
- 7.5.5 If required, the curves can be flipped with **Lane|Flip curves**.
- 7.5.6 Select **File|Save file as** to save the curve gel files in the default directory for the active database. Use for example the name **ABIGEL01**.

The program adds a number 01, 02, 03 or 04 to each gel, depending on the color, and adds the extension .CRV to each gel.

- 7.5.7 Run the Analyze program and create a new Fingerprint type (7.1), e.g. **ABI**, specifying **Densitometric curves** when the wizard asks “*What kind of fingerprint type do you have?*”.
- 7.5.8 Specify **12-bit** OD range (4096 gray levels).

When finished, two new files are listed in the **Files** panel: **ABIGEL01_02** and **ABIGEL01_04**.

- 7.5.9 Process gel **ABIGEL01_04** with **File|Open experiment file (data)**, and assign it to fingerprint type ABI.

We select this gel first because it is the one containing the reference patterns. The lanes are shown as reconstructed pattern images. It may be necessary to adjust the brightness and contrast (**Edit|Change brightness & contrast**), by enabling the **Dynamical preview** and slowly moving down the **Maximum value** until the darkest bands are (nearly) black.

- 7.5.10 Define the lanes with **Lanes|Auto search lanes**.

You will notice that some setting options that apply to TIFF files are not available here: for example the **Gelstrip thickness**, the **Number of nodes**.

- 7.5.11 Move on to the next step with . This shows the densitometric curves.

Here again, **Average thickness** and **Number of nodes** do not apply. You may want to adjust the background subtraction and the filtering as described in 7.2.

- 7.5.12 Move on to the **Normalization** step with .

- 7.5.13 Locate a suitable standard, place the gel in *Normalized view* , and define the reference bands. The example uses the reference mix from ABI, containing 13 bands with known molecular weight.
- 7.5.14 Select *References | Use all lanes as reference lanes* to mark all 16 lanes as *Reference lane*, and align the bands with *Normalization | Auto assign (bands)*.
- 7.5.15 Update the normalization with  and save the normalized reference gel.
- 7.5.16 Do not close the fingerprint data window for gel ABIGEL01_04, but reduce it or move it away from the center of the screen so that the **Experiments** and **Files** panels of the main analyze screen become available.
- 7.5.17 Select the data gel **ABIGEL01_02** with *File | Open experiment file (data)*, and assign it to fingerprint type ABI.
- 7.5.18 Link this gel to the reference gel by dragging the grayed  button of the data gel and dropping it on the window of the reference gel.

You can also use the menu command *File | Link to reference gel*, and enter the name of the reference gel.

The program now asks “*Link this gel to the reference gel ABIGEL01_04?*”.

- 7.5.19 If you answer <Yes>, the linkage button becomes ungrayed .

- 7.5.20 Once the data gel is linked, you can close the reference gel window.

The tracking info, curve settings, and alignment of the reference gel are now automatically superimposed to the data gel. You can run through the different steps till you reach the normalization step: the alignments as obtained in the reference gel are shown. If you wish, you can show the normalized view before you move to the last step, i.e. defining bands.

- 7.5.21 Whenever needed, you can pop up a reference gel to which a data gel is linked by clicking the  button (*File | Open reference gel*).
- 7.5.22 If you have made changes to the reference gel without saving them, you can update the changes to the data gel by right clicking on the  button (*File | Update linked information*). Once you save the changes to the reference gel, the data gel(s) are updated automatically.

B. RFLP gelscans containing internal markers

The way of processing the gels is similar as described for the ABI multichannel files, except that we start from two independent TIFF files here. An absolute condition is that the two TIFF files, containing the references and the data lanes respectively, have exactly the same resolution (dpi). If the images are shifted or rotated, they can be aligned to each other by applying two or more marker points to the gel. These marker points will be visible on the TIFF files, and the software allows such markers to be used to align the images.

- 7.5.23 Read the TIFF image of the reference gel.

If the reference gel and the data gel need to be aligned to each other, you should define marker points as follows:

7.5.24 In the first step (**1. Strips**), select **Lanes|Add marker point** and click on the first marker point of the gel.

7.5.25 Repeat the same action for the other marker points.

At least two marker points should be present before the program can copy the geometry from one gel to another.

If the TIFF images are already aligned (for example, when different fluorescent markers are used in the same gel, which are visualized at the same time), you should not add marker points.

7.5.26 Proceed with the full normalization of the reference gel as described in 7.2. Save the file but do not close the window.

7.5.27 Open the data gel and assign it to the same fingerprint type.

7.5.28 Link this gel to the reference gel by dragging the grayed  button of the data gel and dropping it on the window of the reference gel.

The program now asks “*Link this gel to the reference gel GELNAME?*”.

7.5.29 If you answer <Yes>, the linkage button becomes ungrayed .

7.5.30 Once the data gel is linked, you can close the reference gel window.

The tracking info, curve settings, and alignment of the reference gel are now automatically superimposed to the data gel. In the second step (**2. Curves**), it is still possible to adjust the position of the track splines individually, or to add nodes and distort the curves where necessary. You can run through the different steps till you reach the normalization step: the alignments as obtained in the reference gel are shown. If you wish, you can show the normalized view before you move to the last step, i.e. defining bands.

7.5.31 Whenever needed, you can pop up a reference gel to which a data gel is linked by clicking the  button (**File|Open reference gel**).

7.5.32 If you have made changes to the reference gel without saving them, you can update the changes to the data gel by right clicking on the  button (**File|Update linked information**). Once you save the changes to the reference gel, the data gel(s) are updated automatically.

*NOTE: It is also possible to copy the geometry and normalization from one gel to another without linking them. In the reference gel, go back to the first step (**1. Strips**) with  and select **Lanes|Copy geometry**. In the data gel, use **Lanes|Paste geometry** to copy the gelstrip definition from the reference gel. The normalization from the reference gel is copied with **References|Copy normalization** and **References|Paste normalization** in the normalization step. This approach may offer additional flexibility in special cases, but is not recommended.*

7.6 Import of molecular size tables as fingerprint type

Fingerprinting II Informatix allows the input of band size and band position tables, and reconstruct fingerprints of these, based upon the size and the amplitude (area or height) of the peaks.

7.6.1 In the **Example** database, create a new fingerprint type **ABI-Genescan**. Leave every setting as default except in the second step, where you should specify *Densitometric curves* and *16-bit (65536 values)*.

We will now create a new reference system to allow the import of an ABI Genescan table, part of which is shown in Figure 34 below. The whole file (5 patterns) can be found in the **Examples\TXTfiles** directory on the installation CD-ROM, as **Genescan.txt**. There are two possible approaches to create a new reference system:

- Enter positions on the gel (running distances) and the corresponding band sizes. Based upon the positions and the corresponding sizes, the program is able to establish a regression curve, upon which all imported bands can be mapped. This option is particularly suitable when you know the exact positions of the size markers in a gel system, and you want to reproduce the real regression exactly.
- Allow the program to create its own regression curve between a defined maximum and minimum molecular weight, so that it can map the imported bands on this synthetic regression curve. This method is useful if you want to import band tables of which you know nothing else than the sizes.

We will focus on the example Genescan file Genescan.txt to apply both methods. The file format contains a column with the sample number, a comma and then the band number (Figure 34), next is a column with the running time, next is the size in basepairs, then the height, the volume, and the running time again.

Option 1: Composing a regression curve by entering positions and sizes.

The running distance needed by Fingerprinting II Informatix is reciprocal to the running time given in the Genescan file (second column). Therefore, we will calculate the reciprocal value of the running time, keeping in mind that this value should never exceed 100%. Thus in order to calculate a running distance of a band (RD), we look for the *lowest* running time ($RT_{\min}$) in the file (highest running distance), divide this number by the actual running time (RT) of that band and multiply by 100 to have it in percent:

$$\%RD = RT_{\min}/RT \times 100$$

In the example, $RT_{\min} = 30$. For the reference lane 17B, this yields the extra column under the running time column (Figure 34).

Sample & band no.	Running time	Size in bp	Height	Volume	
17B, 1	33.00	60.47	228	929	330
17B, 2	34.60	67.53	201	815	346
17B, 3	43.30	106.02	113	855	433
17B, 4	52.90	146.14	381	1908	529
17B, 5	88.20	298.95	131	690	882
17B, 6	89.00	302.68	1425	7821	890
17B, 7	155.40	709.46	304	1800	1554
17B, 8	158.50	736.00	182	966	1585
17B, 9	165.10	796.02	121	713	1651



90.9
86.7
69.3
56.7
34.0
33.7
19.3
18.9
18.2

Figure 34. Lane in an ABI Genescan table and conversion of running distances to Fingerprinting II Informatix positions.

Based upon this running distance in percent and the band sizes, we can create a realistic regression curve according to the first approach described above.

7.6.2 In the experiment window, select *Settings | New reference system (positions)* .

The input box shown in Figure 35 allows all known reference bands to be entered.

7.6.3 Press the **<Add>** button and enter all running distances and sizes of lane 17B, as shown in Figure 35.

7.6.4 Enter a name for the reference system, e.g. **ABI**.

7.6.5 When finished, press **<OK>**.

Warning: once a new reference system is defined, it is not possible to change it anymore! If you want to change a self-made reference system once it is saved, you will have to delete it and create it again.

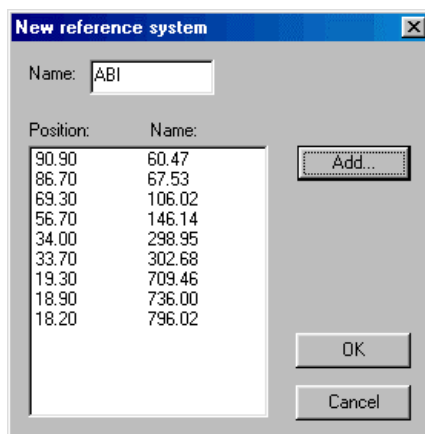


Figure 35. Defining a new reference system based upon known band positions and sizes.

7.6.6 Make the new reference system the *active reference system* by selecting it and **Settings | Set as active reference system** (not necessary if the reference system is the only one available).

7.6.7 Select **Settings | Edit reference system** or double-click to define the molecular weight regression.

7.6.8 In the reference system window, copy the entered molecular weights with **Metrics | Copy markers from reference system**.

Fingerprinting II Informatix is now configured to import the Genescan tables, except that an import macro is needed to read the file type. An import macro for the Genescan table files is given in 24.6, example 6.

7.6.9 Exit the reference system window and the fingerprint type window.

7.6.10 In the main window, select **ABI-Genescan** under **Fingerprint types** in the **Experiment panel** and **File | Import experiment data**.

7.6.11 As Data type, select **Tables of peaks** and press **<OK>**.

7.6.12 On the CD-ROM, select file **Examples\TXTfiles\Genescan.txt** in the file dialog box.

7.6.13 In the "**Import macro**" list box, select **<Create new data import macro>** in order to create a new macro.

7.6.14 Press **<Open>**.

The macro editor window is created. See paragraph 24.6 for a description of the macro editor. You can enter the macro given in 24.6, example 6. Run the macro on the table file as described in 24.6.

7.6.15 When the macro is executed successfully, select **Entries | Import** or  to add the imported entries to the database.

A *Peak import table* dialog box prompts for the file name, the resolution, the peak width, and the maximum amplitude (height) of the peaks. Usually, the latter two parameters will need no changes, but it may be useful to select a resolution which is higher than the default 600 points (e.g. 1000 points).

7.6.16 Press **<OK>** to import the entries in the database.

Option 2: Importing band sizes by using a synthetic regression curve

As an exercise, we will now import the same file using the second option described above, i.e. allowing the program to create its own regression curve.

7.6.17 In the main window, open the fingerprint type window for **ABI-Genescan**.

7.6.18 In the ABI-Genescan experiment window, select **Settings | New reference system (curve)**.

The *New reference system* window (Figure 36) allows the size range to be specified as well as the type and strength of the regression.

7.6.19 Under *Metrics range of fingerprint*, enter 1000 as **Top** and 30 as **Bottom**.

7.6.20 Press the **<Add>** button to add the sizes for all reference bands available in the fingerprint type (see lane 17B, Figure 34).

The reference bands are shown as red dots on the regression curve. This makes the adjustment of the *Calibration curve* easier.

7.6.21 Optimize the *Calibration curve* and the strength in % to obtain the best spread of the reference bands.

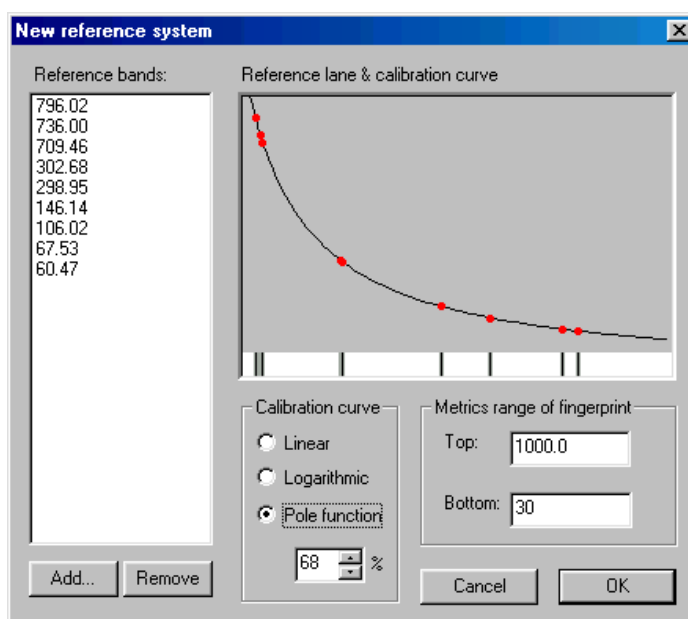


Figure 36. Defining a new reference system using a synthetic regression curve between user-defined size limits.

7.6.22 When finished, press **<OK>** to save the new reference system.

7.6.23 Make the new reference system the *active reference system* by selecting it and **Settings | Set as active reference system**.

Now you can import the same band table as described in 7.6.9 and further. Instead of creating a new data import macro however, you can select the one that already exists for importing Genescan tables. In addition, when creating the database file, you should change the name Genescan.txt into another

name, for example Genescan2.txt, because the program does not allow existing database files to be overwritten.

The two differently imported band size tables are an excellent example to illustrate the *remapping* functions in Fingerprinting II Informatix. Both gels have their bands on different positions because of the different logarithmic function that was used to reconstruct the gels.

7.6.24 Select an entry of the first imported file (should be **G@Example@Genescan@001** or similar if you used other names).

7.6.25 Select the corresponding entry of the second imported file (should be **G@Example@Genescan2@001** or similar if you used other names).

7.6.26 Create a comparison containing these entries and *Layout | Show image*. The patterns look the same except for very minor differences due inevitable error caused by remapping.

7.7 Conversion of gel patterns from GelCompar versions 4.1 and 4.2 and from Molecular Analyst Fingerprinting

The installation CD-ROM contains a directory GCEXPORT, in which the following two files are found: BNEXPORT.EXE and BNEXPORT.HLP.

The program BNEXPORT.EXE and its help file BNEXPORT.HLP should be copied to the home directory of GelCompar 4.1 or GelCompar 4.2.

The file BNEXPORT.HLP is a Windows help file which explains step by step how to proceed to convert patterns from GelCompar to Fingerprinting II Informatix.

To convert gel patterns from Molecular Analyst Fingerprinting software, please contact Bio-Rad Laboratories.

7.8 Defining a new character type

7.8.1 Select *Experiments | Create new character type* from the main menu, or press  and *New character type*.

7.8.2 The *New character type* wizard prompts you to enter a name for the new type. Enter a name, for example "Pheno".

7.8.3 Press <Next> and check the kind of the character data files. Check *Numerical values* if the tests are not just positive or negative but can differ in intensity (choose Numerical values in this example).

7.8.4 For numerical values, enter the number of decimal digits you want to use. If you only want to use integer values, for example between 0 and 10, enter zero (this example).

7.8.5 After pressing <Next> again, the wizard asks if the character type has an open or closed character set.

In an open character set, the number of characters is not defined. For example, studying 10 bacterial strains by means of fatty acids can result in a total of 20 fatty acids found, but if some more strains are

added, more fatty acids may become present in the list. In such cases, *Consider absent values as zero* should be checked, because if a fatty acid is not found in a strain it will not be listed in its fatty acid profile, and thus should be considered as zero.

In a closed character set, the same number of characters are present for all entries studied. This is the case with commercially available test kits. In such cases, *Consider absent values as zero* should not usually be checked.

7.8.6 Answer **No** to the *open character set* and leave the *absent values* checkbox unchecked.

If the character set is *closed*, i.e. when all the tests are predefined, the user is allowed to specify the *Layout* of the test panel. This layout involves a *Number of rows* and *Number of columns* to be specified, as well as the Maximum value for all the tests. By default, the number of rows and columns is set to zero, which means that the character set will be empty initially. In this case, you still can add all the tests one by one or by columns and rows, once the character type is defined. If you are defining a test panel based upon a microplate system (96 wells), you can now enter 8 as *Number of rows* and 12 as *Number of columns*. The program will automatically assign names to the tests: A1, A2, A3, ..., A12, B1, B2, B3, ... These names can be changed into the real test or substrate names afterwards.

Press the **<Finish>** button to complete the setup of the new character type. It is now listed under **Character types** in the experiment type panel.

The new character type exists by now, but the program still doesn't know which, and how many tests it contains. We will further define its tests.

7.8.7 Double click on **Pheno** in the experiment types panel or select **Pheno** and *Experiments | Edit experiment type* or press  in the **Experiments** panel.

The *Character type* window appears (Figure 37), initially with an empty character list. Suppose that the phenotypic character kit exists of 10 tests; we will enter them one by one.

7.8.8 Select *Characters | Add new character*. Enter any name, e.g. **Character 1, Glucosidase...** and press **<OK>**.

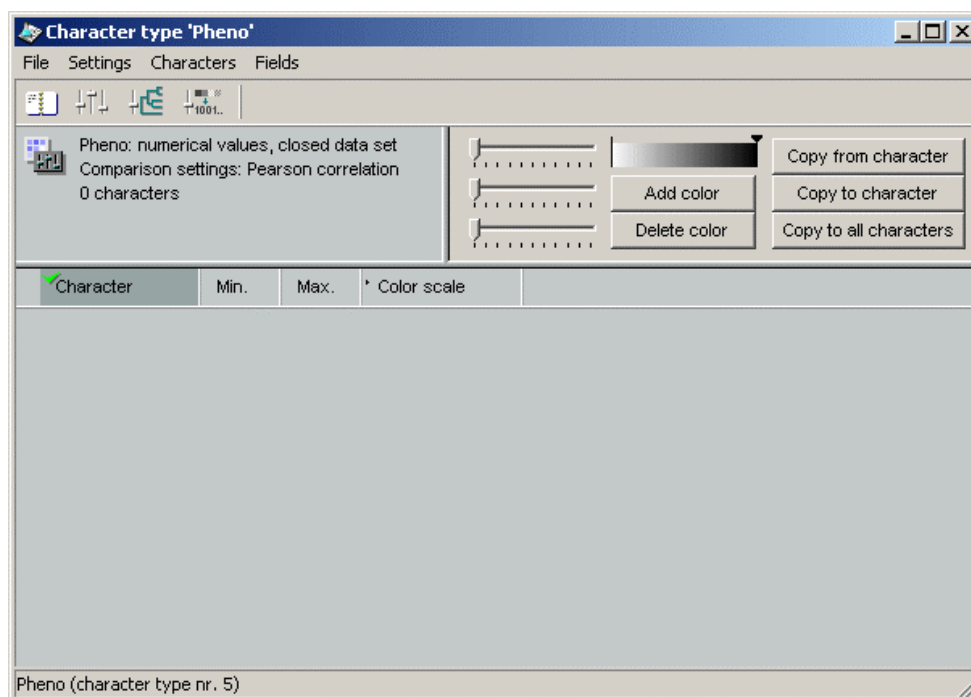


Figure 37. Character type window; no characters defined yet.

The character is now listed in the characters panel (Figure 37). Its default color scale ranges from white (negative) to black (most positive). Its default intensity range is 0 to 100. If you want the character to cover another range, proceed as follows:

- 7.8.9 Select **Character | Change character range**. Enter a minimum and maximum value and press **<OK>**.

NOTE: You can quickly access all menu commands that apply to a character by right-clicking on the character.

- 7.8.10 To change the color for a character, first select it and press **<Copy from character>** in the color setup panel.

- 7.8.11 Then, select the start color left (negative reaction) on the color scale: it is now marked with a black triangle (see arrow on Figure 37).

The three slide bars represent red, green, and blue, respectively. By moving the sliders to the right, the color becomes brighter.

- 7.8.12 Adjust the red, green and blue components individually until you have obtained the desired color for a negative reaction of this character.

- 7.8.13 Select the end color right (positive reaction) on the color scale and adjust the red, green and blue components individually until you have obtained the desired color for a positive reaction of this character.

- 7.8.14 If you want more transition colors, use the **<Add color>** button. A new color mark appears in the middle. You can select this color, drag it to the left or to the right, and adjust it as described.

- 7.8.15 Press **<Copy to character>** to copy the created color scale to the character.

Repeat action 7.8.8 to add more characters, and adjust the colors and the ranges individually as needed. When all characters have the same color range, you can use **<Copy to all characters>**.

A quick method to add a complete array of characters at a time, for example a microplate array, is **Characters|Add array of characters**. The program subsequently asks to enter the number of rows, the number of columns, and the maximum values for the tests. The program automatically assigns names to the tests: A1, A2, A3, ..., B1, B2, B3, ... These names can be changed into the real test or substrate names afterwards.

When a default *Connected Database* is defined for the current database (1), it is possible to define additional information fields for the characters. Additional character fields can be added, renamed or removed with **Fields|Add new field**, **Fields|Rename field**, and **Field|Remove field**, respectively. A string can be entered for a given character by double-clicking on the intersection between the field and the character, or by clicking on the intersection and **Fields|Set field content**. In a comparison window, you can choose to display another field than the character name (first field) as the default to display. Select the field header and **Fields|Use as default field**.

Each character is marked with a green ✓ sign, which means that it is used in comparisons and identifications.

7.8.16 If you want a character to be disabled (not used) in comparisons, uncheck the **Characters|Use character for comparisons** menu item. This may be useful for a blank test which is often present in commercial identification kits. Unused characters are marked with a red cross.

7.8.17 Using the menu **Settings|General settings** or , the character type settings which were entered in the setup wizard can be changed. The **Experiment card** tab, however, lets you define some visual attributes of the experiment. These settings apply to the *experiment card*, which is explained in paragraph 8.2.

With **Represent as Plate** and **Represent as List**, you can choose whether the individual tests are shown graphically on a panel, using colors, or as a list of characters with their name and intensities as a numerical value.

7.8.18 For the example here, choose **Plate**, and enter the number of columns in the test panel (if you entered 10 tests, enter 10 here too).

For test kits on microtiter plates, one would enter 96 tests and 12 columns.

In order to represent existing commercial kits as truthfully as possible, you can choose between three different circular cup types, elliptical cups. For blots and microarrays, you can choose between small blot, large microarray spots and small microarray spots.

7.8.19 Select the type of cells in the **Cell type** pull-down menu, e.g. elliptical, and press **<OK>**.

7.8.20 With the menu command **Settings|Binary conversion settings**, or , you can specify a binary cutoff value in percent.

Whenever converting the numbers to binary states, Fingerprinting II Informatix will consider all values above the cutoff value as positive and those below the cutoff value as negative. If you have entered 50% as cutoff value, you can choose the cutoff level to be 50% of the maximum value found in the experiment, or 50% of the average value from the experiment.

7.8.21 Enter 50% and **Of mean value**.

7.9 Input of character data

There are three possibilities for entering data of a character type:

- A. Importing a character file or importing characters from an ODBC source using the Fingerprinting II Informatix **macro** script language. Such files can be any text format (see Figure 1, step 2). This method is used to import external text formats, e.g. from automated reading devices and other software.
- B. Defining a new character file in Fingerprinting II Informatix and entering the values manually.
- C. Entering the data via the experiment card of the database entry (see 8.3).
- D. Processing and quantification of images scanned as TIFF files.

Importing characters from files or databases

To import characters from ODBC sources (databases or spreadsheets), there are scripts available on the web. These scripts can be launched from the Fingerprinting II Informatix main window, using the menu **Scripts | Browse Internet**, or . The script to import character data can be found under **Import tools** and is called **Import characters (ODBC)**. A full description of how to use this script is available on the website.

A general script to import character data from tab-delimited text files is also available on the website, on the same location. The script is called **Import characters from tab-delimited files**.

Importing external text formats is done as follows:

A macro editor is used to define the format, read the required data from the file, and save the data in the Fingerprinting II Informatix database. Both character data and information fields of the organisms can be imported. See chapter 20.4.6 for a detailed description of the macro editor and the macro language.

To import a character file, we will use the example file PHENOKIT.TXT that is provided on the CD-ROM (directory **EXAMPLES\TXTfiles**).

7.9.1 Select Pheno under Character types.

7.9.2 Choose **File | Import experiment file**.

7.9.3 In the *Input character data* dialog box (Figure 51), select file PHENOKIT.TXT on the CD-ROM.

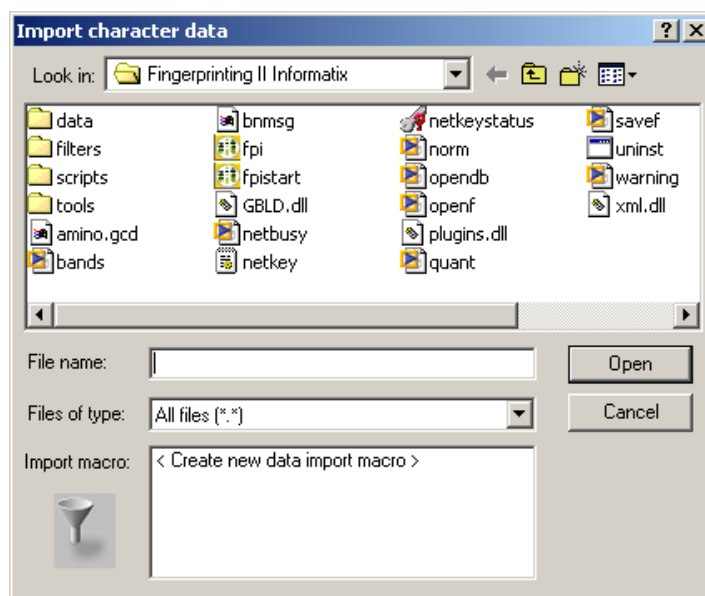


Figure 38. The character import dialog box.

7.9.4 Under *Import macro*, select *Create new data import macro*.

7.9.5 In the macro editor (upper left panel), enter the following macro text (see chapter 20.4.6 for description).

```
read(8)
setfield("genus",readuntil(" "))
if not(endofline) then setfield("species",readuntil(" "))
if not(endofline) then setfield("subspecies",readuntil(""))
nextline
read(8)
setfield("sample",readuntil(""))
nextline
read(8)
repeat 10 times setnextvalue(read(1))
nextline
```

7.9.6 Select *Macro | Run* and then *Entries | Import*.

NOTE: Files to import should contain no more than 300 entries. If the file is larger, the number of imported entries will be truncated after the first 300.

Defining a new character file

A new character file is created as follows:

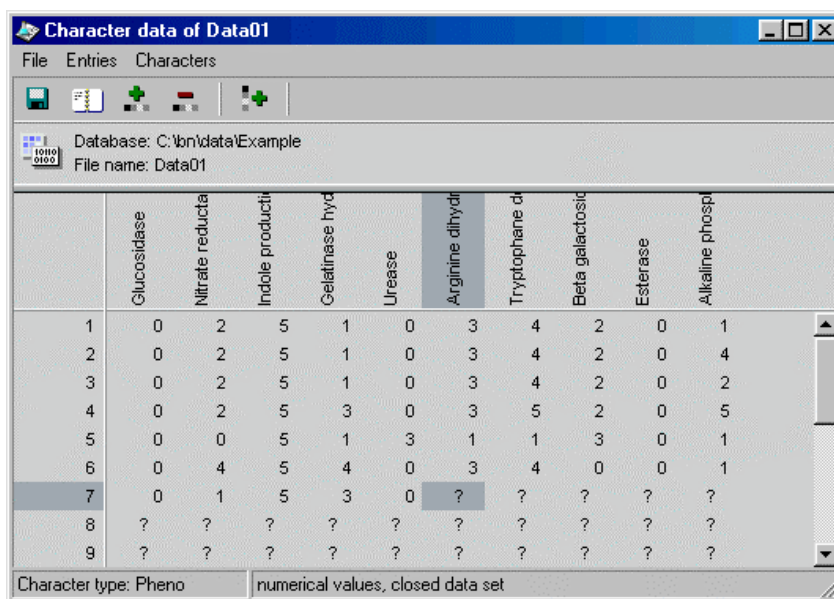
7.9.7 In database **Example**, select the new character type **Pheno**.

7.9.8 Right-click in the **Files** panel, and choose *Add new experiment file* from the floating menu.

7.9.9 Enter a name, e.g. **Data01**, and press <OK>.

7.9.10 Select **Data01** in the **Files** panel, and *File | Open experiment file (data)*.

This opens the *Character data file* window (Figure 39), which is empty initially. The 10 test names which we entered as an example, are shown in the column header. You can add characters here with *Characters | Add new character* or .



	Glucosidase	Nitrate reducta	Indole producti	Gelatinase hyd	Urease	Arginine dihydr	Tryptophane d	Beta galactosid	Esterase	Alkaline phospt
1	0	2	5	1	0	3	4	2	0	1
2	0	2	5	1	0	3	4	2	0	4
3	0	2	5	1	0	3	4	2	0	2
4	0	2	5	3	0	3	5	2	0	5
5	0	0	5	1	3	1	1	3	0	1
6	0	4	5	4	0	3	4	0	0	1
7	0	1	5	3	0	?	?	?	?	?
8	?	?	?	?	?	?	?	?	?	?
9	?	?	?	?	?	?	?	?	?	?

Character type: Pheno numerical values, closed data set

Figure 39. Character data file window.

Before you can enter data, you have to add new entries to the file. Suppose that we want to add character data for all entries of the database except the standard (17).

7.9.11 Select *Entries | Add new entries* or .

7.9.12 You are prompted to enter the number of entries; enter 17, and press <OK>.

Seventeen entries are now present, and all character values are initially represented by a question mark (Figure 39).

7.9.13 To enter values, you can either double click on the question mark, or press the Enter key.

7.9.14 Enter values between 0 and 5 (the range of the characters), and press Enter again.

The next character of the same entry is automatically selected, so that you can directly enter the next value.

7.9.15 If you are entering large character files, we recommend to save now and then with *File | Save* () or the F2 shortcut.

7.9.16 *File | Exit* when you are finished entering the data.

7.9.17 In the main window, double click on the file **Data01** (or *File | Open experiment file (entries)*).

The *Character entry file* window (cf. the Fingerprint entry file window, Figure 31) contains unlinked entries, which you can now link to the corresponding database entries.

A link arrow  for each entry allows you to link an entry to a database entry, by clicking on the arrow and dragging it onto a database entry, and then releasing the mouse button. When the experiment is linked, its link arrow is purple: .

7.9.18 Drag the link arrow of **entry 1** to the first database entry: as soon as you pass over a database entry, the cursor shape changes into .

7.9.19 Release the mouse button on the database entry; entry is now linked to this database entry, and its arrow in the character entry file window has become purple  instead of gray .

NOTE: if you try to link an entry to a database entry which already has an entry of the same experiment type linked to it, the program will refuse the second link with the message:

“The Experiment ‘Pheno’ of this database is already defined in XXX”

where XXX is another lane of the same character type, in the same or another experiment file.

As soon as an experiment is linked to a database entry, the experiment presence panel (see Figure 10) shows a green dot for the experiment of this entry.

You can edit the information fields for this entry in two places: directly in the database (see 6.4.1 to 6.4.2), or in the character entry file window, by double clicking on the entry.

*NOTE: Experiment files added to the Experiment files panel can also be deleted by selecting the file and choosing **File|Delete experiment file** from the main menu. Deleted experiment files are struck through (red line) but are not actually deleted until you exit the program. So long, you can undo the deletion of the file by selecting **File|Delete experiment file** again.*

7.10 Import of character data by quantification of images scanned as TIFF files

Similar as for gel images, Fingerprinting II Informatix can import *character type* data from TIFF images. This happens by quantification of the color intensity and/or color transitions on the TIFF file. Character data from phenotypic test panels often provide color transitions rather than changes in intensity. For example, many test panels have reactions that change from yellow to red, or blue – green – yellow, and hence, quantifying the colors by the intensity would provide no meaningful information. Rather, the program needs to be able to read the files as true RGB images and allow the possibility to define negative colors and positive colors, as well as transition colors. For example, in an acidification reaction with a bromophenol blue dye, non-reactive tests will be blue, whereas weak reactions will show the transition color green, and strongly positive reactions will show up yellow.

Using the same tool, Fingerprinting II Informatix also allows the import of micro-array and gene chip images scanned as TIFF files, offering for each gene or oligo a quantitative reaction value.

The character import tool is provided as a separate program, BNIMA.EXE, that can be started from the Fingerprinting II Informatix program. BNIMA only works when the Fingerprinting II Informatix analyze program is running, in other words, you should either start BNIMA from within Fingerprinting II Informatix, or first launch Fingerprinting II Informatix and then start BNIMA.

Example 1: import of microtiter plate image.

The first example we will use to illustrate the program is EXAMPLES\CHARFILE\PLATE1.TIF on the Fingerprinting II Informatix installation CD-ROM. It is a photograph of a 96 wells microtiter plate with bromophenol blue as reaction indicator dye (see Figure 40).

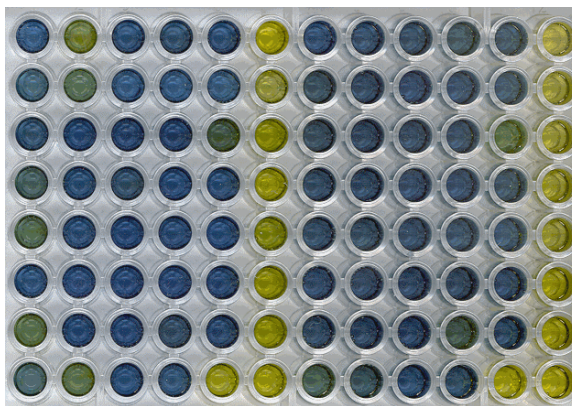


Figure 40. 96-wells microtiter plate with bromophenol blue as reaction indicator dye.

7.10.1 Create a new *closed character type* as described in 7.8.1 to 7.8.6, and call it **Microplate**. Specify 8 rows and 12 columns under *Layout* (third step). Specify a color range from blue to yellow, over green (see 7.8.10).

7.10.2 Double click on an entry to show its Entry edit window.

The experiment type Microplate shows an empty flask.

7.10.3 Click on the flask button. Since this experiment is not defined for the selected entry, the program asks “Do you want to create a new one?”.

7.10.4 Answer <Yes> to create an *Experiment card* (see further, paragraph 8.2), an empty microplate image.

7.10.5 Right-click on the empty microplate image and select *Edit image* from the floating menu.

This loads the BNIMA program.

7.10.6 Select *File|Load image* in BNIMA and load the file EXAMPLES\CHARFILE\PLATE1.TIF from the Fingerprinting II Informatix installation CD-ROM. The resulting window looks as in Figure 41.

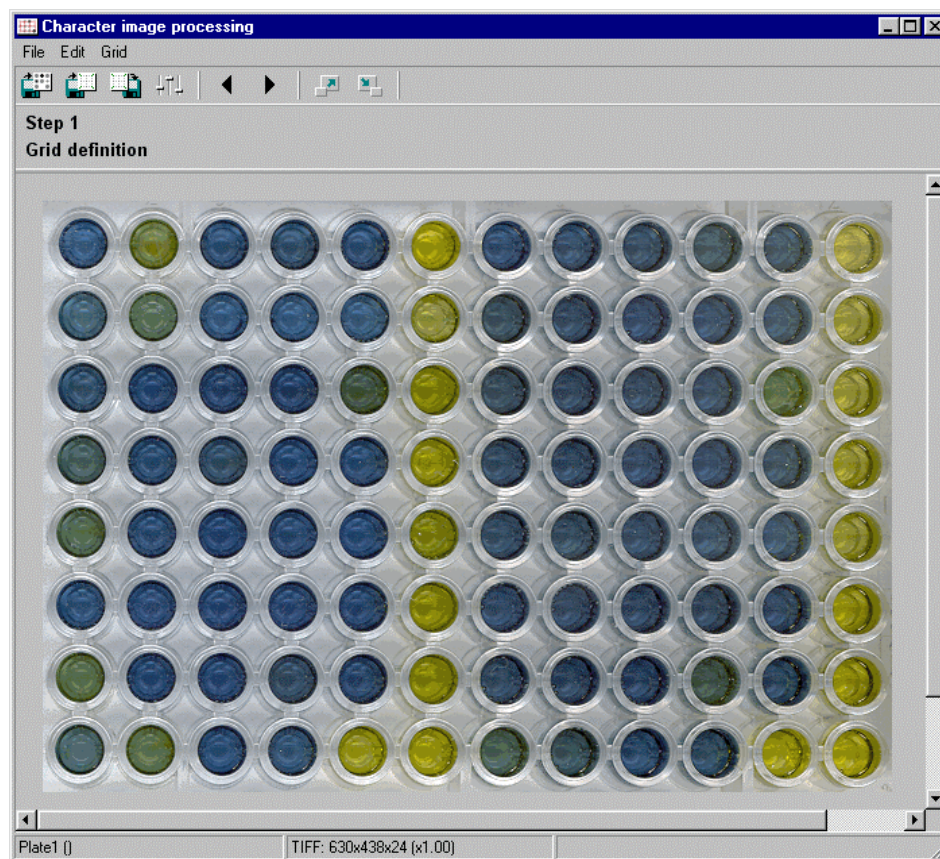


Figure 41. The BNIMA program with a microplate image loaded.

7.10.7 First call the settings dialog box with *Edit | Settings* or .

The *Image* tab offers two choices for the *Image type*: *Densitometric* and *Color scale*.

In case the color reaction can be interpreted as a simple change in intensity (e.g. from light to dark), one should select *Densitometric*. The *Densitometric values* panel offers some additional tools to edit the TIFF file: *Inverted values* is to invert the densitometric values; *Background subtraction* allows a two-dimensional subtraction of the background from the TIFF file, using the rolling ball principle. The *Ball size* can be entered in pixels. Spot removal allows all spots and irregularities below a certain size to be removed from the image, whereas larger structures are preserved. The background subtraction and spot removal changes are only seen when *Edit | Show value scale* is enabled in the main window.

In case the reaction causes a change from one color to another color, as in the above example, *Color scale* is the right option. An additional feature, *Hue only*, is particularly useful when the scanned images differ in brightness (illumination) or contrast. If the images do not contain black or white in their color range, it is better to enable this feature.

7.10.8 Select *Color scale* and *Hue only*.

7.10.9 Press <OK> to proceed with these settings. The other settings will be discussed later.

Like the process of normalization of gels, processing a character panel image exists of a number of steps: (1) Grid definition; (2) Cell layout; and (3) Quantification.

In **Step 1: Grid definition** we will create a grid that defines the wells of the microplate.

7.10.10 Select **Grid | Add new** and enter 8 as *Number of rows* and 12 as *Number of columns*.

7.10.11 Press **<OK>** and the grid appears.

At each edge of the grid, there is a dragging node (green square). The upper left dragging node is to *move* the grid as a whole; the lower right node is to *resize* the grid, and the upper right and lower left nodes are to *distort* the grid in case the image is not perfectly rectangular or not scanned horizontally.

7.10.12 Drag the nodes until the grid matches with all 96 wells.

NOTE: Using the SHIFT key, one can distort the grid locally if needed. The size of the local distortion area is indicated by a circle. It is even possible to reduce or enlarge the size of the distortion area as follows: hold the shift key and left-click on any cell-marking cross of the grid. The area defining circle appears. Hold the left mouse button down and release the SHIFT key: the circle is still visible. While holding the left mouse button down, press the PgDn or PgUp key to reduce or enlarge the area of distortion. The size of the circle will decrease or increase. Using a very small circle, it is possible to correct the grid in any individual cell.

7.10.13 In case you want to remove the grid and define it again, select one of the cells of the grid (becomes red) and **Grid | Delete**.

In case the image consists of two or more subsets of cells (e.g. some more complex test panels or micro-arrays), it is possible to define more than one grid using the **Grid | Add new** command.

7.10.14 Move to the next step using **Edit | Next step** or the  button.

In this step, the layout of the cells is defined: the shape and size of the quantification area within each cell. In this step, we also define which cells we want to use for quantification and which cells not. By default, all cells of the grid are used for quantification.

7.10.15 Click in the upper left corner of the image and while holding the left mouse button down, select the left half of the test panel.

All the selected cells are marked in red.

7.10.16 Select **Cells | Delete selected**. The left half of the panel will not be used for quantification, and hence, cannot be used in the resulting character set.

Cross marks of unused cells are smaller than of used cells.

7.10.17 Select the non-used cells again with **Cells | Add selected**.

Before the program can do the quantification, it needs to know what the averaging area of the cells is. This is done using a *mask* which the user defines. One can define the same mask for all cells, or assign particular masks to individual (groups of) cells. In the case of a microplate it is obvious that all cells should have the same mask.

7.10.18 Select all cells as in 7.10.15.

7.10.19 Add a circular mask to all selected cells with **Cells | Add disk to mask**.

A dialog box prompts to enter a *Radius* for the disk in pixels, the *X offset* (horizontal shift from the cell marking cross) and the *Y offset* (vertical shift from the cell marking cross). For the offsets, a negative value can be entered.

7.10.20 Enter 8 as radius, and leave the offsets zero. Press **<OK>** to confirm.

The masks appear on all used cells of the grid as semitransparent red disks. In order to see the masks, it is important that they are in a color that is complementary to the reaction colors of the cells. One can change the color of the masks as follows:

7.10.21 Select **Edit|Settings** or .

7.10.22 Under **Layout**, pull down the **Mask color** menu and select the appropriate color.

In the example microplate, the most appropriate color is red.

*NOTE: In case of very small cells, e.g. microplate images, you can select **Small cross marks**, so that they don't overlap most of the cells.*

It is possible to add more than one mask to the cells. In case the cells have a more complex layout, i.e. not just circular, one can add two or more disks with different offsets to approach the shape of the cells. By selecting individual cells or groups of cells, it is also possible to change the shape of the masks per cell or per group of cells.

NOTE: Some more advanced features allow the mask of individual cells to be changed manually:

*With **Cells|Add pixels to mask** or  the cursor changes into a pencil which you can use to add pixels to the masks manually. When doing so, it is recommended to zoom in on the cell using the*

***Edit|Zoom in** command or .*

*Similarly, it is possible to remove pixels from the mask with **Cells|Remove pixels from mask** or . Clicking a second time on these buttons or selecting the menu item finishes the pixel editing mode.*

If you selected the image type to be **Color scale**, and not **Densitometric** in the settings (see 7.10.7 to 7.10.8), you can now specify the negative color, the positive color, and any transition colors between negative and positive. For each cell, you can define a unique color scale, which can be necessary for some commercial test panels containing more than one reaction dye.

7.10.23 In the example case, there is only one reaction dye, so select all cells as in 7.10.15.

7.10.24 Select **Cells|Edit color scale** or . This brings up the *Color scale editor* as shown in Figure 42.

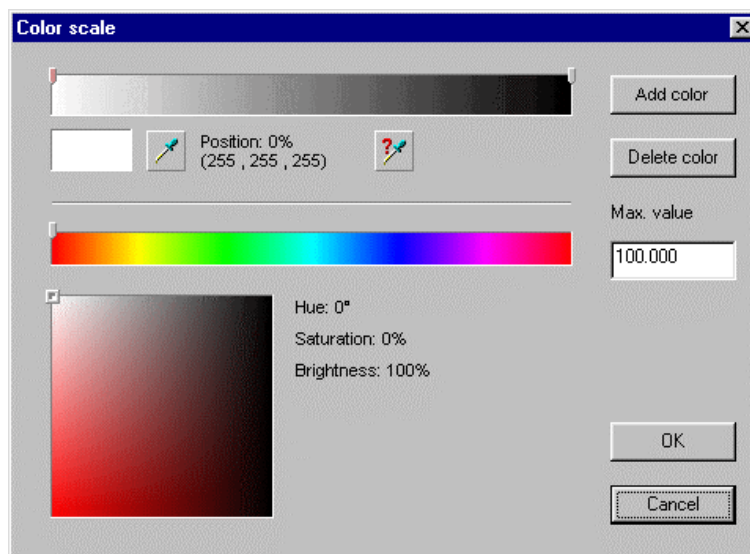


Figure 42. Color scale editor in the Cell layout step of the BNIMAGE program.

By default, the color scale exists of two colors: white as negative and black as positive. In the case of the example microtiter plate, this scale would obviously not work. Since the scale ranges from blueish (negative) over greenish to yellow (positive), we will add a new intermediate color.

7.10.25 Press the **Add color** button. One new color (gray) is defined in the middle of the scale.

7.10.26 Select the color selector of the negative color (left).

7.10.27 Move the slider on the color scale of predefined colors to blue.

7.10.28 Move the slider in the Saturation/Brightness square to the lower left corner to obtain maximum brightness and saturation.

7.10.29 Repeat 7.10.26 to 7.10.28 for the intermediate color (middle), assigning green, and for the positive color (right), assigning yellow.

*NOTE: If you selected **Hue only** in the settings (see 7.10.7 to 7.10.8), changing saturation and brightness has no effect on the obtained color scale. If saturation or brightness transitions within the same color are to be registered, you should disable the **Hue only** feature in the settings.*

The upper color scale now should range from blue over green to yellow (Figure 43).



Figure 43. Appropriate color scale for the example microplate image.

*NOTE: One can also pick up colors from the image in order to define the selected color in the upper color scale. To this end, click and hold the left mouse button on the left pipet button . The mouse pointer shape changes into a pipet which you can drag to the most negative cell, e.g. the blank control. The selected color in the color scale automatically changes into the color at the pipet's position. If **Hue only** is enabled, the closest hue color is selected.*

Once the color scale is defined you can interrogate the reaction of any cell using the right pipet button  as described above. This pipet does not affect the defined color scale, but only shows the position of the pointed cell graphically on the color scale, and the percentage reaction with error indication.

With the *Max. value* field you can enter the maximum value to which all characters will be rescaled.

7.10.30 Enter 100 as *Max. value* and press <OK> to confirm the color settings.

7.10.31 Move to the next step using *Edit|Next step* or the  button.

The next and last step involves quantification of the cells. First of all, the cells to be added to the character set need to be defined. In case one or more cells are intended only for calibration purposes, they can be excluded from the resulting character set, but used as calibration marker.

7.10.32 Select all cells as in 7.10.15.

7.10.33 Select *Quantification|Add cells to character set*.

The cells are now numbered from 1 to 96.

7.10.34 If you click on a particular cell, its quantified value as rescaled according to 7.10.30 is given in the status bar as well as the value after calibration (see further).

Quantification is done by integrating the pixels within the defined mask. There are different options for integration:

7.10.35 Select *Edit|Settings* or  and choose the *Quantification* tab.

Cell integration methods include *Average*, *Median*, and *Sum*. In case the image contains spots that could influence the quantified values, the Median option will provide more reliable results than the arithmetic averages.

7.10.36 Select *Median* integration and press <OK>.

In order to illustrate the calibration feature, we will define one of the cells as negative control (minimum value), and another cell as positive control (maximum value).

7.10.37 Select cell A1 (negative control) and *Quantification|Define calibration point*. Enter 0 as value and press <OK>.

7.10.38 Select cell A12 (positive control) and *Quantification|Define calibration point*. Enter 100 as value and press <OK>.

Since only two calibration points are defined now, it is obvious that the program needs to calculate a linear regression through the defined points, in order to requantify the other cells according to the negative and positive controls:

7.10.39 Select *Edit|Settings* or  and choose the *Quantification* tab.

7.10.40 Under *Calibration*, enter 1 as *Polynomial degree*. This will result in a first degree regression.

7.10.41 Press <OK> to close the settings dialog box.

7.10.42 Select *Quantification | View calibration curve*. This shows a linear regression between the two calibration points zero and 100.

Finally, there is one more thing to do, i.e. to copy the character values in the microplate opened in Fingerprinting II Informatix.

7.10.43 Select *Quantification | Export to clipboard* or .

Before closing the BNIMA program, you can save the entire configuration defined for this microplate system. If you load a next microplate, you can reload the grid and all other settings such as color scale, disabled cells, quantification parameters etc.

7.10.44 Select *File | Save configuration as* or .

7.10.45 Enter a name e.g. microplate, and press <OK> to save the configuration.

For next microplates you can reload the configuration using *File | Load configuration* or .

7.10.46 Close the BNIMA program.

7.10.47 Right click on the *Experiment card* (see also paragraph 8.2), and select *Paste data from clipboard* from the floating menu.

The microplate now is filled with data and looks like in Figure 44.

7.10.48 Click the upper left triangular button to close the experiment card.

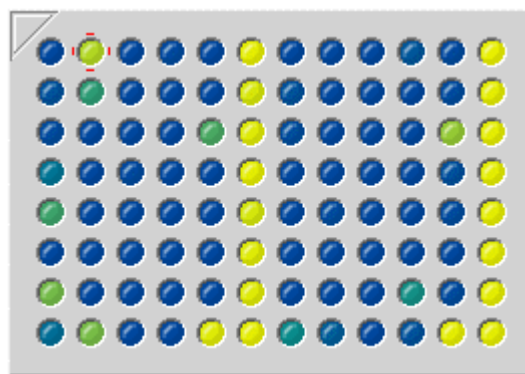


Figure 44. Example microplate experiment card after import of character values using BNIMA.

Example 2: import of gene-array scanning.

The second example we will use to illustrate the BNIMA program is a fragment of a gene array which can be found in EXAMPLES\CHARFILE\ARRAY.TIF on the Fingerprinting II Informatix installation CD-ROM. The array image was generated by chemiluminescent detection of digoxigenin-labeled cDNA¹. Each gene is characterized by two spots (horizontally next to each other), which can be considered as a control measure. For this example, we have used a fragment representing two blocks of 14 x 7 genes (the complete array is composed of six blocks of 14x7 genes, totalling 588 characters). The left and right half are separated by one blank column, and the two bottom rows contain calibration and reference spots (see Figure 45).

¹ Courtesy S.D. Vernon, M.S. Mangalathu and E.R. Unger (J. Histochemistry & Cytochemistry 1999; 47:337-342).

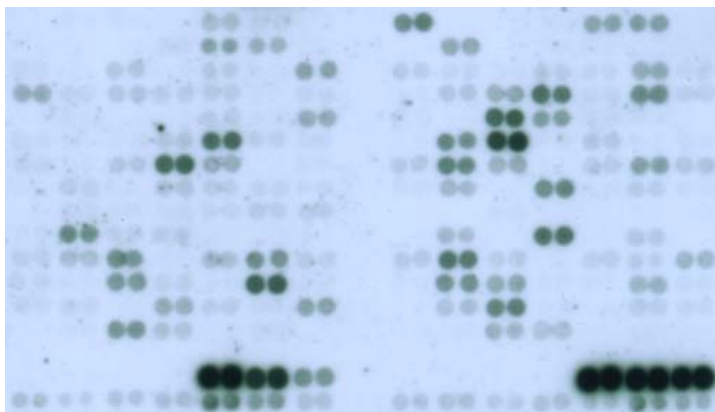


Figure 45. Fragment of gene array scanned as TIFF image (file ARRAY.TIFF)

7.10.49 Create a new *closed character type* as described in 7.8.1 to 7.8.6, and call it **Gene array**. Specify 14 rows and 14 columns under *Layout* (third step).

7.10.50 When the **Gene array** experiment type is created, double click on it in the **Experiments** panel.

7.10.51 In the appearing *Character type* window, select *Settings|General settings*, and click the *Experiment card* tab.

7.10.52 Under *Cell type*, select *Small blot*, which makes it possible to show large data sets in the experiment cards (see paragraph 8.2).

7.10.53 Click <OK> and close the *character type* window.

7.10.54 Double click on an entry to show its Entry edit window.

The experiment type **Gene array** shows an empty flask.

7.10.55 Click on the flask button. Since this experiment is not defined for the selected entry, the program asks “Do you want to create a new one?”.

7.10.56 Answer <Yes> to create an *Experiment card* (see further, paragraph 8.2), an empty 14 by 14 array image.

7.10.57 Right-click on the empty array image and select *Edit image* from the floating menu.

This loads the BNIMA program.

7.10.58 Select *File|Load image* in BNIMA and load the file EXAMPLES\CHARFILE\ARRAY.TIF from the Fingerprinting II Informatix installation CD-ROM. The resulting window looks as in Figure 46.

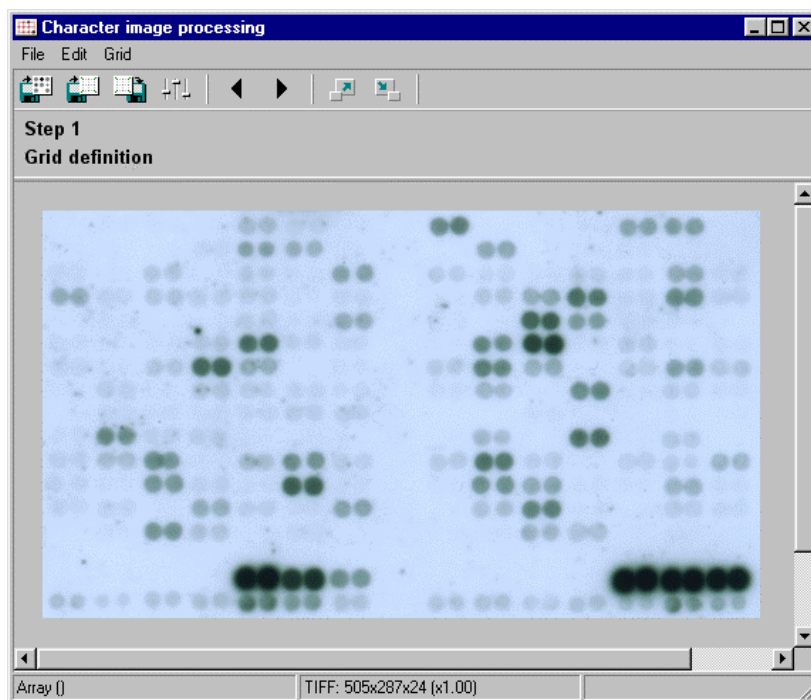


Figure 46. The BNIMA program with a gene array image (fragment) loaded.

7.10.59 First call the settings dialog box with *Edit | Settings* or .

The *Image* tab offers two choices for the *Image type*: *Densitometric* and *Color scale*.

Unlike the first microplate image, the color reaction of this gene array can be interpreted as a simple change in intensity (e.g. from light to dark), hence one should select *Densitometric*.

7.10.60 Select *Densitometric* under *Image type*.

The *Densitometric values* panel offers some additional tools to edit the TIFF file: *Inverted values* is to invert the densitometric values; *Background subtraction* allows a two-dimensional subtraction of the background from the TIFF file, using the rolling ball principle. The *Ball size* can be entered in pixels. Background subtraction is only necessary if the illumination of the image is not uniform, which is not the case in the example image. Spot removal allows all spots and irregularities below a certain size to be removed from the image, whereas larger structures are preserved.

7.10.61 Leave *Background subtraction* disabled, and enable *Spot removal*, with a maximal *Spot size* of 3 pixels.

7.10.62 Press <OK> to quit the settings panel.

The background subtraction and spot removal changes are only seen when *Edit | Show value scale* is enabled in the main window.

7.10.63 Check *Edit | Show value scale*. The image now looks “cleaned up”: spots are removed and the image is shown in grayscale rather than as 24 bit true color image.

In **Step 1: Grid definition** we will create a grid that defines the cells of the array.

7.10.64 Select *Grid | Add new* and enter 17 as *Number of rows* and 15 as *Number of columns*.

Choosing 17 and 15 rather than 14 by 14 is to allow the calibration spots to be included, and to take account of the blank column.

7.10.65 Press **<OK>** and the grid appears.

7.10.66 Move the upper left dragging node until the grid crosses match the middle of each double spot in the upper left area of the array (see Figure 47).

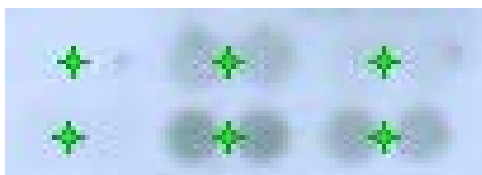


Figure 47. Correct alignment of grid on gene array spots.

7.10.67 Next, move the lower right dragging node until the grid crosses match the middle of each double spot in the lower right area of the array.

7.10.68 Then, move the lower left and upper right dragging nodes of the grid to distort the rectangle so that the grid crosses in the lower left and upper right areas, respectively, match with the double spots.

The grid on the image should now look as in Figure 48.

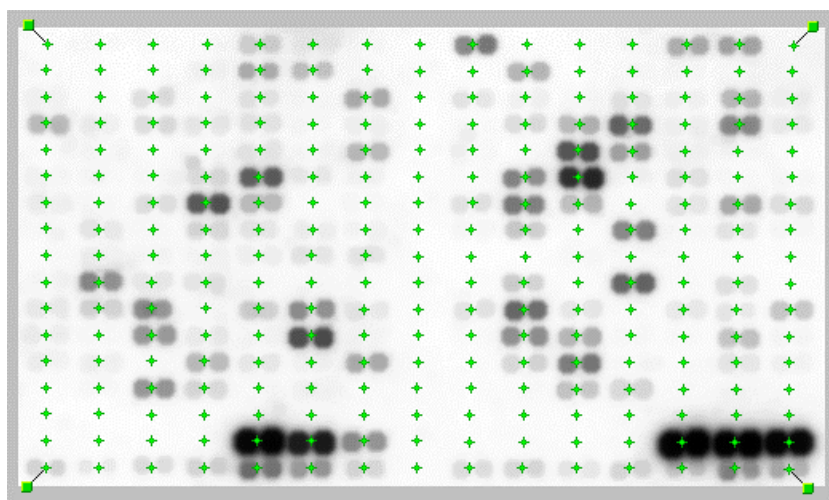


Figure 48. Correctly aligned grid on example gene array.

7.10.69 Move to the next step using *Edit | Next step* or the  button.

In this step, the layout of the cells is defined: the shape and size of the quantification area within each cell. In this step, we also define which cells we want to use for quantification and which cells not. By default, all cells of the grid are used for quantification.

7.10.70 Select the cells in the blank column of the image and *Cells | Delete selected*.

7.10.71 Similarly, select the three lowest rows and *Cells | Delete selected*.

Two cells of the second last row represent 0 and 100% hybridization respectively: the 4th and the 5th cell. We will include these cells for calibration, hence we have to include them again:

7.10.72 Select the 4th and 5th cell of the second last row and *Cells | Add selected*.

Before the program can do the quantification, it needs to know what the averaging area of the cells is. This is done using a *mask* which the user defines. In this case, it is clear that we will have to define two masks per cell, in order to cover the duplicate spots.

7.10.73 Select all cells as in 7.10.15.

7.10.74 Add a circular mask to all selected cells with *Cells | Add disk to mask*.

A dialog box prompts to enter a *Radius* for the disk in pixels, the *X offset* (horizontal shift from the cell marking cross) and the *Y offset* (vertical shift from the cell marking cross). For the offsets, a negative value can be entered.

7.10.75 Enter 6 as radius, and -6 as *X offset*. Press <OK> to confirm.

The masks appear on all used cells of the grid as semitransparent red disks.

7.10.76 Add a second mask to all selected cells with *Cells | Add disk to mask*.

7.10.77 Enter 6 as radius, and 6 as *X offset*. Press <OK> to confirm.

After these steps, the BNIMA window should look like in Figure 49.

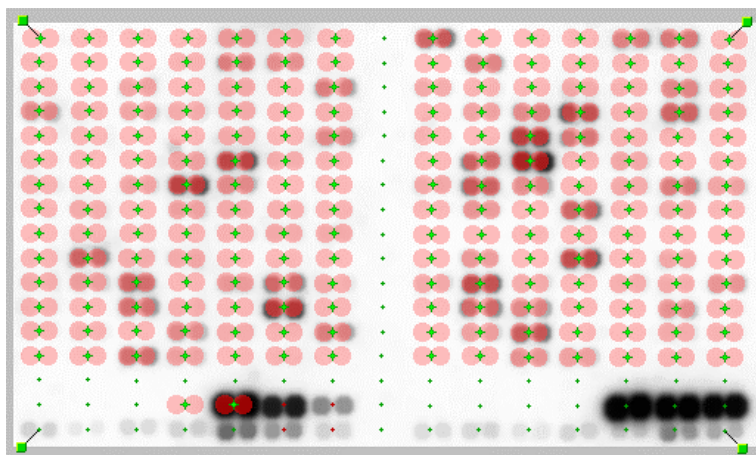


Figure 49. Array editing in BNIMA, with included and excluded cells, and masks defined.

7.10.78 If this is the case, move to the next step using *Edit | Next step* or the  button.

7.10.79 In the *Quantification* step, first call the settings dialog box with *Edit | Settings* or .

7.10.80 Select the *Quantification* tab, specify a first degree polynomial fit and click <OK>.

7.10.81 Select the 4th cell in the second last row and *Quantification | Define calibration point*.

7.10.82 Enter 0 (zero).

7.10.83 Select the 5th cell in the second last row and *Quantification | Define calibration point*.

7.10.84 Enter 100.

All cells are now quantified between the zero and 100% hybridization control, and we now need to specify which cells to add to the character set. Since the calibration cells (second last row) are not part of the character set, these should not be included.

7.10.85 Select all but the three last rows and *Quantification | Add cells to character set*.

The cells to be used in the character set are now numbered 1 to 196.

7.10.86 Copy the quantified cells to the clipboard with *Quantification | Export to clipboard* or



Before closing the BNIMA program, you can save the entire configuration defined for this gene array system:

7.10.87 Select *File | Save configuration as* or .

7.10.88 Enter a name e.g. "Gene array", and press <OK> to save the configuration.

7.10.89 Close the BNIMA program.

7.10.90 Right click on the *Experiment card* (see also paragraph 8.2), and select *Paste data from clipboard* from the floating menu.

The experiment card now is filled with data and looks like in Figure 50.

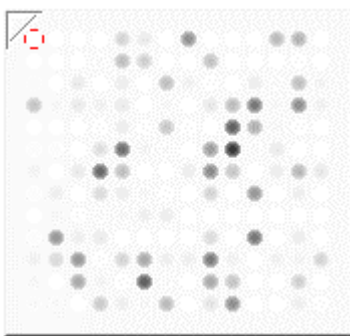


Figure 50. Example gene array experiment card after import of character values using BNIMA.

7.10.91 Click the upper left triangular button to close the experiment card.

7.11 Defining a new sequence type

7.11.1 Select *Experiments | Create new sequence type* from the main menu, or press  and *New sequence type*.

7.11.2 The *New sequence type* wizard prompts you to enter a name for the new type. Enter a name, for example “SSU-Ribo”.

7.11.3 Press <Next> and check the kind of the sequences: *nucleic acid sequences* or *amino acid sequences*. Select nucleic acid sequences.

Press the <Finish> button to complete the setup of the new sequence type. It is now listed under **Sequence types** in the experiment type panel.

The new sequence type exists by now, and we can enter sequence data in three ways:

- A. Importing a sequence file as external format. EMBL, GenBank and Fasta formats are supported, and the information fields can be extracted from the headers using tags.
- B. Defining a new sequence file in Fingerprinting II Informatix and entering the values manually, or pasting them from the clipboard.
- C. Entering or pasting the sequences via the experiment card of the database entry (see 8.3).

To import sequences in EMBL format, we will use the example file EMBL.TXT that is provided on the CD-ROM (directory **EXAMPLES\TXTfiles**).

7.11.4 Select SSU-Ribo under Sequence types.

7.11.5 Choose **File | Import experiment file**.

In the *Open file* dialog box, select file EMBL.TXT on the CD-ROM. This opens the sequence import dialog box (Figure 51).

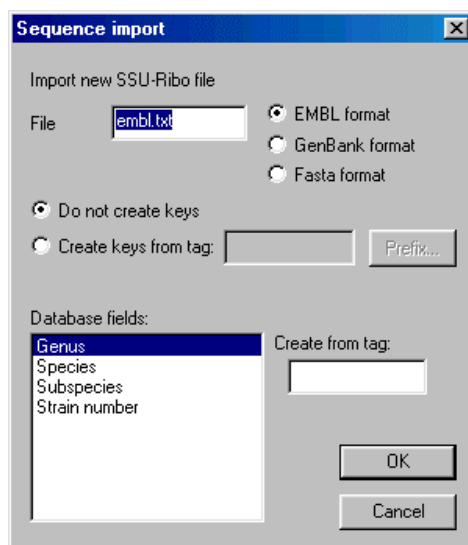


Figure 51. The sequence import dialog box.

In order to understand how the import of information via tags works, you may want to open the file EMBL.TXT in Notepad (Wordpad) or another text editor. EMBL files and GenBank files contain for each sequence a header of which the information is characterized by tags. In EMBL, DE refers to the organism name, AC is the accession number, KW is the keyword, etc. Similar tags are used in the GenBank format. We will specify these tags to extract specific information from the headers.

In the Fasta format, each sequence is preceded by one information line, which starts with a > sign. If this information line contains several fields separated by vertical slashes (/), you can enter a number: for example, the third field is represented by a 3.

7.11.6 The file name as it will be saved in the experiment files directory of Fingerprinting II Informatix is shown in a text box *File*. If desired, you can change this name, for example if a file with the same name already exist in the experiment files directory.

7.11.7 Select *EMBL format* from the supported formats.

With *Do not create keys*, the program will import the sequences without automatically creating new corresponding database entries. Since no entries are created, the *Database fields* list has no importance in this case. If you select this function, you will have to link the sequences manually to existing database entries, as explained for character types (see 7.9.17 to 7.9.19).

If *Create keys from tag:* is selected, the program will automatically create a new database entry for each sequence. If nothing is filled in the *TAG* field, the program will automatically construct keys for the entries. By entering a tag that refers to a unique code, for example the accession number, you can have the program use these numbers to create the entry keys. The *<Prefix>* button allows you to specify a fixed prefix that will precede each key.

7.11.8 Choose *Create keys from tag:* and enter **NI** as tag.

7.11.9 Select *Genus* under *Database fields*, and enter **OS** as tag. In fact, **OS** in the EMBL file displays the full organism name, so that we are unable to read the genus and species names separately from the file.

7.11.10 Select *Strain no* under *Database fields*, and enter **AC** as tag.

7.11.11 Press *<OK>*. Twenty new database entries are created.

7.11.12 You can further edit the sequences by selecting file **EMBL.TXT** in the **Files** panel, and *File | Open experiment file (data)*.

NOTE: Files to import should contain no more than 300 entries. If the file is larger, the number of imported entries will be truncated after the first 300.

Double click on an entry or *Sequence | Edit* to edit the sequence.

A new sequence file is created as follows:

7.11.13 In database **Example**, select the new sequence type **SSU-Ribo**.

7.11.14 Right-click in the **Files** panel, and choose *Add new experiment file* from the floating menu.

7.11.15 Enter a name, e.g. **Seq01**. and press *<OK>*.

7.11.16 Select **Seq01** in the **Files** panel, and *File | Open experiment file (data)*.

This opens the *Sequence data file* window, which is empty initially.

Before you can enter sequences, you have to add new entries to the file. Suppose that we want to add sequence data for three more entries of the database.

7.11.17 Select *Entries | Add new entries* or .

7.11.18 You are prompted to enter the number of entries; enter 3, and press <OK>.

Three entries are now present, and all sequences are initially represented by a blank line.

With *Sequence|Paste from clipboard*, the contents of the clipboard is pasted into the selected sequence.

7.11.19 Double click or *Sequence|Edit* is to edit the sequence or to enter the bases manually.

7.11.20 If you are doing a lot of editing work, we recommend to save now and then with *File|Save* () or the F2 shortcut.

7.11.21 *File|Exit* when you are finished editing the sequences.

7.11.22 In the main window, double click on the file **Seq01** (or *File|Open experiment file (entries)*).

The *Sequence entry file* window (cf. the Fingerprint entry file window, Figure 31) contains unlinked entries, which you can now link to the corresponding database entries.

A link arrow  for each entry allows you to link an entry to a database entry, by clicking on the arrow and dragging it onto a database entry, and then releasing the mouse button. When the experiment is linked, its link arrow is purple: .

7.11.23 Drag the link arrow of **entry 1** to any database entry that doesn't have a sequence linked: as soon as you pass over a database entry, the cursor shape changes into .

7.11.24 Release the mouse button on the database entry; entry is now linked to this database entry, and its arrow in the sequence entry file window has become purple  instead of gray .

NOTE: if you try to link an entry to a database entry which already has an entry of the same experiment type linked to it, the program will refuse the second link with the message:

"The Experiment 'SSU-Ribo' of this database is already defined in XXX"

where XXX is another lane of the same sequence type, in the same or another experiment file.

As soon as an experiment is linked to a database entry, the experiment presence panel (see Figure 10) shows a green dot for the experiment of this entry.

You can edit the information fields for this entry in two places: directly in the database (see 6.4.1 to 6.4.2), or in the sequence entry file window, by double clicking on the entry.

*NOTE: Experiment files added to the Experiment files panel can also be deleted by selecting the file and choosing **File|Delete experiment file** from the main menu. Deleted experiment files are struck through (red line) but are not actually deleted until you exit the program. So long, you can undo the deletion of the file by selecting **File|Delete experiment file** again.*

7.12 Input of sequences using the contig assembler program GeneBuilder

GeneBuilder is a plugin tool to assemble contig sequences from partial sequences which result from sequencing experiments. The program accepts flat text files as well as binary chromatogram files from

ABI and Beckman sequencers. In the latter cases, GeneBuilder allows the user to verify base assignments by inspecting the chromatograms along with the partial sequences and the consensus sequence.

GeneBuilder can be launched only from Fingerprinting II Informatix

7.12.1 Double-click on an entry in the database which does not have a sequence assigned.

The *Entry edit* window of the entry appears.

7.12.2 Click on the  button next to the sequence type (e.g. SSU-ribo or another name you entered).

The program now asks **"The experiment "SSU-ribo" is not defined for this entry. Do you want to create a new one?"**.

7.12.3 By answering **<Yes>**, the program will create a new empty sequence that is linked to this entry. The experiment card for the sequence type of this entry appears: a small empty window (see Figure 52; see also 8.2).

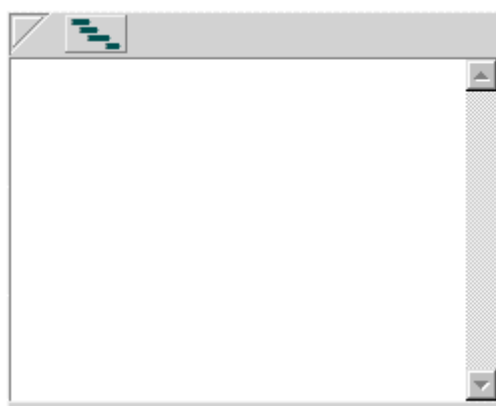


Figure 52. Empty sequence experiment card with button to launch GeneBuilder.

7.12.4 It is now possible to simply paste a sequence in this window; however, pressing the  button will launch the GeneBuilder contig assembler program to assemble a contig sequence from a series of partial sequencing experiments for this entry (see also 8.3).

The GeneBuilder main window, initially empty, looks as in Figure 53.

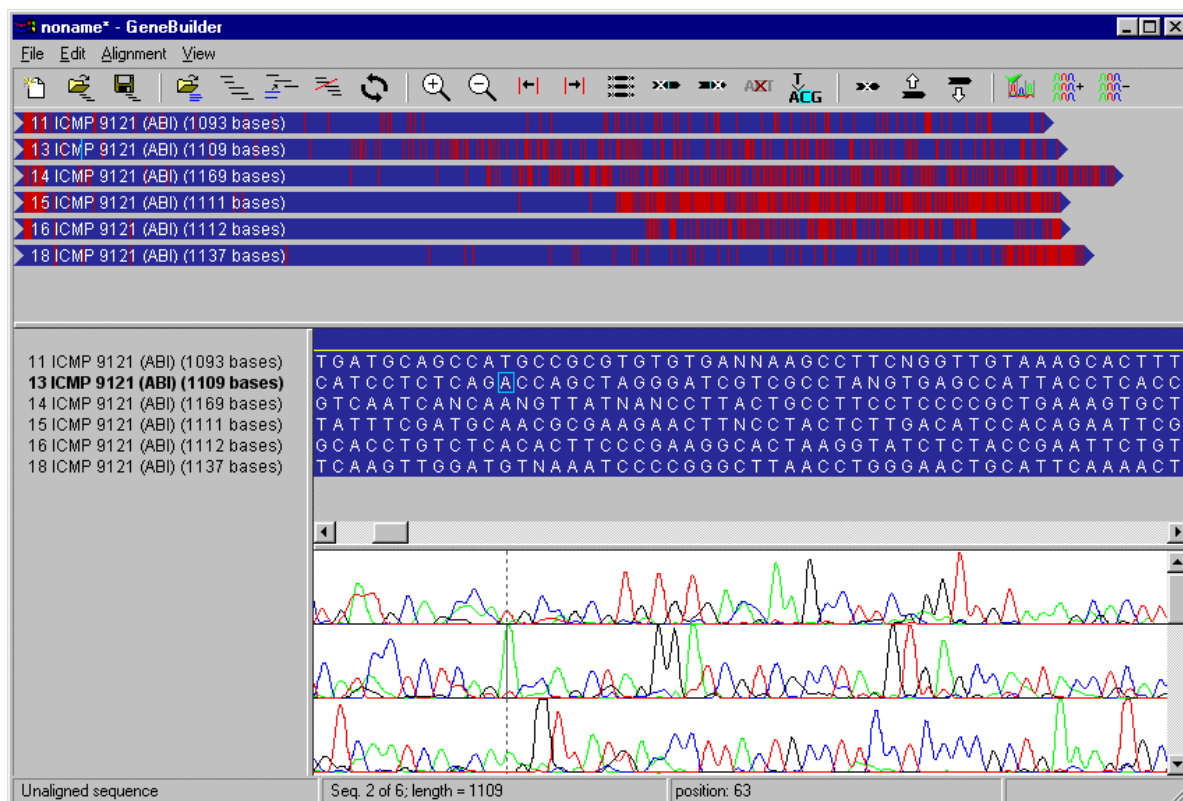


Figure 53. The GeneBuilder main window.

7.12.5 Select **File | Import sequences from file** or .

A set of partial 16S-rRNA sequences from *Xanthomonas* strain ICMP 9121² run on an ABI 370 machine are provided on the installation CD-ROM by selecting all the files under **Examples\Seqassem**.

7.12.6 Select all sequences **11 ICMP 9121 ... 18 ICMP 9121** under **Examples\Seqassem** and press **<Open>**.

The six partial sequences are now shown in the GeneBuilder main window as in as in Figure 53.

The top panel shows the sequences in a graphical representation. The direction corresponds to the direction of the sequences: this can be either unchanged (pointing to the right) or inverted and complemented (pointing to the left). Before an alignment is done, all sequence directions are unchanged. Red vertical lines through a sequence indicate uncertain base positions. A bright blue vertical line indicates the sequence and its position selected.

7.12.7 You can zoom in on the graphical overview panel with **View | Zoom in** or .

7.12.8 To zoom out on the graphical overview panel, use **View | Zoom out** or .

The central panel shows the sequences at base level. When multiple alignment is done, the aligned bases are shown. A bright blue rectangle indicates which position on which sequence being selected.

² Hauben L., L. Vauterin, J. Swings, and E.R.B. Moore. 1997. Int. J. Syst. Evol. Microbiol. 47: 328-335.

The *bottom panel* displays the densitometric curves in four colors for each of the partial sequences. A black vertical line indicates which position on which sequence is being selected. The line is dashed through non-selected sequences and full through the selected sequence. By default, the curves are normalized (stretched or shrunk) so that each peak matches the corresponding base in the written sequence (central panel).

7.12.9 You can toggle between normalized and original display mode by selecting *View|Toggle normalization* or .

7.12.10 It is possible to increase the number of curves shown in this panel with *View|Increase number of curves* or .

7.12.11 To decrease the number of curves shown simultaneously, select *View|Decrease number of curves* or .

7.12.12 The colors for the four bases can be chosen by selecting *View|Customize colors*. The dialog box allows you also to change colors for non-aligned sequences, aligned sequences, text (written sequences), mismatches, corrected bases, and the selection cursors in the different panels.

A small *left panel* shows a list of the file names of the partial sequences in the assembly project, with the currently selected sequence in bold.

When you click on any position in one of these panels (except the list of sequence file names), all other panels are automatically brought to the same focus.

Before we actually align the sequences, we need to cut out the bad parts, i.e. the parts from the curves with unreliable signal or no signal at all.

7.12.13 Identify for each sequence where it becomes reliable and click on the first reliable position on the curve set.

7.12.14 Select *Edit|Cut left from cursor* or .

7.12.15 Repeat the same at the end of each sequence, selecting the last reliable position on the curve set, and *Edit|Cut right from cursor* or .

7.12.16 As an alternative to manually cleaning the partial sequences, the function *Edit|Auto trim sequences* or  automatically cuts bad parts from the sequences, according to minimal quality parameters which you can specify (see 7.12.20 below).

7.12.17 To restore a sequence which has been trimmed incorrectly, use *Edit|Reload sequence*.

7.12.18 To change a base in a partial sequence, place the cursor on the base and type the base, which can be A, C, T, G or any IUPAC code for ambiguous positions.

7.12.19 If you wish to insert a base at a certain position, you can toggle between overwrite and insert mode by selecting *Edit|Toggle overwrite/insert mode* or .

Before aligning the sequences, we will describe the various settings that apply to the alignment and the generation of the consensus sequence.

7.12.20 Select *Alignment* | *Set parameters*. This will result in the *Alignment parameters* dialog box (Figure 54).

Section	Parameter	Value
Alignment parameters	Minimum score	500
	Open gap cost	100
	Unit gap cost	20
Optimization parameters	Number of diagonals	100
	Wordsize	4
Autocorrection parameters	Accept level	50
	Confidence level	20
Auto clean sequences	Percentage	10
	Averaging length	10

Figure 54. Alignment, consensus and 'auto clean sequence' parameters.

Under **Alignment parameters**, the *Minimum score* determines the number of matching bases that need to be found for two sequences to be aligned. Mismatches and gaps are subtracted from this score with a value as specified in the same dialog box. The *Open gap cost* is the cost to create a gap between two sequences. The *Unit gap cost* is the cost to increase an existing gap between two sequences. Default parameters are 500% as minimum score, 100% to open a gap and 20% to increase a gap.

The **Optimization parameters** define how accurate the algorithm should work. In most cases, the default parameters will work fine, but in case of large sequences with many overlapping regions and/or many uncertain positions. *Number of diagonals* relates to the alignment technique that is used, i.e. an fast algorithm based upon Needleman and Wunsch (1970)³. The number of gaps the algorithm can create is proportional to the number of diagonals specified. The larger the number, the more accurate but the slower the calculations. The suggested default setting is 100 diagonals. The *Wordsize* also relates to the same algorithm, which creates a lookup table of groups of bases to accelerate the alignment. Up to a certain extent, the smaller the wordsize is taken, the more accurate but the slower the alignment. However, if the wordsize is set too small (e.g. the minimum of 2), the alignment may become less correct unless the number of diagonals is extremely high. The suggested default setting is a wordsize of 4.

The **Autocorrection parameters** determine how ambiguous positions in the contig sequence will be derived from aligned partial sequences, i.e. for those bases where there is no consensus. *Accept level* determines the minimum percentage of aligned sequences that need to have the same base in order for that base to be written in the consensus sequence. The *Confidence level* determines the minimum percentage difference between the highest number and the second highest number of identical bases. Consider the example, where 5 partial sequences together represent a part of the contig. At a certain position 3 subsequences show 'A' and two others show 'G'. This means that 60% is 'A' and 40% is 'G'. the difference is 20%. With an *Accept level* of 50% and a *Confidence level* of 20%, 'A' will be accepted in the consensus sequence. If the *Confidence level* would be 25%, 'R' would be written into the consensus sequence.

³ Needleman, S., and C. Wunsch. 1970. J. Mol. Biol. 48:443-453.

Whenever the program decides to write a certain base into the consensus based upon the majority rule as specified, it writes this base on a blue background, to show that a decision has been made based upon a discrepancy in the partial sequences. When no consensus can be found with the specified parameters, the IUPAC notation is used to represent the position, and the background is shown in red.

The **Auto clean sequences** parameters specify the maximal *Percentage* of ambiguous positions allowed in a certain region of a partial sequence. If the region has a higher number of ambiguous positions, it will be cut from the sequence in the "*Auto trim sequences*" function. The *Averaging length* is the number of bases which are taken together to determine the percentage of ambiguous positions.

7.12.21 Press <OK> or <Cancel> to exit the *Alignment parameters* dialog box.

7.12.22 Select *Alignment | Align all sequences* or  to align the sequences.

When the alignment is finished, the upper schematic panel shows how the partial sequences are aligned, and also the directions (unchanged or inverted-complemented) are indicated. Problems are shown as blue or red vertical lines: a blue line means that there is no consensus among the partial sequences but that the program has decided upon a majority criterion, and a red line means that the position is unknown (see also **Autocorreccion parameters** above). The central panel shows the aligned sequences written in full, and the consensus sequence is shown on top of the panel. The same colors are used to display ambiguous positions.

7.12.23 To change a base in a partial sequence, place the cursor on the base and type the base, which can be A, G, T, C or any IUPAC code for ambiguous positions.

7.12.24 If you wish to insert a base at a certain position, you can toggle between overwrite and insert mode by selecting *Edit | Toggle overwrite/insert mode* or .

7.12.25 To delete a base in a sequence, select *Edit | Delete base* or press the DEL key.

7.12.26 To change a base in the consensus sequence, place the cursor on the corresponding base of any of the partial sequences, and hold the CTRL key while typing the base, which can be A, G, T, C or any IUPAC code for ambiguous positions.

7.12.27 To check the consensus sequence for correctness, you can let the program jump to each next problem position using *View | Go to next problem* or  (or using the shortcut CTRL+Right arrow key).

7.12.28 To jump to the previous problem, use *View | Go to pervious problem* or  (or using the shortcut CTRL+Left arrow key).

7.12.29 More specifically, to go to the next mismatch, use *View | Go to next mismatch* or the function key F3.

7.12.30 To go to the previous mismatch, use *View | Go to previous mismatch* or SHIFT+F3.

7.12.31 To go to the next gap, use *View | Go to next gap* or F4.

7.12.32 To go to the previous gap, use *View | Go to previous gap* or SHIFT+F4.

7.12.33 In case the program has incorrectly aligned two sequences, you can place the cursor within the aligned region and select *Align | Break alignment*.

7.12.34 New sequences can be added at any moment to the existing alignment project with *File|Import sequences from file* or *File|Import sequences from clipboard*, and subsequently selecting *Align|Align incremental*.

Although GeneBuilder automatically inverts and complements subsequences wherever necessary to obtain the consensus sequence, the program cannot know the correct orientation of the consensus sequence. Hence, it may be necessary to invert and complement the consensus sequence before entering it into the database.

7.12.35 To invert-complement the consensus sequence, select *Align|Invert and complement group* or .

NOTE: In case the program couldn't find one single consensus for all subsequences, two or more groups will exist. Therefore you will need to select a group of aligned subsequences before executing the invert-complement function.

7.12.36 When the aligned sequences are ready for importing in the sequence database, select *File|Save project*, *File|Save project as*, or .

7.12.37 A project can be opened within GeneBuilder using *File|Open project* or .

7.12.38 From a sequence in Fingerprinting II Informatix assembled using GeneBuilder, the project can be opened in GeneBuilder pressing the  button in the sequence experiment card (7.12.4). Such projects can be changed at any time and are updated automatically in the Fingerprinting II Informatix database.

7.13 Defining a new matrix type

7.13.1 Select *Experiments|Create new matrix type* from the main menu, or press  and *New matrix type*.

7.13.2 Enter a name for the new type. Enter a name, for example “DNA-homol”.

Press the <OK> button to complete the setup of the new matrix type. It is now listed under **Matrix types** in the experiment type panel.

Unlike other experiments, a matrix type does not provide an experiment for each entry. Instead, it contains *similarities between entries*. Hence, the “data file” which contains the experiment data, and the “entry file” which links the experiments to database entries are the same here. There are two ways to enter similarity values: by importing a matrix as a whole, and by entering the values from the keyboard.

To import a matrix, it must have the following format:

```
ENTRY KEY<tab>VALUE<eol>
ENTRY KEY<tab>VALUE<tab>VALUE<eol>
ENTRY KEY<tab>VALUE<tab>VALUE<tab>VALUE<eol>
Etc.
```

<eol> means “end of line”, a simple return in MS-DOS text, which corresponds to ASCII character #13 followed by ASCII character #10.

Matrix files can be imported by selecting the matrix type in the experiment types panel, and *File|Import experiment file*. The program compares the entry keys as provided in the import file with the entry keys in the database and assigns values to the corresponding keys. If entry keys are not found in the database, it will automatically create new database entries.

To enter similarity values manually, you first have to select the entries in the database for which you want to create a matrix.

7.13.3 Select some entries in the database by holding the CTRL key and left-clicking (see further, paragraph 9.2). Selected entries are marked with a blue arrow.

7.13.4 Double click on the file **DNA homol** in the **Files** panel (not in the **Experiment types** panel). This opens the Matrix file window (Figure 55).

The diagonals, i.e. the similarity values of the entries with themselves, are filled in already and cannot be changed.

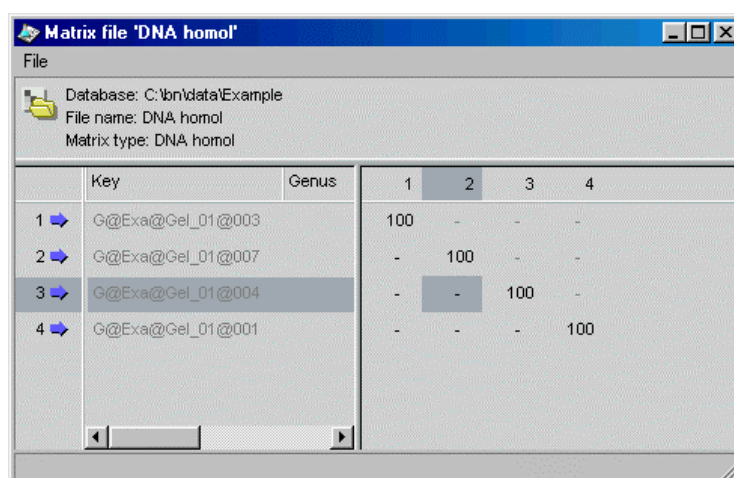


Figure 55. The Matrix file window to enter and edit similarity values.

7.13.5 To enter a value, press Enter or double click on a field.

7.13.6 When finished, exit the window with *File|Exit*.

8. Experiment display and edit functions

In paragraph 6.4, we have explained how you can edit the information fields for each database entry by double clicking on the entry (6.4.1), which pops up the *Entry edit* window. It is possible to enter and view experiment data directly from the entry edit window.

In order to explain the edit functions, we will use the DemoBase database.

8.1.1 Exit the main program

8.1.2 Back in the Startup screen, select DemoBase and start Analyze again.

8.2 The experiment card

8.2.1 If we open the entry edit window for any database entry (except a standard) here, the window lists all available experiment types for this entry, each of which contains two buttons (Figure 56) .

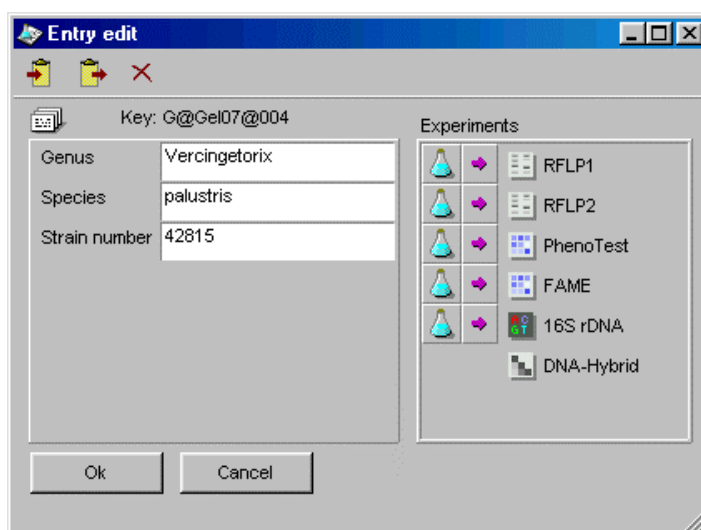


Figure 56. Entry edit window.

8.2.2 With the  button, you can display the *Experiment card* of an experiment (Figure 57).

An Experiment card can also be opened from the main window, by clicking on the green dot that indicates that an experiment is present for an entry (see 6.1).

These experiment cards always remain in front of any window regardless of whether it is a Fingerprinting II Informatix window or not.

- 8.2.3 You can move the experiment card by clicking and holding the left mouse button on the card, and then dragging it to its new position. For sequence experiment cards, move the window in the caption.
- 8.2.4 When you hover over the image card with the mouse, a small tag displays additional information. In case of a fingerprint, it shows the key of the entry, and the gel name and lane number. In case of a character type shown as a plate it shows the key of the entry, and the name and the value of the character being pointed to.
- 8.2.5 Close an experiment card by clicking in the small triangle-shaped button in the left upper corner.

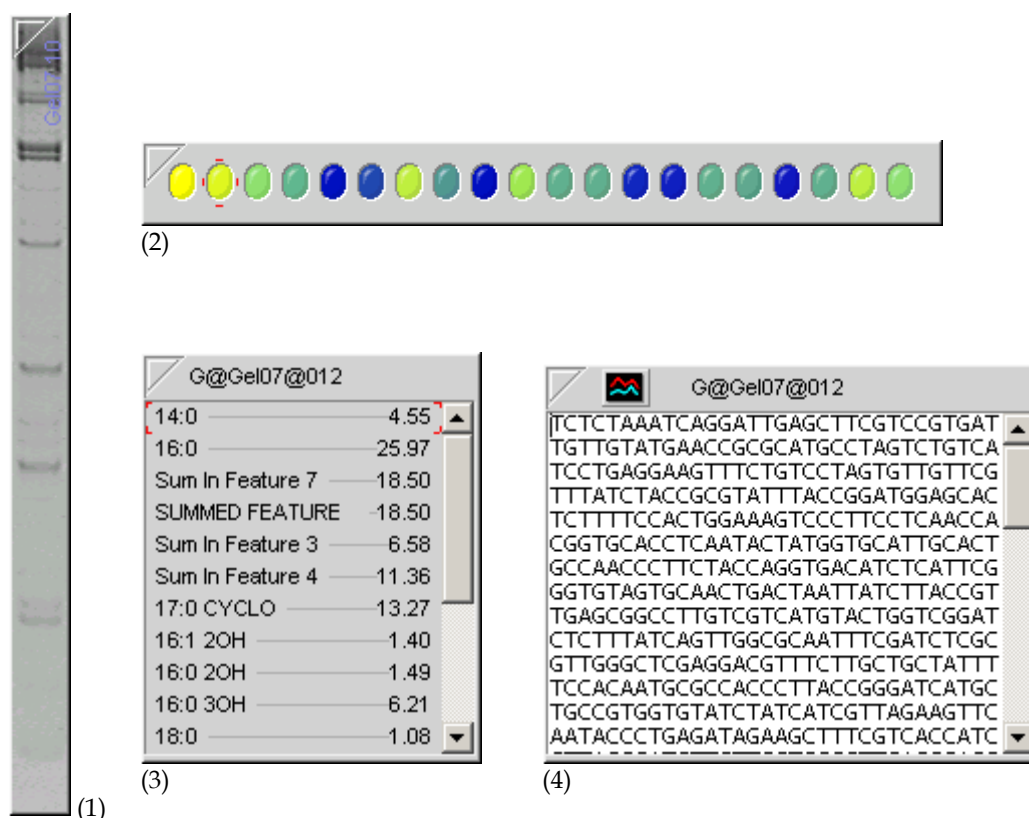


Figure 57. Experiment cards of fingerprint type (1), character type with fixed number of characters (closed type) (2), character type with non-fixed number of characters (open type) (3), and sequence type (4) .

You can open an experiment card for an entry, close its entry edit window, and then show the corresponding experiment card for another entry, to arrange and compare them side by side. Only the screen size will be the limiting factor as to the number of experiment cards that can be shown together.

- 8.2.6 In case of a fingerprint type experiment card, you can increase or decrease the size of the card using the keyboard, by pressing the **numerical + key** (increase) or the **numerical - key** (decrease). You can right-click on the experiment card to pop up a floating menu, from which you can *Export normalized curve*, *Export normalized band positions*, and *Export band metrics*.
- 8.2.7 In case of a character type, you can right-click on the experiment card to pop up a floating menu, from which you can call the character image import program BNIMA (*Edit image*),

copy the data set to the clipboard (*Copy data to clipboard*), or paste data from the clipboard into the experiment (*Paste data from clipboard*). *Export character values* creates a similar output, but provides the names of the characters in case of an *open character set* (see 7.8).

8.3 Entering experiment data via the experiment card

For character types and sequence types, it is possible to enter the experiment data directly on the experiment card.

8.3.1 Open the entry edit window. In this example, we will open the entry edit window for a STANDARD.

For experiments that are not available for the entry, an empty flask is shown: .

Character types

Depending on whether the character type is closed (defined number of tests) or open (undefined number of tests), the input method is different.

8.3.2 Click the  button of an empty character type; in the example, we choose PhenoTest, which is a closed character type.

A message displays “The experiment ‘PhenoTest’ is not defined for this entry. Do you wish to create a new one?”.

8.3.3 Answer <Yes> to this question. An empty experiment card appears.

8.3.4 In case of binary (plus or minus) data, you can enter the values using the **numerical + and - keys**. The cursor automatically jumps to the next test if you have entered a value.

8.3.5 You can move the cursor using the **Left and Right arrow keys**.

NOTE: if you use the + and - keys to enter non-binary data, the defined maximum for the character type is used if + is entered.

8.3.6 In case of non-binary values (real or integer values), each test can be varied continuously between the minimum and the maximum using the **PgUp key** (increase intensity) and the **PgDn key** (decrease intensity).

8.3.7 Press the close button of the experiment card. The program asks to save the changes made.

8.3.8 Open an entry edit window of an entry which contains a FAME experiment (shown as a green dot in the experiment presence panel).

8.3.9 In the entry edit window, press the  button of FAME. The card lists all fatty acids that are present in this entry, as percentages.

8.3.10 Double click on a fatty acid to change its value. A dialog box prompts to “Enter a value of character XXX”.

8.3.11 Press <Cancel> if you do not wish to modify the data of the demobase.

8.3.12 At the bottom of the list, the last line shows <Add new character>. If you double click on this line, a dialog box shows all known characters for this character type which are not yet available in this entry, from which you can select one.

8.3.13 Press the <Create new> button to create a new character.

Sequence types

8.3.14 Press the  button of an empty sequence type; in the example, 16S rDNA.

8.3.15 An empty sequence editor appears. You can enter bases or amino acids manually, or by pasting from the clipboard (SHIFT + INS).

8.3.16 When the sequence has been generated using the Fingerprinting II Informatix contig assembler program "GeneBuilder", pressing the  button will launch GeneBuilder with the contig project associated with this sequence.

8.3.17 When no contig project is available for this entry, pressing the  button will launch GeneBuilder with a new project associated. See 7.12 to work with GeneBuilder.

8.4 Entering experiment data via the experiment file

8.4.1 Using the  button, you can open the *Experiment entry file* window, with the entry selected. With *File|Edit fingerprint data* () , *File|Edit character data* or *File|Edit sequence data*, you can edit the experiment data for the entry directly in the file.

9. Comparison functions

9.1 Definition

A *Comparison* in Fingerprinting II Informatix includes every function to compare multiple database entries. This involves displaying experiment images of selected entries, calculating and showing cluster analyses, aligning sequences, and calculating principal component analysis (PCA) and multi-dimensional scaling (MDS).

The *Comparison window* in Fingerprinting II Informatix presents a comprehensive overview of all available experiments for a selection of entries and enables the user to show and compare any combination of images of experiments. A comparison window is always associated with a selection of entries from the database. To select entries, several search and selection functions are available.

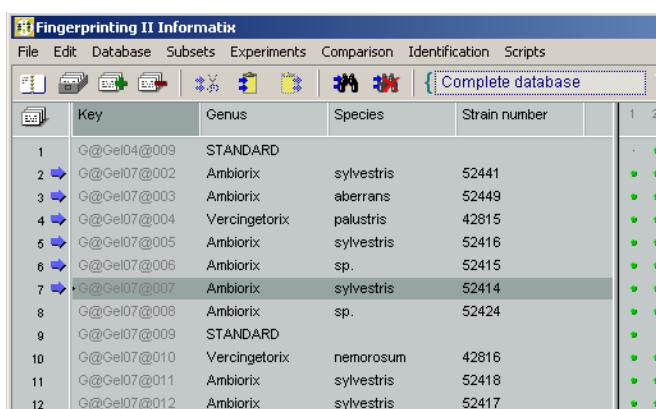
In order to explain the comparison functions, we will use the DemoBase database. If the DemoBase is not the current database loaded, execute the following two steps.

9.1.1 Exit the main program

9.1.2 Back in the Startup screen, select DemoBase and start Analyze again.

9.2 Manual selection functions

A single entry can be selected by holding the CTRL key and left-clicking. Selected entries are marked by a blue arrow (Figure 58). Selected entries are unselected in the same way.



	Key	Genus	Species	Strain number	1	2
1	G@Gel04@009	STANDARD				
2	G@Gel07@002	Ambiorix	sylvestris	52441		
3	G@Gel07@003	Ambiorix	aberrans	52449		
4	G@Gel07@004	Vercingetorix	palustris	42815		
5	G@Gel07@005	Ambiorix	sylvestris	52416		
6	G@Gel07@006	Ambiorix	sp.	52415		
7	G@Gel07@007	Ambiorix	sylvestris	52414		
8	G@Gel07@008	Ambiorix	sp.	52424		
9	G@Gel07@009	STANDARD				
10	G@Gel07@010	Vercingetorix	nemorosum	42816		
11	G@Gel07@011	Ambiorix	sylvestris	52418		
12	G@Gel07@012	Ambiorix	sylvestris	52417		

Figure 58. Database panel in the main program, showing selected entries (blue arrows).

9.2.1 Select the first non-standard lane (CTRL + mouse click). The entry is now marked by a blue arrow.

9.2.2 In order to select a group of entries, hold the SHIFT key and click on another entry.

- 9.2.3 If you wish to select entries using the keyboard, you can scroll through the database using the Up/Down arrow keys, and select or unselect entries using the space bar.
- 9.2.4 A single entry can be selected or unselected from its entry edit window (6.4) using the  button. When the entry is selected, this button shows as .
- 9.2.5 To make viewing of selected entries easier in a large database, you can bring all selected entries to the top of the list with *Edit | Bring selected entries to top*.
- 9.2.6 Clear all selected entries with *Edit | Clear selection list* (F4 key) or .

9.3 Automatic search and select functions

Besides the manual selection functions as described above, Fingerprinting II Informatix possesses more advanced database search functions.

- 9.3.1 Select *Edit | Search entries* (F3) or . This pops up the *Entry search* dialog box (Figure 59).

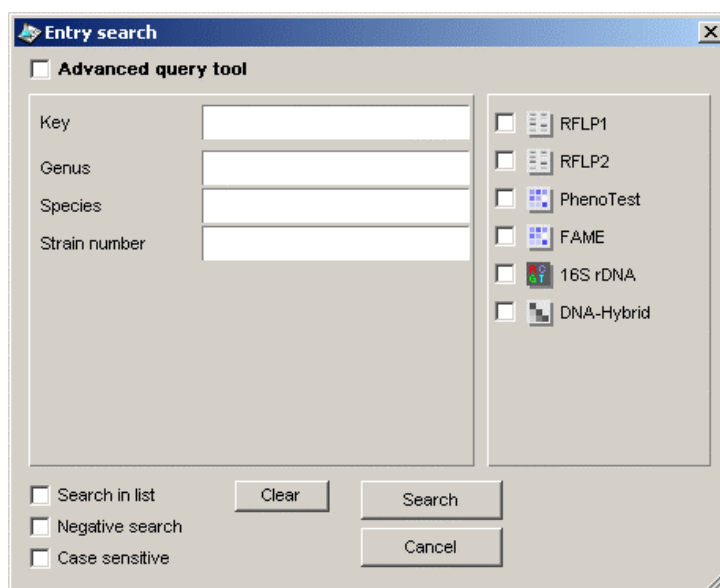


Figure 59. Entry search dialog box.

You can enter a specific search string for each of the database fields defined in the database (left panel). Wildcards can be used to search for substrings: an **asterisk** * replaces any range of characters in the beginning or the end of a string, whereas a **question mark** ? replaces one single character.

It is also possible to search for all entries that contain a certain experiment (right panel). Both the string search and the experiment search can be combined.

Normally, successive searches are additive: new searches are added to the selection list. The *Search in list* checkbox allows you to refine the search within a list of selected entries.

With *Negative search*, all entries that do not match the specified criteria will be selected.

Case sensitive lets the program make a distinction between uppercase and lowercase.

The <Clear> button clears all entered search criteria.

9.3.2 As an example, enter *L* in the *Species* field.

9.3.3 Press <Search>. All entries having a L in their species name are selected: *Ambiorix sylvestris* and *Vercingetorix palustris*.

9.3.4 Call the entry search dialog box again, and press the <Clear> button.

9.3.5 Enter STANDARD in the Genus field, and check the *Negative search* checkbox.

9.3.6 Press <Search> to select all database entries, except the entries used as standard lanes in the RFLP techniques.

9.3.7 Clear the selection with the F4 key or click the  button (*Edit | Clear selection list*) .

9.4 The advanced query tool

Fingerprinting II Informatix contains an advanced query tool that allows searches of any complexity to be made within the database, based on information fields and experiment data.

9.4.1 Call the query tool again, by selecting *Edit > Search entries* or pressing .

The *Entry search dialog box* (Figure 59) contains a checkbox *Advanced query tool*.

9.4.2 Check *Advanced query tool*. The normal search window changes into the *Advanced query tool* (Figure 60).

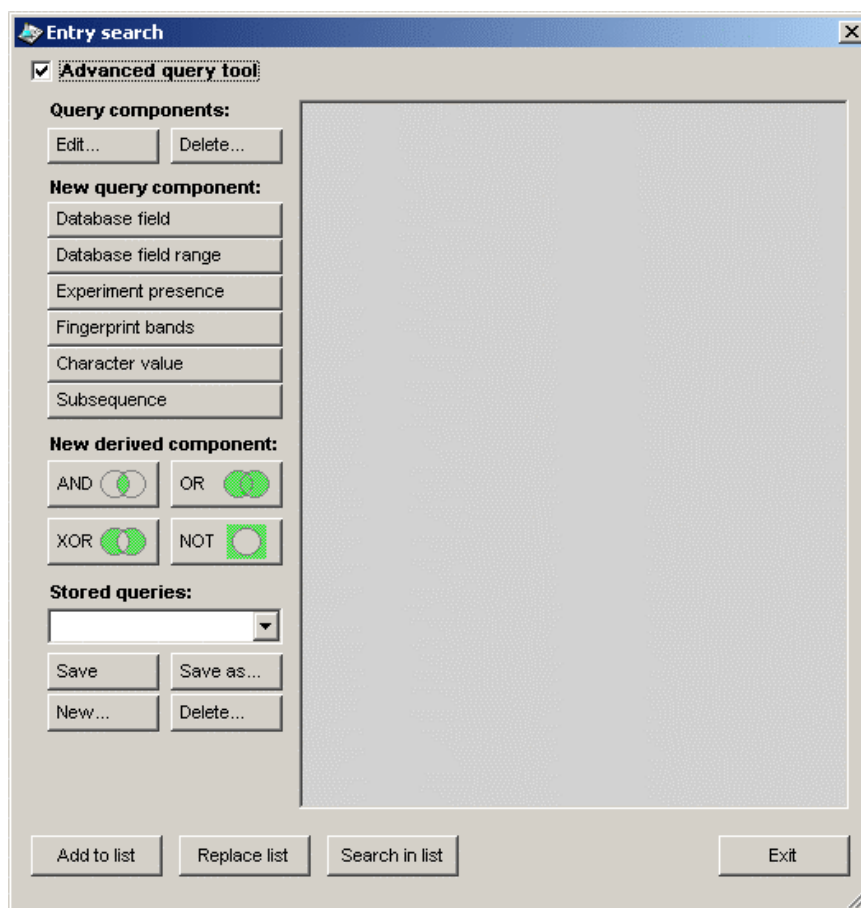


Figure 60. The advanced query tool.

The advanced query tool allows you to create individual *query components*, which can be combined with *logical operators*. The available targets for query components are *Database field*, *Database field range*, *Experiment presence*, *Fingerprint bands*, *Character value*, and *Subsequence*.

Database field

Using this component button, you can enter a (sub)string to find in any database field (<Any field>) or in any specific field that exists in the database (Figure 61). Note that the wildcards * and ? are not used in the advanced query tool.

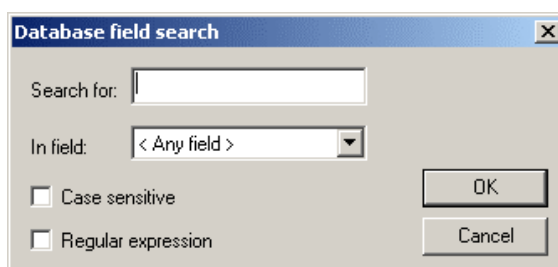


Figure 61. Database field search component dialog box.

The search component can be specified to be *Case sensitive* or not. In addition, a search string can be entered as a regular expression (see 25.2).

Database field range

Using this component button, you can search for database field data within a specific range, which can be alphabetical or numerical. Specify a database field, and enter the start and the end of the range in the respective input boxes (Figure 62). A range should be specified with the lower string or value first. Note that, when only one of both limits is entered, the program will accept all strings above or below that limit, depending on which limit was entered. For example, when only the first (lower) limit of the range is entered and the upper limit is left blank, all strings (values) *above* the specified string (value) will be accepted.

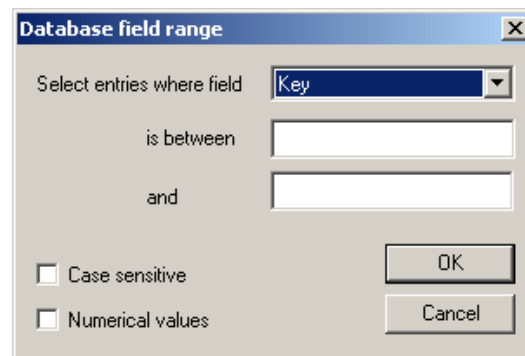


Figure 62. Database field range component dialog box.

The search component can be specified to be *Case sensitive* or not. When *Numerical values* is checked, the search component will look only for numerical values and ignore any other characters.

Experiment presence

With this search component, you can specify an experiment to be present in order for entries to be selected.

Fingerprint bands

The Fingerprint bands search component allows specific combinations of bands to be found in the database entries. The dialog box that pops up (Figure 63) allows you to enter a *Fingerprint experiment type*, and specify an **Intensity filter**, a **Target range**, and a **Number of bands present**.

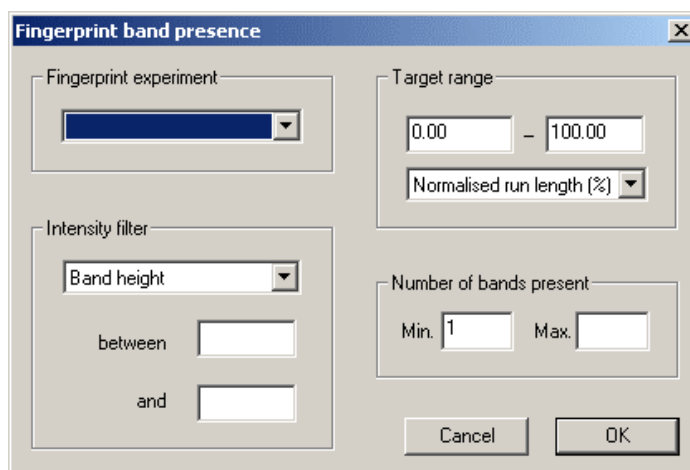


Figure 63. Fingerprint bands presence component dialog box.

Under **Intensity filter**, you can choose which intensity parameter to be used: *Band height*, *Band surface* or *Relative band surface*. When a 2-D quantification analysis is done, you can also choose *Volume*, *Relative volume* or *Concentration*. A range should always be specified with the lower value first. Note that, when only one of both limits is entered, the program will consider all bands above or below that limit, depending on which limit was entered. For example, when only the first (lower) limit is entered and the upper limit is left blank, all bands *above* the specified intensity will be accepted. When both fields are left blank, no intensity range will be looked for, i.e. all bands will be considered.

Under **Target range**, you can search for bands with specific sizes, either entered as *Normalized run length (%)* or as *Metric values*. A target range should always be entered with the lower value first. Note that, when only one of both limits is entered, the program will consider all bands above or below that limit, depending on which limit was entered. For example, when only the first (lower) limit is entered and the upper limit is left blank, all bands *above* the specified size will be accepted. When both fields are left blank, no size range will be looked for, i.e. all bands will be considered.

Under **Number of bands present**, you can enter a minimum and a maximum number of bands the patterns should contain. Note that, when only one of both limits is entered, the program will consider all patterns with band numbers above or below that limit, depending on which limit was entered. For example, when only the first (lower) limit is entered and the upper limit is left blank, all patterns having *at least* the specified number of bands will be accepted. At least one of both limits must be entered.

Character value

With the Character value component, you can search for characters within certain ranges. You should select a Character experiment type, specify a character or **<All>** characters, and enter a maximum and minimum value. A range should always be specified with the lower value first. Note that, when only one of both limits is entered, the program will consider all characters above or below that limit, depending on which limit was entered. For example, when only the first (lower) limit is entered and the upper limit is left blank, all characters with values *above* the specified value will be accepted.

Subsequence

With the Subsequence component you can perform a search for a specific subsequence in a sequence type experiment. The sequence type experiment should be chosen, and a subsequence entered. A mismatch tolerance can be specified with *Maximum number of mismatches*.

Logical operators

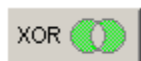
NOT, operates on one component. When a component is combined with NOT, the condition of the component will be inverted.



AND, combines two or more components. All conditions of the combined components should be fulfilled at the same time for a sequence to be selected.



OR, combines two or more components. The condition implied by at least one of the combined components should be fulfilled for a sequence to be selected.



XOR, combines two or more components. Exactly one condition from the combined components should be fulfilled for a sequence to be selected.

NOTE: the buttons for the logical operators contain a helpful Venn diagram icon that clearly explains the function of the operator.

To create a search component, you can select to search in the database fields, fingerprint bands, characters, and sequences. As an example, we will select all entries from the genus *Ambiorix* that have no RFLP1 bands in the range 71-72 basepairs, and of which the 16S ribosomal RNA sequences contains a subsequence "TGGTGCATTG".

9.4.3 Press **<Database field>**. In the box that appears you can choose the genus field or leave **<Any field>**.

9.4.4 Enter "**ambiorix**" and press **<OK>**.

A query component now appears in the right panel, stating "**Database field:** Search 'ambiorix' in field 'Genus'".

9.4.5 Press **<Fingerprint bands>**. The Fingerprint bands presence component dialog box appears (Figure 63).

9.4.6 Select **RFLP1** from the *Fingerprint experiment* pull-down list.

9.4.7 Under *Target range*, enter 71 - 72, and specify **Metric values**.

9.4.8 Press **<OK>**. A second component appears in the query window, saying "**Fingerprint bands:** 'RFLP1' has at least 1 bands in the range 71.00 - 72.00".

9.4.9 Select this **Fingerprint bands** component by clicking on it (purple highlighted when selected), and press the **<NOT>** button .

9.4.10 Select the first component by clicking on it.

9.4.11 Hold down the CTRL key and click on the **NOT** box resulting from the second component to select it together with the first one.

9.4.12 Press the **<AND>**  button to combine the created components with AND.

9.4.13 Press **<Subsequence>**. This box allows you to type or paste a sequence that will be searched for.

9.4.14 Type **TGGTGCATTG** in the input field, and select **16S rDNA**. Press **<OK>**.

A third query component appears in the right panel, stating “**Subsequence:** Search ‘TGGTGCATTG’ in the sequence ‘16S rDNA’”. We will now combine the resulting AND box from the first two components with this last component, using an AND operator, to restrict the selection to those sequences that fulfill the ‘Ambiorix’ and RFLP1 conditions *AND* contain the specified subsequence.

9.4.15 Select the **AND** box by clicking on it.

9.4.16 Hold down the CTRL key and click on the **Subsequence** component to select it together with the **AND** box.

As both components are not selected, we can combine them with a *logical operator*.

9.4.17 Press the **<AND>** button to combine the created components with AND.

This is now shown graphically in the right panel (Figure 64). Note that:

- Individual components can be re-edited at any time by double-clicking on the component or by selecting them and pressing **<Edit>**.
- Selected components can be deleted with **<Delete>**.
- The result of a logical operator as obtained in this example can be combined again with other components (or logical operators) to construct more complex queries.
- Queries can be saved with **<Save>** or **<Save as>**.
- Saved queries can be loaded using the pull-down listbox under *Stored queries*.
- Existing queries can be removed with loading them first and pressing **<Delete>**.

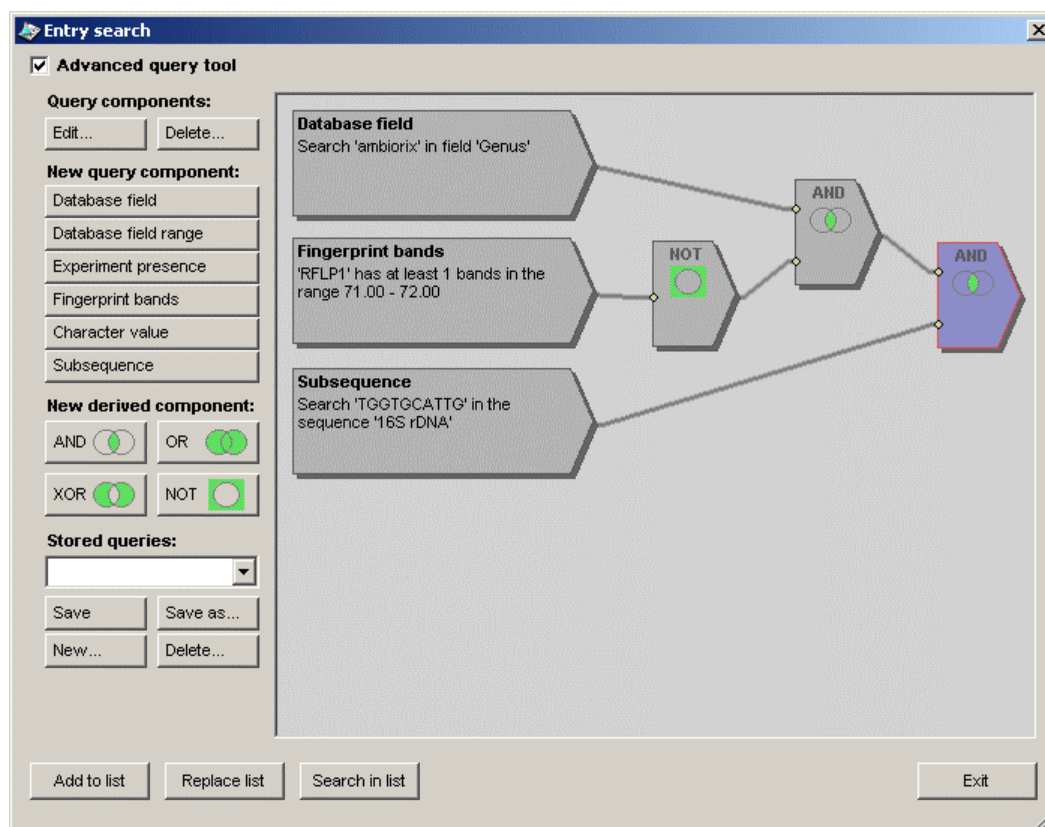


Figure 64. Combined query constructed in the *Advanced query tool* (see text for explanation).

9.4.18 To view the selected entries, press <Add to list>.

The entries that were found are highlighted with a blue arrow left from them.

NOTES:

- (1) In order to speed up the search function in case of large databases, it is important to know that searching through the database fields is extremely quick, while searching through sequences or large character sets can be much slower. Using the AND operator, it is always recommended to define the quickest search component as the first, since the searching algorithm will first screen this first component and subsequently screen for the second component on the subset that match the first component.
- (2) When combined with a logical operator, query components contain a small node at the place where they are connected to the logical operator box (AND, OR, XOR). By dragging this node up or down, you can switch the order of the query components, thus making it possible to move the most efficient component to the top in AND combinations, as explained above under (1).
- (3) Multiple components/operators can also be selected together by dragging the mouse over the boxes in the right panel.

9.5 Subsets

A selection of entries from the database can be saved as a *subset*. Subsets can include a certain target group in a database, for example, a single species in a database containing many species, or any selection of relevant strains for a certain purpose. Selecting the defined subset displays a view of the database containing only the entries of the subset. Search functions, copy and select functions will be restricted only to the displayed subset, and new comparisons, when created, will only contain the selected entries from the subset.

9.5.1 In database **Demobase**, make sure no entries are selected using *Edit | Clear selection list* (F4 key) or .

9.5.2 Selecting *Edit | Search entries* or press .

9.5.3 In the Entry search dialog box, enter “**Ambiorix**” under **Genus**, and press <OK>.

All *Ambiorix* entries are now selected. When we create a new subset, the selected entries will be automatically placed in the subset.

9.5.4 Select *Subsets | Create new* or press . Alternatively, you can also click on the  button which will drop down a list of currently defined subsets (initially empty), and an option <Create new subset>. Selecting this option has the same effect.

9.5.5 Enter a name for the subset, e.g. the name of the selected genus “**Ambiorix**”.

The created subset is now displayed, and the name of the current subset is displayed in the subset selector button .

9.5.6 Selecting the complete database or another subset, when available, can be done by pressing  and selecting *Complete database* or the other subset in the list.

Once a subset exists, it remains possible to add or remove entries, using the copy and paste functions. The following example will illustrate this.

9.5.7 Select subset *Ambiorix* from the subset selector button.

9.5.8 We want to remove all the “sp.” entries from this subset. Clear any selected entries and select the 3 “sp.” entries by manual selection or using the search function.

9.5.9 Press  or select *Edit | Cut selection* to cut the selected entries from the current subset.

9.5.10 We can place them in a new subset by pressing , entering a name, e.g. “**Unknowns**” and in this new subset, pressing  or selecting *Edit | Paste selection*.

9.5.11 If you want to copy entries from a subset to another subset, without removing them from that first subset, there is also a command *Edit | Copy selection* or .

NOTES:

- (1) A selection that is copied or cut from a subset or copied from the database is placed on the Windows clipboard as the keys of the selected entries, separated by line breaks. You can paste them in other software when desired.
- (2) The commands *Cut selection*, *Paste selection* and *Delete selection* are not available in the **Complete database**.

When you want to remove entries from a subset without overwriting the contents of the clipboard, you can use the command *Edit | Delete selection*.

9.5.12 The current subset can be renamed using *Edit | Rename current*.

9.5.13 The current subset can be deleted using *Edit | Delete current* or .

9.6 Pairwise comparison between two entries

From within any window where you can select entries, you can display a detailed comparison between two entries. This pairwise comparison includes all the images of the experiment types as well as the similarities obtained using the specified coefficients.

9.6.1 In the main window, clear any selection with F4.

9.6.2 Select any two entries you want to compare.

9.6.3 In the main menu, select *Comparison | Compare two entries* or use the **CTRL+2** shortcut (numerical 2).

The CTRL+2 shortcut works from within any window. The *Pairwise comparison* window appears (Figure 65).

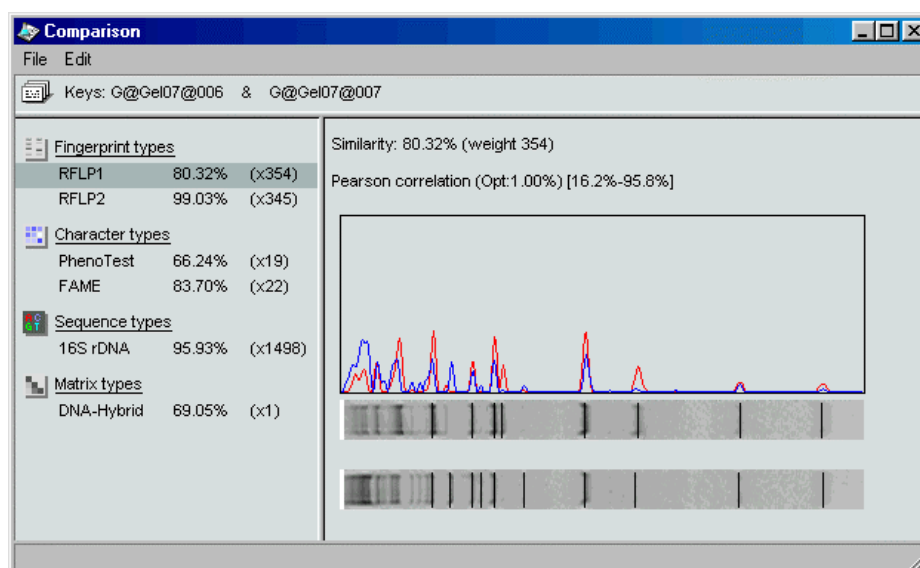


Figure 65. Pairwise comparison window.

9.6.4 Click on an experiment type in the left panel to display the images and the similarity in the right panel.

In case of sequences, the aligned sequences are shown.

In case of a matrix type, the similarity value is shown, if available.

9.7 The Comparison window

The Comparison window is the basis for all grouping and comparison functions. When a new comparison is created, it will automatically use the list of selected entries in the database. With all entries except the standards selected, we will create a new comparison.

9.7.1 Select *Comparison | Create new comparison* (ALT+C) or . A comparison window is created, with the selected database entries (Figure 66).

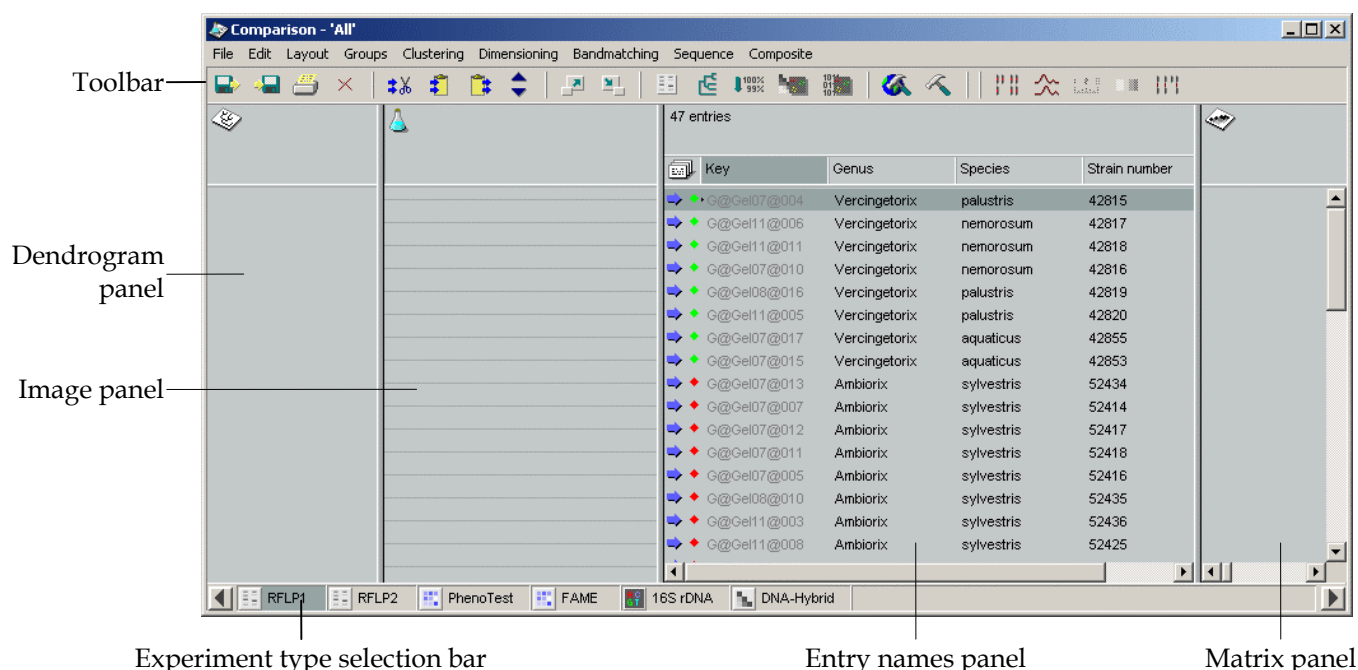


Figure 66. The Comparison window, initial view.

The comparison window is divided in four main panels: the *Dendrogram panel*, which shows the dendrogram if calculated, the *Image panel*, showing the images of the experiments, the *Entry names panel*, which shows the database fields in the same layout as in the database (see 6.5), and the *matrix panel*, which shows the similarity values. Initially, the dendrogram panel, the image panel and the matrix panel are empty.

9.7.2 You can drag the separator lines between the four panels to the left or to the right, in order to divide the space among the panels optimally.

The bottom of the window is the *Experiment type selection bar*, from where you can select one of the available experiment types, to show an image, to calculate a dendrogram, or to show a matrix. Each experiment type on the selection bar contains two objects: a **button**, and the **experiment type name** right from the button. In case of an electrophoresis (fingerprint) experiment type, the button is shown as ; character experiments as ; sequence experiment types as , and matrix types as . This button is to show the image of the experiment type.

9.7.3 Press the  button of RFLP1; the pattern images are shown for RFLP1. When the image of an experiment type is shown, the button shows like .

9.7.4 Press the  button of RFLP2; the pattern images of RFLP2 are shown right from those of RFLP1. When the image of an experiment type is shown, the button shows like .

NOTES:

(1) To display more than one image at a time, we recommend to maximize the comparison window, and to use maximal space for the image panel by minimizing the dendrogram and matrix panels (see 9.7.2).

(2) When preferred, the image of patterns can be shown with a space between the gelstrips. To do so, open the Experiment type window in the program's main window (under **Fingerprint types**) and select **Layout|Show space between gelstrips**.

9.7.5 If insufficient space is available to show both images at the same time, you can scroll through the image panel, or use the zoom functions  and  (**Layout|Zoom in** and **Layout|Zoom out**).

9.7.6 In the caption of the image panels, you can drag the separator line between the images to the left or to the right, to reserve more or less horizontal space for a particular experiment image. Contrary to the zoom function, the original aspect ratio (proportion height to width) of the image will not be maintained by this action.

9.7.7 To select an experiment type in the comparison window, you can either click on the experiment type name in the selection bar (bottom), or on the image itself.

9.7.8 Press the  button or **Layout|Show metric scale** to display the molecular weight scale of the selected fingerprint type.

9.7.9 Press  or **Layout|Show bands** to show or hide the band positions in the image panel. One can show band positions without showing the image.

9.7.10 When bands are shown on the image, they can be exported as a tab-delimited file with **File|Export bands**. The export file, popped up as RESULT.TXT in Notepad, contains the key of the entry, and a list of band positions as relative run lengths (in percent) and molecular weight (if determined).

9.7.11 Press  or **Layout|Show densitometric curves** to show or hide small densitometric curves in the image panel. One can also show curves without showing the image.

9.7.12 When densitometric curves are shown on the image, they can be exported as a tab-delimited file with **File|Export densitometric curves**. The export file, popped up as RESULT.TXT in Notepad, contains the list entry keys separated by tabs, and a list of densitometric curves, of which the curves are listed as columns, separated by tabs.

In case only densitometric curves are available, it can be useful to display the curves as pseudo gelstrips (reconstructed images). This option is selected in the fingerprint type window as follows:

9.7.13 In the main window, double-click on a fingerprint type (e.g. RFLP1) in the **Experiments** panel.

9.7.14 In the *Fingerprint type* window, select **Layout|Show curves as images**. If densitometric curves are now shown in a comparison (9.7.11), they will be displayed as pseudo gelstrips.

In case densitometric curves have different intensities, the densitometric curves can be rescaled so that each curve fills the full available intensity range specified for the fingerprint type. This can be achieved as follows.

9.7.15 In the *Fingerprint type* window, select **Layout|Rescale curves**. If densitometric curves are now shown in a comparison (9.7.11), they will all be displayed with equal intensity.

When an experiment type is selected, both the image caption and its name in the selection bar are highlighted. Functions like *Clustering*, *PCA*, and *Bandmatching* as well as some *Layout* functions, apply to the selected experiment type.

Selections of entries made in the database are also shown in the comparison. The entries in the current comparison are all marked with a blue selection arrow, since they were all selected in the database. You can manually select and unselect entries in the Entry names panel (see Figure 66), using the CTRL and SHIFT keys as described in 9.2.1 and 9.2.2.

9.7.16 First unselect all entries by pressing the F4 key (9.2.6).

9.7.17 Select some entries from the comparison (see 9.2.1 and 9.2.2).

9.7.18 With **Edit|Cut selection** or , the selected entries are removed from the comparison, and are copied to the clipboard.

9.7.19 With **Edit|Paste selection** or , the same entries are placed back in the comparison. If no dendrogram is present, they are placed at the position of the selection bar.

This tool can be used to rearrange entries in the comparison window. Some more other convenient functions are available for rearrangement of entries in a comparison, as explained below.

9.7.20 Select **Edit|Arrange entries by database field** to sort the entries according to the highlighted database field.

When two or more entries have identical strings in a field used to rearrange the order, the existing order of the entries is preserved. As such it is possible to categorize entries according to fields that contain information of different hierarchical rank, for example *genus* and *species*. In this case, first arrange the entries based upon the field with the lowest hierarchical rank, i.e. *species*, and then upon the higher rank, i.e. *genus*.

9.7.21 When a field contains numerical values, which you want to sort according to increasing number, use **Edit|Arrange entries by database field (numerical)**.

In case numbers are combined numerical and alphabetical, for example entry numbers [213, 126c, 126a, 126c], you can first arrange the entries alphabetically (**Edit|Arrange entries by database field**), and then numerically using **Edit|Arrange entries by database field (numerical)**. The result will be [126a, 126b, 126c, 213].

9.7.22 A group of selected entries (blue arrows) can be placed at the position of the cursor (the entry you last clicked on) with **Edit|Bring selected entries to cursor**.

9.7.23 A group of selected entries (blue arrows) can be moved to the top of the comparison with **Edit|Bring selected entries to top**.

9.7.24 An individual entry can be moved up and down by left-clicking on it to select it, and **Edit|Move entry up**  or **Edit|Move entry down** .

9.7.25 When using the up/down buttons  and , you can move an entry to the top or the bottom at once by holding the CTRL key.

Database fields can be abbreviated using **Edit|Set database field length** (see also 6.5.4).

9.7.26 To save the comparison, select *File | Save as* or  and enter a name, e.g. **All**.

9.7.27 Close the comparison with *File | Exit*. Comparison “**All**” is now listed in the *Comparisons* panel of the main window.

9.7.28 You can open the existing comparison by double clicking on **All** in the comparisons panel.

Entries can be added to an existing comparison at any time. The entries first need to be copied to the clipboard from the main window or from another comparison.

9.7.29 To copy entries to the clipboard, use the *Edit | Copy selection* command or .

9.7.30 To cut entries from one comparison into another, use *Edit | Cut selection* or  in the one comparison and *Edit | Paste selection* or  in the other comparison.

10. Band matching and polymorphism analysis

Band matching is a comparison function which applies only to fingerprint types. It can be executed on any selection of entries from the database. In a first step, Fingerprinting II Informatix divides all the bands found among the selected patterns into *classes of common bands*. As such, every band of a given pattern belongs to a class, and conversely, every band class is represented by a band on one or more patterns. The result is shown in Figure 67.

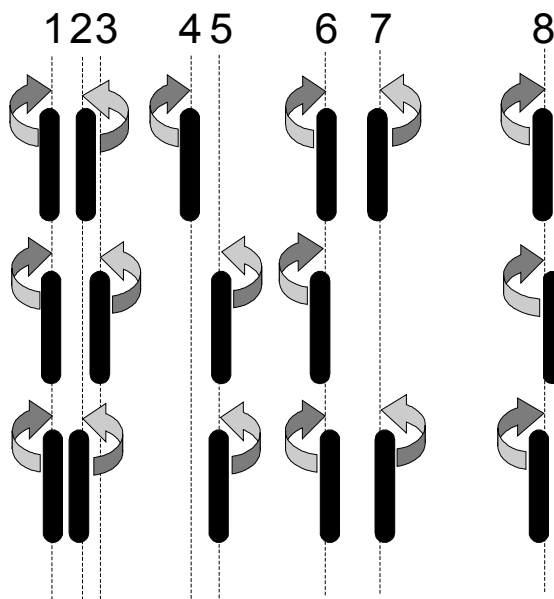
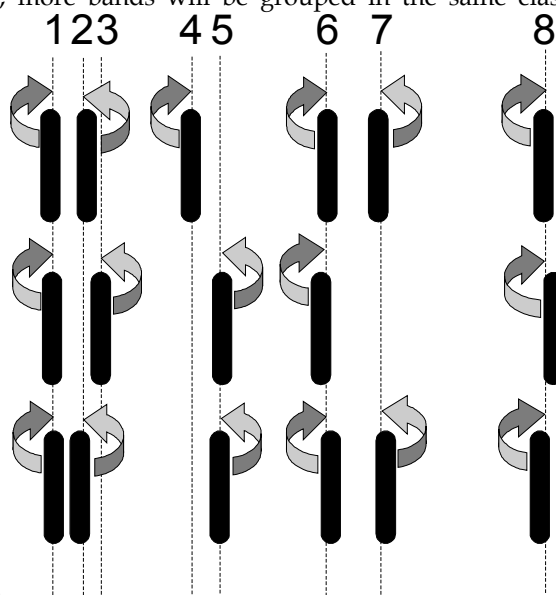


Figure 67. Comparative quantification: bands are assigned to classes.

Clearly, the number of band classes distinguished will depend on the *optimization* and the *position tolerance* that is allowed between bands considered as matching. For example, when a larger position

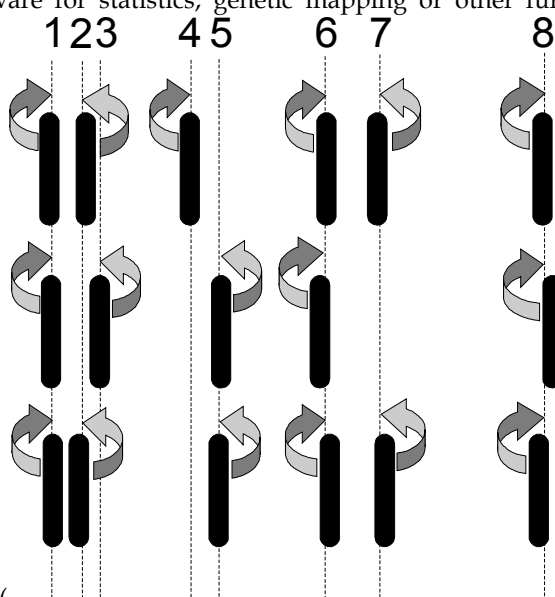
tolerance is specified, more bands will be grouped in the same class than when a small position



tolerance is chosen. In

Figure 67, taking a larger position tolerance would have resulted in the merging of band classes 2 and 3, whereas a smaller position tolerance would have resulted in two separate classes for band class 8.

For each pattern, a particular band class can have two states: present or absent. This is the basis for *polymorphism analysis*, a tool which allows comparative binary (+/-) tables to be generated, displaying polymorphic bands between the selected patterns. These tables, created as text or tab-delineated files, are ready for export to other specialized software for statistics, genetic mapping or other further



analysis. The binary table for the above example (

Figure 67) is shown in Figure 68.

	1	2	3	4	5	6	7	8
Pattern 1	+	+	-	+	-	+	+	+
Pattern 2	+	-	+	-	+	+	-	+
Pattern 3	+	+	-	-	+	+	+	+

Figure 68. Binary presence/absence table of banding patterns.

Instead of using binary (+/-) data, the same tables can be generated using band intensities obtained from curves (band heights or surfaces) or from two-dimensional pattern contours (volumes or concentrations).

10.1 Composite data sets

Before you can create a (binary) presence/absence table as shown in Figure 68, you will need to define a *Composite data set*, containing the fingerprint type as input. A composite data set is a character table that contains all the characters of one or more experiment types. Such a character table is necessary to convert the band classes and represent them as presence/absence tables.

10.1.1 In the main window, with the database DemoBase loaded, select *Experiments | Create new composite data set*, or .

10.1.2 Enter a name, for example **RFLP1-table** and press <OK>.

The *Composite data set window* is shown for “RFLP1-table” (see Figure 69). All experiment types defined for the database are listed, and when they are marked with a red cross, they are not selected in the composite data set.

10.1.3 Since we want to create a character table for the fingerprint type RFLP1, we select RFLP1 and *Experiment | Use in composite data set*.

When the experiment type is selected, it is marked with a green ✓ sign. The scroll bar that appears in the **Weights** column as well as the other menu commands under *Experiment* apply to cluster analysis and will not be discussed here.

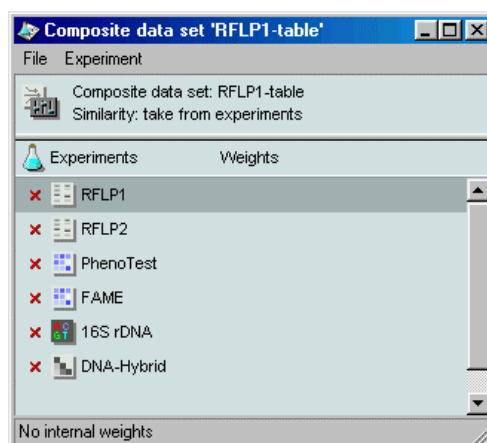


Figure 69. Composite data set window.

In order to create a band presence/absence table using multiple *fingerprint types* (for example, obtained by using different primers or restriction enzymes), one can simply include the other fingerprint types in the composite data set.

10.1.4 Close the composite data set window with *File|Exit*. The new composite data set is shown in the experiment types panel of the main window.

10.2 Creating a band matching

10.2.1 In the main window, with the database DemoBase loaded, select a number of entries, for example all but the “STANDARD” entries. If the comparison **All** is present, you can open this one (see previous chapter).

10.2.2 Select RFLP1 in the experiment type selection bar (bottom), and press the  button or *Layout|Show image*.

10.2.3 Choose *Bandmatching|Perform band matching*.

The *Position tolerance settings* dialog box for the fingerprint type is popped up (Figure 70). This is the same dialog box which can be called from the experiment type window settings (7.3.26).

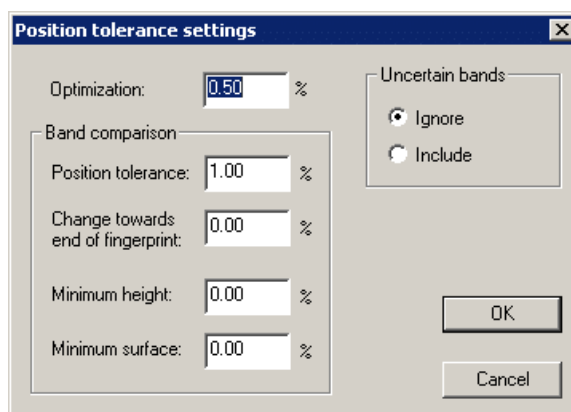


Figure 70. Position tolerance settings dialog box of a fingerprint type.

The *Position tolerance* is the maximal shift (in percentage of the pattern length) between two bands allowed to consider the bands as matching. With *Change towards end of fingerprint*, you can specify a gradual increase or decrease in tolerance.

The *Optimization* is a shift that you allow between any two patterns and within which the program will look for the best possible matching. To understand the utility of optimization in addition to tolerance, see the example in Figure 71.

With minimum height and minimum surface, you can exclude weak or irrelevant bands.

The *Uncertain bands* option allows you to either include uncertain bands or ignore them (see 7.3.1). When *Ignore* is chosen, uncertain bands are ignored. This means that in composing a band matching table, the software will omit the uncertain bands. When *Include* is chosen, uncertain bands are treated in the same way as certain bands, which means that uncertain bands will contribute to the band classes of a band matching tables in the same way as certain bands.

10.2.4 Enter a position tolerance of 1%, an optimization of 1%, a change of 0%, and a minimum height and minimum surface of 0%, and press **<OK>**.

*NOTE: Depending on whether or not there are selected entries in the comparison, the program will ask "Search only inside the selection?". If you want to include all patterns for band class searching, answer **<No>** to this question.*

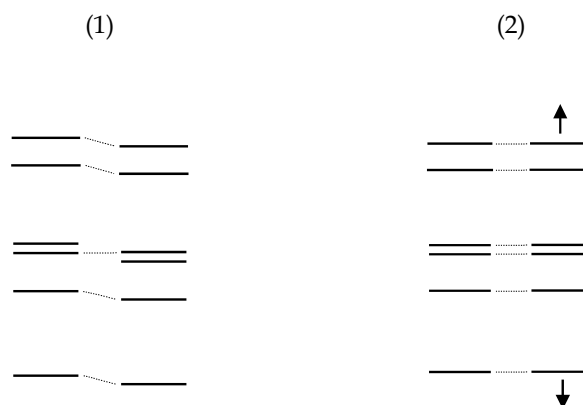


Figure 71. Effect of position tolerance (1) and optimization (2) on the matching between shifted patterns.

The program has now defined the band classes and has associated each band with a class. The band classes are shown as blue lines (Figure 72) and the bands are linked to a class in red. On top of the image are the band class selectors. If a band class is selected, this selector is blue.

10.2.5 Zoom in on the image as necessary using the zoom functions  and  (**Layout | Zoom in** and **Layout | Zoom out**).

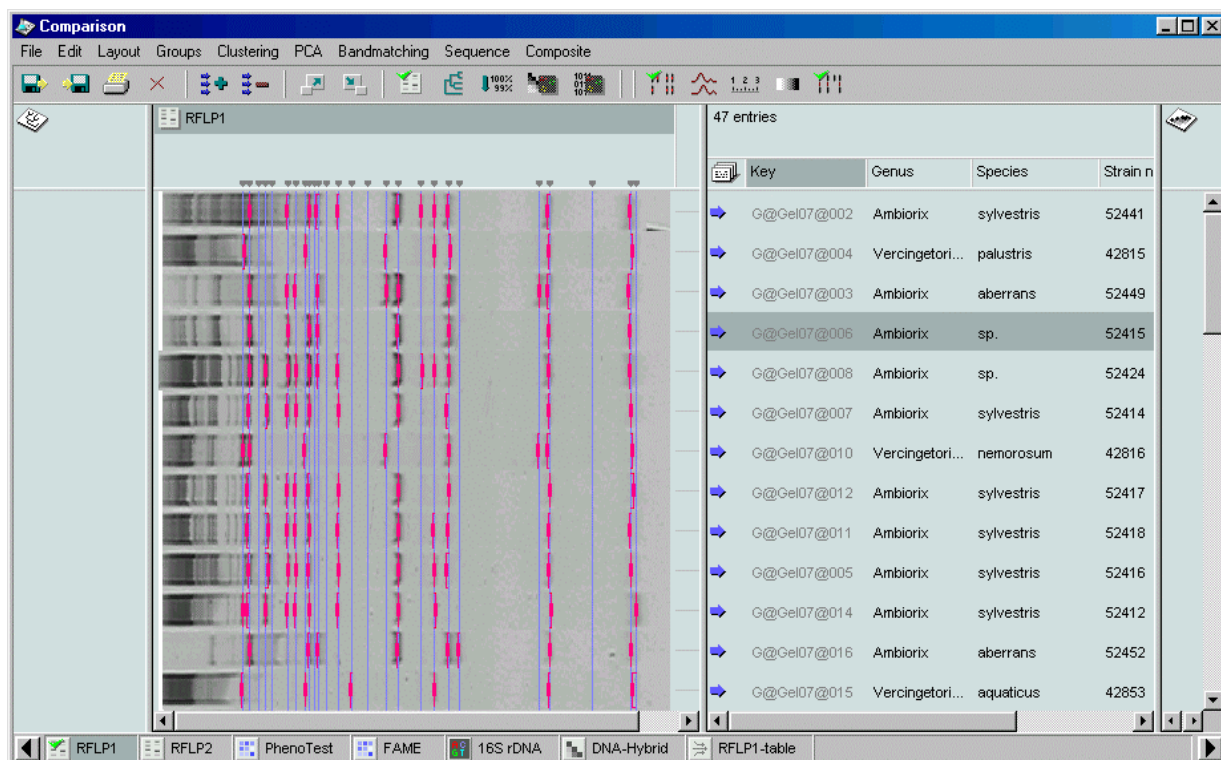


Figure 72. Band matching analysis.

10.2.6 An interesting option for long patterns with numerous small bands is *Layout | Stretch (X dir)*.

This causes the image to be enlarged in the horizontal direction only, so that sharp bands become better visible, without losing the overview of a large number of patterns.

10.2.7 Restore the view with *Layout | Compress (X dir)*.

10.2.8 Press the  button or *Layout | Show metric scale* to display the molecular weight scale of the fingerprint type.

The program does not always assign the bands to the correct class. Therefore, you can manually correct the assignments. In Figure 73, the band marked with the arrow is assigned to the left of two close classes, whereas it should be assigned to the right class.

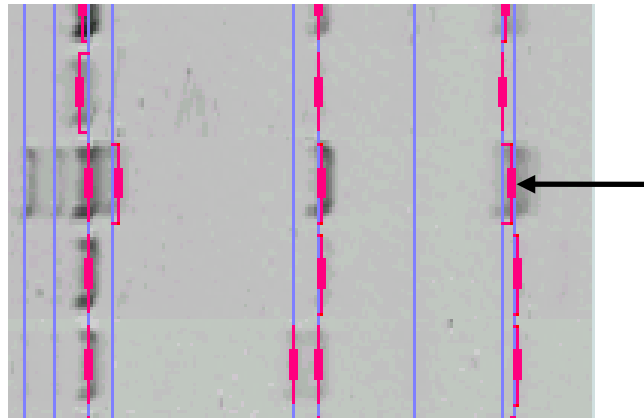


Figure 73. Detail of band class assignments.

NOTE: you can easily see which bands that belong to a given band class by double clicking on the vertical blue line that represents the class: all bands that belong to the class are selected with a green flag.

Manual editing of a band matching

In order to reassign a band to another class, proceed as follows:

10.2.9 Select the band class to which the band ought to be assigned. The selector becomes blue.

10.2.10 Right-click on the band, and select ***Band classes | Assign band to class*** from the floating menu.

A whole band class is deleted as follows:

10.2.11 Click on a band belonging to the band class.

10.2.12 Right-click on the band, and select ***Band classes | Remove band class*** from the floating menu.

If different bands are incorrectly assigned to the same class, you can create a second class as follows (Figure 74):

10.2.13 Select a band which should belong to a new class (left-click) (see Figure 74).

10.2.14 Right-click on the band, and select ***Band classes | Add new band class*** from the floating menu.

10.2.15 Right-click again on the band, and select ***Band classes | Auto assign bands to class*** from the floating menu.

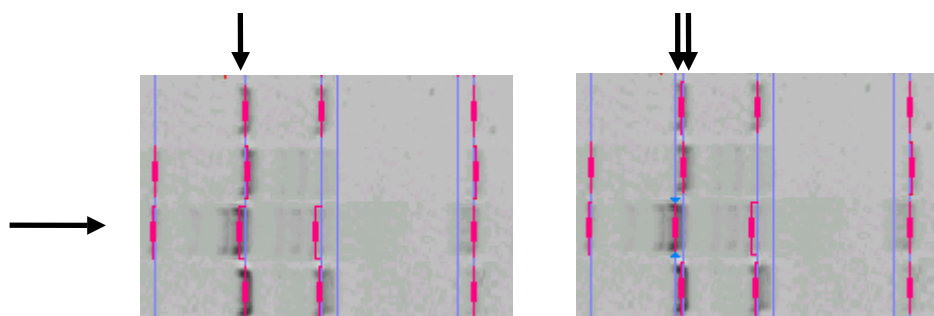


Figure 74. Splitting up a band class into two band classes.

All bands that are closer to the new band class are automatically reassigned to that new class. In order to reassign bands to the other class, follow the procedure explained in 10.2.9 to 10.2.10.

If bands are incorrectly assigned to different classes, you can merge the classes as follows (Figure 74):

10.2.16 Choose a band which occurs quite in the middle of the two classes.

10.2.17 Right-click on the band, and select **Band classes | Add new band class** from the floating menu.

10.2.18 Choose a band which belongs to the left class.

10.2.19 Right-click on the band, and select **Band classes | Remove band class** from the floating menu, or press SHIFT + DEL.

10.2.20 Choose a band which belongs to the right class (left-click).

10.2.21 Right-click on the band, and select **Band classes | Remove band class** from the floating menu, or press SHIFT + DEL.

10.2.22 Right-click again on the band, and select **Band classes | Auto assign bands to class** from the floating menu.

10.2.23 Select the new band class to which all the bands should belong (left-click). The selector becomes blue.

10.2.24 Right-click, and select **Band classes | Auto assign bands to class** from the floating menu.

If you do not wish to use a single band in a band matching analysis, you can undo its assignment it as follows:

10.2.25 Right-click on the band and select **Band classes | Remove band from class** or DEL.

In defining a band class, the program automatically calculates the average position of all bands belonging to that class to define the position of the band class. After reassigning bands, removing and adding bands etc. the band class position may not be the center anymore. You can correct the position of the band class:

10.2.26 Select the band class (left-click) and call the floating menu (right mouse button) to select **Band classes | Center class position**.

NOTE: these commands are also accessible from the main menu, but they are much easier using the floating menu.

10.2.27 If all assignments are corrected, you can save the band matching with **File | Save** or .

Analyzing polymorphic bands only

Some more advanced tools are available for specific purposes. Suppose that it is only important to show and compare the bands that are polymorphic in a subset of the entries in the comparison (for example, two parent lines in a plant breeding study).

10.2.28 First clear any selected entries, and then select the entries that are of interest to find polymorphic bands on.

10.2.29 Select **Bandmatching | Polymorphic classes only (for selection list)**.

The table of band classes is reduced to contain only the bands that are polymorphic within the selected entries (i.e. not present in all of them and not absent in all of them). Obviously, this feature can also be used to select only polymorphic bands from the complete study: in that case, first select all entries in the comparison, and **Bandmatching | Polymorphic classes only (for selection list)**.

Adding entries to a band matching

Since a band matching analysis and the associated table can be saved, it should be possible to delete entries from, or add entries to the band matching at any time.

10.2.30 To delete some entries, simply select some entries and **Edit | Cut selection** or .

If entries are added however, it is possible that those new entries contain bands that are not defined as a band class yet. If you have done some editing work to the band classes already, it would be beneficial to preserve the existing band classes, and simply associate the bands of the new entries to the existing classes, and introduce new classes in those cases where the new entries have bands that do not fit in any of the existing classes. This is achieved as follows:

10.2.31 Select a few entries in the database. If you have executed the previous step (10.2.30) there are still some entries selected and placed on the clipboard.

10.2.32 [In case you would have copied something else in the meantime, select **Edit | Copy selection** or  in the main window.]

10.2.33 With **Edit | Paste selection** or  in the other comparison, the selected entries are placed back in the band matching.

10.2.34 Select **Bandmatching | Search band classes**. The *Position tolerance settings* dialog box as in Figure 70 is shown. Leave the settings as they are and press **<Ok>**.

The program now asks "Remove existing band classes?".

10.2.35 In order to preserve the existing band matching, it is important to answer **<No>** to this question.

The program asks one more question: "Search only inside the selection?".

10.2.36 If you wish to search only for new band classes that might occur in the new patterns added to the band matching, answer **<Yes>** to this question. Otherwise, any band classes that you may have deleted manually from the other entries would appear again.

Band and band class filters

When searching bands in complex patterns, especially those for which the terminal step is a PCR reaction such as AFLP patterns, it is sometimes difficult to define objective criteria as to what is a band and what is not a band. However, when the user examines a set of patterns by eye, it often becomes easier to decide whether a band is valid or not, because the user automatically compares the band with those on neighboring patterns, thus obtaining information which cannot be obtained by inspecting the pattern alone. This is more or less the way the band filters work in the band matching application of Fingerprinting II Informatix: in a first step, band classes are defined over all patterns; then the relative areas of all bands of a given class are averaged, and if a band deviates more than a certain percentage from this average, it is not considered as being a matching band for this class.

Using this tool, it is possible to define more bands on the gels than one would usually do, without spending a lot of time deleting and adding bands manually. Using the band matching filters, weak bands or artifacts that do not reflect the expected intensity will be filtered out automatically, and the assignment of bands is often as reliable as after hours of band editing work.

10.2.37 In the band matching analysis created in the current paragraph, select ***Bandmatching | Band class filter***.

This pops up the *Band filtering settings* dialog box (Figure 75). It consists of two parts: the upper part "***Remove all bands below...***" is to filter individual bands within a found band class, and the lower part "***Remove all band classes that have no bands exceeding...***" is to remove all band classes that do not contain any significant band.

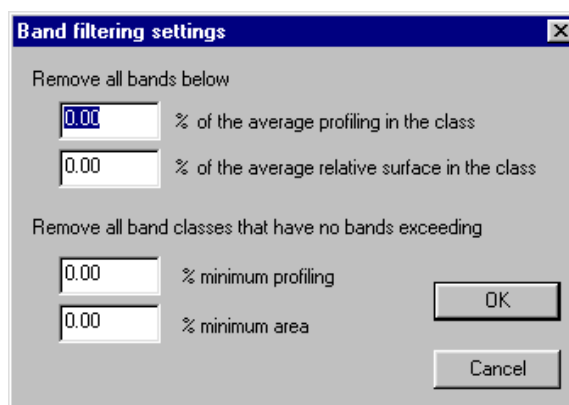


Figure 75. Band filtering settings dialog box for band matching.

Similar as for band searching, the band class filters consist of two separately working components: a *profiling* component, which is the height of the band or class, and an *area* component, which is the relative area (surface).

10.2.38 Within a band class, you can ***Remove all bands below*** a certain ***% of the average profiling in the class***.

If you entered 80%, this means that, if the height of a band is lower than 80% of the average profiling calculated for its class, it will not be matched with that class, and the band will be recorded negative in the band matching table. Note that the profiling of a band is an absolute measure: if a pattern as a

whole is rather weak, many of its bands may be excluded from the band matching just by this fact. In such cases, we recommend to take the area as filtering factor.

10.2.39 Within a band class, you also can ***Remove all bands below*** a certain ***% of the relative surface in the class***.

In this case, if you entered 80%, all bands that have a *relative* surface less than 80% of the average surface for the band class will not be matched with that class, and the bands will be recorded negative in the band matching table. Since the surface is relative to the total surface of a pattern, weak patterns in principle will not be treated differently compared to dark patterns.

In case of complex patterns such as AFLP, many band classes consist of just one weak band, spot or artifact and have no phylogenetic or taxonomic relevance. Such band classes are just filling up the band matching table, and being treated equally important, they are disturbing the information provided by the band matching table. Therefore, Fingerprinting II Informatix offers the possibility to have all band classes that do not contain at least one clear relevant band excluded from the band matching table.

10.2.40 With ***Remove all band classes that have no bands exceeding*** a certain ***% minimum profiling***, you can remove all irrelevant band classes based upon the minimum height of the bands included.

If you entered 20%, this means that a band class for which the highest band is less high than 20% of the OD range of the fingerprint type will be considered irrelevant and will be removed.

NOTE: This is again a non-relative parameter. If by incidence a band class is formed by a set of weak patterns, it may be excluded incorrectly. If this happens to be a problem, we recommend to use the more reliable feature of % minimum area only.

10.2.41 With ***Remove all band classes that have no bands exceeding*** a certain ***% minimum area***, you can remove all irrelevant band classes based upon the minimum area of the bands included. The minimum area is defined as the area relative to the total area of a pattern.

If you enter 20% here, a band class that contains no band with an area bigger than 20% of its pattern's total area will be removed from the band matching table.

10.3 Creating a band matching table for polymorphism analysis

After the composite data set RFLP1-table had been defined (see paragraph 10.1), it is listed in the experiment type selection bar in the bottom of the comparison window. Since we defined RFLP1 as being the experiment type used in this composite data set, the band matching are automatically filled in as character values.

10.3.1 Press the  button of RFLP1-table. The binary band matching table appears as in Figure 76.

10.3.2 In order to reveal the complete information on the band classes, it may be necessary to drag the separator line between the table and its header (see Figure 76) downwards.

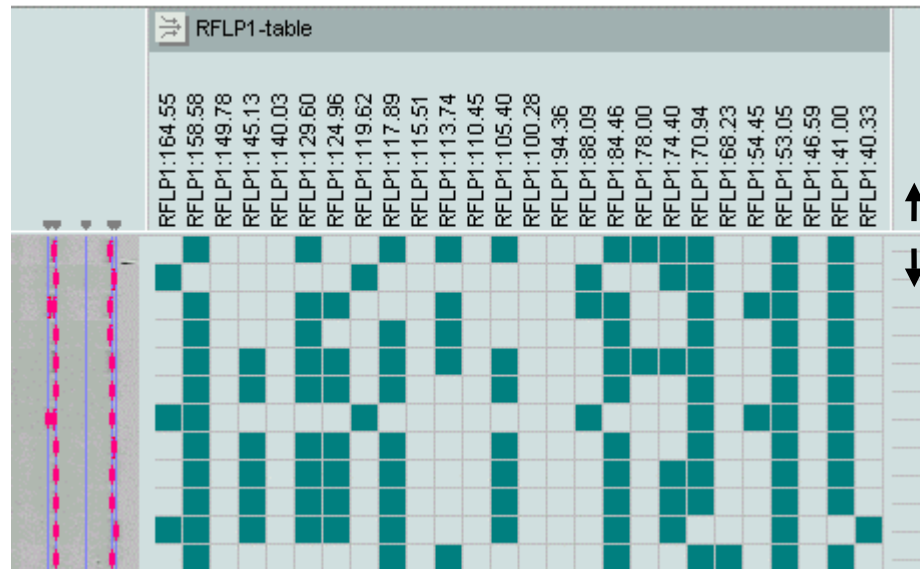


Figure 76. Binary band matching table; detail.

NOTES:

(1) You can scroll between the image of gel patterns and the character table using the scroll bar at the bottom of the image panel. Once the character table is present, it is still possible to edit the band class assignments on the patterns. The character table is updated automatically.

(2) Band classes that have been created by the user are marked with an asterisk (*).

10.3.3 Use **Composite|Export character table** to export a space or tab-delineated text file of the binary band matching table.

When the program asks “*Use tab-delineated fields*”, you should answer **<Yes>** to produce a tab-delineated text file. The space-delineated table looks as follows (Figure 77).

[illegible]

Figure 77. Binary band matching character table exported from Fingerprinting II Informatix.

In the tab-delineated format, the band classes (header) and the band presence/absence table are given in columns separated by tabs. This format is the easiest to import in spreadsheet or database software packages.

10.3.4 To show the intensity of the bands, choose *Composite | Show quantification (colors)*.

The color ranges from blue (weakest bands) over cyan, green, yellow, orange to red (darkest bands). The intensity is based upon the *Comparative quantification settings* for the used fingerprint type. This option can be defined in the fingerprint type window (see 7.3.26), but it can also be changed in the comparison window, as follows:

10.3.5 Select RFLP1 in the experiment type selection bar.

10.3.6 Make sure the bands are shown for RFLP1. Check whether *Show bands* is marked with ✓ in the *Layout* menu.

10.3.7 Select *Bandmatching | Comparative Quantification settings*. This opens the Comparative Quantification settings dialog box (Figure 78).

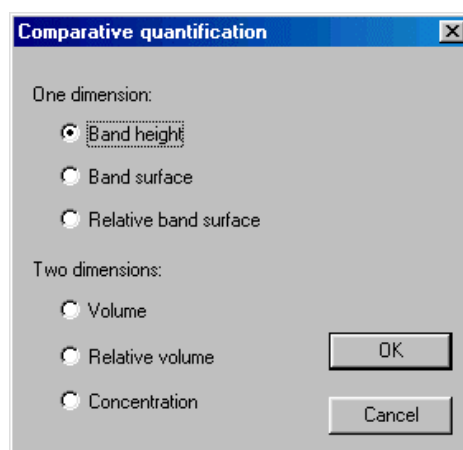


Figure 78. The Comparative quantification settings dialog box.

One dimension quantification is based on the densitometric curves extracted from the patterns (see paragraph 7.3): *Band height* is the height of the peak; *Band surface* is the area under the Gaussian curve approximating a band; *Relative band surface* is the same as band surface, as a percentage of the total band area of the pattern.

Two dimensions quantification is based on the band contours of the two-dimensional pattern images (see paragraph 7.3): *Volume* is the absolute volume within the contour; *Relative volume* is the same as a percentage of the total band volume of the pattern; *Concentration* is the physical concentration unit the user has assigned based upon regression through known calibration bands.

If no two-dimensional quantification is performed for the gels, it is obvious that one should select among the first three options.

10.3.8 Close the comparative quantification settings dialog box with <OK> or <Cancel>.

10.3.9 Select *Composite | Show quantification (values)* to display the numerical intensities of the bands.

10.3.10 With *Composite | Export character table*, a numerical band matching table is created in text format, separated by tabs (Figure 79).

The meaning of the numbers depends on the selected comparative quantification parameter (see 10.3.5 to 10.3.7).

RFLP1:94.36														
RFLP1:88.09														
RFLP1:84.46														
RFLP1:78.00														
RFLP1:74.40														
RFLP1:70.94														
RFLP1:68.23														
RFLP1:54.45														
RFLP1:53.05														
RFLP1:46.59														
RFLP1:41.00														
RFLP1:40.33														
0.00	0.00	25.27	3.48	2.99	8.95	0.00	0.00	8.31	0.00	13.31	0.00		TABLE: Rows=entries	
0.00	10.39	0.00	0.00	8.85	10.46	0.00	0.00	11.78	0.00	10.23	0.00			
0.00	13.49	25.32	0.00	0.00	15.51	0.00	9.28	7.11	0.00	5.05	0.00			
0.00	0.00	19.79	0.00	0.00	13.71	0.00	0.00	5.13	0.00	5.55	0.00			
0.00	0.00	21.67	2.36	2.42	6.39	0.00	0.00	6.29	0.00	6.61	0.00			
0.00	0.00	25.06	0.00	0.00	3.92	0.00	0.00	6.33	0.00	5.27	0.00			
0.00	11.89	0.00	0.00	0.00	10.34	0.00	5.47	7.44	0.00	3.69	0.00			
0.00	0.00	26.11	0.00	0.00	4.97	0.00	0.00	5.85	0.00	3.65	0.00			
0.00	0.00	26.20	0.00	3.76	2.59	0.00	0.00	5.95	0.00	4.56	0.00			

Figure 79. Numerical band matching character table exported from Fingerprinting II Informatix.

10.4 Tools to display selective band classes

10.4.1 First, remove the existing band matching analysis by selecting **RFLP1** in the experiment type selection bar and *Bandmatching | Perform band matching* or .

10.4.2 Clear any selection made by pressing F4, and manually select some entries in the comparison.

10.4.3 Select *Bandmatching | Perform band matching* to create a new band matching. Press <OK> to the position tolerance settings.

The program now asks “*Search only inside the selection?*”. With this option, the program will perform the band matching only for the selected entries. This means that it will only create band classes for bands found on the entries in the selection.

10.4.4 Answer <Yes> to perform a band matching on the selected entries only.

10.4.5 With *Bandmatching | Auto assign all bands to all classes*, you can let the program assign the bands of the not selected entries to the corresponding band classes.

Fingerprinting II Informatix offers another interesting tool to display only the polymorphic bands. To make this tool as flexible as possible, the polymorphic bands are only looked for within the selection list. For genetic mapping purposes, the user can select the patterns from two (or more) parent entries, and have the program display only the polymorphic band classes between these two patterns. This reduces the size of the band matching table to contain only the polymorphic bands of interest. Of course, the user can add or delete band classes afterwards, as desired.

10.4.6 Clear any list of selected patterns with F4.

- 10.4.7 First, remove the existing band matching analysis by selecting **RFLP1** in the experiment type selection bar and *Bandmatching | Perform band matching* or .
- 10.4.8 Select *Bandmatching | Perform band matching* to create a new band matching, including all band classes.
- 10.4.9 Select two entries having a few different bands.
- 10.4.10 Select *Bandmatching | Polymorphic bands only (for selection list)*. Only the band classes that are polymorphic between the selected two patterns are now displayed.

10.5 Finding discriminative bands between entries

- 10.5.1 In database **Demobase**, have a comparison window open with all non-“STANDARD” entries selected (e.g. comparison ‘**All**’, see 9.7.26) and the *composite data set RFLP1-table* shown (see 10.2 and 10.2.39).
- 10.5.2 Make sure that the image of the composite data set is shown, by pressing the  button of **RFLP1-table**.
- 10.5.3 Minimize or reduce the comparison window so that the main window (at least the menu and toolbar) becomes visible.
- 10.5.4 Press F4 to make sure that no entries are selected.
- 10.5.5 In the main window, select *Edit | Search entries* (F3) and enter **Vercingetorix** in the *Genus* field.
- All *Vercingetorix* entries are selected in the database (and in the comparison window).
- 10.5.6 To group the selected entries, choose *Edit | Bring selected entries to top* in the comparison window.
- 10.5.7 Select *Composite | Discriminative characters*.

The characters (bands) are reorganized in such a way that those characters positive for the selected entries and negative for the other entries occur left, and those characters negative for the selected entries and positive for the other entries occur right (see Figure 80).

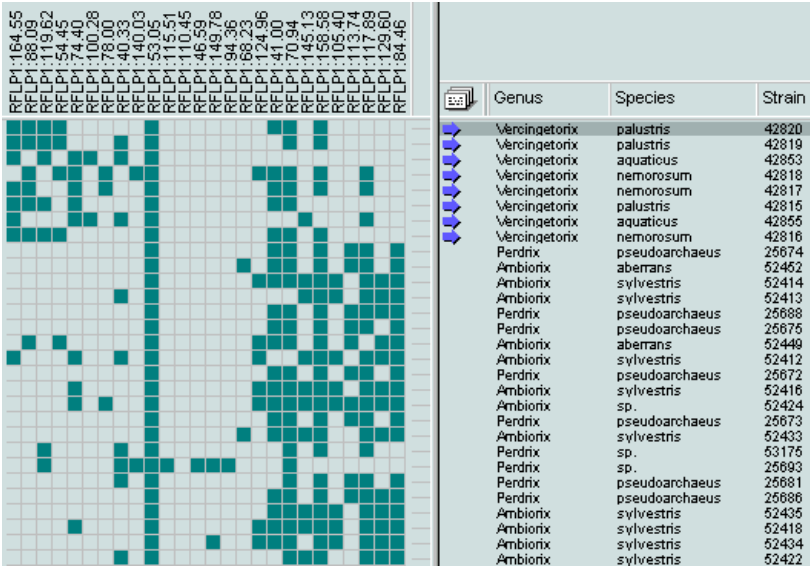


Figure 80. Discriminative bands for selected entries, positive discrimination left, negative discrimination right.

In composite data sets, it is also possible to list the entries according to the amount of a selected character. In case of banding patterns, the entries will be ordered by the intensity of a selected band. This feature allows for a particular bands the entries to be found in which the band is present or not.

10.5.8 Show the band table as intensity table with *Composite | Show quantification (colors)*.

10.5.9 Select a band class in the band classes header (Figure 76) and *Composite | Sort by character*.

The entries are now sorted by increasing intensity of the selected band class.

11. Cluster analysis

11.1 Principles

A clustering can be performed on an individual experiment type or on a combination of experiment types. The methods that Fingerprinting II Informatix uses to arrive at dendrograms representing combined techniques are schematized in Figure 81.

- Flows 1 and 2 represent the steps to obtain dendrograms for two single experiments, experiment 1 and experiment 2, respectively. The steps involve the creation of a similarity matrix and the calculation of a dendrogram based on this matrix.
- Flow 3 is the first method to calculate a combined dendrogram from multiple experiments: the individual similarity matrices are first calculated and from these matrices, a combined matrix (A) is calculated by averaging the values. The averaging can happen in two ways: each value can be considered equally important, or the program can assign a weight proportional to the number of tests in an experiment. In addition, the user can define an extra weight for each experiment manually.
- Flow 4 starts directly from the character tables, and merges all characters from different experiment types to obtain a *composite data set*. From this composite data set, a similarity matrix is calculated (combined matrix B), resulting in combined dendrogram B.

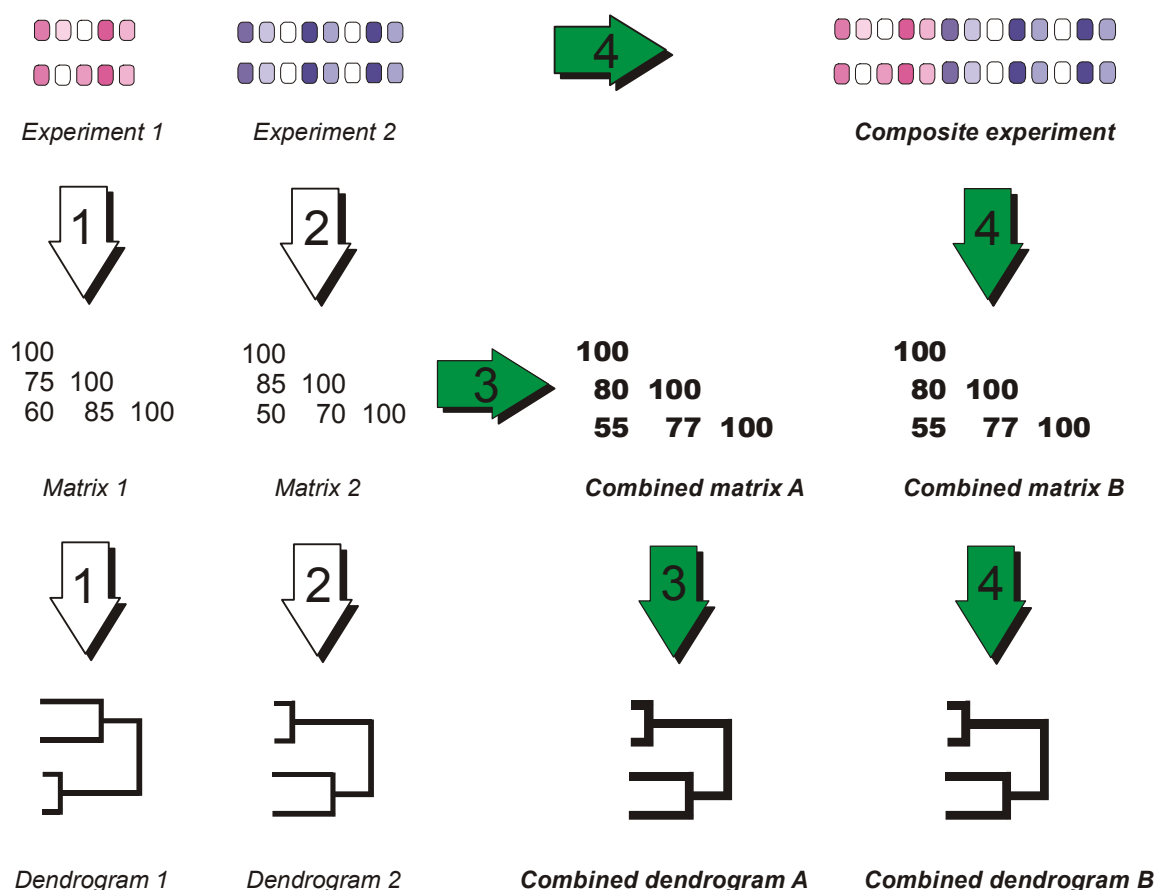


Figure 81. Scheme of possibilities in Fingerprinting II Informatix to obtain combined dendrograms from multiple experiments.

Both steps 3 and 4 require a *composite data set* to be generated.

11.2 Composite data sets

A composite data set is a character table that contains all the characters of one or more experiment types. We have briefly described the use of composite data sets in order to create a character table for a band matching analysis (10.1).

*NOTE: In addition to the obvious reasons of (i) creating band matching tables and (ii) creating clusterings of multiple data sets, a composite data set also offers some additional interesting features compared to single character types. These include a function to discriminate groups based upon differential characters in the comparison window (**Composite|Discriminative characters**) and a function to sort the entries in a comparison by the intensity of a selected character (**Composite|Sort by character**). Both features are described in 10.5. Further, the characters in a composite data set can be displayed as numerical values which can be exported as tab-delineated tables in a comparison (**Composite|Export character table**). This feature is described in 10.2.*

We will now describe the use of composite data sets in function of *cluster analysis* based upon multiple experiments.

11.2.1 In the main window, with the database DemoBase loaded, select *Experiments | Create new composite data set*, or .

11.2.2 Enter a name, for example **All-pheno** and press <OK>.

The *Composite data set window* is shown for “All-Pheno” (see Figure 82). All experiment types defined for the database are listed, and when they are marked with a red cross, they are not selected in the composite data set. We will now describe the use of composite data sets in function of cluster analysis based upon multiple experiments.

11.2.3 We want to create a character table for all phenotypic tests defined in the database, so we select **Phenotest** and *Experiment | Use in composite data set*. We repeat this for **FAME**.

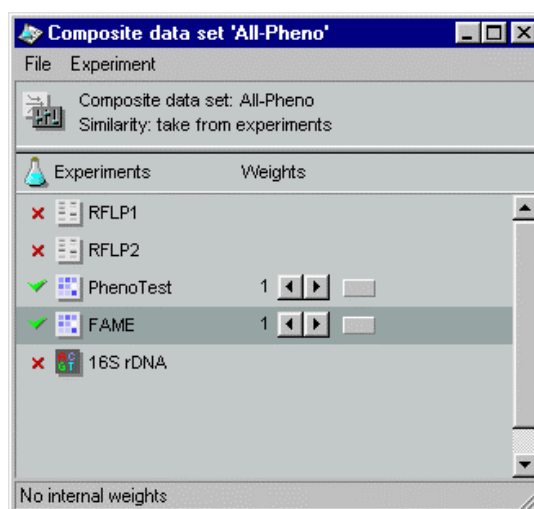


Figure 82. Composite data set window.

When an experiment type is selected in the composite data set, it is marked with a green ✓ sign.

The scroll bar that appears in the **Weights** column allows the user to manually assign weights to each of the selected experiment types (see step 3 described in 11.1). If the individual matrices of the experiments are averaged to obtain a combined matrix, the similarity values will be multiplied by the weights the user has specified for each experiment.

In order to treat individual characters on an equal basis while averaging matrices, the program can automatically use weights proportional to the number of tests each experiment contains. This correction is achieved as follows:

11.2.4 Select *Experiment | Correct for internal weights*. The header now shows **Weights (x internal weight)**.

NOTES:

1. The correction for internal weights also applies to banding patterns: if technique RFLP1 reveals 10 bands between entries A and B, whereas RFLP2 only reveals 5 bands, the similarity value resulting from RFLP1 will be twofold more important in averaging similarity between entries A and B.

2. Both functions **Correct for internal weights** and the manual weight assignment can be combined. The program will then multiply the weights obtained after correction by the weights assigned by the user.
3. In case step 4 described in 11.1 is chosen further in the analysis, i.e. the character sets are merged to a combined character set to which a similarity coefficient is applied, the user defined weights also have their function: in this case, the program multiplies each character of a given experiment with the weight assigned to that experiment. This feature is useful in case the ranges of combined experiments are different; for example when one experiment has a character value range between 0 and 1 and another experiment has a range between 0 and 100, a quantitative coefficient such as the correlation coefficients, Gower, Euclidian distance, would in practice only rely on the second experiment. Assigning a weight of x100 to the first experiment makes them equally important for quantitative coefficients.

11.2.5 The comparison settings for the composite data set are entered with **Experiment | Comparison settings**.

The *Composite data set comparison* dialog box (Figure 83) allows you to choose between step 3 (averaging the matrices of the experiments) and step 4 (merging the experiments to a composite experiment). With the **Similarity coefficient** option *Take from experiments*, the matrices from the individual experiments are averaged according to the defined weights (step 3 in Figure 81). With one of the coefficients under **Use binary character table** or **Use character table value** (non-binary coefficients), step 4 in Figure 81 will be followed using a composite character table.

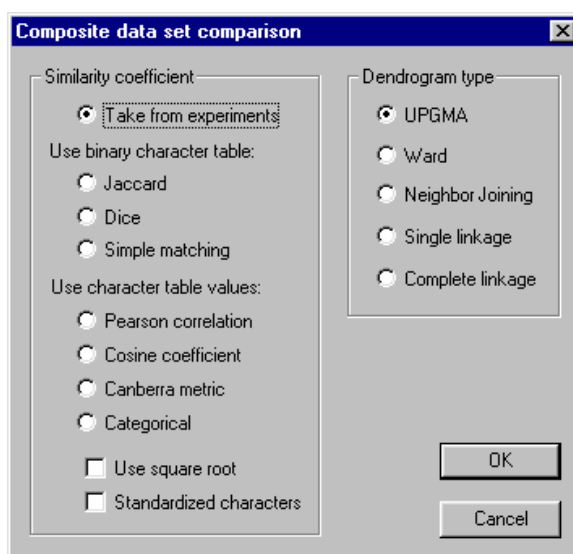


Figure 83. Composite data set comparison dialog box.

If non-binary characters (values) are used, it may be meaningful to enable the feature **Standardized characters** in the following cases. (1) For some techniques, e.g. fatty acid methyl ester analysis, it is common that some fatty acids occur in high amounts, whereas other fatty acids occur only in very small amounts. It is likely that the major fatty acids will account for most of the discrimination between the organisms studied, whereas the minor fatty acids, which may be as valuable from a taxonomic point of view, are masked. (2) When creating composite character sets from different experiments, the ranges of the experiment may be different. When using a coefficient such as the correlation coefficient, characters with a higher range will have more influence on the similarity and the dendrogram. The feature **Standardized characters** standardizes each character by subtracting its mean value and dividing by its standard deviation. The result is that all characters have equal influences on the similarity.

The feature *Use square root* is intended for character sets that yield high similarities within groups. In such cases, it may be useful to combine *Use square root* with correlation, cosine coefficient (or Euclidean distance in case of non-composite data sets).

Select *Pearson correlation* with *Standardized characters* and *UPGMA* as clustering method.

11.2.6 Close the composite data set window with *File|Exit*. The new composite data set is shown in the experiment types panel of the main window.

11.3 Calculating a dendrogram for a single experiment

11.3.1 Run the Analyze program with database **DemoBase** loaded.

11.3.2 Select all entries except STANDARD (9.3) and create a new comparison (9.7), or open comparison **All** if existing.

11.3.3 Select RFLP1 in the experiment type selection bar (bottom of window), and show the normalized gel image by pressing the  button.

11.3.4 Select *Clustering|Calculate cluster analysis*.

The *Comparison settings* dialog box allows you to specify the similarity coefficient to calculate the similarity matrix, and the clustering method (Figure 84).

Two coefficients provide similarity based upon densitometric curves; the Pearson product-moment correlation (*Pearson correlation*) and the *Cosine coefficient*.

Four different binary coefficients measure the similarity based upon common and different bands: the *Jaccard*, *Dice*, *Jeffrey's X*, and *Ochiai* coefficients. A fifth coefficient, *Different bands*, is essentially a distance coefficient as it simply counts the number of different bands in two patterns. It is converted into a similarity by subtracting this distance value from 100. If you select one of these binary coefficients, you can enable the *Fuzzy logic* option: instead of a yes/no decision whether two bands are matching or not, the program lets the matching value gradually decrease with the distance between the bands. The *Area sensitive* option makes the coefficient take into account differences in area between two matching bands: if for each matching band the areas on both patterns are exactly the same, the coefficient reduces to a normal binary coefficient; the more the areas differ, the lower the similarity will be.

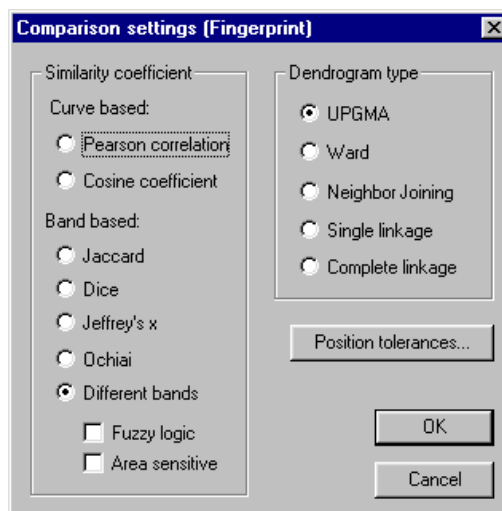


Figure 84. The Comparison settings dialog box.

Among the dendrogram types, the program offers the Unweighted Pair group Method using Arithmetic averages (*UPGMA*), the *Ward* algorithm, the *Neighbor Joining* method, and two variants of UPGMA, namely *Single linkage* and *Complete linkage*.

11.3.5 Select *Dice* and *UPGMA*.

The *Position tolerances* button allow you to specify the maximum allowed distance between the positions of two bands on different patterns, for these bands to be considered as matching.

11.3.6 Press the <*Position tolerances*> button.

The *Position tolerance settings* dialog box for the fingerprint type is popped up (Figure 70).

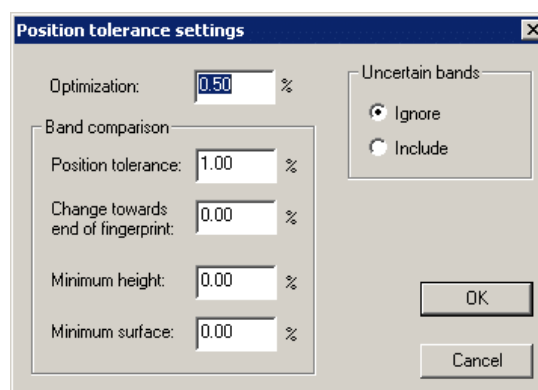


Figure 85. Position tolerance settings dialog box of a fingerprint type.

The *Position tolerance* is the maximal shift (in percentage of the pattern length) between two bands allowed to consider the bands as matching. This parameter only applies to band matching coefficients. With *Change towards end of fingerprint*, you can specify a gradual increase or decrease in tolerance. In paragraph 12.2, we discuss how to have the program automatically calculate the optimal position tolerance settings for your fingerprint type.

The *Optimization* is a shift that you allow between any two patterns and within which the program will look for the best possible matching. This parameter applies for both curve-based and band

matching coefficients. To understand the utility of optimization in addition to tolerance, see the example in Figure 86.

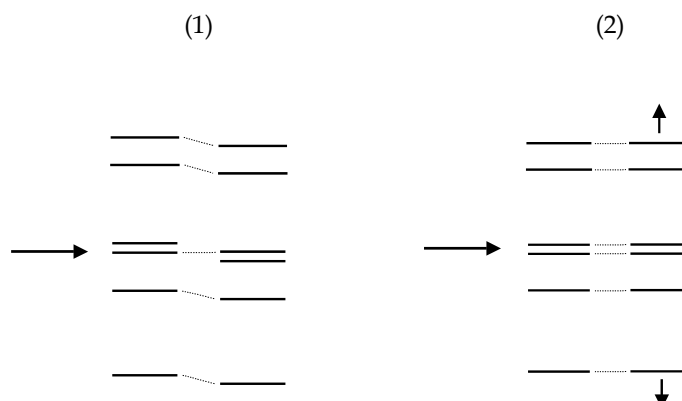


Figure 86. Effect of position tolerance (1) and optimization (2) on the matching between shifted patterns.

In paragraph 12.2, we discuss how to have the program automatically find the best optimization value for your fingerprint type.

With minimum height and minimum surface, you can exclude weak or irrelevant bands.

NOTE: Both the Comparison settings and the Position tolerance settings are stored along with the fingerprint type. The same dialog boxes can be called from the experiment type window settings (7.3.26).

The **Uncertain bands** option allows you to either include uncertain bands or ignore them (see 7.3.1). When **Ignore** is chosen, uncertain bands are ignored. This means that in a pairwise comparison, an uncertain band is not penalized if there is no matching band on the other pattern. Conversely, if there is a band on the other pattern that matches an uncertain band, it will also be ignored in that comparison. When **Include** is chosen, uncertain bands are treated in the same way as certain bands, which means that an uncertain band which is not complemented by a band in the other pattern, is penalized.

*NOTE: The **Ignore** option will only work when both Fuzzy logic and Area sensitive are disabled in the Comparison settings dialog box (Figure 84).*

11.3.7 Enter a position tolerance of 1%, an optimization of 1%, a change of 0%, and a minimum height and minimum surface of 0%, and press **<OK>**.

11.3.8 Press **<OK>** again in the comparison settings dialog box to start the cluster analysis.

During the calculations, the program shows the progress in the Comparison window's caption (as a percentage), and there is a green progress bar between the toolbar and the window panel that proceeds from left to right.

When finished, the dendrogram and the similarity matrix are shown (Figure 87). For more information about the panels in the comparison window, see 9.7.

The experiment type (or composite data set) from which the dendrogram is generated, is shown in the header of the dendrogram panel. The parameters and settings of the cluster analysis are shown in the header of the matrix panel.

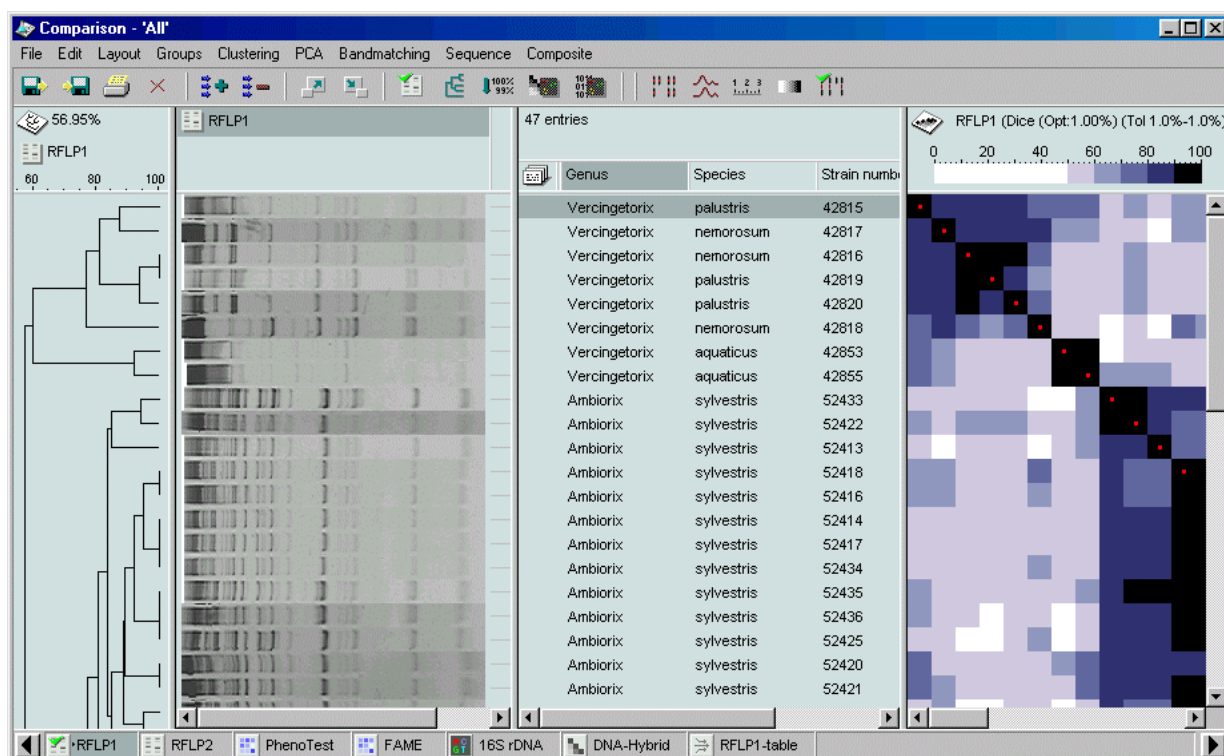


Figure 87. Comparison window with dendrogram, image, entry names, and similarity matrix.

11.4 Calculation priority settings

Fingerprinting II Informatix performs almost all its calculations in multithreaded mode. This means that you can further use Fingerprinting II Informatix or any other programs while the often time-consuming calculations are going on (especially sequence alignments and phylogenetic clustering can take long). In order to speed up the calculations, or make multitasking smoother, you may want to select one of the priority settings for the calculations.

11.4.1 In the main window, select *File|Calculation priority settings*. The *Calculation settings dialog box* (Figure 88) allows 5 priority levels.

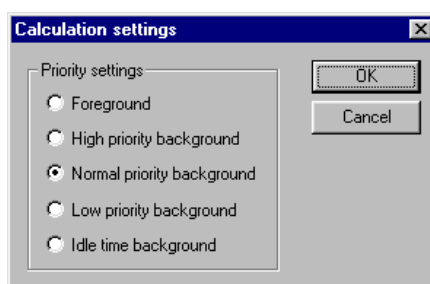


Figure 88. Calculation priority settings dialog box.

If *Foreground* is chosen, it will not be possible to run other applications while the calculations are going on. *Idle time background* means that the computer will only process the Fingerprinting II Informatix calculations while it has nothing else to do.

11.4.2 Select *Normal priority background* and <OK>.

11.4.3 While the program is calculating, you can abort the calculations at any time using the  button.

11.5 General edit functions

11.5.1 You can drag the separator lines between the four panels to the left or to the right, in order to divide the space among the panels optimally.

11.5.2 Similarly, you can drag the separator lines between the information field columns to the left or to the right, in order to divide the space among the information fields optimally.

11.5.3 Left-click on the dendrogram to place a cursor on any node or tip (where a branch ends in an individual entry). The average similarity at the cursor's place is shown in the upper left corner.

11.5.4 You can also move the cursor with the arrow keys.

Many of the display and edit functions, for example zooming in and out, showing bands, are described in paragraph 9.7 (9.7.5 to 9.7.10).

11.6 Adding and deleting entries

11.6.1 First make sure that no entries are selected by pressing the F4 key (9.2.6).

11.6.2 Select some entries from the comparison (see 9.2.1 and 9.2.2).

This can either be done in the information fields panel (use CTRL and Shift keys) or groups can be directly selected on the dendrogram.

11.6.3 To select a cluster on the dendrogram at once, hold the CTRL key and left-click on a branch node. Repeat this action to unselect a branch. Alternatively, right-click on a branch and choose *Select branch into list* from the floating menu.

11.6.4 With *Edit | Cut selection* or , the selected entries are removed from the comparison, and are copied to the clipboard.

11.6.5 With *Edit | Paste selection* or , the same entries are placed back in the comparison. If no dendrogram is present, they are placed at the position of the selection bar.

Note that in this way, you can add new database entries to an existing dendrogram: select the new entries in the database, open an existing comparison with dendrogram, and paste the selection into the comparison. Both the similarity matrix and the dendrogram will be updated, which uses considerably less time than recalculating the whole cluster analysis. The entries first need to be copied to the clipboard from the main window or from another comparison.

11.6.6 To copy entries to the clipboard, use the *Edit | Copy selection* command or .

11.6.7 To cut entries from one comparison into another, use **Edit | Cut selection** or  in the one comparison and **Edit | Paste selection** or  in the other comparison.

11.6.8 To save the comparison with the dendrogram, select **File | Save as** or  and enter a name, e.g. **All**. Comparison “All” now contains a dendrogram for fingerprint type RFLP1.

11.7 Dendrogram display functions

In some cases, it may be necessary to select the root of a dendrogram, for example if you want to (un)select all the entries of the dendrogram. In case of large dendrograms, selecting the root may be difficult using the mouse.

11.7.1 With **Clustering | Select root**, the cursor is placed on the root of the dendrogram.

Two branches grouped at the same node can be swapped to improve the layout of a dendrogram or make its description easier:

11.7.2 Select the node where two branches originate and **Clustering | Swap branches**.

To simplify the representation of large and complex dendrograms, it is possible to simplify branches by abridging them as a triangle.

11.7.3 Select a cluster of closely related entries and **Clustering | Collapse/expand branch**.

Another function, **Clustering | Reroot tree**, only applies to so-called *unrooted trees*, i.e. neighbor joining, parsimony and maximum likelihood. These clustering methods produce trees without any specification as to the position of the root or origin. Since users will want to display such trees in the familiar dendrogram representation, the tree is to be rooted artificially. “Rerooting” is usually done by adding one or more unrelated entries (so-called *outgroup*) to the clustering, and using the branch connecting the outgroup to the others as root. The result is a *pseudo-rooted tree*.

To illustrate the rerooting of an unrooted tree, we will create a second dendrogram, based upon neighbor joining of the fingerprint type RFLP2.

11.7.4 In the experiment selection bar, select experiment RFLP2.

11.7.5 Select **Clustering | Calculate cluster analysis** and specify **Neighbor Joining** in the dialog box. A neighbor joining tree is calculated for RFLP2.

If you scroll through the tree, you will notice that two entries, i.e. *Perdrix sp.* strain numbers 53175 and 25693 protrude on a very long branch. These two entries are ideally suited as “*outgroup*”.

11.7.6 Click somewhere in the middle of the branch. A secondary, X-shaped cursor appears.

11.7.7 Select **Clustering | Reroot tree**, and the new root connects the outgroup with the rest of the entries.

11.7.8 The software automatically limits the similarity range to the depth of the dendrogram. If you want to change this range, select **Clustering | Set minimum similarity value**.

11.7.9 The similarity scale can be displayed in similarity (default for most clustering types) or in distance. To toggle between similarity and distance modes, select **Layout | Show distances**.

Another display function, the creation of *groups*, make the comparison between dendrograms obtained from different experiments much easier.

11.7.10 Show the UPGMA dendrogram of RFLP1 as follows: right-click on RFLP1 in the experiment type selection bar, and select *Show dendrogram*.

This dendrogram reveals three major groups: *Vercingetorix*, *Ambiorix*, and *Perdrix* (with some exceptions).

11.7.11 First make sure that no entries are selected by pressing F4.

11.7.12 Hold the CTRL key and click on the node that connects all the entries belonging to the *Vercingetorix* cluster. The entries of this cluster are now selected.

11.7.13 In the menu, select *Groups | Assign selected to*, and choose a color, for example green (the first color).

11.7.14 Press F4 and CTRL click on the node connecting all *Ambiorix* entries.

11.7.15 Select *Group | Assign selected to*, and choose the second color, red.

11.7.16 Repeat actions 11.7.11, 11.7.12, and 11.7.13 for the third group mainly composed of *Perdrix*. Use blue as color.

11.7.17 You can repeat these actions for two outliers of *Perdrix*, using yellow.

Whatever dendrogram you display now, you will be able to recover the groups of the RFLP1 dendrogram at a glance.

11.7.18 Right-click on RFLP2 in the experiment type selection bar, and select *Show dendrogram*. The *Perdrix* and *Ambiorix* strains are not well separated by this technique: red and blue are mixed up.

The group assignments are saved along with the cluster analysis.

An alternative method to define groups is by selecting a database field and having the program automatically create groups based upon the different names that exist in this database field. One should be aware however, that any misspelled name or typographic error will result in a different group. The method works as follows.

11.7.19 Select a database field by clicking on the database field name, for example **Genus**.

11.7.20 In the *Groups* menu, select *Create from database field*.

The program now answers "*Remove existing groups?*". If you answer **<Yes>**, the existing groups will first be removed and the program will assign the new groups based upon the selected database field only. If you answer **<No>**, the program will keep the groups that are already defined, and split existing groups into more groups if differences in the selected database field are found.

11.7.21 Press **<Yes>** to remove the existing groups. The program creates three groups according to the genus names.

11.7.22 Select the **Species** database field and *Group | Create from database field* again.

11.7.23 Press **<No>** to keep the existing groups.

Every unique species name now is assigned to a different group. In addition, if two different general would have the same species name, they would belong to a different group too, since we kept the existing groups based upon the genus database field.

11.8 Cluster significance tools

A dendrogram tells you something about the groups among a selection of entries, but nothing about the *significance*, i.e. the reliability, the value of these groups. Therefore, the software offers a range of methods that express the stability or the error at each branching level.

The Standard Deviation of a branch is obtained by reconstructing the similarity values from the dendrogram branch and comparing the values with the original similarity values. The standard deviation of the derived values versus the original values is a measure of the reliability and internal consistence of the branch.

11.8.1 Right-click on RFLP1 in the experiment type selection bar, and select *Show dendrogram*.

11.8.2 Select *Clustering | Calculate error flags*.

An error flag is drawn on each branch. The average similarity and the exact standard deviation is shown at the position of the cursor (see Figure 89). The smaller this error flag, the more consistent a group is. For example, the *Perdrix* group has a small error flag, meaning that this group is very consistent. This group will for example not disappear by incidental changes such as tolerance settings, adding or deleting entries etc.

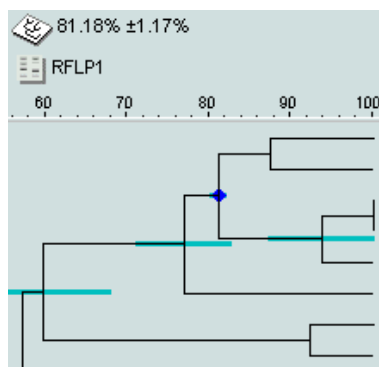


Figure 89. Dendrogram with error flags, detail. The average similarity and standard deviation is shown at the cursor's position (top).

The Cophenetic Correlation is also a parameter to express the consistence of a cluster. This method calculates the correlation between the dendrogram-derived similarities and the matrix similarities. The value is usually calculated for a whole dendrogram, to have an estimation of the faithfulness of a cluster analysis. In Fingerprinting II Informatix, the value is calculated for each cluster (branch) thus estimating the faithfulness of each subcluster of the dendrogram. Obviously, you can obtain the cophenetic correlation for the whole dendrogram by looking at the cophenetic correlation at the root.

11.8.3 Select *Clustering | Calculate error flags* again to remove the error flags.

11.8.4 Select *Clustering | Calculate cophenetic correlations*.

The cophenetic correlation is shown at each branch (Figure 90), together with a colored dot, of which the color ranges between green-yellow-orange-red according to decreasing cophenetic correlation. Thus, it is easy to detect reliable and unreliable clusters at a glance.

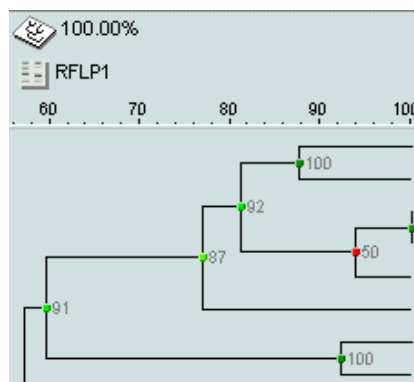


Figure 90. Dendrogram showing cophenetic correlation values, detail.

Bootstrap analysis measures cluster significance at a different level. Instead of comparing the dendrogram to the similarity matrix, it directly measures the influence of characters on the obtained dendrogram. The concept is very simple: “sampling with replacement”, i.e. characters are randomly left out from the character set and replaced with another. For each sampling case, the dendrogram is recalculated, and the relative number of dendrograms in which a given cluster occurs is a measure of its significance. This method requires the characters to be independent and equally important.

Since bootstrap analysis requires a closed character set, the method can only be performed on aligned sequences and composite data sets.

11.8.5 Select **All-Pheno** (a composite data set; see 11.2) in the experiment type selection bar and **Clustering | Calculate cluster analysis**.

11.8.6 When the dendrogram appears, Select **Clustering | Bootstrap analysis** and enter the number of simulations (samplings) to perform. A reasonable number of samplings is 100.

11.8.7 Press **<OK>** and wait till the sampling and calculation process is finished. No need to explain that calculating 100 matrices and dendrograms can take some computing time.

The bootstrap values are shown in a similar way as the cophenetic correlation values (see Figure 90).

Another way of looking at dendrograms is to try to delimit, by objective means, the relevant clusters from the non-relevant clusters. The simplest and most arbitrary method is to draw a vertical line through the dendrogram in a way that it cuts most homogeneous clusters from most heterogeneous clusters. However, there are more statistically founded methods to draw either straight lines, or to evaluate cluster by cluster and delimit relevant clusters at different similarity levels. The *Cluster Cutoff method* in Fingerprinting II Informatix is one of these statistical methods. The method draws a line through the dendrogram at a certain similarity level, and from the resulting number of clusters defined by that line, it creates a new, simplified, similarity matrix, in which all within-cluster values are 100%, and all between-cluster values are 0%. Then, the *Point-bisectional correlation* is calculated, i.e. the correlation between this new matrix and the original similarity matrix. The same is done again for other cutoff similarity levels, and the level offering the highest PBC is the one offering the most relevant groups.

In Fingerprinting II Informatix, this standard method is even refined, as the cutoff values can be different per cluster, to allow even more reliable clusters to be defined.

- 11.8.8 Select *Clustering|Calculate cluster cutoff values*. The branches that were found to be below the cluster cutoff value are shown in dashed lines.

11.9 Matrix display functions

NOTE: If the similarity matrix is not shown for the selected experiment, you can display it with Layout|Show matrix.

The matrix panel is at the right side of the comparison window (see Figure 87).

- 11.9.1 It may be necessary to reduce the space allocated for the image and for the information fields, in order to increase the space for the matrix panel, by dragging the separator lines between the panels.

Initially, the matrix is displayed as differentially shaded blocks representing the similarity values. The interval settings for the shadings is graphically represented in the caption of the matrix panel (Figure 91).

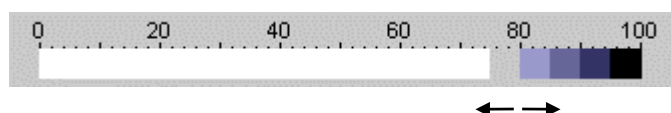


Figure 91. Adjustable similarity shading scale.

There are two ways to change the intervals for shading:

- 11.9.2 Drag the interval bars on the scale; the matrix is updated instantly.
- 11.9.3 Select *Layout|Similarity shades* in the menu. The maximum/minimum values for each interval can be entered as numbers.
- 11.9.4 To show the similarity values in the matrix, select *Layout|Show similarity values*. If it is difficult to read the values on the shaded background, you can remove the shades with *Layout|Similarity shades* and entering 100% for each interval.
- 11.9.5 By double-clicking on a similarity block or value, you can pop up the detailed comparison between the two entries (9.6).
- 11.9.6 To export a tab-delineated text file of the similarity matrix, select *File|Export similarity matrix*.

This text file contains the entry keys as descriptors. You can export a text file which contains the same descriptors with the corresponding information fields:

- 11.9.7 Export the information fields with *File|Export database fields*. If a maximum is specified for a database field, the export file will be abbreviated too for this information field.

11.10 Group statistics

The groups statistics functions are based upon the groups the user has defined. We have explained earlier how to define groups (see 11.7.11 to 11.7.17), and if you have gone through the dendrogram display functions (11.7), the groups are already present on the dendrogram.

K-means partitioning

One function to let the software automatically determine groups is the mathematical function *K-means partitioning*. The user first creates groups based upon one or more strains (e.g. type strains). Then, the program automatically calculates for each entry of the cluster analysis in which group it fits best. This fitting can be based upon *Average* similarity with the group, upon the highest similarity (*Nearest neighbor*), or upon the lowest similarity (*Furthest neighbor*). Obviously, the partitioning process must be iteratively executed, since by adding an entry to a group, the average similarity of the group as well as the highest and lowest similarities with entries may change.

11.10.1 To illustrate the partitioning method, we select RFLP1.

11.10.2 Select the root and select all entries on the dendrogram with CTRL + left-click.

11.10.3 Remove all group assignments with **Groups | Assign selected to | None**.

11.10.4 Select one or a few entries per cluster, each time assigning a different group color to them (see Figure 92 for an example). Do not forget to unselect all entries before you start defining a next group.

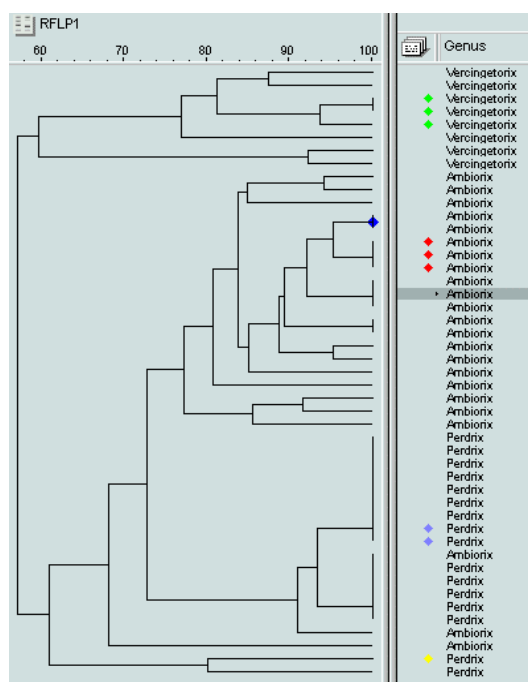


Figure 92. Example of manual group assignment in preparation of a partitioning process.

11.10.5 In the menu, select **Group | Partitioning of groups**, which allows you to choose between the three options described above (Figure 93).

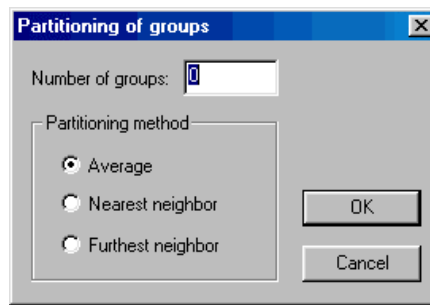


Figure 93. Partitioning of groups dialog box.

11.10.6 Leave *Number of groups* on zero so that the program will only use the groups we have defined manually.

11.10.7 Select *Nearest neighbor*, which will place a new entry in the group containing the highest individual similarity with that entry.

11.10.8 Press <OK> to execute the partitioning. After partitioning, all entries belong to one of the defined groups.

Note that these groups do not necessarily correspond exactly to the visual clusters on the dendrogram. This is not the case if the clusters on the dendrogram are not well-defined or inconsistent. A second reason is the oversimplification of complex matrices by the UPGMA algorithm.

11.10.9 As an alternative, you can also select *Groups|Partitioning of groups*, specifying a predefined number of groups, e.g. 3.

11.10.10 Press <OK> to partition into 3 groups. The program now has defined the 5 3 most relevant groups in the comparison.

Group separation statistics

These statistical methods determine the stability of the defined groups, whether they are defined manually, derived from clusters, using K-means partitioning, or created from an information field. They involve the *Jackknife* method and the “*group violation*” measurement.

11.10.11 With *Groups|Group separation*, the separation between the defined groups are investigated.

The *Group separation settings* dialog box is shown, allowing a number of choices to be made (Figure 94).

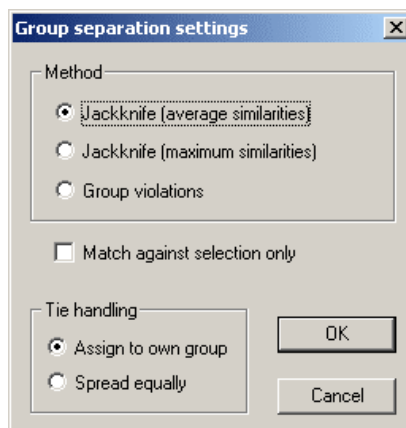


Figure 94. Group separation settings dialog box.

The principle of the *Jackknife* method is to take out one entry from the list, and to identify this entry against the different groups. This can be done by calculating the *Average similarities* with each group, or finding the *Maximum similarities* with each group. This is done for all entries (when *Match against selection only* is not checked). The percentage of cases that entries are identified to the group they were assigned to, is a measure of the internal stability (significance) of that group. Between the groups, the percentage of cases that entries are identified to another group are given.

Using *Match against selection only*, you can let the program calculate the matches against a selection you made in the comparison, rather than against all entries of the groups.

In cases where an entry has an equal match with a member of its own group and a member of another group (a “tie”), there are two equally valid interpretations possible. The program can handle such ties in an ‘optimistic’ way, i.e., by always assigning equal matches to their own group, or in a ‘realistic’ way, by spreading ties equally between their own groups and the other groups.

The way ties are handled can be chosen in the settings dialog box under *Tie handling*. This includes two options, *Assign to own group* and *Spread equally*.

11.10.12 Click <OK> with the default settings to display the *Group separation* window (Figure 95).

Note that the values in the matrix are not reciprocal, i.e., the matrix is not symmetric! The number of misidentifications for members of group A are given in column 1 (Figure 95), for members of group B in column 2, etc. In Figure 95 for example, 25% of group B members are identified as group C, but only 3% of group C members are identified as group B. To facilitate the interpretation of this matrix, the columns are separated by black lines.

When the Jackknife method is used, a value (or cell) in the group separation matrix can be selected, and with the  button or *File|Select cell members*, the entries contributing to this cell will be selected in the comparison window. The method is useful to identify entries that fit well or do not fit well in their assigned groups.

*NOTE: The interpretation of matching and non-matching entries is less easy when the **Spread equally** function has been chosen, since in that case, some entries may fall outside their group “unexpectedly” when they have an equally high score with another group.*

11.10.13 Click  or select *Settings|Statistics* to call the settings dialog box again.

11.10.14 Under *Method*, select *Group violations*. Figure 95 is based on group violations between three groups partitioned as above (RFLP1).

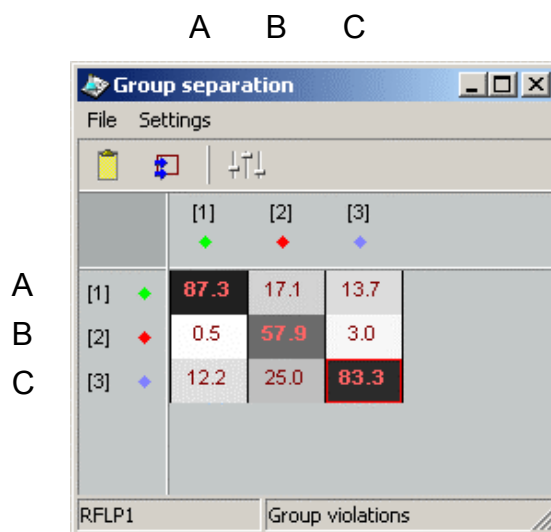


Figure 95. Group separation statistics window.

The group violations method compares all the similarity values within a group with those between the group and the other groups. All the values occurring in the overlap zones (see Figure 96) are considered “violations” of the integrity of the group.

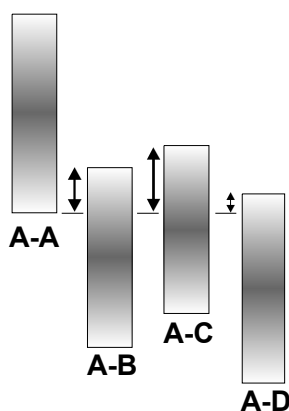


Figure 96. Schematized representation of internal similarity range of group A (A-A), and similarity ranges with other groups (A-B, A-C, and A-D). The overlapping values are group violations.

The percentage of group violations for group A is the number of external entries scoring higher than the lowest internal values over the total number of similarity values considered. The percentages seen in the diagonal of the matrix are the percentages of **non-violations**.

11.10.15 The *Group statistics* can be copied to the clipboard using *File | Copy to clipboard*.

11.11 Printing a cluster analysis

When printing from the comparison window, Fingerprinting II Informatix first shows a print preview. This print preview shows the same as is shown in the panels of the comparison window: for example, a dendrogram, one or more images from different experiments, metrics scale, etc. One exception is the similarity matrix: the print preview does not print matrices unless you explicitly select it in the print preview. The preview looks exactly as it will look on printed pages. You can edit the layout of the print preview by adjusting the space allowed for the different items (dendrogram, image(s), information fields), by changing the size of the figure to fit on one or more pages, etc.

11.11.1 In the comparison window, select **File|Print preview**, which opens the *Comparison print preview window* (Figure 97).

The print preview window is divided in two panels: left, it shows an overview of the pages that will be printed, with the actual page in yellow. Right, the actual page is shown.

11.11.2 With the PgUp and PgDn keys or **Edit|Previous page**  and **Edit|Next page** , you can thumb through the pages that will be printed out.

11.11.3 It is possible to zoom in or out on a page using **Edit|Zoom in**  and **Edit|Zoom out** , or the + and - keys.

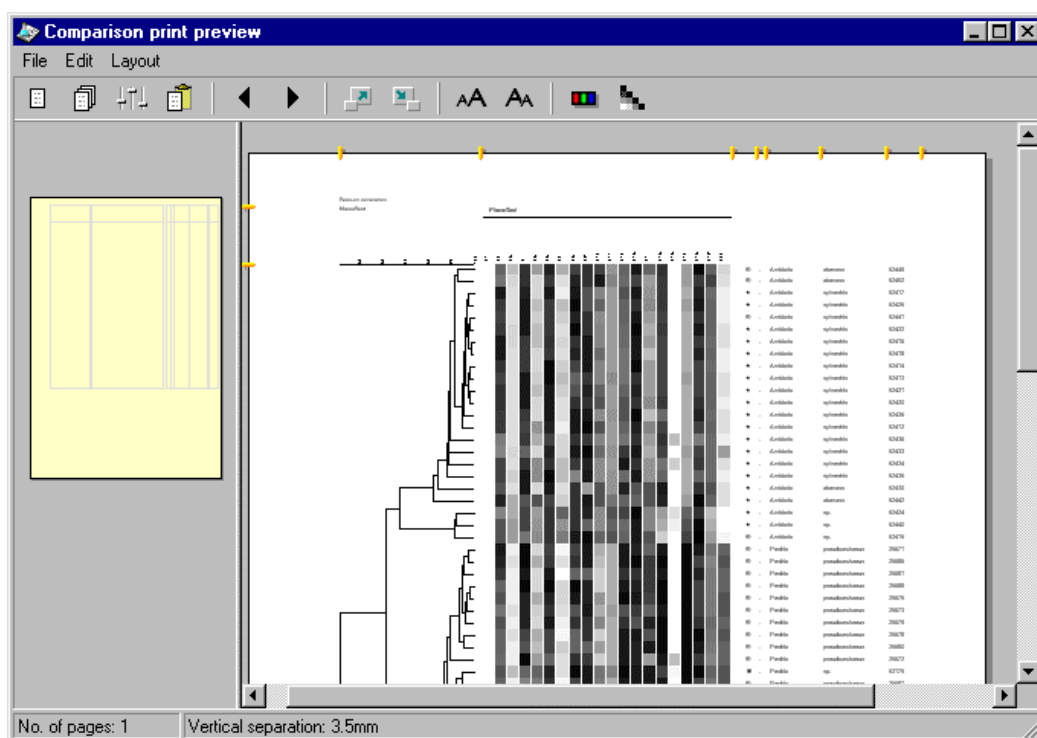


Figure 97. Comparison print preview window.

11.11.4 When zoomed, the horizontal and vertical scroll bars allow you to scroll through the page.

11.11.5 The whole image can be enlarged or reduced with **Layout|Enlarge image size**  or **Layout|Reduce image size** .

11.11.6 If a similarity matrix is available, it can be shown and printed with **Layout|Show similarity matrix** or .

On top of the preview page, there are a number of small yellow slide bars (Figure 97). These slide bars represent the following margins, respectively:

- Left margin of the whole image;
- If dendrogram shown, right margin of dendrogram;
- If image shown, right margin of image;
- If groups are defined, right margin of groups;
- Right margin of entry keys or group codes (if not set to zero length);
- Right margins of different information fields (except those set to zero length);
- If similarity matrix shown, right margin of matrix.

Left on the first preview page, there are two slide bars: representing the top margin of the whole figure and lower margin of the header, respectively. Left on the last page, there is one slide bar representing the bottom margin of the image.

Each of these slide bars can be shifted individually to reserve the appropriate space for the mentioned items. The image is printed exactly as it looks on the preview.

11.11.7 You can preview and print the image in full color with **Layout|Use colors** or .

11.11.8 In addition, the menu command **File|Printer setup** or  allows you to set the paper orientation, the margins, and other printer settings for the default printer.

11.11.9 If the preview is taking more than one page, you can click on a page in the left page preview panel to select a page from the range.

11.11.10 With **File|Print this page** or , the current page is printed.

11.11.11 Use **File|Print all pages** or  to print all pages at once.

11.11.12 If you want to export the image to another software package for further editing, use **File|Copy page to clipboard** or .

This function provides a choice between the Windows *Enhanced Metafile* format, i.e. the standard clipboard exchange format between native Windows applications (default), or a *bitmap* file with 75 dpi, 150 dpi, 300 dpi or 600 dpi resolution. Many software applications, although supporting the enhanced metafile format, are unable to properly import some advanced Fingerprinting II Informatix clipboard files that make use of mixed vector, bitmap and (rotated) text components. If you experience such problems, you should select a bitmap file to be exported, or use another software application to import the graphical data.

With the latter function, only the current page is copied to the clipboard. If you want the whole image to be copied to the clipboard, first reduce the size of the image (11.11.5).

*NOTE: When preferred, the image of a Fingerprint type can be shown and printed with a space between the gelstrips. To do so, open the Experiment type window in the program's main window (under **Fingerprint types**) and select **Layout|Show space between gelstrips**.*

11.11.13 Select **File|Exit** to close the comparison print preview window.

11.12 Analysis of the concordance between techniques

gAs soon as multiple techniques are used to study the relationships between organisms, the question arises how concordant the groupings obtained using the different techniques are. It is also interesting to compare the techniques by the level at which they discriminate the entries, in other words, the taxonomic depth of the techniques.

An evident way to perform such a study is by comparing the similarity matrices obtained from the different experiment types used. By plotting the corresponding similarity values in an X-Y coordinate system, one can easily observe the kind and degree of concordance at a glance. Fingerprinting II Informatix even calculates a regression curve through the plot.

11.12.1 In the main window with **DemoBase** loaded, open comparison **All**, or a comparison with all entries except those defined as STANDARD.

11.12.2 Check whether a cluster analysis is present for each experiment type (except the composite data sets) by right-clicking on the experiment name in the experiment type selection bar (bottom). The floating menu appears.

If the menu items *Show dendrogram* and *Show matrix* display in black (enabled), a cluster analysis is present for the experiment and no further calculation is needed.

11.12.3 If the menu items *Show dendrogram* and *Show matrix* display in gray (disabled), it means that no cluster analysis is present for the experiment. In that case, select **Calculate cluster analysis** from the floating menu.

11.12.4 In the comparison window's menu, select **Clustering|Congruence of experiments**.

The *Experiment congruence* window (Figure 98) shows both a matrix of congruence values between the techniques (experiment types) and a dendrogram derived from that matrix.

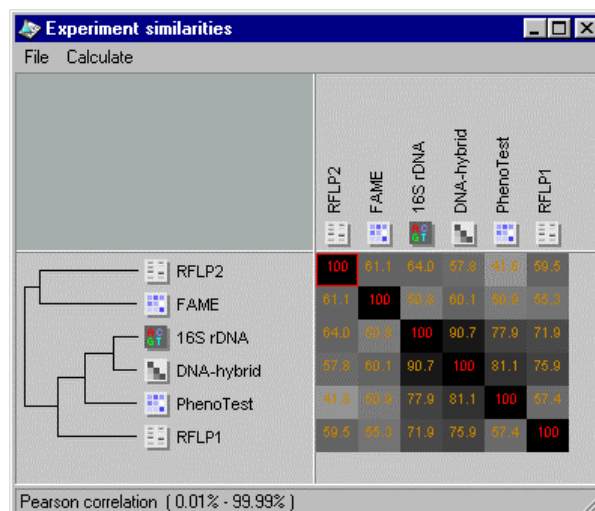


Figure 98. Experiment congruence window.

The default method to calculate the congruence between two experiment types is by using the Pearson product-moment correlation coefficient. An alternative coefficient is *Kendall's tau*. The principle of Kendall's tau is as follows: if value A is higher than value B in experiment 1, then corresponding value A of experiment 2 should also be higher than corresponding value B of experiment 2. The less infringements on this statement, the more congruent the techniques are. Kendall's tau has the advantage over Pearson correlation that non-linear correlations still have good scores. In addition, the significance of the correlation between techniques is shown (green values) when the Kendall's tau is selected, as well as the standard deviation on the values.

11.12.5 Select *Calculate* | *Experiment correlations*.

The *Correlation between experiments* dialog box that pops up shows the settings for the calculation of correlation between the experiment types (Figure 99).

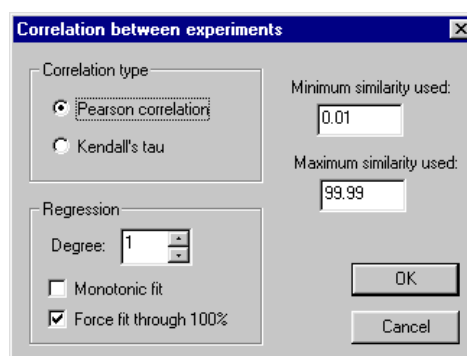


Figure 99. Correlation between experiments settings dialog box.

11.12.6 In the *Correlation type* box, select *Kendall's tau*.

The *Minimum similarity used* and *Maximum similarity used* allow a range of similarity values to be specified within which the analysis is done.

11.12.7 In order to exclude the similarities of the entries with themselves, enter 99.99 as maximum value.

The *Regression* determines the kind of best-fitting curve that is calculated through a plot of two experiment types. You can enter the *Degree* of the regression (first degree is linear, second degree is a quadratic function, etc.). If there is any concordance between techniques, one should expect that the function increases monotonously; with *Monotonous fit*, only such functions are allowed. With *Force through 100%*, the program will force the regression curve to pass through the 100% for both techniques. In other words, if entries are seen identical in one technique, you would expect that they are seen identical in another technique as well.

11.12.8 Select a 3rd degree, *Monotonous fit*, and *Force through 100%*, and press <OK>.

11.12.9 Click on a value in the similarity matrix and *Calculate | Similarity plot*.

The similarity plot between the two selected experiments appears (Figure 100), with a third degree regression drawn through it.

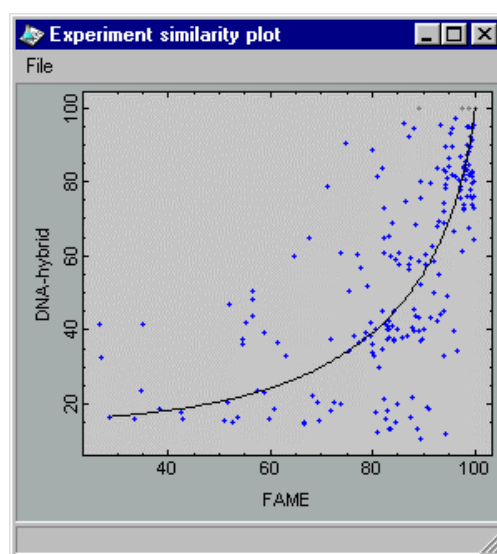


Figure 100. Similarity plot between two experiments.

11.12.10 You can click on any dot in the similarity plot to pop up a detailed pairwise comparison between the two entries (see 9.6).

11.13 Calculating a dendrogram from a composite data set

Calculating a dendrogram from a composite data set is almost the same as for a single experiment. The creation of a composite data set and its functions is described in 11.2, and if you have gone through that paragraph, a composite data set **All-Pheno** should be available in the database, including the character type experiments **Phenotest** and **FAME**.

11.13.1 Run the Analyze program with database **DemoBase** loaded.

11.13.2 Select all entries except STANDARD (9.3) and create a new comparison (9.7), or open comparison **All** if existing.

11.13.3 Select **All-Pheno** in the experiment type selection bar (bottom of window), and show the character image by pressing the  button of All-Pheno.

11.13.4 Right-click on the image to select *Show quantification (colors)* .

11.13.5 Select *Clustering | Calculate cluster analysis*.

The *Comparison settings* dialog box allows you to specify the similarity coefficient to calculate the similarity matrix, and the clustering method (Figure 83). It also allows you to take the similarity matrices from the individual experiments and to calculate an average matrix from these (*Take from experiments*). The meaning of the options are described in 11.2.

If the combined experiments are comparable in terms of biological meaning, reaction type and numerical range, it is possible to use one of the binary coefficients *Jaccard*, *Dice*, *Simple matching*, or one of the numerical coefficients *Pearson correlation*, *Cosine correlation*, or *Canberra metric*. For example, if both experiments involve substrate utilization tests and are recorded either positive or negative, the best option is to *Use binary character table* with Jaccard, Dice or Simple matching as coefficient. If both character tests are registered quantitatively as numerical values between 0 and 100, a suitable option is to *Use character table values* with Pearson correlation, Cosine correlation, or Canberra metric as coefficient.

*NOTE: It can be proven that in case of binary data sets the option **Take from experiments** offers exactly the same results when **Correct for internal weights** is enabled in the composite data set settings (see 11.2.4).*

In the case however, that the *ranges* of the combined experiments are different, e.g. a range between 0 and 10 for one experiment and between 0 and 100 for another experiment, the numerical coefficients Pearson correlation, Cosine correlation, or Canberra are not suitable, as they would assign much more weight to the second experiment than to the first. In such cases, you should either take the similarity values from the individual experiments (*Take from experiments*) and average them into a new matrix, or specify user-defined weights for the experiments, so that their final weights are comparable (see notes on page 141).

The *Categorical* coefficient can be chosen in case all the characters of the individual experiment types are *multistate* characters. As opposed to *binary*, where only two states are known, multistate characters are defined as characters that can take more than two states. However, as opposed to *numerical* characters, the different states represent discrete categories, which cannot be ranked somehow. Examples are Phage types, Multilocus Sequence types (MLST), colors, etc.

11.13.6 In the example data sets Phenotest and FAME, the character sets have different ranges, so select perhaps *Take from experiments*.

11.13.7 Press <OK> to calculate the cluster analysis.

The resulting dendrogram is based upon the average matrix of both similarity matrices. In this composite data set, we have chosen the averaging to correct for internal weights (see 11.2.4), so since the experiment type FAME contains more characters than Phenotest, it is assigned more weight proportionally. Hence, we can expect that the composite clustering will have a higher congruence with FAME than with Phenotest. You can check this as follows:

11.13.8 First make sure that a matrix is present for both FAME and Phenotest: select *Calculate cluster analysis* for both experiments.

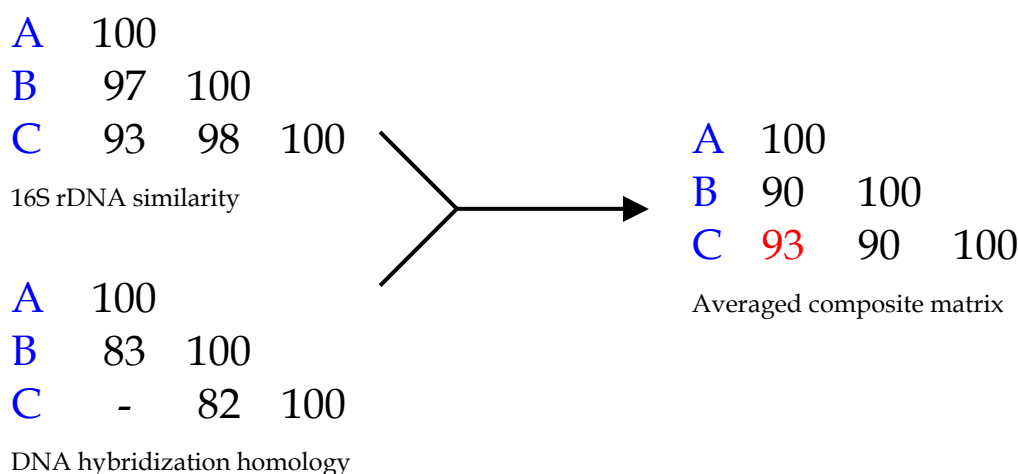
11.13.9 Select *Clustering | Congruence of experiments* (see 11.12). All-Pheno shows 96.4% similarity with FAME and only 71.6% with Phenotest.

*NOTE: If you want to see the difference when **Correct for internal weights** is not enabled, save and close the comparison window, open All-pheno in the Experiments panel, and uncheck **Experiment | Correct for internal weights**. Open the comparison again, Calculate cluster analysis*

again for *All-pheno*, and **Clustering**|**Congruence of experiments**. The similarity of *All-pheno* now is 88.6 and 84.7 with *Phenotest* and *FAME*, respectively.

It is obvious that the possibility of approach 4 described in 11.2, i.e. merging two character sets to a combined character set, is only applicable to comparable character sets. It makes no sense, and is even impossible to combine a phenotypic test panel with a sequencing experiment in this way. When such experiments of different nature are to be used for consensus groupings, the only remaining approach is to combine the obtained individual similarity matrices (approach 3 in 11.2). However, the option to create an *average* matrix from individual experiment matrices only works well in case two conditions are fulfilled: (i) the expected *similarity range* for both experiments is comparable, and (ii) the matrices are complete, i.e. for each experiment there is a similarity value present for each pair of entries. Suppose that two experiment types are to be combined which generate strongly different similarity levels, e.g. DNA homology values on the one hand and 16S rDNA similarity on the other hand. In many cases, DNA homology values will range from 100% to 40% or less, whereas 16S rDNA similarity will range between 100% and 90% or even higher. It is clear that the small but very significant similarity differences in 16S rDNA homology will be masked by the much larger differences (including experimental error) of DNA hybridization, and will have no contribution to the clustering based upon averaging of matrices. In such cases, other methods are needed to compose a consensus matrix, that “takes the best of it all”.

The principle of averaging matrices is even worse when one or more matrices are incomplete. Suppose three entries in Fingerprinting II Informatix, A, B, and C. Consider the following matrices for these three entries, generated from 16S rDNA aligned sequence similarity and DNA hybridization. The DNA hybridization matrix is incomplete, a situation which happens often.



The averaged matrix created in the composite data set from these two experiments shows averaged values for (AB) and (BC) but for (AC) it has taken the only available value, 93%. The resulting matrix provides a completely distorted view of the relationships between these three organisms, as it suggests A and C to be closest related. In reality however, one can predict, based upon the lower 16S rDNA similarity, that (AC) will be much less related than (AB) and (BC).

This is an obvious example where averaging similarity matrices is not a good approach, and therefore, another algorithm has been incorporated in Fingerprinting II Informatix, based upon linearization of the consensus matrix with respect to the individual experiment matrices. The consensus matrix is composed in such a way that it constitutes a third degree function of each individual experiment matrix, and the result is that it reflects each of the constituent matrices as closely as possible.

The *consensus matrix* can be calculated in Fingerprinting II Informatix as follows:

11.13.10 If not existing yet, create a new composite data set “All”, in which you add all experiments available in DEMOBASE.

- 11.13.11 Open a comparison containing all but the STANDARD lanes, and create a matrix (*Calculate cluster analysis*) for each experiment.
- 11.13.12 Select *Composite|Calculate consensus matrix*. The consensus matrix and a corresponding consensus dendrogram is calculated. The resulting groupings can be considered as the most faithful “compromise” from all available data.

NOTE: The feature to correct for internal weights (11.2.4) does not apply to a consensus matrix.

12. Cluster analysis of fingerprints

12.1 Defining 'active zones' on fingerprints

When clustering fingerprints, one is not necessarily interested in comparing the complete patterns. For example, when the loading well or the loading dye is comprised within the fingerprints, it may be better to exclude such a region from the cluster analysis.

It is possible to define *excluded regions* which are applied for all comparisons.

12.1.1 Select any entry in the database that contains a fingerprint of RFLP1.

12.1.2 Open the fingerprint type window for RFLP1 in the experiment types panel.

At the bottom of the window, the fingerprint of the selected database entry is shown (Figure 101).

12.1.3 To exclude a region for comparison, hold the left mouse button and the SHIFT key at the same time while dragging the mouse pointer over the fingerprint.

The excluded region becomes cross-hatched in red. The header of the window shows the parts of the fingerprints that are included for comparison, as percentages (see Figure 101).

12.1.4 To include a region, hold the left mouse button (without holding the SHIFT key), while dragging the mouse pointer over the fingerprint.

12.1.5 You can for example exclude the top 15% and the end 15% of the fingerprints.

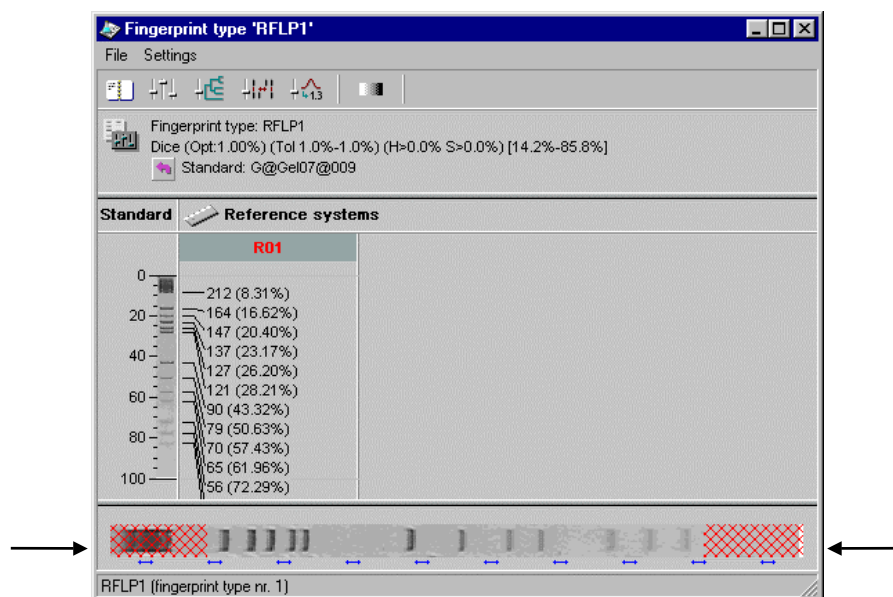


Figure 101. Fingerprint type window with excluded regions defined (see arrows).

NOTE: you can exclude / include multiple regions. The defined regions apply both to comparisons based on densitometric curves and to comparisons based on band matching. Bands falling within an excluded region will not be considered for cluster analysis and band matching analysis

- 12.1.6 Back in the comparison window, select **RFLP1** and *Clustering|Calculate cluster analysis*, to recalculate the dendrogram using the excluded regions.

12.2 Calculation of optimal position tolerance optimization and settings

Fingerprinting II Informatix possesses a very interesting option to calculate the optimal settings for position tolerance and optimization automatically for a given fingerprint type. The principle is as follows: the user selects a number of entries, which he or she wants to cluster into a comparison, and the program will calculate similarity matrices with a range of position tolerance values. We have found that the optimal position tolerance value yields the matrix with the highest group contrast: scores as high as possible within groups and as low as possible between groups. This translates in the highest standard deviation on the matrix of similarity values. The same process can be done to find the best optimization range. Given the principle of the method, it is important to select entries belonging to different groups or showing enough heterogeneity.

The best way to proceed is to create a comparison with *groups* (see 11.7) already defined, e.g. based upon cluster analysis or partitioning (see 11.10). The program will then optimize the intergroup separation based upon these groups. If no groups are defined, the standard deviation of the whole matrix is optimized, which also works in case the comparison contains some groups of more related patterns.

In case you choose a correlation coefficient based on densitometric curves, only the optimization value is needed, and the program will calculate this value. However, in case you apply a band matching coefficient, for example Dice or Jaccard, both the tolerance and optimization values are important. Therefore, the program can also calculate the optimal setting for both values in combination with each other. If n matrices are to be calculated for the tolerance value, and n matrices for the optimization, the combined process requires $n \times n$ matrices to be calculated. In addition, each value from each matrix is to be calculated a number of times within the tolerance/optimization boundaries, in order to find the highest value. No need to argue that this process is extremely time-consuming; it should only be executed on very small numbers of entries.

Given the time needed to calculate $n \times n$ matrices with increasing tolerance applied, we recommend to first calculate the optimization value using Pearson coefficient, and then, using this value, calculate the optimal position tolerance setting. This is done as follows:

- 12.2.1 In the main window with **DemoBase** loaded, open comparison **All**, or a comparison containing all entries except the STANDARDS.
- 12.2.2 In case no groups are defined, select the **Genus** database field and *Groups|Create from database field*.
- 12.2.3 Select **RFLP1** in the experiment type selection bar, and *Clustering|Tolerance & optimization analysis*.

The *comparison settings* dialog box appears (see Figure 84) where you can select the coefficient and clustering method. Only the coefficient is important to calculate the optimization.

- 12.2.4 Select *Pearson correlation* under *Similarity coefficient* and press <OK>.

The program now calculates the best optimization value. When finished, the Position tolerance analysis window appears (Figure 102) showing the group separation in function of the allowed optimization in the right diagram.

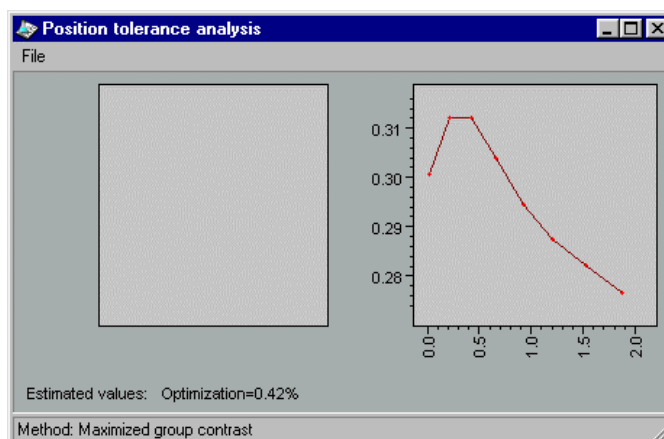


Figure 102. Position tolerance analysis. Optimization analysis shown for curve-based coefficient.

The ideal optimization value is shown (bottom) and is automatically saved in the settings for the experiment type.

12.2.5 Close the window with *File|Exit* and select *Clustering|Tolerance & optimization analysis* again.

12.2.6 This time, select the *Dice* coefficient and press <OK>.

12.2.7 The program asks "*Do you wish to estimate the optimization parameter?*". Answer <No>.

The program now calculates the best position tolerance value for band matching. When finished, the Position tolerance analysis window (Figure 103) shows the group separation in function of the allowed band matching tolerance in the left diagram.

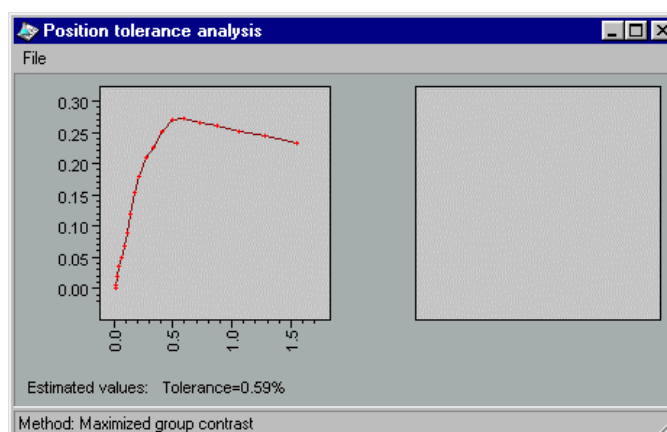


Figure 103. Position tolerance analysis. Position tolerance analysis shown for band matching coefficient.

The position tolerance value is shown (bottom) and is automatically saved in the settings for the experiment type.

12.2.8 Close the window with *File|Exit*.

13. Cluster Analysis of characters

13.1 Coefficients for character data and conversion to binary

In terms of parameter settings, character sets are the simplest class of data to analyze. The various types of character sets that exist, however, require that a large number of coefficients should be available for analyzing character tables.

13.1.1 When selecting a character set to analyze in the *comparison* window, for example Phenodata, and selecting *Clustering*|*Calculate cluster analysis*, the *Comparison settings* dialog box appears (Figure 104).

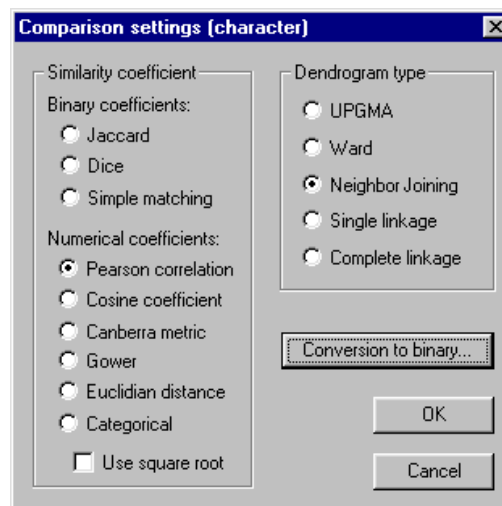


Figure 104. Comparison settings dialog box for character data.

Binary coefficients include *Jaccard*, *Dice*, and *Simple matching*. Dice and Jaccard are very related to each other whereas Simple matching is more fundamentally different. The Jaccard and Dice coefficients only consider "scoring characters" being two positive characters in both data sets, whereas the Simple matching coefficient also two negative characters as scoring. When dealing with a non-binary (numerical) data set, a conversion needs to be done from numerical values to binary values (positive or negative).

13.1.2 The *<Conversion to binary>* button lets you specify how this conversion is done (Figure 105).

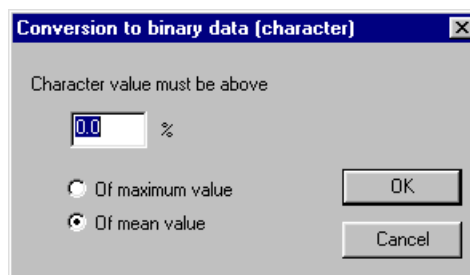


Figure 105. Conversion to binary dialog box.

By default, every character that has a value above zero will be converted to positive. Alternatively, one can specify a certain percentage of either the *maximum value* or the *mean value* from the experiment.

Numerical coefficients include *Pearson correlation* (product-moment correlation) and the related *Cosine coefficient*, the *Canberra metric* and *Gower* coefficients, and *Euclidian distance*. For comparisons between highly related organisms, it can be useful to check the option *Use square root*, especially when using Pearson correlation or Euclidian distance. This has the effect that narrow branches on a dendrogram are stretched out relatively more than distant links.

The *Categorical* coefficient is neither binary nor numerical, since it treats each different value as a different *state*. This coefficient is useful for analyzing *multistate* character sets, for example colors (red, green, blue etc.) are each categorical states. Typical multistate characters used in typing, taxonomy and phylogeny are phage typing, Multilocus Sequence Typing (MLST), Variable Number Tandem Repeats (VNTR) typing. The types or categories assigned to the different phage reactions, allele numbers, or repeat numbers, respectively, in the aforementioned techniques, are good examples of categorical or multistate data which can be analyzed using the categorical coefficient.

13.2 Advanced analysis of massive character sets using GeneMaths

The analysis of huge data sets such as produced by gene chips or high-density gene arrays (micro-arrays) requires special clustering algorithms capable of processing many thousands of entries or characters. In addition, the successful exploration of such data sets also depends on the ability to associate certain clusters of characters (observations) with groups of entries (samples). Although these features are available in the Comparison functions of Fingerprinting II Informatix, the flexibility of handling and clustering extremely large matrices as well as some sophisticated functions are something which is provided in a separate program, GeneMaths. The GeneMaths program is capable of clustering data sets of up to a million characters per entry.

GeneMaths is available a standalone program, but can also be added as a module to the Fingerprinting II Informatix software. Please contact Bio-Rad Laboratories for further information.

14. Cluster analysis of sequences

Among all types of experimental data, cluster analysis of sequence data is by far the most complex in steps and possibilities. The fact that sequences need to be *aligned* before one can estimate similarity requires a number of additional steps before a dendrogram is achieved. Furthermore, sequence data are a suitable substrate for a number of phylogenetic clustering algorithms which can seldom be applied to other types of data.

There are two ways to obtain a dendrogram from sequence data: by aligning the sequences *pairwise* (steps 1-2 in Figure 106), or by first obtaining a *multiple alignment* of all sequences (steps 1-6 in Figure 106).

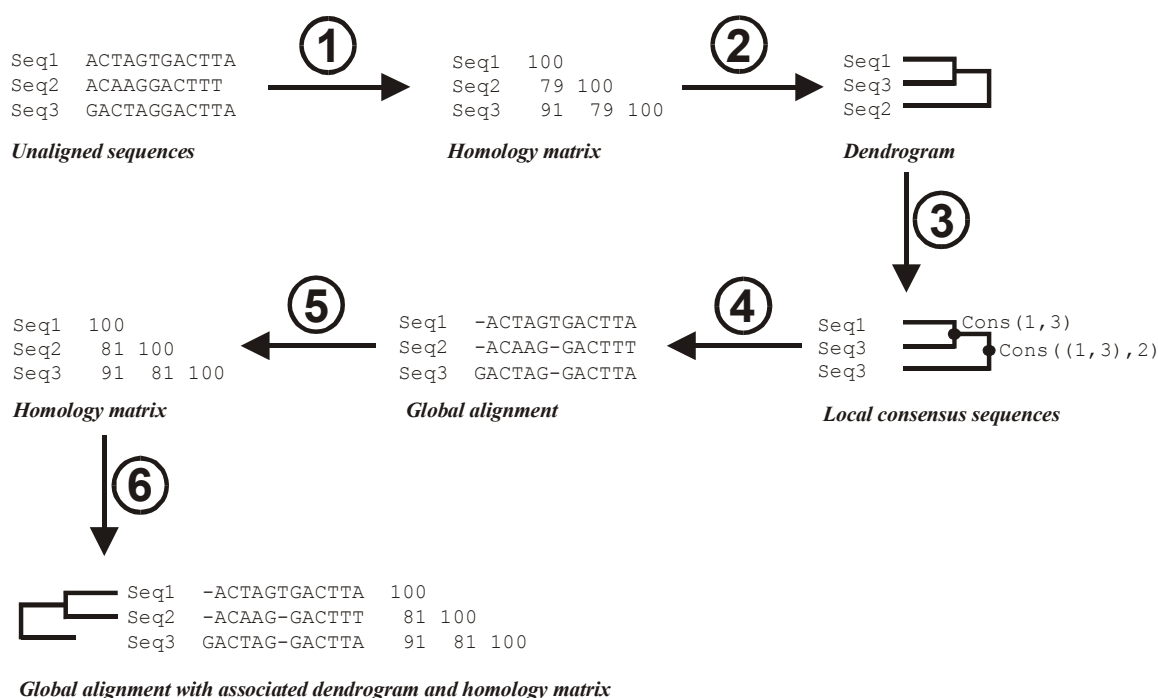


Figure 106. Steps in a cluster analysis of sequences: dendrogram based on pairwise alignment (steps 1 to 2), and dendrogram based on multiple alignment (steps 1 to 6).

The best multiple alignments that can be achieved, particularly for large numbers of sequences, involve the following steps.

1. Pairwise alignment and calculation of similarity of all possible pairs of sequences, resulting in the *Pairwise alignment similarity matrix*.
2. Construction of a *UPGMA dendrogram* based on the similarity matrix.
3. Determination of *consensus sequences* at each linkage node of the dendrogram, down to the root.
4. Alignment of all sequences based on the local and the root consensus sequences.
5. Calculation of a similarity matrix based on the aligned sequences, the *Multiple alignment similarity matrix*.
6. Construction of a Neighbor Joining dendrogram based on the multiple alignment similarity matrix.

In step 1, each individual sequence is aligned with each other sequence, and for each pair of aligned sequences, the similarity value is calculated into a similarity matrix. The obtained matrix of similarity values based on pairwise alignment (*pairwise alignment similarity matrix*) will serve as the basis for cluster analysis by the Unweighted Pair Group Method using Arithmetic averages (UPGMA) (step 2). Neighbor Joining or other algorithms resulting in unrooted dendrograms would not be suitable here, as in such dendrograms, the closest linked sequences are not necessarily the most related ones. This is a requirement for step 3 discussed below.

Steps 3 and 4 are very important for obtaining a sensible global alignment. Each linkage node on the UPGMA dendrogram represents a *local alignment* of the sequences linked at the node, resulting in a *local consensus*. These local consensus sequences are calculated downwards, i.e. starting from the highest related sequences down to the dendrogram root (step 3). In the above example, the highest linkage observed is between sequences 1 and 3, leading to consensus (1,3). The next linkage level is the branch that links sequences 1 and 3 with sequence 2. At this node, the consensus (1,3) is aligned with sequence 2. This results in a consensus ((1,3),2), which will in turn be aligned with the consensus of another group linked to this one. For each sequence or local consensus, the program keeps track of the positions of the gaps that are introduced to align it with the branch it is linked to. Finally, a *global consensus* for the whole dendrogram is inferred.

The program now introduces to each individual sequence all the gaps that were introduced on the subsequent consensus sequences following the path from the sequence itself down to the global consensus (step 4). This results in a *global* or *multiple alignment*.

The multiple alignment in turn can be used as the basis for the calculation of a similarity matrix. Now, instead of aligning each sequence with each other sequence to determine their mutual similarity, the multiple alignment is used to calculate the *multiple alignment-based similarity* between each pair of sequences (step 5). Once the multiple alignment is present, this step is extremely fast. The *multiple alignment-based similarity matrix* can be used for Neighbor Joining or UPGMA clustering, or other clustering algorithms (step 6).

14.1 Calculating a cluster analysis based on pairwise alignment (steps 1 and 2)

14.1.1 Open comparison **All**, or any selection containing the three genera in the database **DemoBase**.

Select 16S rDNA and **Layout** | *Show image*.

Initially, the sequences are not aligned and no similarity matrix exists.

*NOTE: it is possible that a dendrogram (and a matrix) are still displayed in the comparison window. This is the dendrogram of the last clustered experiment, which you can remove with **Layout**|**Show dendrogram** and **Layout**|**Show matrix**.*

14.1.2 The similarity matrix is calculated with **Clustering** | *Calculate cluster analysis* or the  button.

The *Pairwise comparison settings* dialog box appears (Figure 107), showing three groups of settings: the *Pairwise alignment settings*, the settings for *Similarity calculation*, and the *Clustering* method.

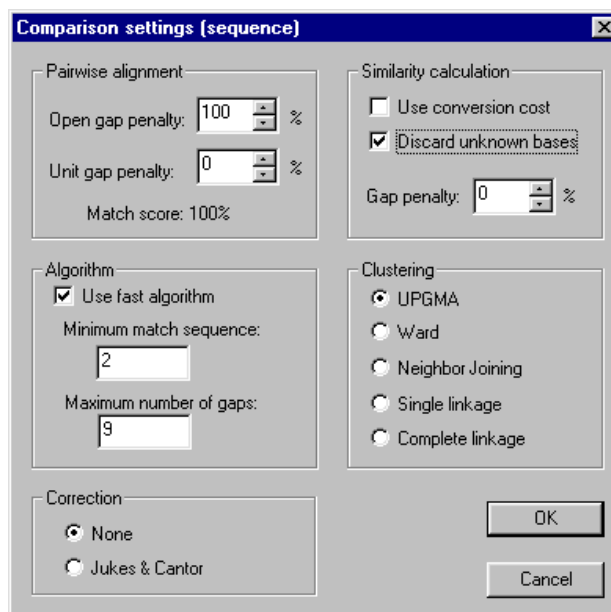


Figure 107. The pairwise similarity comparison settings dialog box.

The *pairwise alignment settings* involve an *Open gap penalty* and a *Unit gap penalty*. A match between bases on two sequences, e.g. A with G, is considered as 100% score. The open gap penalty is the percentage cost of that score if one single gap is introduced in one of both sequences. The unit gap penalty is the percentage cost of that score to increase the gap by one base position. The default setting is 100% open gap penalty and 0% unit gap penalty, which means that introducing a gap in one of both sequences has the same cost as a mismatch, whereas there is no extra cost for gaps of multiple positions. It should be emphasized that the pairwise alignment settings will only determine the way the alignment is done: if a large unit gap cost is set (e.g. 350%), the program would not easily introduce gaps between sequences; for example, the program would rather allow three successive mismatches than one single gap. If no gap cost is chosen (0%) the program would introduce gaps to match every single base. The pairwise alignment settings have no direct influence on the similarity values, but of course, if the obtained alignments differ, the similarity values may differ too.

Use fast algorithm offers an interesting accelerated algorithm, with two adjustable parameters: the *Minimum match sequence* and the *Maximum number of gaps*. The program creates a *lookup table* of groups of bases for both sequences. The *minimum match sequence* is the size of such a group. The smaller the groups are, the more precise the alignment will be, but the longer the alignment will take. The parameter can be varied between 1 and 5, with 2 as default. The *maximum number of gaps* is the maximum number of possible gaps that you allow the algorithm to introduce in one of both sequences. The values can be varied between 0 and 99 with 9 as default. The larger the number, the more gaps the program can create to align every two sequences, but the longer the alignment will take. If zero is selected, no gaps at all would be introduced. Thus, you can custom-define its accuracy between very fast and fairly rough to slow and very accurate.

Contrary to the pairwise alignment settings, the *Similarity calculation* parameters will not influence the alignments, but they determine the way the similarity is calculated. The *Gap penalty* is a parameter which allows you to specify the cost the program uses when one single gap is introduced. This cost is relative to the score the program uses for a base mismatch, which is equal to 100%. The program uses 0% as default. When *Discard unknown bases* is disabled, the program will use a predefined cost table for scoring uncertain or unknown bases. For example, N with A will have 75% penalty, as there is only 25% chance that N is A. Y and C will be counted 50% penalty because Y can be C or T with 50% chance each. If this setting is disabled, all uncertain and unknown bases will not be considered in calculating the final similarity. *Use conversion cost* is a parameter which makes calculation of the pairwise similarity matrix faster. Both described alignment methods work in two

steps: first they determine the total maximal conversion score to convert one sequence into the other (given the current alignment settings) and then they realize the alignment using the minimal gap cost and maximal matching score. If *Use conversion cost* is enabled, the calculated conversion cost is transformed into a similarity value. This method is two times faster than the usual similarity calculation, but the obtained values cannot be described as real "similarity".

Under *Correction*, one can select the *one parameter* correction for the evolutionary distance as calculated from the number of nucleotide substitutions as described by *Jukes and Cantor* (1969)⁴. The resulting dendrogram displays a distance scale which is proportional to an evolutionary time, rather than a similarity scale.

As *Clustering* method, you can choose between UPGMA, Ward, Neighbor Joining, single linkage and complete linkage.

14.1.3 Select an *Open gap penalty* of 100, a *Unit gap penalty* of 0, *Minimum match sequence* of 2, *Maximum number of gaps* of 9, enable *Discard unknown bases*, with a *Gap penalty* of 0 for similarity calculation, *None* for correction, and select *UPGMA* as Clustering method.

14.1.4 Press <OK> to calculate the matrix and the dendrogram.

When the calculations are finished, the dendrogram and the matrix are shown. The sequences are still unaligned since no multiple alignment is calculated yet.

14.2 Calculating a multiple alignment (steps 3 and 4)

14.2.1 Select *Sequence | Multiple alignment* or . The *Global alignment settings dialog box* (Figure 108) appears.

When a multiple alignment is calculated, individual sequences and local consensus sequences are aligned pairwise, down to the root, to obtain a global consensus (see steps 3 and 4 on Figure 106). It is this pairwise alignment of local consensus sequences that uses the same two parameters as explained before: the *Open gap penalty* and the *Unit gap penalty*.

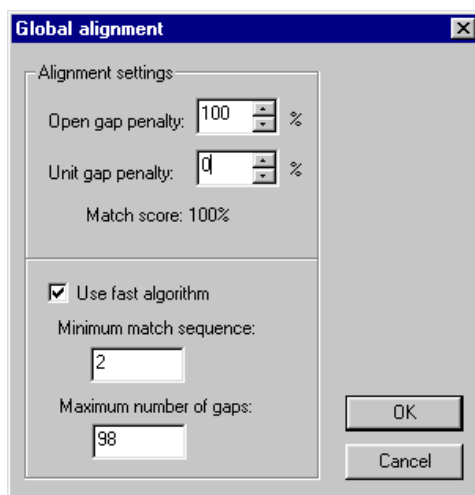


Figure 108. Global alignment settings dialog box.

⁴ Jukes, T.H. and C.R. Cantor. 1969. In "Mammalian Protein Metabolism III" (H.N. Munro, ed.), p. 21. Academic Press, New York.

The significance of the open and unit gap penalties is the same as explained for pairwise alignment: they are the percentage of the mismatch cost to create a gap, and to increase the gap by one base position, respectively. The default setting is 100% open gap penalty and 0% unit gap penalty, which means that introducing a gap in one of both sequences has the same cost as a mismatch, whereas there is no extra cost for gaps of multiple positions. These pairwise alignment settings will only determine the way the alignment of the local consensus sequences is done: if a large unit gap cost is set (e.g. more than 100%), the program would not easily introduce gaps between sequences. If no gap cost is chosen (0%) the program would introduce gaps in order to match single bases. The pairwise alignment settings have no direct influence on the similarity values obtained from a global alignment, but if the eventual multiple alignment differs, the derived similarity values may differ too.

Use fast algorithm is an algorithm with two adjustable parameters: the *Minimum match sequence* and the *Maximum number of gaps* (see also under pairwise alignment). The *minimum match sequence* can be varied between 1 and 5, with 2 as default. The *maximum number of gaps* can be varied between 0 and 198 with 98 as default. The smaller the first number and the larger the number, the more accurate the multiple alignment should be. If the default values are not satisfactory, some experimenting is recommended.

Note that the *Global alignment settings dialog box* does not contain settings for similarity calculation, unlike the *Pairwise alignment settings dialog box*. The similarity matrix based upon the global alignment is not calculated automatically by the program, but requires a further command by the user (step 5 in Figure 106).

14.2.2 Press <OK> to start the multiple alignment. When the calculations are done, the sequences are aligned in the image panel.

Display options

In order to facilitate visual interpretation of global alignments there are three methods to highlight homologous regions.

Select *Sequence | Neighbor blocks* to show the *Neighbor match* representation.

This representation shows bases as blocks (highlighted) if at least one of the neighboring sequences has the same base at the corresponding position. Between two different groups of consensus, a small black line is drawn (Figure 109)

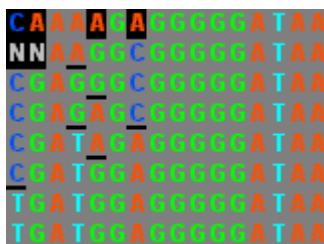


Figure 109. Neighbor match representation.

The *Consensus match* first requires a consensus sequence to be present. A consensus sequence is defined from one or more sequences, and in case more than 50% of the sequences have the same base at a given position, this base will be defined in the consensus. Usually, one will select the root to calculate a consensus from. This method highlights bases (shown as blocks) on the aligned sequences if they are the same as on the consensus sequence.

14.2.3 Select the root and *Sequence | Create consensus of branch*. A consensus sequence of the root is shown on the header of the image panel.

14.2.4 Select *Sequence | Consensus blocks* to show the consensus match representation (Figure 110).

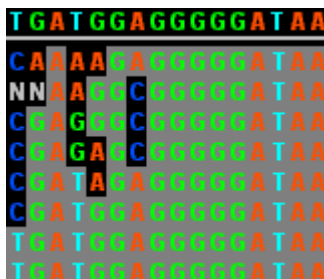


Figure 110. Consensus match representation.

The *Consensus difference* also displays the consensus sequence in the editor caption, and only shows bases that differ from the consensus while bases that are the same as the consensus are shown as |. Example:

14.2.5 Select *Sequence | Consensus difference*. The consensus difference representation is as in Figure 111.

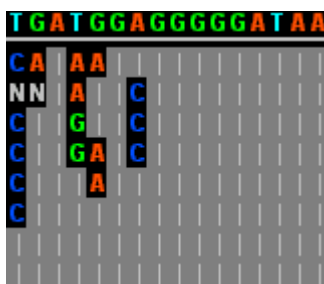


Figure 111. Consensus difference representation.

14.2.6 A consensus sequence can be copied to the clipboard with *Sequence | Copy consensus to clipboard*.

Bases for which there is a consensus in more than 50% of the sequences are named, the other bases are unnamed (N).

Editing a multiple alignment

A multiple alignment can be edited manually and is saved along with the comparison.

14.2.7 Select *File | Save* or  to save the multiple alignment.

In order to rearrange the multiple alignment as desired, any sequence can be moved up or down:

14.2.8 Left-click on the entry you want to move up or down.

14.2.9 Press the  button to move the entry up, or the  button to move it down.

14.2.10 To move a sequence to the top or the bottom of the alignment, hold the CTRL key and press the up or down button, respectively.

Note that, as soon as an entry is moved up or down, the dendrogram disappears: a dendrogram imposes a certain order to the entries, which is not compatible with freely moving sequences up or down. You can display the dendrogram again using *Layout|Show dendrogram*, however, this will reorder the entries again so that any manual changes you made to the sequence order is lost.

14.2.11 A cursor, visible as a white rectangle can be placed on any base of any sequence, and can be moved up, down, left, and right using the arrow keys.

All changes made to a sequence, i.e. inserting gaps or deleting gaps, cause shifts no further than the next gap. You can consider an aligned sequence as a series of blocks with some space in between (the gaps), just like carriages on a railway: if one block is shifted to the right, it will move alone until it touches the next block, which will then move together, until they touch the next block etc. The following alignment editing tools are available:

A. Inserting gaps

 Inserts a gap at the position of the cursor, by shifting the block right from the cursor position to the right. This function can be used on a gap as well as on a base. In the latter case, the base at the cursor position will also shift to the right, i.e. the gap will be inserted left from it. Keyboard: INSERT.

Example: 



Result: 

 Inserts a gap at the position of the cursor, by shifting the block left from the cursor position to the left. This function is similar to the previous function. Keyboard: HOME.

Example: 



Result: 

 Inserts gaps at the position of the cursor, by shifting the block right from the cursor position to the right, until it closes up with the next block. Keyboard: SHIFT+INSERT.

Example: 



Result: 

 Inserts gaps at the position of the cursor, by shifting the block left from the cursor position to the left, until it closes up with the next block. Keyboard: SHIFT+HOME.

Example:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCTTCTCT~TGA~GAGCGGCGGAC



Result:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCTTCTCTTGA~GAGCGGCGGAC

B. Deleting gaps

 Deletes a gap by shifting the block right from the gap to the left. Keyboard: DEL.

Example:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCT~TCTC~TTGA~GAGCGGCGGAC



Result:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCT~TCTC~TTGAGAGCGGCGGAC~

 Deletes a gap by shifting the block left from the gap to the left. Keyboard: END.

Example:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCT~TCTC~TTGA~GAGCGGCGGAC



Result:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCT~TCTC~TTGAGAGCGGCGGAC

 Deletes all gaps right from, and including the cursor, by shifting the block right from the gap to the left. Keyboard: SHIFT+DEL.

Example:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCTTCTCTTGA~GAGCGGCGGAC



Result:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCTTCTCTTGA~GAGCGGCGGAC~

 Deletes all gaps left from, and including the cursor, by shifting the block left from the gap to the right. Keyboard: SHIFT+END.

Example:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ABCTTTCCTTCTCTTGA~GAGCGGCGGAC



Result:

CBCTTTCGCTTTTGTGACGAGCGGCGGAC
 ~AGCTTTCCTTCTCTTGA~GAGCGGCGGAC

14.2.12 To insert and delete gaps or move blocks of a group of sequences as a whole, it is possible to lock a branch on the dendrogram by selecting the branch on the dendrogram (click on the dendrogram node) and *Sequence | Lock / unlock dendrogram branch*.

Locked branches are displayed in red and can be unlocked using the same command.

When no dendrogram is present for a set of aligned sequences, it is also possible to create groups of locked sequences, as follows.

14.2.13 Make sure no dendrogram is present with the multiple alignment. If a dendrogram is present, right-click in the dendrogram panel and select *✓Show dendrogram*.

14.2.14 Select a consecutive group of entries using CTRL + left-click or SHIFT + left-click. When selected, the entries are marked with blue arrows.

14.2.15 In the *Sequence* menu, select *Create locked group*. Locked sequences are connected by a red brace in the left panel.

14.2.16 To unlock locked groups of sequences, click on any of the entries within the group, and select *Sequence | Unlock group*.

Note that locked groups are not the same as locked branches on the dendrogram (14.2.12). When the dendrogram is shown, the locked groups will not be seen anymore, whereas clusters on the dendrogram that were locked previously, become visible and active. When the dendrogram is removed again, the locked groups become visible and active again.

Locked groups have the advantage over locked dendrogram branches that the sequences within a locked group are not restricted to clusters from the dendrogram. One can rearrange the sequences in the multiple alignment as desired, and then create groups of locked sequences.

Changing sequences in a multiple alignment

In some cases, it is possible that ambiguous positions in certain sequences can be filled in when a multiple alignment of highly homologous sequences is present. Fingerprinting II Informatix offers the possibility to change bases in sequences within a multiple alignment.

14.2.17 Place the cursor on any base in the multiple alignment.

14.2.18 Hold the CTRL key and type a base letter, a space (gap) or any letter corresponding to the IUPAC nucleotide naming code.

The sequence is now changed in the multiple alignment, but not yet in the Fingerprinting II Informatix database.

14.2.19 In order to reload the original sequence, select *Sequence | Reload sequence from database*.

14.2.20 To save the changed sequence to the database, select *Sequence | Save changed sequences*.

As an alternative, a base in a sequence can also be changed by double-clicking on that base, which will pop up the experiment card of the sequence, with the clicked base selected. To change the base, simply type in another base from the keyboard. Upon exiting the experiment card, the software will ask to save the changes. These changes, however, are not updated in the multiple alignment. You have to close and reopen the multiple alignment for the changes to become visible.

Finding a subsequence

In order to find certain subsequences in a sequence from a multiple alignment, e.g. restriction sites, primer sequences, repeat patterns etc., you can perform a subsequence search.

14.2.21 First, select a sequence within the multiple alignment (white rectangular cursor).

14.2.22 The subsequence search dialog box (Figure 112) is popped up with *Sequence|Find sequence pattern*.

You can enter any sequence including unknown positions, which are entered as a question mark. You also can allow a number of mismatches to occur in matching subsequences, by specifying a number under *Mismatches allowed*.

For rare subsequences which you do not expect to occur more than once, select *Complete sequence*. For frequently occurring subsequences, you can place the cursor at the start of the sequence, and check *Right from cursor*. By successively pressing *Find*, all subsequent matching patterns will be shown. Similarly, *Left from cursor* shows the first matching pattern left from the cursor, whereas *Closest to cursor* only shows the matching pattern closest to the cursor, in any direction.

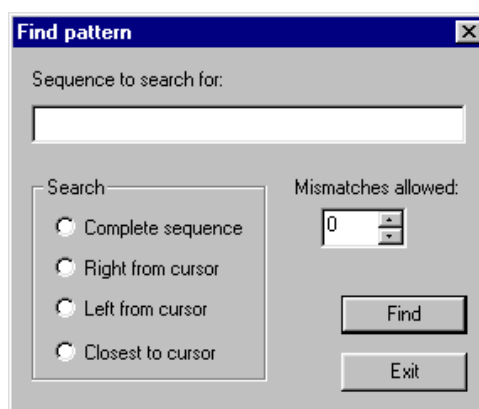


Figure 112. Subsequence search dialog box.

14.3 Calculating a clustering based on the multiple alignment (steps 5 and 6)

The mutual similarities between all the sequences are calculated from the aligned sequences as present in the multiple alignment.

14.3.1 Select *Sequence|Calculate global cluster analysis* or ; the *Global alignment similarity* dialog box is shown (Figure 113):

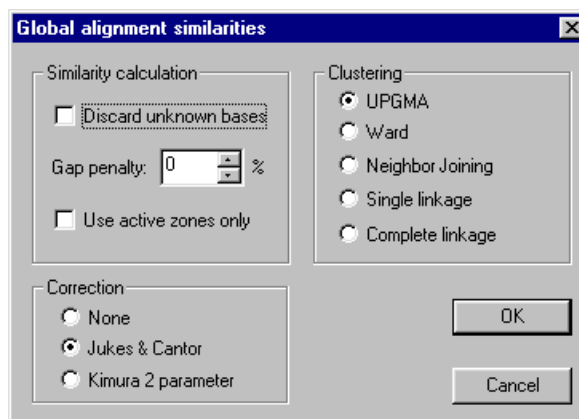


Figure 113. Global alignment similarity dialog box.

These settings determine the way the similarity is calculated between the pairs of sequences. The *Gap penalty* is a parameter which allows you to specify the cost the program uses when one single gap is introduced. This cost is relative to the score the program uses for a base matching, which is equal to 100%. The program uses 0% as default. When *Discard unknown bases* is disabled, the program will use a predefined cost table for scoring uncertain or unknown bases. For example, N with A will have 75% penalty, as there is only 25% chance that N is A. Y and C will be counted 50% penalty because Y can be C or T with 50% chance each. If this setting is disabled, all uncertain and unknown bases will not be considered in calculating the final similarity. To obtain a dendrogram, you can choose between the available clustering algorithms.

The checkbox *Use active zones only* is only applicable when a reference sequence is defined, and when certain zones on this reference sequence are excluded for analysis (see further on page 182).

Under *Correction*, one can select the *Jukes and Cantor* (1969)⁵ correction, a *one parameter* correction for the evolutionary distance as calculated from the number of nucleotide substitutions as. Alternatively, the *Kimura 2 parameter* correction (Kimura, 1980)⁶ can be selected. In either case, the resulting dendrogram displays a distance scale which is proportional to an evolutionary time, rather than a similarity scale.

14.3.2 Check *Discard unknown bases*, select a *Gap penalty* of 0, and leave *Use active zones only* unchecked. Apply no *Correction* and select *Neighbor Joining* as clustering method.

14.3.3 Press <OK> to start calculating the multiple alignment-based dendrogram. This calculation is usually fast.

14.4 Adding entries to and deleting entries from an existing global alignment

The feature of Fingerprinting II Informatix that makes it possible to add entries to (or delete entries from) an existing cluster analysis also applies to sequence clusterings: it is not necessary to recalculate the complete similarity matrix because the program calculates the similarity of the new sequence(s) with each of the other sequences and adds the new similarity values to the existing matrix. Particularly in case of sequence clusterings, this feature is extremely time-saving and causes no degeneration of the clusterings.

⁵ Jukes, T.H. and C.R. Cantor. 1969. In "Mammalian Protein Metabolism III" (H.N. Munro, ed.), p. 21. Academic Press, New York.

⁶ Kimura, M. J. 1980. Mol. Evol. 16: 111.

In case a multiple alignment exists, the problem is more complex. As soon as sequences are added, the program will have to recalculate the multiple alignment (steps 3 and 4 of scheme in Figure 106) to find the optimal alignments again for the new set of sequences. This would cause corrections in the alignment made by the user to be lost each time sequences are added. Therefore, the program offers some additional features to add sequences to existing multiple alignments without affecting the alignments (and manual corrections) between the other sequences.

14.4.1 In database **DemoBase**, open comparison **All**, and display the multiple alignment.

14.4.2 Calculate a dendrogram from the multiple alignment with *Sequence | Calculate global cluster analysis*.

14.4.3 Select some entries and cut them from the analysis with *Edit | Cut selection*.

The dendrogram is recalculated immediately, and the multiple alignment is preserved, since deleting entries does not influence the multiple alignment.

14.4.4 Paste the selection (which is still on the clipboard) again with *Edit | Paste selection*.

The matrix based upon pairwise alignments and the corresponding dendrogram are now being updated, and when finished, the pasted sequences are shown in the multiple alignment. However, the program has NOT aligned them. The multiple alignment dendrogram does not appear, since the multiple alignment is not updated yet.

14.4.5 To show that the pairwise dendrogram is updated, select *Sequence | ✓Show global cluster analysis* to undo displaying the global cluster analysis.

14.4.6 The pairwise dendrogram appears; if not, choose *Layout | Show dendrogram*.

14.4.7 Inspect where the pasted sequences are inserted in the dendrogram (blue arrows).

14.4.8 If the pasted sequences constitute one single branch, select that branch on the dendrogram, and *Sequence | Align internal branch*.

14.4.9 The sequence within the branch are now being aligned internally, and once this is finished, you can select *Sequence | Align external branch* to align the sequences from the rest of the dendrogram with the selected branch.

The advantage of this approach is that by using the *Align internal branch* feature, only the sequences within the selected branch are aligned. This is useful to update a part of a multiple alignment without affecting the non-selected branches. With the *Align external branch* feature, the selected branch is aligned to the rest of the dendrogram as a whole: all sequences within the branch are treated as one block, and all the other sequences are treated as another block. The two blocks are aligned to each other. These features give the user full control over how new sequences are added to a multiple alignment without affecting any editing.

14.5 Excluding regions from the sequence comparisons

Similar as for the comparison of fingerprints, it is possible to exclude regions from the sequences to be clustered. First, one needs to define a reference sequence, and next, one can indicate the zones to be excluded and included on the reference sequence. The exclusion of regions is only possible when calculating a cluster analysis based upon globally aligned sequences (multiple alignment) and when the reference sequence is included in the multiple alignment.

14.5.1 In the main window, open the sequence type window of **16S rDNA** by double-clicking on **16S rDNA** in the experiment types panel.

Initially, there is no reference sequence present. A *link arrow*  allows you to link a reference sequence to a database entry, by clicking on the arrow and dragging it onto a database entry, and then releasing the mouse button. When the experiment is linked, its link arrow is purple: .

14.5.2 Drag the link arrow to database entry *Vercingetorix palustris* strain no. 42819: as soon as you pass over a database entry, the cursor shape changes into .

14.5.3 Release the mouse button on database entry *Vercingetorix palustris* strain no. 42819.

This entry is now defined as reference sequence, and the arrow in the sequence type window has become purple  instead of gray . The reference sequence is shown in the sequence type window (Figure 114).

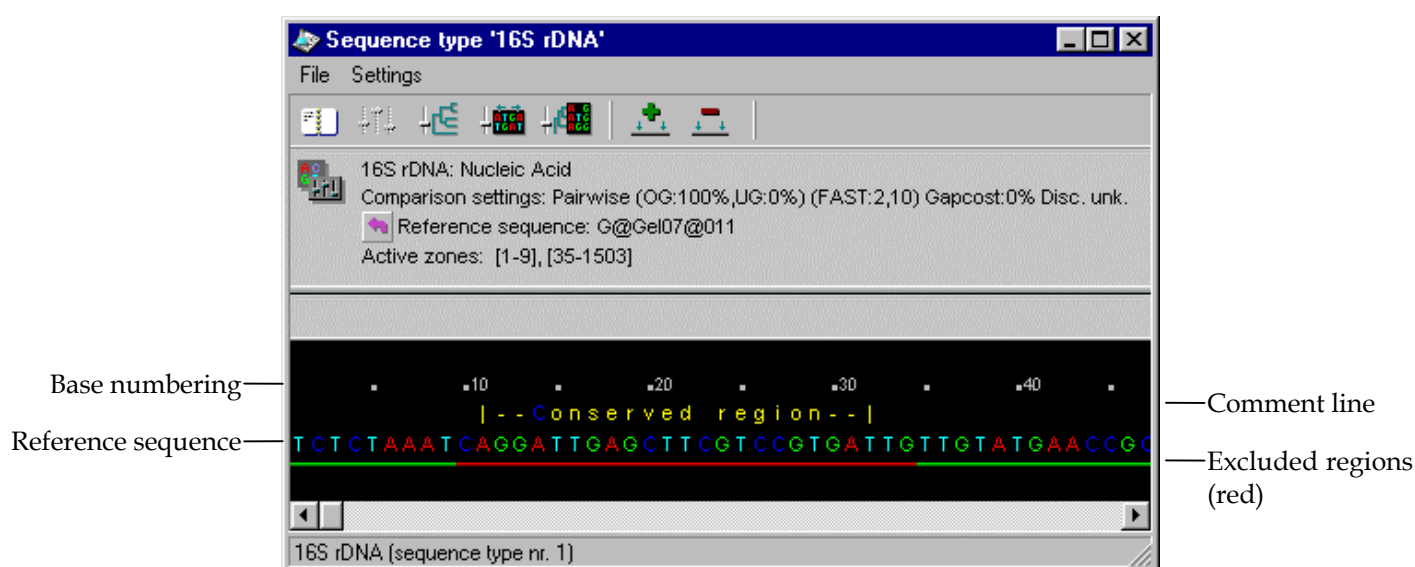


Figure 114. Sequence type window with reference sequence defined, region excluded, and comments added.

14.5.4 Select **Settings | Exclude active region** or  to exclude a region for comparison.

14.5.5 Enter start and end base number of the region to be excluded, and press the **<OK>** button.

The included regions are marked with a green line whereas the excluded regions are marked with a red line (Figure 114).

14.5.6 In order to remove all excluded regions at a time, select **Settings | Include active region** or . Enter 1 as **From** number, and enter the length of the sequence, or a number which certainly exceeds the sequence's length, as **To** number.

14.5.7 Open comparison **All** (or another comparison which you have saved with the aligned sequences).

14.5.8 Show the image of the aligned sequences.

14.5.9 Select the branch top of *Vercingetorix palustris* strain no. 42819 in the dendrogram panel (the reference sequence) and *Sequence | Create consensus of branch*.

By creating a consensus of a single sequence, you can display the reference sequence in the consensus sequence line (Figure 115). At the same time, the excluded and included regions are indicated, and the base numbering, according to the reference sequence appears (Figure 115).

14.5.10 In order to see the base numbering it may be necessary to drag the horizontal line that separates the header from the image panels downwards.

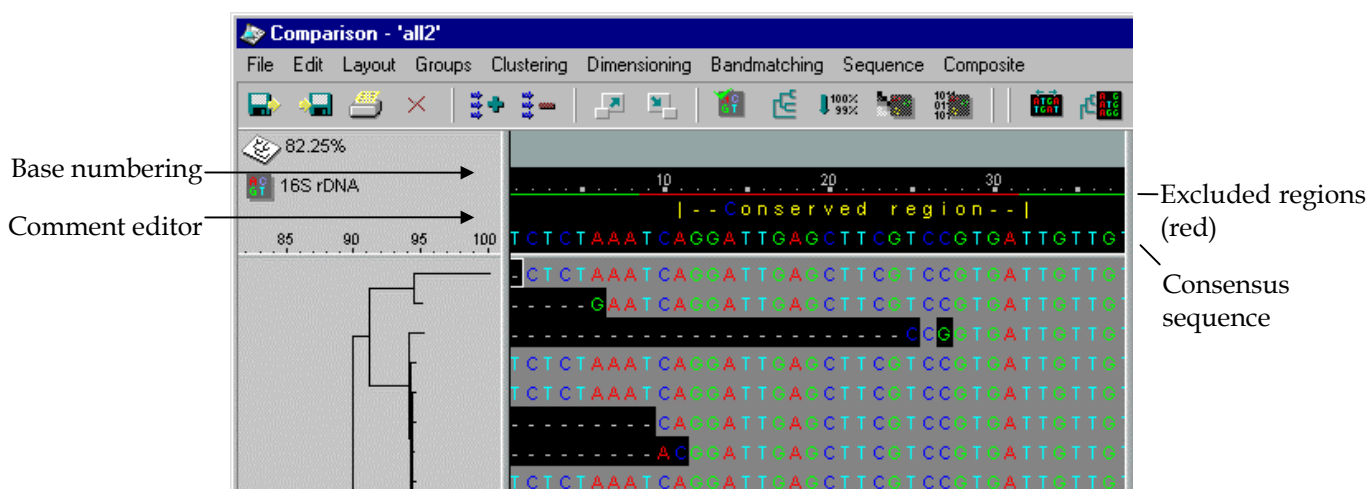


Figure 115. Comparison window with image of aligned sequences in consensus blocks view, detail.

14.6 Writing comments in the alignment

In order to mark special regions on the reference sequence or on the multiple alignment, a simple comment editor allows you to add any comment to the comparison. The comments can only be added when a reference sequence is present, when the sequences are aligned, and when a consensus sequence is shown. The comment line is saved along with the sequence type, and new comments can be added at any time.

14.6.1 Click the cursor on one of the aligned sequences in the image. At the position of the cursor, you can start writing comments.

The comments appear in the image header, above the consensus sequence (Figure 115). Any character input is supported. A's, C's, G's and T's are written in the colors of the bases.

14.6.2 To delete a comment, place the cursor on any sequence at the position of the first character of the comment and enter spaces.

14.6.3 Aligned or not aligned sequences in a comparison can be exported as text-file with the command *File | Export sequences*.

The program now asks "Do you want to export the database fields?".

14.6.4 Answer <Yes> to export tab-delineated database fields along with the sequences.

Next, the program asks “*Do you want to include regions with gaps?*”.

14.6.5 Answer <Yes> if you want to preserve the gaps introduced in the multiple alignment.

This allows aligned sequences to be exported from Fingerprinting II Informatix to other software applications. Gaps are represented as spaces.

15. Phylogenetic clustering methods

In addition to the *Neighbor Joining* method, which we already described in this chapter, Fingerprinting II Informatix offers two alternative phylogenetic clustering methods, based on the concepts of *parsimony* and *maximum likelihood*, respectively. Usually, these clustering methods will only be applied on sequence data, and we will describe them with the sequence type **16S rDNA** of database **DemoBase**. Parsimony and maximum likelihood clustering only work on aligned sequences: a multiple alignment must be present.

15.1 Maximum parsimony clustering

15.1.1 In **DemoBase**, open the comparison window with the aligned sequences shown.

First, we will reduce the number of entries in the comparison, in order to make maximum likelihood (15.2) possible in a reasonable time.

15.1.2 Select all entries in the comparison, and then unselect a couple of entries per cluster, so that some 10 entries from all clusters are unselected in total.

15.1.3 Remove the selected entries from the comparison with *Edit | Cut selection*.

15.1.4 Select *Clustering | Maximum parsimony cluster analysis*.

The *Maximum parsimony clustering* dialog box (Figure 116) allows you to specify a cost for each base conversion (mutation). The default settings is 100% for each possible conversion.

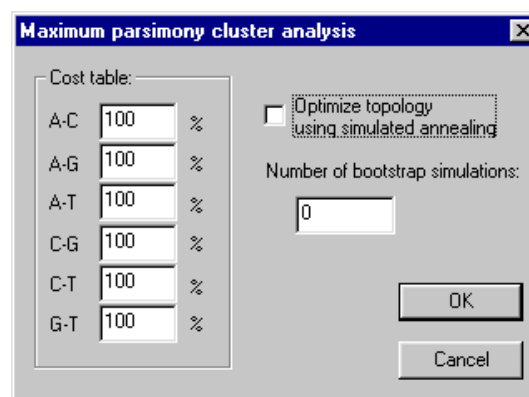


Figure 116. Maximum parsimony clustering dialog box.

Fingerprinting II Informatix uses methods that are described in the literature to optimize the topology of parsimonious trees. An alternative method, which finds even more parsimonious trees, but which is considerably slower, is the mathematical principle of *Simulated annealing*.

In addition, Fingerprinting II Informatix can do a *Bootstrap* analysis on the parsimony clustering, of which you can enter the number of simulations. **Caution:** enabling simulated annealing and at the same time entering a number of bootstrap simulations will increase the computing time dramatically!

15.1.5 Leave the *Cost table* unaltered, enable *Optimize topology*, and leave the *Number of bootstrap simulations* zero.

15.1.6 Press <OK> to start the calculations, which will take some time.

The result is an *Unrooted tree window*, of which the parsimony (the total number of base conversions over the three) is given in the status bar (Figure 117). The entries are represented in the colors of the groups we have defined earlier.

15.1.7 You can toggle between the colors and the black-and-white representation mode with *Layout | Show group colors* or .

In black-and-white mode, the groups are represented (and printed) as symbols.

15.1.8 With *Layout | Show keys or group numbers* or , the entry keys are displayed next to the dendrogram entries.

However, the entry keys may be long and uninformative for the user, so the entry keys can be replaced by a group code. The program assigns a letter to each defined group, and within a group, each entry receives a number. The group codes are shown as follows:

15.1.9 In the parent comparison window, select *Layout | Use group numbers as keys*.

15.1.10 A legend to the group numbers can be obtained with *File | Export database fields* in the parent comparison window.

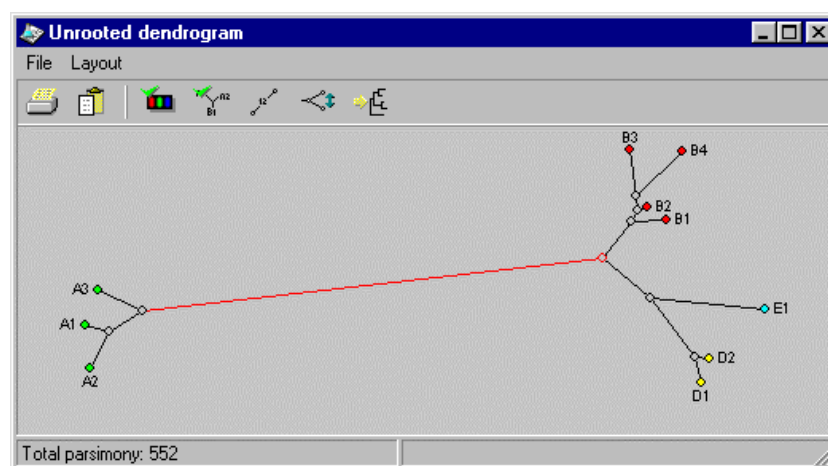


Figure 117. Unrooted maximum parsimony or maximum likelihood tree.

15.1.11 With *Layout | Show branch lengths* or , the lengths of the branches, in number of base conversions, are shown.

In more complex trees, the spread of the branches may not be optimal. The program can iteratively optimize the spread of the branches:

15.1.12 In the unrooted tree window, select *Layout | Optimize branch spread* or .

The user can rotate and swap the branches manually if the tree layout is not satisfactory.

15.1.13 Left click in the proximity of a node or a branch tip.

15.1.14 While holding down the mouse button, rotate the branch to the desired position.

15.1.15 If you select entries in the parent comparison window or in the main window, these entries are shown within a square in the unrooted tree window.

15.1.16 You can also select entries directly in the unrooted tree window, by holding the CTRL key while clicking in the proximity of a node. All entries branching off from this node will be selected.

15.1.17 Repeat this action to unselect entries.

15.1.18 To copy the unrooted tree to the clipboard, select *File | Copy image to clipboard* or .

15.1.19 The unrooted tree can be printed with *File | Print image* or .

Since interpreting unrooted trees is not always easy, especially with large numbers of entries, it is possible to create a rooted dendrogram from the unrooted tree. This process requires an artificial root to be defined as follows:

15.1.20 Select a branch by clicking in the proximity of one of the two nodes it connects. The selected branch is red.

15.1.21 In the menu, choose *Layout | Create rooted tree* or .

The dendrogram in the parent comparison window now is a rooted version of the maximum parsimony or maximum likelihood tree.

NOTE: all functions described under cluster analysis apply to such unrooted trees, except the incremental clustering: one cannot delete or add entries while the tree is automatically updated.

15.1.22 Close the unrooted tree window.

15.2 Maximum likelihood clustering

In the comparison window, select *Clustering | Maximum likelihood cluster analysis*.

The *Maximum likelihood clustering* dialog box shows up (Figure 118).

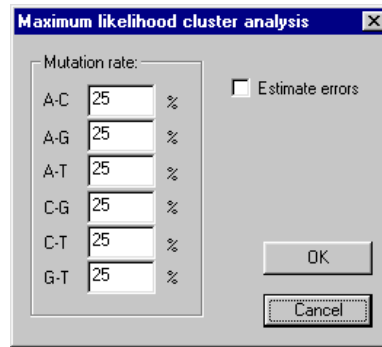


Figure 118. Maximum likelihood clustering dialog box.

Similar as for the maximum parsimony clustering, a *Mutation rate* can be defined for each individual base conversion. The default of the program is 25% for each possible mutation. The maximum likelihood clustering algorithm also allows a standard deviation to be calculated for each branch (*Estimate errors*). This is only an approximate error estimation, but since maximum likelihood clustering is absurdly slow, it is absolutely impossible to perform bootstrap analysis.

15.2.1 Leave the *Mutation rate* unaltered, and enable the *Estimate errors* checkbox.

15.2.2 Press <OK> to start the calculations.

Maximum likelihood clustering is an extremely time-consuming process; depending on the length of the sequences, clustering 30-50 entries may take several hours on a powerful computer. The calculation time increases with the third power of the number of entries included.

When the calculations are finished, an unrooted tree is shown which has all the same functions as described for maximum parsimony (15.1).

15.2.3 If you want to see the estimated errors on the branch lengths, use *Layout | Show branch lengths* or .

16. Dimensioning techniques (PCA, MDS and SOM)

Principal Components Analysis (PCA) and Multi-Dimensional Scaling (MDS) are two alternative grouping techniques that can both be classified as *dimensioning techniques*. In contrast to dendrogram inferring methods, they do not produce hierarchical structures like dendrograms. Instead, these techniques produce two-dimensional or three-dimensional plots in which the entries studied are spread according to their relatedness. Unlike a dendrogram, a PCA or MDS plot does not provide "clusters". The interpretation of the obtained comparison is, more than in cluster analysis, left to the user.

PCA assumes a data set with a known number of characters and analyzes the characters directly. PCA is applicable to all kinds of character data, but not directly to fingerprint data. Fingerprints can only be analyzed when converted into a band matching table (see 10.2.39).

MDS does not analyze the original character set, but the matrix of similarities obtained using a similarity coefficient. Rather than being a separate grouping technique, MDS just replaces the clustering step in the sequence *characters > similarity matrix > cluster analysis*. However, it is a valuable alternative to the dendrogram methods, which often oversimplify the data available in a similarity matrix, and tend to produce overestimated hierarchies.

16.1 Calculating an MDS

Whatever experiment type for which a complete similarity matrix is available can be analyzed by MDS. Matrix types are not suitable for MDS clustering if the matrices are incomplete.

16.1.1 In the main window with **DemoBase** loaded, open comparison **All**, or a comparison with all entries except those defined as STANDARD.

16.1.2 Select **FAME** in the experiment type selection bar, and check whether a matrix is available for this experiment type by looking in the **Layout** menu if the menu command **Show matrix** is enabled (not grayed).

16.1.3 If **Show matrix** is grayed, first calculate a dendrogram with **Clustering | Calculate cluster analysis**.

16.1.4 Select **Dimensioning | Multi-dimensional scaling** or .

The program now asks "**Optimize positions**". Fingerprinting II Informatix iteratively recalculates the MDS, each time again optimizing the positions of the entries in the space to resemble the similarity matrix as closely as possible. If you allow the optimization to happen, the calculations take longer.

16.1.5 Press <Yes> to optimize the positions.

The MDS is calculated and the *Coordinate space* window is shown.

16.2 Editing an MDS

The *Coordinate space* window (Figure 119) shows the entries as dots in a cubic coordinate system.

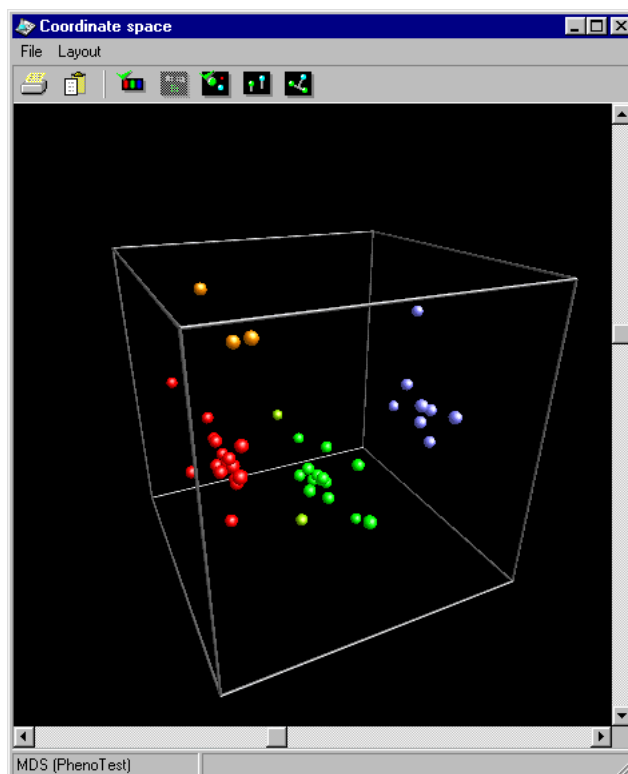


Figure 119. Coordinate space window, resulting from a PCA or MDS analysis.

16.2.1 To zoom in and zoom out on the image, use the Pge Dn and Pge Up keys, respectively.

16.2.2 The image can be rotated in real time using the arrow keys or the horizontal and vertical scroll bars.

By default, the entries are represented as small dots, in the colors as defined for the groups on the dendrogram (11.7).

16.2.3 With *Layout | Show keys* or , you can display the database keys of the entries instead of the dots.

However, the entry keys may be long and uninformative for the user, so the entry keys can be replaced by a group code. The program assigns a letter to each defined group, and within a group, each entry receives a number. The group codes are shown as follows:

16.2.4 In the parent comparison window, select *Layout | Use group numbers as keys*.

16.2.5 A legend to the group numbers can be obtained with *File | Export database fields* in the parent comparison window.

16.2.6 With *Layout|Show group colors* or , you can toggle between the color representation and the non-color representation, in which the entry groups are represented (and printed) as symbols instead of colored dots.

On the screen, it is easier to evaluate the groups using colors.

16.2.7 With *Layout|Show construction lines* or , the entries are displayed on vertical lines starting from the bottom of the cube. This may facilitate the three-dimensional perception. Disable this option to view the next features.

16.2.8 *Layout|Show rendered image* or , displays the coordinate system in realistic three-dimensional perspective.

16.2.9 Another very interesting option is *Layout|Show dendrogram* or .

When this option is enabled, the entries in the coordinate system are connected by the dendrogram branches from the parent comparison window. This is an ideal combination to co-evaluate a dendrogram and a coordinate system (PCA or MDS).

16.2.10 To copy the coordinate space image to the clipboard, select *File|Copy image to clipboard* or .

16.2.11 The image can be printed with *File|Print image* or . The image will print in color if the colors are shown on the screen.

16.3 Calculating a PCA

PCA is typically executed on complete character data. It does not work on sequence types. Fingerprints can only be analyzed by PCA if a band matching table is first generated (see 10.2).

16.3.1 In the main window with **DemoBase** loaded, open comparison **All**, or a comparison with all entries except those defined as STANDARD.

16.3.2 Select **FAME** in the experiment type selection bar, and *Dimensioning|Principal Components Analysis* or .

The *Principal Components Analysis* dialog box (Figure 120) allows a number of more advanced choices to be made.

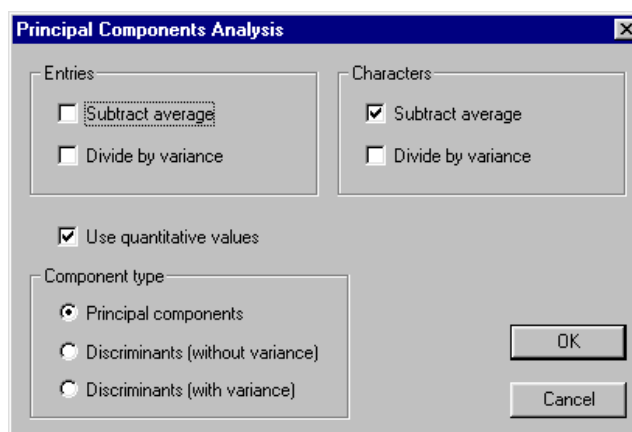


Figure 120. Principal Components Analysis dialog box.

The simplest choice is “*Use quantitative values*”. By default, this choice is checked, and if the technique provides quantitative information (not just absent/present), one will normally want to use this information for the PCA calculation. If this option is unchecked, the character values will be converted to binary as specified in the *Conversion to binary* settings (see 7.8.20).

More sophisticated options are the possibilities to *Subtract average* character value over the *Entries*, and to *Subtract average* character value over the *Characters*. Figure 121 explains how the averaging works.

- Subtraction of the *averages* over the *characters* (green in the figure) results in a PCA plot arranged around the origin, and therefore, it is recommended for general purposes.
- Division by the *variances* over the *characters* (green in the figure) results in an analysis in which each character is equally important. Enabling this option can be interesting in a study containing characters of unequal occurrence. For example, if fatty acid extractions are analyzed for a set of bacteria, some fatty acids may be present in abundant amounts, whereas others may occur only in very small amounts. It is well possible that the “minor” fatty acids are as informative or even more informative than the abundant ones, taxonomically seen. If no correction is applied, those minor fatty acids will be completely masked by differences in the abundant fatty acids. Dividing by the variance for each fatty acid normalizes for such range differences, making each character equally contributing to the total separation of the system.
- Subtraction of the *averages* over the *entries* (red in the figure) results in character sets of which the sum of characters equals zero for each entry. This feature has little meaning for general purposes.
- Division by the *variances* over the *entries* (red in the figure) results in character sets for which the intensity is normalized for all entries. For example, suppose that you have scanned phenotypic test panels for a number of strains and want to calculate a PCA. If some strains are less grown than others, the overall reaction in the wells will be less developed. Without correction, well developed and less developed panels will fall apart in the study. Dividing by the variances normalizes the character sets for such irrelevant differences, making character sets with different overall character developments fall together as long as the relative reactions of the characters are the same.

*NOTE: The two latter features are exactly what is done by the Pearson product-moment correlation coefficient. This coefficient subtracts each character set by its average, and divides the characters by the variance of the character set. The feature **Divide by variance** under **Entries** should not be used in character sets where the characters are already expressed as percentages (for example, fatty acid methyl esters).*

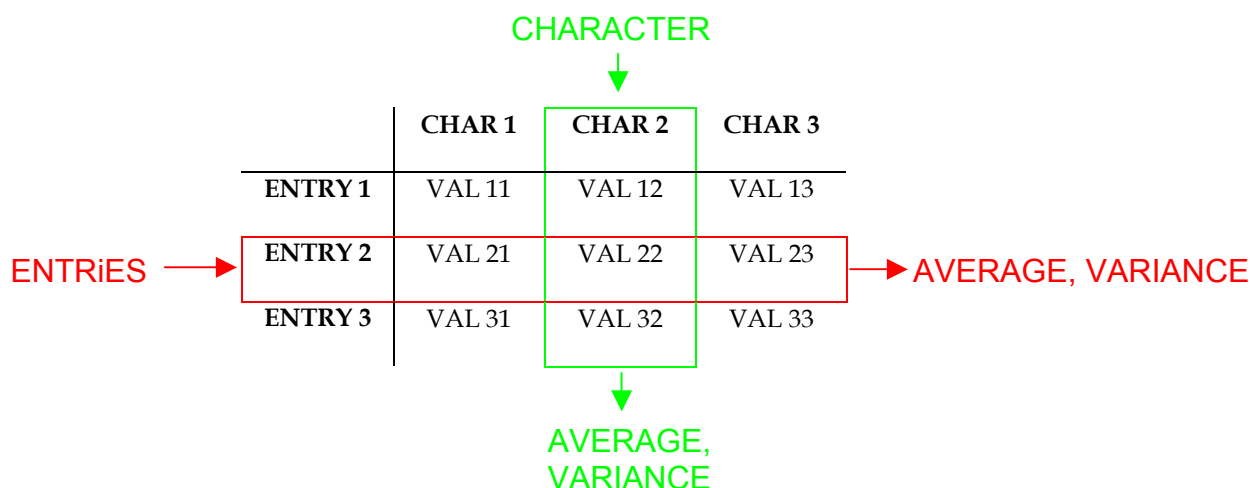


Figure 121. Character table showing the meaning of *Average* and *Variance* correction at the *Entries* and *Characters* level.

The lower panel of the dialog box (Figure 120) displays the *Component type*. This can be *Principal components*, *Discriminants (without variance)*, or *Discriminants (with variance)*. The first option is to calculate a principal components analysis, whereas the *Discriminants* options are to perform discriminant analysis. These options are described in paragraph 16.6.

16.3.3 In the *Entries* and *Characters* panels, check *Subtract average* under *Characters*, and leave the other options unchecked.

16.3.4 In the *Component type* panel, select *Principal components*, and press <OK>. Calculation of the PCA is started.

The resulting window, the *Principal components analysis* window, is shown in Figure 122.

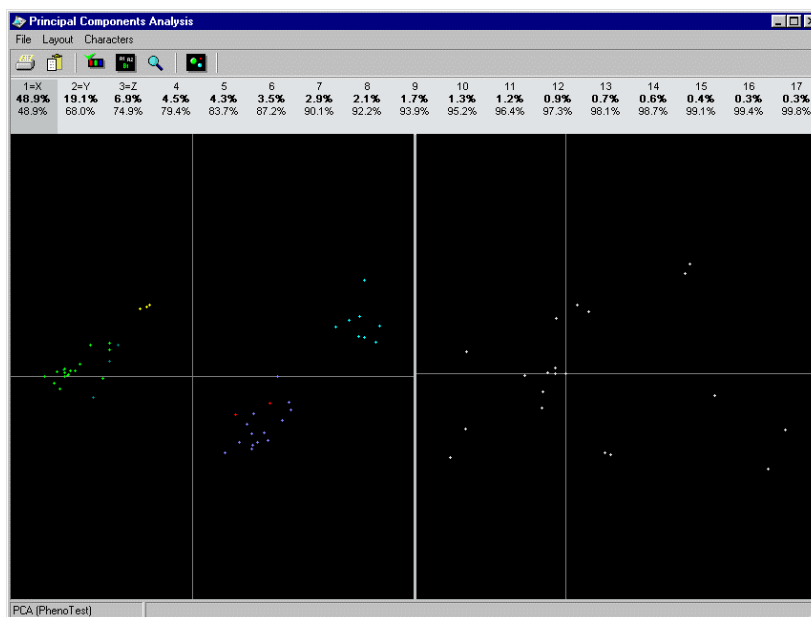


Figure 122. Principal components analysis window.

The window is divided in two panels: the left panel shows the *entries* plotted in an X-Y diagram corresponding to the first two components. In the caption of the window, the first 20 components are shown, with their relative contribution and the cumulative contribution displayed. Also, the components used as X, Y and Z axes are indicated. The right panel shows the characters plotted in the same X-Y diagram. From the right panel, one can see the contribution each character has to the two displayed components, and hence, what contribution it has to the separation of the groups along the same components. For example, if a group of entries appears left along the X-axis whereas the other entries appear right, those characters occurring left on the X-axis are positive for the left entries and negative for the right entries, and *vice versa*.

By default, the first component is used for the X axis, the second component is used for the Y axis, and the third component is used for the Z axis. The Z axis is not shown here, but can be shown in the three-dimensional representation with **Layout | Show 3D plot** (see further).

16.3.5 If you want to assign another component as one of the axes, select the component in the caption, and **Layout | Use as X axis**, **Layout | Use as Y axis**, or **Layout | Use as Z axis** (or right-click on the component).

Layout tools:

16.3.6 Switching from color indication for the groups to symbol indication with **Layout | Show group colors** or .

16.3.7 Showing the keys or a unique label based upon the groups for the entries with **Layout | Show keys** or .

*NOTE: In case keys are assigned automatically by the program, they are not very informative, so one should select **Layout | Use group numbers as key** in the underlying Comparison window. A list of the group codes and the corresponding entry names can be generated in the underlying Comparison window with **File | Export database fields**.*

16.3.8 The option **Layout | Preserve aspect ratio** allows you to either preserve the aspect ratio of the components, i.e. the relative discrimination of the component on the Y axis with respect to the component on the X axis, or to stretch the components on the axes so that they fill the image optimally.

16.3.9 With **Layout | Zoom in / zoom out** or , you can zoom in on any part of the *entries* or *characters* panel of the PCA plot: drag the mouse pointer to create a rectangle; the area within the rectangle will be zoomed to cover the whole panel.

16.3.10 In order to restore the original size of the image, simply left-click within the panel. Disable the zoom-mode afterwards.

16.3.11 If you move the mouse pointer over the right panel (characters), the name of the pointed character is shown.

Editing tools:

16.3.12 Entries can be selected in a PCA window by holding the SHIFT key down and selecting the entries in a rectangle using the left mouse button. Selected entries are encircled in blue.

A PCA is automatically saved along with its parent *Comparison* window. It is possible to add entries to an existing PCA or remove entries from it. The feature to add entries to an existing PCA is an interesting alternative way of identifying new entries. They can be placed in a frame of known

database entries, and in this way, identifying is just looking at the groups they are closest to. **Since the components are not recalculated when entries are added to an existing PCA, the PCA does not reflect the full data matrix anymore!**

16.3.13 If you want to add entries to an existing PCA, you can select new entries in the main window and copy them to the clipboard using *Edit | Copy selection* or .

16.3.14 In the *Comparison* window, select *Edit | Paste selection*. The new entries are placed in the comparison window and in the PCA window.

16.3.15 To delete entries from a PCA, select some entries as in 16.3.12 and in the *Comparison* window, select *Edit | Cut selection*.

If you started the PCA from a composite data set, you can order the characters according to the selected component in the underlying *Comparison* window. This is an interesting feature to locate characters that separate groups you are interested in. The feature works as follows (only composite data sets).

16.3.16 First determine the component that best separates the groups.

16.3.17 Select that component in the caption and *Characters | Order characters by component* (or right-click on the component). The characters are now ordered by the selected component in the *Comparison* window.

16.3.18 The entry plot can be printed with *File | Print image (entries)* or  and the character plot can be printed with *File | Print image (characters)*.

16.3.19 Alternatively, the entry plot can be copied to the clipboard with *File | Copy image to clipboard (entries)* or  and the character plot can be printed with *File | Copy image to clipboard (characters)*.

If you want to reconstruct or analyze the PCA system in other software packages, it is possible to export the coordinates of the entries along a selected component (for example the X-axis):

16.3.20 Select a component and *File | Export entry coordinates*.

If you want to reconstruct the PCA with the first two components, you should also export the second component (Y-axis), by selecting that component and *File | Export entry coordinates*.

Similarly, one can export the coordinates for the characters for a certain component:

16.3.21 Select a component and *File | Export character coordinates*.

Fingerprinting II Informatix allows you to display three components at the same time, by plotting the entries in a 3-dimensional space.

16.3.22 To create a three-dimensional plot from the PCA, select *Layout | Show 3D plot* or .

The *Coordinate space* window is shown. See 16.2 to edit a PCA in 3-D representation mode.

16.3.23 Close the coordinate space window with *File | Exit*.

16.3.24 Close the PCA window with *File | Exit*.

16.4 Calculating a discriminant analysis

Discriminant analysis is very similar to PCA. The major difference is that PCA calculates the best discriminating components for the character table as a whole, without foreknowledge about groups, whereas discriminant analysis calculates the best discriminating components *for groups that are defined by the user*. In case of discriminant analysis, these principal components are then called *discriminants*. Like PCA, discriminant analysis is executed on complete character data. It does not work on sequence types. Fingerprints can only be analyzed by discriminant analysis if a band matching table is first generated (see 10.2). Discriminant analysis also forms the basis for multivariate analysis of variance (MANOVA), which is explained in paragraph 16.6.

16.4.1 In database DemoBase, we first create a new composite data set (see 11.2) for experiment type **PhenoTest**. You can name it **Phenodata**, for example.

16.4.2 Open comparison **All**, or a comparison with all entries except those defined as STANDARD.

Since discriminant analysis work on user-delineated groups, the comparison should contain groups (see 11.7).

16.4.3 Select **Phenodata** in the experiment type selection bar, and **Dimensioning|Principal Components Analysis** or .

The *Principal Components Analysis* dialog box (Figure 120) allows a number of choices to be made under *Entries* and *Characters*, which are described under PCA (16.3). These choices also apply for discriminant analysis. However, the *Divide by variance* option under *Characters* makes no difference whether it is enabled or disabled for discriminant analysis.

The following two options are available for discriminant analysis: *Discriminants (without variance)*, and *Discriminants (with variance)*. If you select “*with variance*”, each character is divided by its variance. In order to understand what this implies, consider Figure 123.

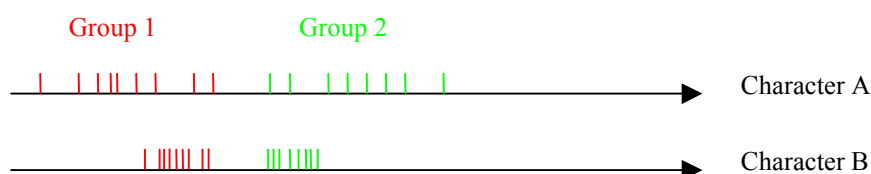


Figure 123. The influence of character spread on discriminant analysis.

This example shows two groups, 1 (red) and 2 (green), that are separated by two characters, A and B. On the average, group 1 is less positive both for characters A and B. Character A seems to be better discriminating between the two groups than character B, because the centers of the groups are lying further from each other in case of character A. However, if the internal spread of groups are considered, then the groups are found much more coherent for character B, which may render this character at least as much value for discriminating as character A. In a non-corrected discriminant analysis, character A will account for most of the discrimination, just by the fact that the centers of the groups are more distant. This is the case in option *Discriminants (without variance)*. When the characters are divided by the variances of the groups, the internal spread is compensated for, and character B will become at least as important as character A. This is achieved with option *Discriminants (with variance)*.

16.4.4 Select *Discriminants (with variance)* and <OK>.

The resulting window is identical as the PCA window described before (Figure 120), and the same features apply.

If you started the discriminant analysis from a composite data set, you can order the characters according to the selected discriminant in the underlying *Comparison* window. This is an interesting feature to locate characters that separate groups you are interested in. The feature works as follows (only composite data sets).

16.4.5 First determine the discriminant that best separates two groups you have in mind. You can examine the discriminants by right clicking on them in the caption and *Use as Y-axis* (or X-axis).

16.4.6 Select that discriminant in the caption and *Characters | Order characters by component* (or right-click on the discriminant). The characters are now ordered by the selected discriminant in the *Comparison* window.

16.5 Self organizing maps

A self organizing map (SOM, also called Kohonen map) is a neural network that classifies entries in a two-dimensional space (map) according to their likeliness. Since the technique which is used for grouping, i.e. the training of a neural network, is completely different from all previously described methods, SOMs are an interesting alternative to conventional grouping methods, including cluster analysis, principal component analysis, and related techniques. Also, similar as in PCA, a SOM can start from the characters as input, thus avoiding the choice of one or another similarity coefficient. Unlike PCA, the distance between entries on the map is not in proportion to the taxonomic distance between the entries. Rather, a SOM contains areas of high distance and areas of high similarity. Such areas can be visualized by different shading, for example when a darker shading is used in proportion to the distance in the SOM.

When the similarity values with all of the other entries of a comparison are considered as the character set, a SOM can also be applied on similarity matrices, which makes the technique also suitable for grouping of electrophoresis patterns that are compared pair by pair using a band matching coefficient such as Dice.

To calculate a self-organizing map based on character data, use for example the character set FAME in DemoBase.

16.5.1 In the main window with **DemoBase** loaded, open comparison **All**, or a comparison with all entries except those defined as STANDARD.

16.5.2 Select **FAME** in the experiment type selection bar, and *Dimensioning | Self-organizing map* (the command *Dimensioning | Self-organizing map (similarities)* is to calculate the som from the similarity matrix).

An input box asks to enter the map size. This is the number of nodes of the neural network in each direction. For the default size 10, a neural network containing 10x6 nodes is generated. The larger the map is taken, the longer the training takes. Note that the optimal size of the map depends on the number of entries compared. For a small number of entries, a small map size will usually provide better results.

16.5.3 Enter 7 as map size and press <OK>.

The SOM is calculated and shown (Figure 124). Areas of high similarity are black. Selected entries in the parent comparison window are also selected on the map. Note that the SOM as shown in will not necessarily correspond to the one you have calculated.

16.5.4 To show the information of a particular entry in the SOM, right-click on the entry and select *Edit database fields*.

16.5.5 You can (un)select entries on the SOM by left-clicking on an entry while pressing the CTRL key, or groups of entries by left-clicking and moving the mouse while pressing the SHIFT key.

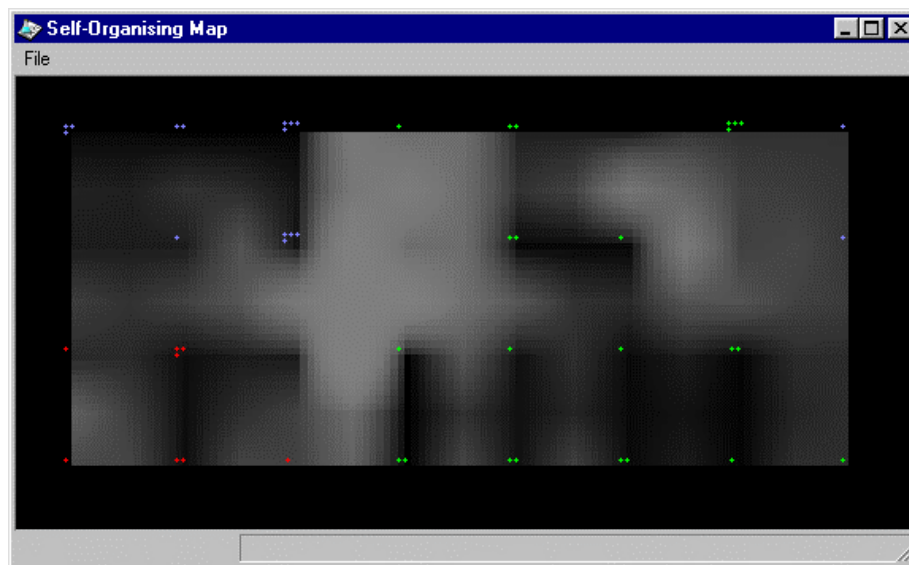


Figure 124. Self Organizing Map calculated from data set FAME in the DemoBase of Fingerprinting II Informatix.

NOTE: When a SOM is calculated on fingerprint type data, the densitometric curves are used as character data sets for training of the SOM.

A SOM is automatically saved along with its parent *Comparison* window. It is possible to add entries to an existing SOM or remove entries from it. The feature to add entries to an existing SOM is an interesting alternative way of identifying new entries. Added entries are placed in a frame of known database entries in the SOM, and in this way, identifying is just looking at the groups they are joining.

16.5.6 If you want to add entries to an existing SOM, you can select new entries in the main window and copy them to the clipboard using *Edit | Copy selection* or .

16.5.7 In the *Comparison* window, select *Edit | Paste selection*. The new entries are placed in the comparison window and in the SOM window.

NOTES:

1. *An identification based upon a self-organizing map is only reliable if the new entries belong to one of the groups the SOM is based upon. A SOM will always produce a “positive” identification: an unknown profile is **always** find a place in the SOM, i.e. the cell having the highest similarity with the new entry. If, after adding a new entry to a SOM, the entry falls next to a known entry of that SOM, this means only that the new entry has the highest similarity with that particular cell compared to the other cells; it does **not** mean that it is highly related to that*

entry. Hence, identification based upon a SOM is only recommended if you are sure the unknown entries belong to one of the groups composing the SOM.

2. *Since no new cells can be created in a SOM, one should never add new entries which are known to constitute a group that is not represented in the SOM.*

16.5.8 To delete entries from a SOM, select some entries and in the *Comparison* window, select **Edit | Cut selection**.

16.5.9 Close the SOM window with **File | Exit**.

16.5.10 To create a Self Organizing Map from a similarity matrix obtained after cluster analysis, select **Dimensioning | Self-organizing map (similarities)**.

In this case, the result of the SOM is based on similarity values of the entries with each other and hence is dependent on the similarity coefficient used, and the tolerance and optimization settings in case of fingerprint types. Obviously, this method only works if a cluster analysis of the selected experiment is available. If not, first create a cluster analysis with **Clustering | Calculate cluster analysis**.

16.5.11 The SOM can be printed with the **File | Print** command, or exported via the clipboard as enhanced metafile using **File | Copy to clipboard**.

In these cases, the map colors are inverted, i.e. white corresponds with areas of high similarity, whereas darker shading corresponds with areas of low similarity.

A SOM is saved along with a comparison. In order to display a previously calculated SOM in a comparison, select **Dimensioning | Show map**.

16.6 Multivariate analysis of variance (MANOVA) and discriminant analysis

Multivariate Analysis Of Variance (MANOVA) is a statistical technique which allows the significance of user-delineated groups to be calculated. Since it is extremely difficult to prove that delineated groups are significant, statistical methods usually are based on the reverse approach, i.e. to prove that the chance (likelihood) to obtain equally good separations with randomly generated groups approaches zero. In addition, a statistical technique related to PCA, discriminant analysis, allows the determination of characters that are responsible for the separation of the delineated groups.

The MANOVA technique only applies to composite data sets. If you want to find the discriminating characters for an experiment, you should first create a composite data set containing that experiment.

MANOVA cannot be applied to incomplete data sets. In other words, all characters must be filled in for each entry. In case of "open" character types (in which the character set may grow dynamically), absent values should be considered as zero (see 7.6).

16.6.1 In database DemoBase, we first create a new composite data set (see 11.2) for experiment type **PhenoTest**. You can name it **Phenodata**, for example.

16.6.2 Open comparison **All**, or a comparison with all entries except those defined as **STANDARD**.

Since MANOVA and discriminant analysis work on user-delineated groups, the comparison should contain groups (see 11.7).

16.6.3 Select **Phenodata** in the experiment type selection bar, and *Groups | Multivariate Analysis of Variance*.

The program now asks "*Account for covariance structure?*". This question relates to the possibility of the discriminant analysis to explore correlations between characters in order to achieve a better discrimination. Most statistical approaches assume that the covariance is accounted for, however, its use becomes dangerous in case the number of characters is close to, or larger than the number of entries studied. One should consider discriminant analysis as solving a system of n equations with p parameters (n being the number of entries, p the number of characters). If p (the degrees of freedom) equals n or is larger than n , there are always perfect solutions, regardless of what groups being defined. The result of the discriminant analysis will be that the delineated groups will be perfectly separated. To avoid such unrealistic separations, you should only allow the program to account for the covariance when the number of entries is significantly larger than the number of characters.

16.6.4 Answer **<No>** to account for covariance.

The program now asks a second question "*Use variance?*".

16.6.5 Unless the number of entries is much larger than the number of characters in the data set, you should answer **<No>** to this question.

A third question is now asked: "*Calculate relative character importance?*". When this feature is applied, the program will repeat the discriminant analysis, each time leaving out one character. The quality of the separation when a character is left out is then compared to the quality when the character is not left out, and this is a direct measure for the importance of that character. Obviously, the calculations take much longer when the discriminant analysis is to be calculated p times, p being the number of characters.

16.6.6 Answer **<Yes>** to calculate the relative character importance.

If one or more characters are identical for all the entries, this will be reported in a message box and such characters will be left out from the discriminant analysis. The resulting *MANOVA & discriminant analysis* window is shown in Figure 125. This example is based upon four groups (red, blue, green and yellow).

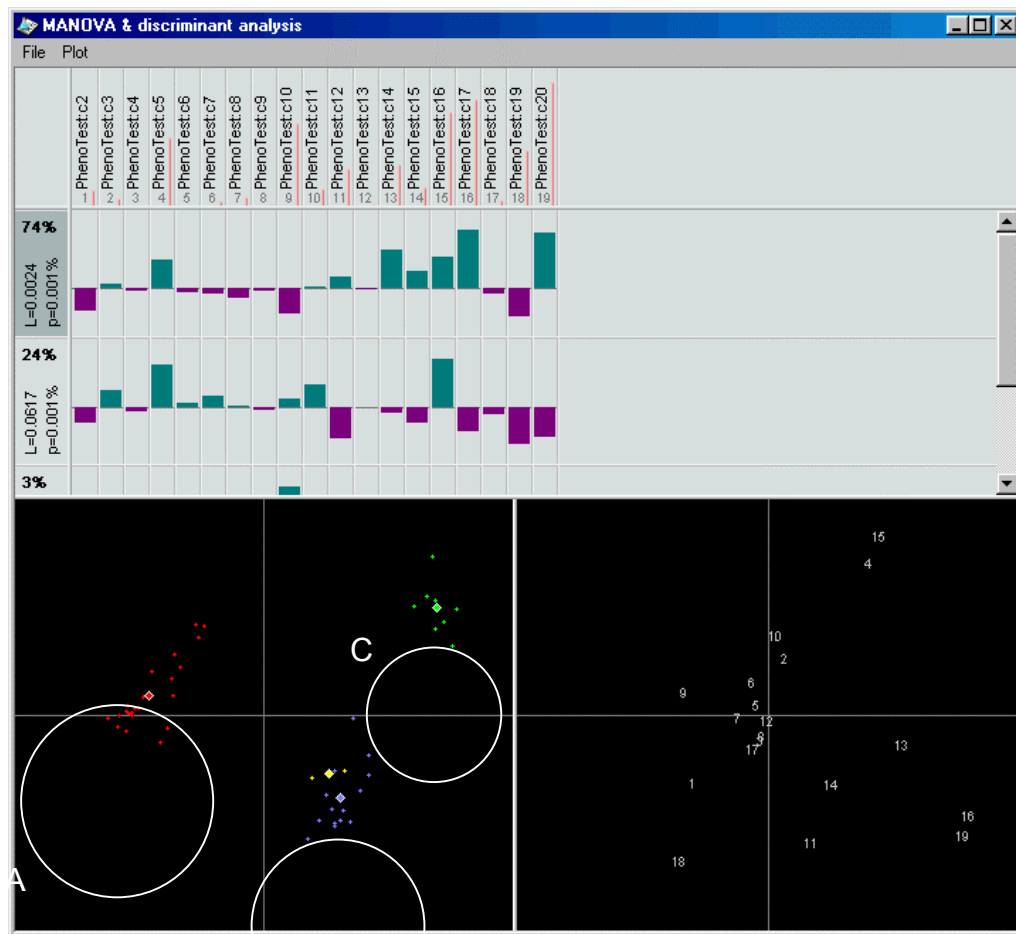


Figure 125. MANOVA & discriminant analysis window. The circles delineating groups A, B, and C are added to this figure to illustrate the interpretation of discriminant analysis.

The window is divided in three panels: the upper panel, covering the full width of the window, shows the relative discriminatory value of the characters for each of the discriminants. A character can have a contribution to the discrimination in the positive sense (green) or in the negative sense (purple). The larger the bar, the greater the contribution, irrespective of the sense. If character contributions have a different sense, it means that the one character will be positive in the groups where the other character will be negative and *vice versa*.

The relative importance for each character is shown as a red line, right from the character name.

Note that the total number of discriminants will always be the number of groups less one. For two groups, there is only one discriminant; for three groups, there are two discriminants, etc.

16.6.7 If there are more than two discriminants, you can scroll through the list of discriminants.

The first discriminant is always the most important, i.e. it accounts for most of the discrimination; the second discriminant is the second most important etc. The percentage discrimination of a discriminant is shown in bold (left). The sum of the percentages equals 100.

In addition, the parameter L (Wilkinson's likelihood for normal distributions) predicts the likelihood of the obtained discrimination in the assumption that the groups are drawn from the same population. If L is low, the entries of the different groups are likely to be drawn from different populations, in other words, the existence of the groups is justified. The parameter p is the probability that a random subdivision in groups would yield the same degree of discrimination.

The left lower panel maps the entries on the first two discriminants (first = X axis, second = Y axis). On this image you can see that the X axis accounts for most of the discrimination (Figure 125). The right panel maps the characters on the entry groups. To interpret this very informative panel, you should inspect it together with the left panel. The more distant a group occurs from the center along a discriminant axis, the better it may be characterized by one or more characters. These characters can be found in the right panel, shown by their number. Characters are very positive for the group if they fall in the same direction of the axis as the group; if they occur in the opposite direction, they are very negative for the group. The further a group and one or more characters occur in either direction of an axis, the more pronounced these characters are either positive or negative for that group. For example, group A occurs in the negative half of the X axis (first discriminant), whereas group C is the most positive group on this axis. Characters 16 and 19 (pronounced positive position) discriminate group C from group A in that they are much more positive for group C members than for group A members. Another example: group C is positive on both the X and Y discriminant, whereas groups A and B are negative on either the X or Y axis. From this, one can conclude that characters 4 and 15 discriminate group C from both groups A and B. The rhomb in the center of each group in the left panel is the average position of the group.

16.6.8 The third discriminant can be plotted on the images by selecting it in the upper panel (third table), and selecting *Plot | Use discriminant as X axis* or *Plot | Use discriminant as Y axis*.

The various results of a MANOVA analysis can be printed or exported as enhanced metafile to the clipboard for further processing in another package:

16.6.9 Use *File | Print report* to print a detailed numerical report of all characters and their contribution along the discriminants. Similarly, *File | Export report* is used to export this report to the clipboard, tab-delineated or space-delineated.

16.6.10 Use *File | Print discriminants* to print the upper graphical panel, representing the selected discriminants and the relative importance of the characters shown as bar graphs. *File | Copy discriminants to clipboard* is to export this report to the clipboard as enhanced metafile.

16.6.11 Use *File | Print correspondence plot* to print the lower two-dimensional plots, representing the entries (left) and characters (right) plotted along two discriminants. *File | Copy correspondence plot to clipboard* is to export this report to the clipboard as enhanced metafile.

17. Identification with database entries

There are two methods for identification available in Fingerprinting II Informatix. The simplest one, as described in this section, is to compare and identify unknown patterns against a selection of database patterns stored on disk. The more sophisticated method is to identify unknown patterns against an identification library (see chapter 18).

17.1 Creating lists for identification

- 17.1.1 In DemoBase, select all *Ambiorix* entries except the *Ambiorix sp.* entries: First perform a search with *Ambiorix* as genus name, and then perform a second search with *Search in list* and *Negative search* enabled, and *sp.* as species string.
- 17.1.2 Create a comparison with the selected entries, and save it as **Ambiorix**.
- 17.1.3 Exit the comparison window.

17.2 Identifying unknown entries

First we select some entries which we want to identify. We consider the *Ambiorix sp.* entries (those without species name) as unknown, and we will identify them against the known *Ambiorix* entries (the list **Ambiorix**).

- 17.2.1 In the main window, press F4 to clear the selection.
- 17.2.2 Select all *Ambiorix sp.* entries (in the entry search dialog box, disable *Search in list* and *Negative search* and enter *Ambiorix* in the Genus field and *sp.* in the species field).
- 17.2.3 Copy the selected entries to the clipboard using *Edit | Copy selection* or .
- 17.2.4 Open the saved comparison **Ambiorix**.
- 17.2.5 Paste the selected *Ambiorix sp.* entries with *Edit | Paste selection*.
- 17.2.6 For the identification purpose, we do not need the dendrogram panel (left, see paragraph 9.7 and Figure 66), which you can minimize.
- 17.2.7 Create sufficient space for the matrix panel (right, see paragraph 9.7 and Figure 66), where the similarity values will appear.
- 17.2.8 In the experiment type selection bar, select an experiment by means of which you want to identify the unknown entries. Select for example FAME (fatty acid methyl esters).
- 17.2.9 Click on the first unknown *Ambiorix sp.* entry in the entry names panel (to place the selection bar on it).

17.2.10 In the menu of the comparison window, choose *Edit | Arrange entries by similarity*.

The entry with the selection focus stands on top and all the other entries in the comparison are arranged by decreasing similarity with that entry. The similarity values are shown in the matrix panel.

17.2.11 You can click on the  button of **FAME** to display the images and drag the horizontal separator line down to show the complete names of the fatty acids.

The *Arrange entries by similarity* function can be repeated for each experiment type and for composite data sets, in order to compare the different results. The program uses the similarity coefficient which is specified in the experiment type window.

17.2.12 A printout of the list of similarity values can be obtained with *File | Print database fields*.

17.2.13 An export file of the similarity values is created with *File | Export database fields*.

*NOTE: In case of a fingerprint type, you can also show the number of different bands between a highlighted entry and the other entries, by selecting **Different bands** as the default similarity coefficient. Before selecting **Edit|Arrange entries by similarity**, you should enable **Layout|Show distances**.*

18. Identification using libraries

A *library* is a collection of *library units*, which in turn is a selection of database entries. A library unit is supposed to be a definable *taxon*. When generating libraries for identification, a new library is first created. Then are library units within that library defined, to which the names of the taxa are given. Within each library unit, a selection of representative entries for that taxon is entered.

18.1 Creating a library

18.1.1 In the main window with **DemoBase** loaded, select *Identification|Create new library* from the menu.

18.1.2 Enter a name for the library, for example **Demolib**.

The *Library window* of new library appears (Figure 126). The left panel shows the available experiment types and composite data sets, and the right panel shows the library units defined within the library. This panel is initially empty.

Within the library, you can include and exclude experiment types and composite data sets. Excluded experiments will not be used for identification.

18.1.3 Select an experiment which you do not want use for identification, for example a composite data set.

18.1.4 In the menu, choose *Experiment|Use for identification*. Experiments that are used for identification are marked with ✓; experiments that are not used are marked with a red cross.

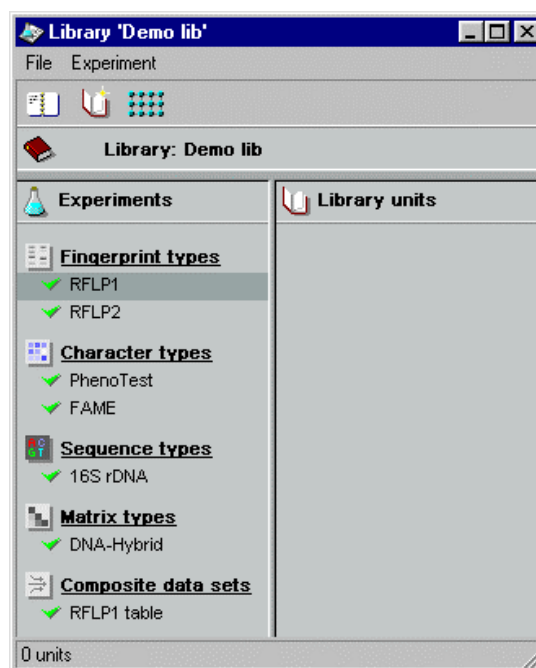


Figure 126. The library window of a new library.

18.1.5 Select **File | Add new library unit** or .

18.1.6 Enter a name of one of the species in the database, for example *Ambiorix sylvestris*.

The library unit now shows up in the right panel.

18.1.7 Double click on the unit, or select it and **File | Edit library unit**.

The Library unit window which appears is very similar to the comparison window, and allows all the same clustering functions as in the comparison window. This allows you to cluster the members of a library unit internally in order to check the homogeneity of a defined taxon.

18.1.8 In the database, select all *Ambiorix sylvestris* entries and copy them to the clipboard using **Edit | Copy selection** or .

18.1.9 Paste the entries in the library unit with **Edit | Paste selection**.

18.1.10 Save the library unit with **File | Save**.

18.1.11 Repeat 18.1.5 to 18.1.10 to create library units for the named species.

18.1.12 When finished, close the library with **File | Exit**.

The library is now listed in the libraries panel of the main window. You can open the library and add or edit units whenever desired.

18.2 Identifying entries against a library

18.2.1 Clear any selected entries in the database with F4.

18.2.2 Select a list of entries, for example all unnamed species (*Ambiorix* sp. and *Perdrix* sp.) and a few entries of the other species.

18.2.3 Select **Demolib** under **Libraries** and **Identification** | **Identify selected entries**.

A dialog box appears, asking to *Calculate quality factors*.

The *Quality value* is an indication of the confidence of the identification. It is achieved by comparing the average similarity between the unknown entry and the library unit's entries with the average similarity of the library unit's entries with each other. If the first value is as high or higher than the second one, the unknown entry fits well within the library unit. Thus this quality indication takes into account the internal heterogeneity of the taxon defined in the library unit.

18.2.4 Answer <Yes> to this question.

The *Identification window* appears, showing the progress of the calculations in the caption. Once the calculations are done, the window is divided in three panels (Figure 127). The left panel lists the unknown entries, and the right panel the experiment types (columns) by which the unknown entries were identified. You can select one of the experiment types (columns), and the central panel shows the best matching library units for the selected experiment. For the not selected experiments, it is obvious that the best matching library unit may be another one. Therefore, the place of the library unit in decreasing order of match is indicated for each experiment (gray number between square brackets). An ideal identification is when all experiment types show [1].

Identification library: Demolib				RFLP1	RFLP2	FAME	PhenoTest	16S rDN
Genus	Species	Strain n°	Library units					
Ambiorix	sylvestris	52441	Ambiorix sylvestris	77.8 [1]	82.2 [1]	93.5 [1]	96.1 [1]	97.3 [1]
Ambiorix	sp.	52415	Perdrix pseudoarchaeus	95.1 [1]	88.5 [1]	93.0 [1]	43.2 [3]	90.5 [1]
Vercingetorix	nemorosum	42816	Vercingetorix palustris	92.3 [1]	78.8 [3]	87.3 [1]	72.0 [3]	95.4 [1]
Vercingetorix	aquaticus	42853	Vercingetorix aquaticus	95.5 [1]	90.1 [1]	99.9 [1]	93.9 [1]	98.8 [1]
Ambiorix	sp.	52424	Ambiorix sylvestris	82.9 [1]	77.6 [1]	86.6 [3]	70.1 [1]	95.7 [1]
Perdrix	pseudoarchaeus	25671	Perdrix pseudoarchaeus	97.2 [1]	87.6 [1]	99.0 [1]	92.0 [1]	99.0 [1]
Perdrix	sp.	53175	Perdrix pseudoarchaeus	70.6 [1]	17.9 [4]	55.6 [2]	85.7 [1]	92.4 [1]
Vercingetorix	palustris	42819	Vercingetorix palustris	92.3 [1]	93.7 [1]	99.9 [1]	88.3 [1]	99.6 [1]
Perdrix	sp.	25693	Perdrix pseudoarchaeus	55.8 [1]	28.9 [3]	51.5 [3]	83.5 [1]	93.1 [1]
Ambiorix	aberrans	52452	Perdrix pseudoarchaeus	89.9 [1]	64.0 [2]	75.9 [3]	57.6 [3]	91.5 [1]
Ambiorix	sp.	52440	Ambiorix sylvestris	83.5 [1]	86.8 [2]	90.1 [2]	72.5 [1]	95.9 [1]

11 unknowns Mean similarity RFLP1 : Dice (Opt:1.00%) (Tol 1.0%-1.0%) (H>0.0% S>0.0%) [15.0%-85.0%]

Figure 127. Identification window.

18.2.5 You can move the vertical separator lines between the panels.

The identification scores are the similarity values obtained using the coefficient which is specified in the settings of the experiment type. The program calculates the similarity of the unknown entry with each of the library unit's entries and uses the **average** of these values.

18.2.6 Select **Show** | **Maximum similarities** to display the maximum similarity found within a library unit instead of the average similarity.

Obviously, the best identification may be different if you choose the maximum similarity instead of the average similarity.

The quality factors appear as colored dots next to the identification scores. They range from red (improbable identification) over orange, yellow (doubtful identification) to green (faithful identification).

18.2.7 Print the global identification report with *File|Print report*, or create a text report of it with *File|Export report to file*.

For routine identification purposes, it can be useful to be able to store the identification results for each unknown entry. Thereto, one can create a dedicated field in the database, and use the following command:

18.2.8 Click on the dedicated field where you want to store the identification result, and select *File|Fill information field*.

18.3 Detailed identification reports

For each unknown entry, you can display a more detailed identification report.

18.3.1 Select one of the unknown entries and *Show|Detailed report* (or double click).

The *Detailed identification* window (Figure 128) displays the best matching library units in decreasing order for the selected experiment type (columns). The experiment type which was selected in the identification window will be selected here too. For the other experiment types, the order of best matching library units is likely to be different. Therefore, the place of the library unit in decreasing order of match is indicated for each experiment type (gray number between square brackets). In the ideal case, all experiment types would show [1].

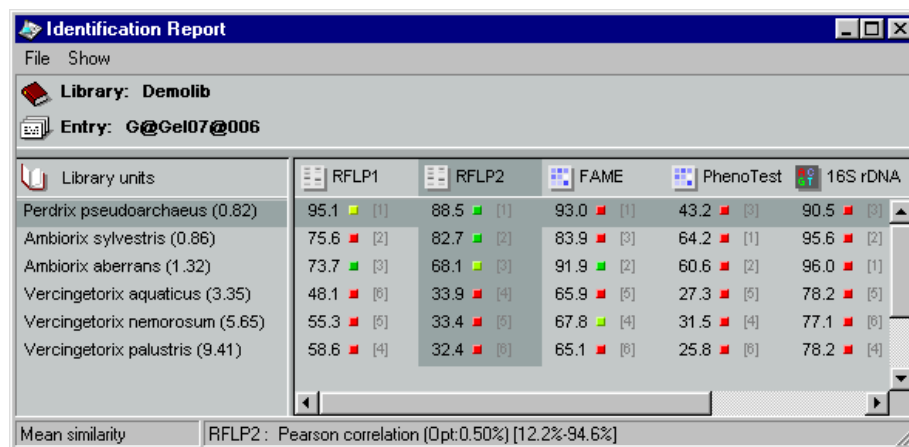


Figure 128. Detailed identification window.

18.3.2 You can print or export the report with *File|Print report* or *File|Export report to file*.

18.3.3 With *Show|Identification comparison*, an *Identification comparison* is created.

This window is in fact a comparison window which lists the unknown entry and the entries of the library unit. The unknown entry is displayed in red, except when it is one of the library unit's members; in that case it is displayed in blue.

18.4 Creating a neural network

Theory

A neural network is a means of calculating a function of which one doesn't have a clear description, but of which many examples with known input and output are present. Typically, the input is a set of characters for each example, and the output is the name of a group to which the example belongs. The neural network can be trained with the examples, and if the training succeeds well, the neural network can be used to perform the same calculation with other data of which the output is not known. Usually, all the examples that are fed to the neural network are divided randomly in a *training set* and a *validation set*. The training set is the part of the example set that will be used to calculate the neural network and the validation set is the part that will be used to validate the network, i.e. check its correctness on other examples than the ones used for training.

A neural network consists of several *layers* of *neurons* or *nodes*; mostly there are 2 or 3 layers. The first layer is the *input layer*, the last one is the *output layer*, and the intermediate ones - if present - are called the *hidden layers*. Usually there are 0 or 1 hidden layers. Every neuron or node has a value that is calculated by the neural network. The values of the neurons in the input layer are simply the input of the function. Every neuron in the successive layers takes the value of all the neurons in the previous layer and performs a calculation on it, to obtain its own value. Mostly this calculation is a weighted sum, in which the weights can be different for every neuron. That value will be used by neurons in consecutive layers. The number of nodes in the input layer is equal to the number of characters available for the data set, i.e. the number of characters in the experiment which is used to calculate the neural network. The number of nodes in the output layer is equal to the number of groups defined in the identification system. The number of nodes in the hidden layer - of any - can be chosen and is dependent on the nature and complexity of the data set and identification system.

During the training cycle the input of a known example is fed in the neural network and the calculation is performed. Initially the calculated output will most likely be very different from what it should be. The weights between every pair of consecutive neurons are then adjusted slightly, so that the calculated output becomes closer to the correct output. This is done using a process called *back-propagation*. This means that in the output layer the errors are calculated, which are the difference between the correct output and the calculated output. These errors are then back-propagated to the neurons in previous layers by multiplying the error by the weight that connects two neurons, and summing for every neuron. The weights of the neurons are then adjusted by the error times a number called the *learning ratio*. Furthermore the weight correction of the previous training cycle times a number called the *momentum* is added. The higher the learning ratio and the momentum, the faster the training, but the higher the risk that the error doesn't decrease.

This training process is repeated many times (typically a few thousand times), each time with another known example chosen randomly from the training set. After sufficient *iterations* the calculated outputs will be very close to what they should be, provided that the number of layers and number of nodes per hidden layer is chosen correctly. A higher number of layers and/or neurons means that training and calculation will take longer, so a trade-off has to be made. Furthermore, there is a danger of *overtraining* when there are too many layers and/or neurons, which means that the neural network would be very good for the examples, but not at all for other inputs. To have an estimate of this, one usually divides the known examples in a *training set* and a *validation set*. The validation set is not used for training, but only to check how well the neural network performs on this set. If it is significantly worse than for the training set, one knows that there are too many layers and/or neurons.

Application

A neural network can be applied to many problems, such as control theory, character recognition, statistical analysis and distinguishing patterns. In practice, a neural network is very useful to set up an identification or recognition system based upon complex data sets in which it is not easy or impossible to identify discriminatory keys based upon conventional methods such as calculation of similarity using coefficients, cluster analysis, principal components analysis etc. An important requirement for successfully applying neural networks is that the example data set is sufficiently large and that many examples are present for each group of the identification system.

In our software, it is used for determining to what predefined group or taxon an unknown database entry belongs, based on measurements that could be a character set or a fingerprint. This is thus an example of distinguishing patterns. In this case the output of the neural network is n values, where n is the number of predefined groups. Every group is given a number from 1 to n , and thereby corresponds to one of the outputs. The higher a value in the output, the more likely the sample belongs to that group. In the training and validation set the output values are zero, except for the output that corresponds to the group, which will be one. After the training has succeeded one can use it with measurements on unknown samples. In these, the highest output will be decisive for what group it is.

In Fingerprinting II Informatix, the choice in hidden layers is limited to none or one, because more hidden layers usually don't give any advantage. In extensive tests, one hidden layer was always sufficient, in many cases no hidden layer worked just as well. The number of nodes in the hidden layer can be chosen if the user wants to do so. If the user doesn't specify this, the neural network will start without a hidden layer. If it doesn't succeed in lowering the error, a hidden layer will be created. If this still doesn't lower the error, the hidden layer is expanded until the error is below a predefined threshold.

The learning rate and momentum cannot be specified. Instead we fixed these to 0.5 and 0.1 respectively, because in our tests these values gave the optimal trade-off between speed and success.

To train a neural network, a library must be present. See 18.1 to create a new library. To obtain a reliable neural network, each of the library units must have sufficient members, for example many more than just two or three. The number of entries requires also depends on the heterogeneity of the group: the more heterogeneous a group, the more entries that will be needed to create a reliable neural network.

18.4.1 Double-click on a library to open it.

18.4.2 Select *Experiment | Train neural network* or .

The *Neural network training* dialog box appears, as shown in Figure 129.

18.4.3 Under "**Select experiment to be used in the neural network**", you can select the experiment to train the neural network.

18.4.4 With "**Validation samples**", it is possible to specify the percentage of the library entries (i.e. the example data) to be used as validation set. By default this value is 25%.

18.4.5 With "**Max. number of iterations**", you can specify the maximum number of training cycles to be performed. By default this value is 20000.

18.4.6 "**Number of hidden nodes**" allows you to manually specify the number of hidden nodes. If you leave this field blank, the program will automatically determine whether a hidden layer is required, and if so, the optimal number of hidden nodes. If you enter zero, no hidden layer will be created.

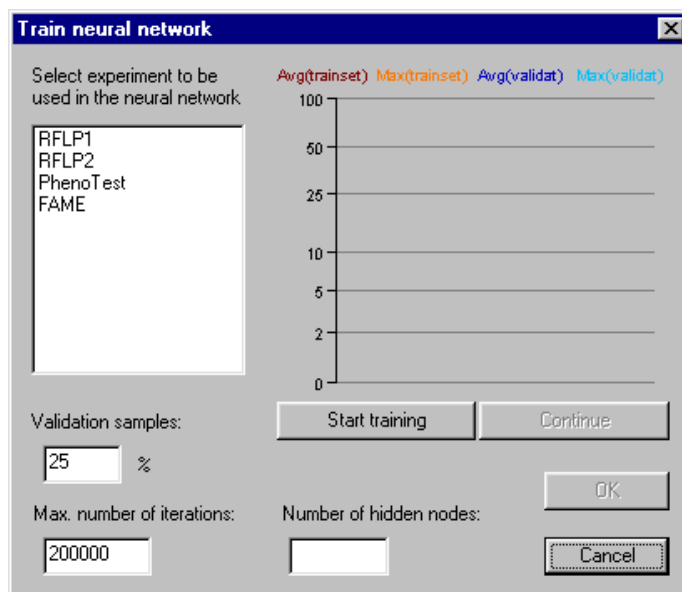


Figure 129. The Neural network training dialog box.

18.4.7 When all parameters are entered, press **<Start training>** to start the training process. Depending on the size of the library, the training proces can take several minutes. An animation of the progress of the training is shown in the x-t diagram (Figure 129).

18.4.8 During the training, it is possible to interrupt or abort the process by pressing **<Stop>**.

18.4.9 If you wish to resume the training process, press **<Continue>**. The program will continue the iteration process untill the max number is achieved.

18.4.10 The save the neural network, press **<OK>**.

19. Database exchange tools

Fingerprinting II Informatix offers a simple and powerful solution to exchange database information between research sites on a peer-to-peer basis. For a selection of database entries, selected information (e.g., experiment types, information fields) can be bundled into a *Bundle*. Such a bundle is a compact data package contained in one file, which can be sent to other research sites over the Internet, as e-mail attachment or via FTP. The receiver can open the bundle directly in Fingerprinting II Informatix and compare the entries contained in it with the own database.

Besides the numerical information of the experiments, a bundle contains all the information of the experiment type, so that Fingerprinting II Informatix can check whether the experiment types contained in the bundle are compatible with those of the receiver's database. If an experiment type in a bundle is not compatible, this experiment type will be automatically created in the receiver's database. If the bundle contains a database information field which is not defined for the database, this information field will be added to the database.

In case of fingerprint types, the bundle holds the complete information about the reference system used and the molecular weight regression, so that Fingerprinting II Informatix can automatically remap the bundle fingerprints to be compatible with the database fingerprints.

19.1 Creating a new bundle

19.1.1 In the main window with **DemoBase** loaded, select all entries belonging to *Vercingetorix*.

19.1.2 Select *File | Create new bundle* or .

The *Create new bundle* dialog box (Figure 130) lists the available database information fields in the left panel and all available experiment types in the right panel.

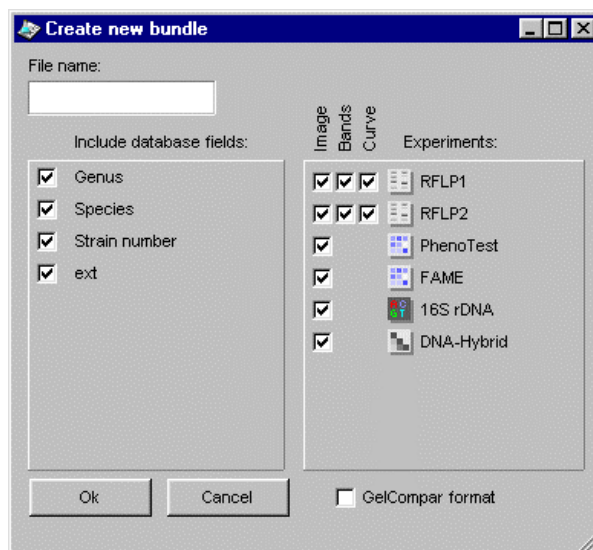


Figure 130. The Create new bundle dialog box.

You can check each of the database information fields and experiment types to be incorporated in the bundle. For fingerprint types, the fingerprint images, band information, and densitometric curves can be incorporated separately.

19.1.3 Leave all checkboxes checked.

*NOTE: With the checkbox **GelCompar format**, one can save bundles in the format of GelCompar versions 4.1 and 4.2 and Molecular Analyst Fingerprinting versions 1.12 through 1.60. Only Fingerprint information can be saved in this format. Fingerprinting II Informatix also recognizes and reads GelCompar and Molecular Analyst Fingerprinting bundles.*

19.1.4 Enter a name for the bundle, for example **Vercingetorix**, and press <OK> to create the bundle.

A bundle file **Vercingetorix.BDL** is created in the **BUNDLES** directory of **DemoBase** (see Figure 3 for the directory structure).

19.1.5 As an example for database exchange, copy **Vercingetorix.BDL** to the **BUNDLES** directory of database **Example**.

19.2 Opening an existing bundle

19.2.1 Close Fingerprinting II Informatix and restart the main program under database **Example**.

19.2.2 In database **Example**, select **File | Open bundle** or .

In the *Open/close bundles* dialog box (Figure 131), you can select a bundle in the list of available bundles in the **BUNDLES** directory.

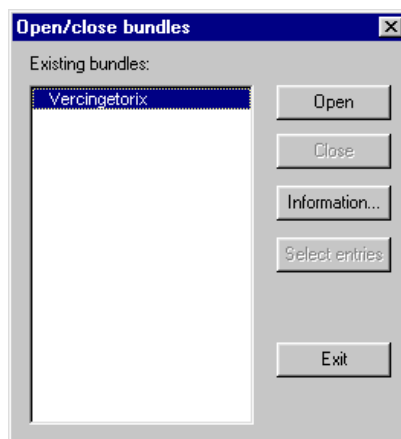


Figure 131. The Open/close bundles dialog box.

19.2.3 Select **Vercingetorix** and press the **<Information>** button.

This opens the *Bundle information* dialog box for the selected bundle (Figure 132). It shows the available information fields in the bundle, as well as the experiment types contained in it. If an information field or an experiment type is recognized as one of the fields or experiment types in the database, a green dot is shown left from it. If not, a red dot is shown left from it. As soon as the bundle is opened, the missing information fields and experiment types are automatically added to the database.

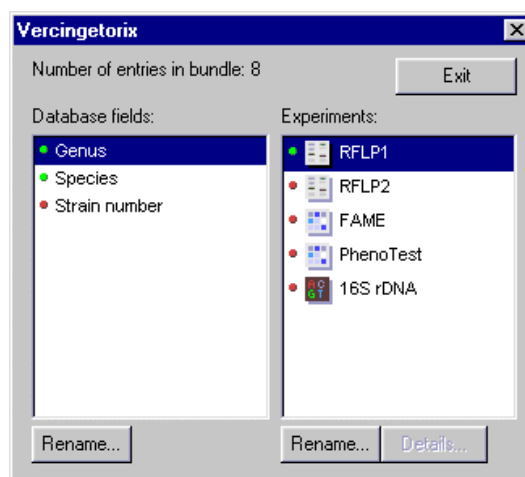


Figure 132. The bundle information dialog box.

For example, in the **Example** database, we have created an information field **Strain no.** This clearly corresponds to the information field **Strain number** in the bundle, but since the names are different, Fingerprinting II Informatix would add a new information field to the database. To avoid this, you can rename the information fields in the bundle.

19.2.4 Select **Strain number** and press the **<Rename>** button under the information fields panel.

19.2.5 Enter **Strain no** and press **<OK>**.

The information field **Strain no** now has a green dot left from it, indicating that it corresponds to the information field in the database.

A similar problem can happen for the experiment types: another user may have given a different name to the same technique, and this would Fingerprinting II Informatix cause to consider the techniques as different experiment types. If you know a technique in a bundle is the same as one of the experiment types defined in the database, you can also rename it using the **<Rename>** button under the experiment types panel.

In addition, in character types, the characters may have received different names by other users. For example, institution 1 may have named a character "**Alpha-Glucosidase**", and institution 2 "**a-Glucosidase**". Obviously, Fingerprinting II Informatix will consider these different names as different tests. To avoid this, you can select the character type and press the **<Details>** button. A list of all characters in the experiment type is shown, and those corresponding to characters in the database's experiment type are marked with a green dot; the characters not recognized in the database's experiment type are marked with a red dot. You can rename such characters with the **<Rename>** button.

19.2.6 **<Exit>** the bundle information dialog box, and press **<Open>** to load the bundle into the database.

If a bundle is loaded, it is marked with a square dot in the *Open/close bundle* dialog box.

19.2.7 To close a loaded bundle, select it in the list and press the **<Close>** button.

19.2.8 Press **<Exit>** to close the *Open/close bundle* dialog box.

In the database, entries from a bundle are recognizable by the bluish color of their entry keys. For all functions, they behave like normal database entries. If you exit Fingerprinting II Informatix, they are not automatically loaded when you run the software again. If you know a saved comparison contains bundle entries, you should load the bundles before opening the comparison, in order to avoid an error message.

NOTE: If you want a bundle to be always opened with the database when Fingerprinting II Informatix is started up, you should rename it to contain the prefix @_ before its name.

20. The Fingerprinting II Informatix client functions

20.1 Introduction

The enormous potential offered by the internet is taken advantage of in the Fingerprinting II Informatix software with its integrated *Database Sharing Tools*. The key component behind the exchange of data is the so-called *Bundle*. A bundle is a packed file that contains a selection of Fingerprinting II Informatix or Fingerprinting II database entries (e.g. bacterial strains), along with one or more experiments and information fields, as specified by the user. The creation and use of bundles is described in chapter 19.

Fingerprinting II Informatix and Fingerprinting II users having the Database Sharing Tools can exchange information at a peer-to-peer level as described in chapter 19. However, in major collaborative research projects, the generation of one centrally coordinated database will be preferred over uncontrolled data flow between research sites. It is here that the Fingerprinting II Informatix Client-Server solution comes in, based upon a powerful *Server suite* on the one hand and the *Client tools* on the other hand.

In collaboration with the company Applied Maths BVBA (St Martens-Latem, Belgium), Fingerprinting II and Fingerprinting II Informatix offer the possibility to organize structured Client-Server interactions. Each Fingerprinting II Informatix and Fingerprinting II software package that contains the Database Sharing Tools comes as a *client* version, which possesses all functionality to connect and communicate with a *Server*. The **server** is basically a Windows NT service, which runs on a dedicated computer available to other computers via Internet. The **server** program runs in background below the user level while processing tasks and queries from client users. The **server** provides two fundamentally different solutions for sharing its databases and offering its services: a *direct TCP/IP based* connection and a *web-based* communication (see Figure 133).

Solution 1: Direct TCP/IP connection. This solution provides a direct connection between the **server** and the client which allows uploading and downloading complete database entries, interactive querying, and batch identification by the client (see Figure 133, upper part). The key components for this direct and powerful client-server communication are the bundle concept and TCP/IP internet communication. The **server**, which contains the Fingerprinting II Informatix reference database(s), runs BNSERVER, a Windows NT service which automatically performs identifications, answers queries, and maintains logfiles. The communication happens through TCP/IP, and bundles are the database information packages transferred. The client is a Fingerprinting II Informatix or Fingerprinting II user possessing the Database Sharing Tools module. Once a client has obtained permission from the **server** to query a database, e.g. by entering a correct login and password, it can connect to the server database and perform one or more of the following functions, according to permittance settings specified by the **server**:

- Perform interactive searches on the server database and download queries;
- Send data to an *Incoming* directory of the **server**;
- Request the **server** to perform remote identifications and receive back detailed reports with possibility to request full information and download.

Solution 2: Web-based communication using HTML. This solution requires Microsoft's *Internet Information Server* (IIS) at the **server** site and an additional Fingerprinting II Informatix tool BNWWW, which is an ISAPI extension translating HTML requests into commands for BNSERVER (see Figure 133, lower part).

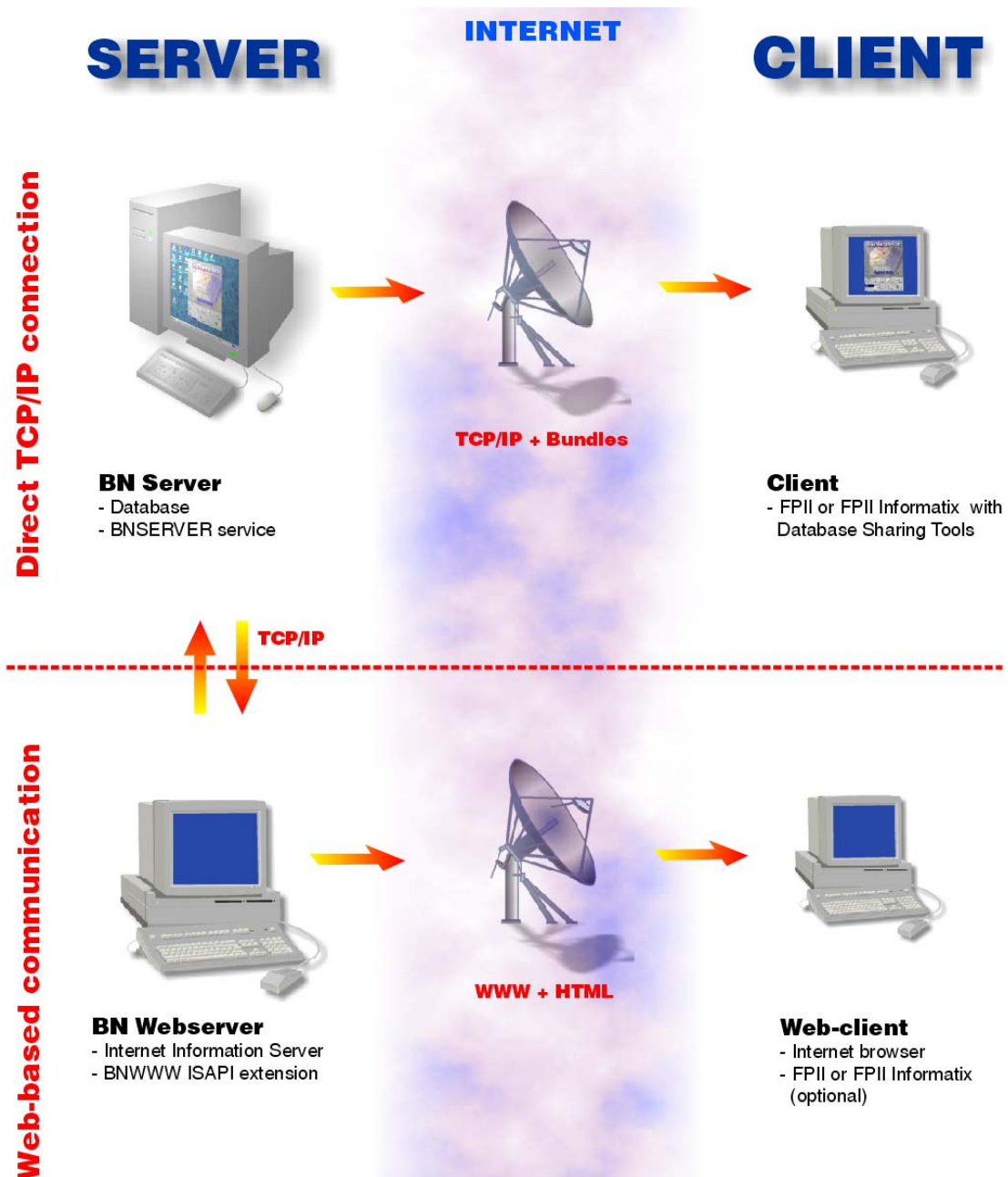


Figure 133. Schematic representation of the two approaches for Internet exchange used by the Server: direct TCP/IP based and HTML web-based.

The *Webserver* can be a separate computer, for example, placed outside the firewall. In this case, the **server** becomes an HTTP-server and communication happens through HTML. The client only needs a web browser to submit fingerprints, profiles or sequences to the **server**; the result is returned as an HTML page documented with identification reports, fingerprint images, and links for requesting detailed information about particular closest matches. If allowed by the **server**, the client can download bundles from its web browser.

It will only be possible to execute the features described in this chapter when you have a valid client login on the **server**.

20.2 Connecting to the server database

20.2.1 In the Fingerprinting II Informatix main window, select **Database | Connect to server** or



The *Server connection* dialog box appears (Figure 134), prompting for the *Server computer name or IP address* and a *Port number*.

20.2.2 Under *Server computer name or IP address*, you can either fill in the *full* computer name of the server computer (computer name + domain name, separated by a period) or its IP address. When working under the same domain name as client, you do not need to enter the domain name.

Leave the default port number unless a different port number was specified on the **server**.

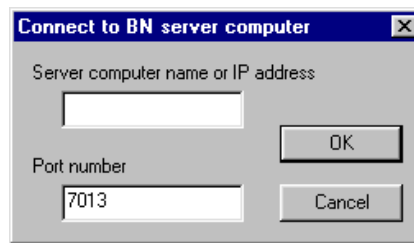


Figure 134. The server connection dialog box.

20.2.3 Press **<OK>** to establish the connection.

Next, the database connection dialog box (Figure 135) prompts you for a *Login name* and a *Password*.

20.2.4 Enter your *Login name* and, if required, your *Password*.

20.2.5 Press **<OK>** to establish the connection with the database.

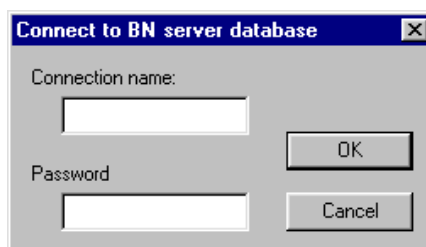


Figure 135. The database connection dialog box.

If the connection is successfully established, the *Server login* window appears (Figure 136), showing the *database components* of the server database in the **Configuration** tab. The database components seen in this window are only the components which are made accessible for your login by the **server**. If a

component is underlined in green, it means that this component is found back in your own client database. If it is not underlined and gray, the component could not found in your client database.

Each server database component contains a checkbox, which is checked by default. The **Fingerprint types** contain three checkboxes, corresponding to the *images*, *bands* and *curves*, respectively.

20.2.6 You can uncheck a certain component if you do not wish to download that component.

For example, if you do not wish to download the gelstrip images for the fingerprint types RFLP1 and RFLP2 in the demobase (Figure 136), you should uncheck the first checkbox of RFLP1 and RFLP2.

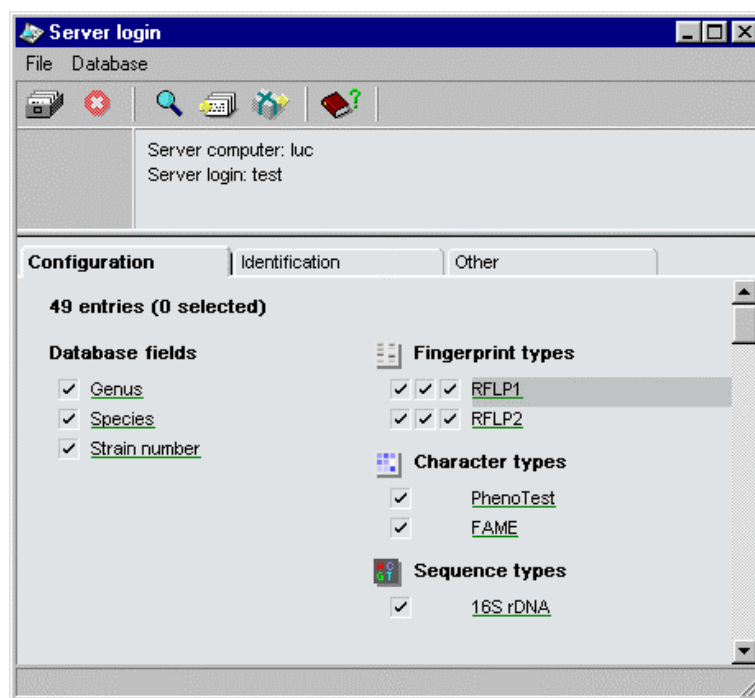


Figure 136. The Server login window.

20.3 Searching and downloading entries on the server database

If permitted by the **server** for the login, the client can perform a search on the server database, using the database fields as search criterion. The **Configuration** panel in the server login window displays the total number of entries in the server database and between brackets the number of entries selected by the client (see Figure 136).

20.3.1 Select **Database | Search** or press the  button.

The *Server database search* dialog box appears (Figure 137). Left, you can select a **Database field**, and right you can fill in a **Search string**. Wildcards can be used to search for substrings: an **asterisk *** replaces any range of characters in the beginning or the end of a string, whereas a **question mark ?** replaces one single character.

Normally, successive searches are additive: new searches are added to the selection list. The **Search in list** checkbox allows you to refine the search within a list of selected entries, whereas **Replace list** replaces the existing selection list with the new one.

With *Negative search*, all entries that do not match the specified criteria will be selected.

Case sensitive lets the program make a distinction between uppercase and lowercase.

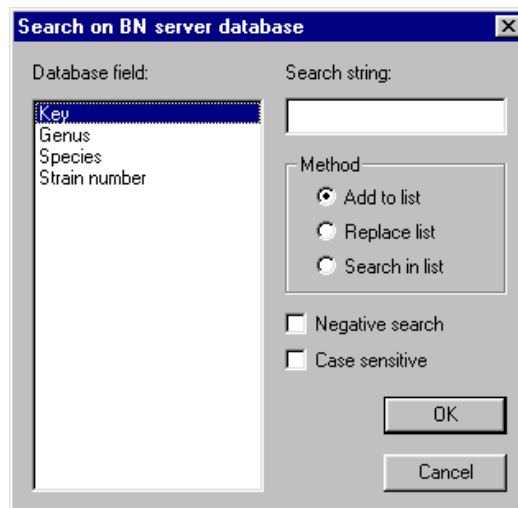


Figure 137. The Server database search dialog box.

20.3.2 Press <OK> to perform a search on the server database. When entries are selected, the number is shown as "**selected**" between brackets.

20.3.3 The selected entries can be downloaded from the **server** with *Database|Download selection list* or .

NOTE: Downloading is only possible when permitted by the server for the login.

20.3.4 A time-consuming data transfer can be stopped at any time by pressing the  button.

When downloading is complete, the downloaded server entries are added to the client database and shown in blue-green color. In addition, they are selected (blue arrows) and a *comparison* window with these entries is automatically created. Note that only the components that were checked in the **Configuration** tab are downloaded (see page 222).

Thus entries from the server are downloaded in a *temporary bundle*; after closing the Fingerprinting II Informatix session, the entries will not reappear. To save them, you should either save them in a bundle during the Fingerprinting II Informatix session (see 19.1) or rename the temporary bundle in the **temp** subdirectory of your client database, also before closing Fingerprinting II Informatix. After a default installation, this directory is found as

C:\Program files\Fingerprinting II Informatix\data\[DatabaseName]\temp.

The downloaded bundle has the following name format:

YYYY-MM-DD__HHhMMmSSs__XXXX.BDL

You can copy this bundle to the **bundles** subdirectory of the same database, and rename it as necessary.

- 20.3.5 If you want a bundle to be always opened with the database when Fingerprinting II Informatix is started up, you should rename it to contain the prefix @_ before its name.

20.4 Uploading bundles and gels to the server

Uploading a bundle

Before you can send a bundle, it must be created in the client database. To create a bundle, see 19.1. Bundles and gel files can only be uploaded when allowed for this login by the server.

- 20.4.1 Select *Database | Send bundle to server* or press the  button.

- 20.4.2 Select a bundle from the file dialog box.

The default directory shown is the bundles subdirectory of the client database. If you wish to select a bundle from elsewhere, you can do so.

Uploading a gel

Uploading a gel to the server is not done from the *server login* window, although a connection to the server must be established first as described in 20.2, which automatically pops up the server window.

- 20.4.3 In the Fingerprinting II Informatix main window, select the appropriate **Fingerprint type** in the **Experiments** panel.

- 20.4.4 In the **Files** panel, double-click on the gel you want to upload, to open the gel file.

- 20.4.5 Select *File | Upload file to server* from the menu.

A *Fingerprint file upload* dialog box (Figure 138) prompts you to enter some comments (optional). In this comment editor you can write or paste some text.

Furthermore, the option is offered to send the following files:

- *Fingerprint lanes*: densitometric curves, normalization info, gelstrips, bands, and quantification, all contained in the **.DEF** file
- *Image file*: The raw image or curve file, usually a **.TIF** file
- *Database fields*: the definition of the database entries for the fingerprint labels, contained in the **.DBS** file.

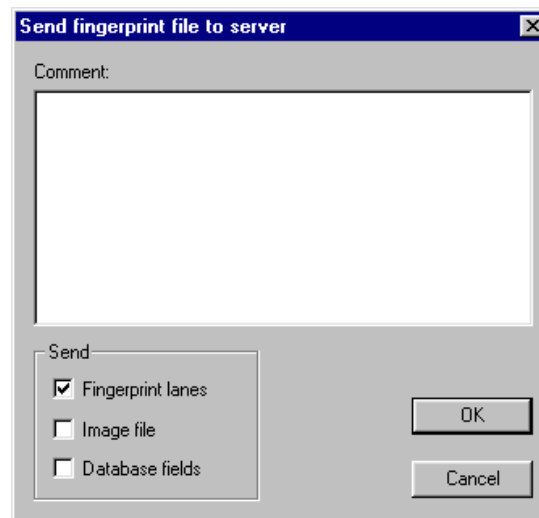


Figure 138. Fingerprint file upload dialog box.

20.4.6 Press <OK> to upload the fingerprint file.

20.5 Performing identifications on the server

When allowed for your login by the server, you can perform batch identifications on the server database.

20.5.1 In the client database, select one entry, or create a selection of entries which you wish to identify. Selected entries are marked with a blue arrow.

20.5.2 In the server login window, select *Database|Identify selection list*, or simply press the



button.

The *Server database identification* dialog box (Figure 139) allows you to select the *Experiment* which you want to use for identification.

It further allows a *Number of matches*, i.e. the number of best matching server database entries, to be displayed ranked according to similarity. If you enter 10 (default), the 10 best matching server database entries will be shown for each unknown entry you have identified.

The Method includes choices *Use complete database*, and *Use selection list only*. In the latter case, any selection you have made in the server database (see paragraph 20.3) will be used to identify your unknown entries.

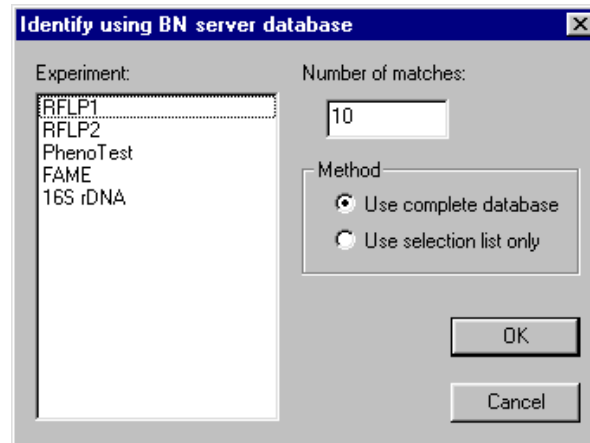


Figure 139. Server database identification dialog box.

20.5.3 Press **<OK>** to perform the identification.

When the identification is complete, the *Identification* tab of the server login window pops up (Figure 140). This window shows the experiment used, the similarity coefficient used, and a list of all unknown entries (represented by their keys) with their closest server database matches (indented). Of the server database entries, the scores, keys and database fields are shown.

Left from each unknown entry and each server entry, an orange arrow is shown (Figure 140).

20.5.4 If you click on the orange arrow left from a particular matching server database entry, it is downloaded in a temporary bundle (see page 223) to your client database (shown in the database in blue-green), and its entry edit window is popped up.

20.5.5 If you click on the orange arrow left from an unknown entry, all the best matching server database entries are downloaded in a temporary bundle (see page 223) to your client database (shown in the database in blue-green); they are selected (blue arrows) and a comparison window is popped up with the downloaded entries.

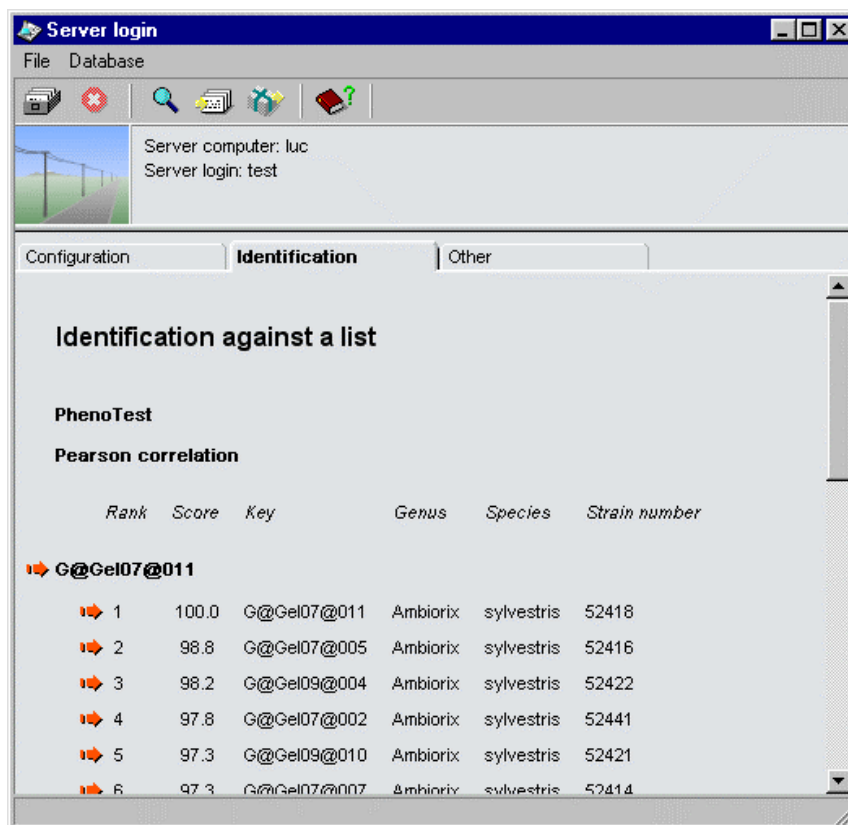


Figure 140. The Identification tab of the server login window after identification.

*NOTE: After closing Fingerprinting II Informatix, the downloaded entries will not reappear since they were saved in a temporary bundle in the **temp** subdirectory. To preserve these entries, see page 223.*

By default, the identifications performed on the server are done using the server's **Comparison Settings** (i.e. the similarity coefficient used, the position and optimization settings applied). However, it is possible to perform identifications using the comparison settings specified in your local database.

20.5.6 To upload your own local comparison settings for the selected experiment in the server login window, select **Database|Upload comparison settings** in the server login window prior to performing the identification.

20.5.7 To close a connection with the server, simply exit the server login window.

*NOTE: Depending on the time-out setting specified on the server, it is possible that a connection is closed after a certain time of inactivity from the client side. Use the menu **File|Open database** or press the  button to establish a new connection.*

21. Import of data from external databases

The databases in Fingerprinting II Informatix are stored in a binary format and can only be exported via the available export functions rather than by reading or copying database files directly. Within Fingerprinting II Informatix, an additional Fingerprinting II Informatix database can be opened quite easily using the menu command *File|Open additional database* in the Fingerprinting II Informatix main window. When the additional database contains experiments and/or database fields that are not available in the own database, Fingerprinting II Informatix will automatically create these components in the own database in order to be able to display them.

Non-Fingerprinting II Informatix databases can only be accessed in Fingerprinting II Informatix through import functions (e.g. scripts) or a common exchange language. Therefore, Fingerprinting II Informatix allows one to establish a link with an external relational database using the *Open Database Connectivity* (ODBC) protocol. This protocol is supported by almost any commercial relational database: Access, Excel, FoxPro, Dbase, Oracle, SQL server, etc... By establishing such a link between Fingerprinting II Informatix and an external data source, the user can import data in a completely transparent way into Fingerprinting II Informatix. Moreover, the Fingerprinting II Informatix database can be brought up to date using the external data source by performing automatic downloads.

The database records in the external database are mapped into Fingerprinting II Informatix entries by making use of the *database key*. The user should specify a field of the external database that corresponds to this key, and then the software is able to automatically determine which external record corresponds to which local Fingerprinting II Informatix database entry.

21.1 Setting up the ODBC link

Use the menu item *Database|Configure external database link* in the main window to call the ODBC configuration dialog box. This dialog box contains two information fields, which are to be filled in:

1. The ODBC data source. This field is to be filled in with a string that defines the external database that will be linked using ODBC. If you are familiar with ODBC, you can specify a string manually. Alternatively, you can press the button **<Select>**. This action pops up the standard Windows dialog box that allows one to select an ODBC data source. In this dialog box, double click on the name of the appropriate available database software and select the database file on the hard disk.

2. The Database table or query. In this field, you should fill in the name of the table or query in the external database that you want to use to import data from. If you are importing data from a spreadsheet program (e.g. Microsoft Excel), you should first create a "table" in the spreadsheet. This can be done by selecting a range of cells that you want to export and assign a name to this selection (read the documentation of the spreadsheet software on how to export data using an ODBC link).

Pressing **<OK>** in this dialog creates the ODBC database import window. This window allows the user to specify how each field in the external database should be mapped to a particular field in the Fingerprinting II Informatix database. On the left side, the Fingerprinting II Informatix fields are listed, while on the right side the external database fields are listed. Initially, all fields are unlinked. You can link two fields by selecting the local Fingerprinting II Informatix field from the left column and the external field from the right column, and pressing the **<Link>** button. At this time, both fields are displayed at the same height, and a green arrow indicates the established link. You can remove any existing link by selecting it and pressing **<Unlink>**.

Before you will be able to perform any exchange action, you should make sure that the Fingerprinting II Informatix “Key” field, which corresponds to the local database keys, is linked to a field from the external database. This link is obligatory, because the software needs to know which record in the external database corresponds to which entry in Fingerprinting II Informatix.

If the necessary links are established between external and local database fields, press <OK> to validate the ODBC link configuration. At this moment, Fingerprinting II Informatix is ready to download information from the external data source.

21.2 Import of database fields using ODBC

Update all Fingerprinting II Informatix database entries from the external data source

It is possible to automatically update all the information fields from each Fingerprinting II Informatix entry, using the data provided by the external database. To this end, select **Database|Copy from external database** in the main window. After confirmation, the software downloads, for each entry, all the database fields that have been linked to the external data source. If the external data source contains records that do not have a corresponding entry in the Fingerprinting II Informatix software, the program automatically creates new entries in the Fingerprinting II Informatix database (after confirmation by the user). In this way, information fields of existing Fingerprinting II Informatix are updated and new entries are automatically added.

Download a database field from the external data source

It is possible to temporarily download an extra database field from the external data source, into an empty database field of Fingerprinting II Informatix. To this end, select the empty database field in the main window (or create a new one), and use the menu command **Database|Download field from external database**. A dialog box pops up, showing all the fields present in the external database. Select the appropriate field and press <OK> to download the information in the local field. Note that the downloaded information is only held temporarily and not stored on disk. The next time you re-open the same database in Fingerprinting II Informatix, the field will be again in its initial state.

Selection of a list using a query in the external data source

The software allows you to perform a query in the external database, and to visualise the result as a selection list in Fingerprinting II Informatix. In the main window, use the command **Database|Select list from external database**. In the dialog box, you can specify a table that should be used to search in (alternatively, you can specify the name of a pre-defined query that is present in the external database). In the next field, you can write an SQL WHERE clause that should be used to build the selection. A complete description of the possible variants is beyond the scope of the manual, and can be found in books on the SQL language. Some possibilities are: “GENUS=‘Ambiorix’” or “GENUS like ‘Amb%’”. The WHERE clause is applied to the records of the external database, and the resulting selection is visualised as a selection of the corresponding entries in the Fingerprinting II Informatix database (supposed that they are present in the local database).

Getting a detailed report of the external database record

For each entry in the Fingerprinting II Informatix database, you can obtain a complete list of all information present in the external data source. To this end, you should first open the *Entry edit* window, e.g. by double clicking on the name in the entry list. Then use the button  to create a new window that shows a list of all information fields that are present for this entry in the external database. Note that there is no limit to the number of fields that can be viewed and edited in this way, and that each field may consist of several lines and can contain up to 5000 characters.

Moreover, you can change some of these fields, and upload these changes to the external database using the button .

21.3 Import of character data using ODBC

One can use an ODBC link to an external database for importing character data into the Fingerprinting II Informatix database. Open the *Character type* that you want to import by double clicking on its name in the experiment window. Then use the command **File|Import from external database**. A dialog box pops up, showing a complete list of all the database fields that are present in the external data source. The program determines automatically if any of the characters in the character type corresponds to a database field in the external database. If so, the field is written in boldface, and the character will be filled with the values from this field during the import. You can add new characters to the character type by selecting an unmatched field and pressing **<Create character>**. Groups of characters to add can be selected using the SHIFT key. To import the data, press **<Ok>**. For every local entry that has a matching key in the external database, the corresponding characters of this character type will be filled with information from the external database.

21.4 Import of sequence data using ODBC

The ODBC link to an external database can also be used to import sequence data into the Fingerprinting II Informatix database. Open the *Sequence type* that you want to import by double clicking on its name in the experiment window. Then use the command **File|Import from external database**. A dialog box pops up, showing a complete list of all the database fields that are present in the external data source. Select the database field that contains the required sequence information and press **<OK>** to import the data. For every local entry that has a matching key in the external database, the corresponding sequence will be filled with the data contained in the selected external database field.

22. Connected databases

Fingerprinting II Informatix offers two possibilities to store its databases: the program's own local database engine (the *local database*) or an external ODBC compatible database engine. The latter solution is called a *Connected Database*. Currently supported database engines are Microsoft Access, SQL Server, and Oracle. Others may work as well but are not guaranteed to be fully compatible in a standard setup.

Connected Databases are particularly useful in the following cases:

- (1) Environments where several users need to access the same database simultaneously. When the Connected Database engine is set up to support multi-user access, Fingerprinting II Informatix will allow multiple users to access and modify the database simultaneously. Note, in this respect, that Fingerprinting II Informatix takes a "snapshot" of the database when the program is launched. As such, changes to the database made by others while you have a Fingerprinting II Informatix session open will not be seen in your current session, until you reload the database during your session (see 22.7).
- (2) When sample information and/or experiment data is already stored in a relational database.
- (3) Laboratories where vast amounts of data are generated. In cases where many thousands of experiment files are accumulated, a powerful database structure such as Oracle or SQL Server will be faster and more efficient in use than Fingerprinting II Informatix' own database.
- (4) When a more flexible database setup is to be achieved, for example with different access/permission settings for different users, and with built-in backup and restore tools.

In a Connected Database, Fingerprinting II Informatix will require a number of tables with specific columns to be available (see 25.1). Fingerprinting II Informatix can either construct its own tables and appropriate fields or link to existing tables and fields in the Connected Database. The latter option is particularly interesting to create a setup where Fingerprinting II Informatix hooks on to an existing database. As soon as a valid Connected Database is defined, the user can start entering information in the Connected Database. Fingerprinting II Informatix writes and reads the information directly into and from the external database, without storing anything locally. Since every Connected Database has a local Fingerprinting II Informatix database associated with it, the user has the option to store and analyze local entries together with entries in the Connected Database. To make the difference between locally and externally stored data, all the entries and experiment data stored in a Connected Database are underlined. Although the use of Connected Databases and associated local databases is transparent, it is not recommended to store entries and experiments in a mixed way.

NOTE: A number of tables in a Fingerprinting II Connected Database deal with Character types, Sequence types and Matrix types (Fingerprinting II Informatix). These tables are also required by Fingerprinting II, in order to assure compatibility with Fingerprinting II Informatix databases and to allow upgrading from Fingerprinting II to Fingerprinting II Informatix.

Fingerprinting II Informatix local databases can be converted into a Connected Database at any time. This process is irreversible: once a local database has been converted into a connected database, the local database is removed, and Connected Databases cannot be back-converted into local databases.

The combined use of local and Connected Databases is limited to avoid possible conflicts between the two database systems, especially when experiment types have the same name but different settings. Therefore, a few approved possibilities for working with Connected Databases are supported:

- (1) Creating a new database in Fingerprinting II Informatix, linking to a new Connected Database. Fingerprinting II Informatix is allowed to construct the database layout.
- (2) Creating a new Connected Database in Fingerprinting II Informatix, by linking to an existing database that has a table structure already in a Fingerprinting II Informatix compatible format (e.g., linking to an existing Fingerprinting II Informatix Connected Database).
- (3) Creating a new Connected Database, linking to an existing database which is not created using Fingerprinting II Informatix.
- (4) Converting a local database to a new Connected Database.

These possibilities are described in further paragraphs.

Note that Connected Databases are only available with the **Database Sharing Tools** module. The following paragraphs describing the use of Connected Databases require Microsoft Access (97 or later), Microsoft SQL Server or Oracle to be installed.

22.1 Setting up a new Connected Database

Fingerprinting II Informatix can automatically create a new database in Microsoft Access. When you are using SQL Server or Oracle, however, you will have to create a new blank database before proceeding with the following steps.

22.1.1 In the Fingerprinting II Informatix Startup program, click **<New>** to create a new database.

22.1.2 Enter **ConnectedBase** as database name.

22.1.3 In the next step, choose **Yes** to automatically create the required directories, since a local database associated with the Connected Database is required.

22.1.4 In the next step, click **Yes** to enable the creation of log files, and press **<Finish>**.

A new dialog box pops up, prompting for the type of database: *Use the local database*, *Create a new, empty connected database*, or *Connect to an existing connected database* (Figure 141).

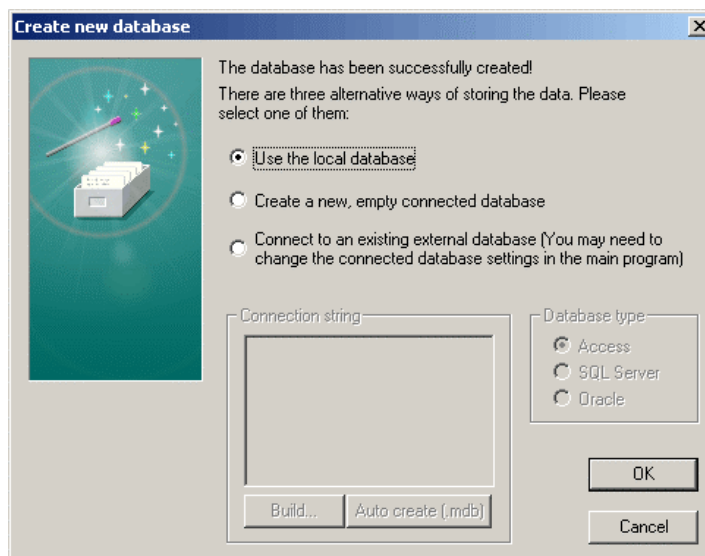


Figure 141. Database selection dialog box.

22.1.5 Select *Create a new, empty connected database* and the database engine of choice.

22.1.6 When the database engine is Access, press **<Auto create (.mdb)>** to create a new Access database.

22.1.7 When the database engine is SQL Server or Oracle, you will need to build a Connection String using the **<Build>** button.

22.1.8 Further to 22.1.7 (non-Access databases) the dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed. Therefore we refer to the Windows manual or help function to select or create a DSN file (ODBC Data Source) that specifies the ODBC driver, and to set up a connection to the database.

22.1.9 Once the database connection is properly configured, you can press **<OK>** to quit the database setup.

22.1.10 Press the **<Analyze>** button.

The Fingerprinting II Informatix main window will open with a blank database.

22.2 Configuring the Connected Database link in Fingerprinting II Informatix

In the Fingerprinting II Informatix main program, you can set up a connection to a Connected Database, or configure an existing connection. In case the program reports database linkage problems when opening the database, you will need to use this configuration to create the required tables in the database.

22.2.1 Select *Database | Connected databases*.

This opens a list of all currently defined Connected Databases for this Fingerprinting II Informatix database (normally just one; see Figure 142).

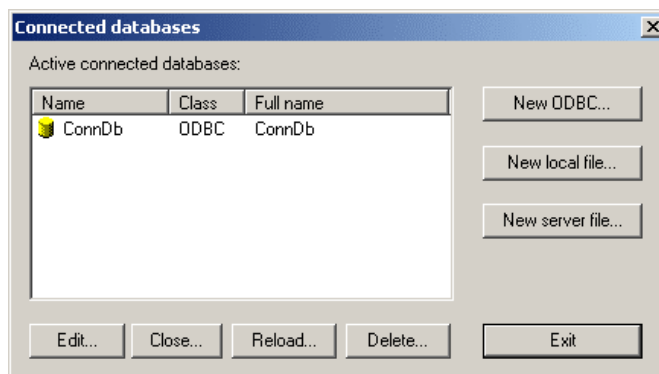


Figure 142. Connected Databases list window.

22.2.2 Select the Connected Database of choice and click **<Edit>**, or double-click on the name.

This results into the *Connected Database configuration* dialog box (Figure 143).

The upper left input field shows the connection description file, which can be found in the local database directory. When Fingerprinting II Informatix has created a new connected database in the Startup program, the file has the default name **ConnDb.xdb**. In the case of the example Connected Database **ConnectedBase**, it occurs as:

c:\Program files\Fingerprinting II Informatix\Data\ConnectedBase\ConnDb.xdb.

This text file can be edited in Notepad or another text editor.

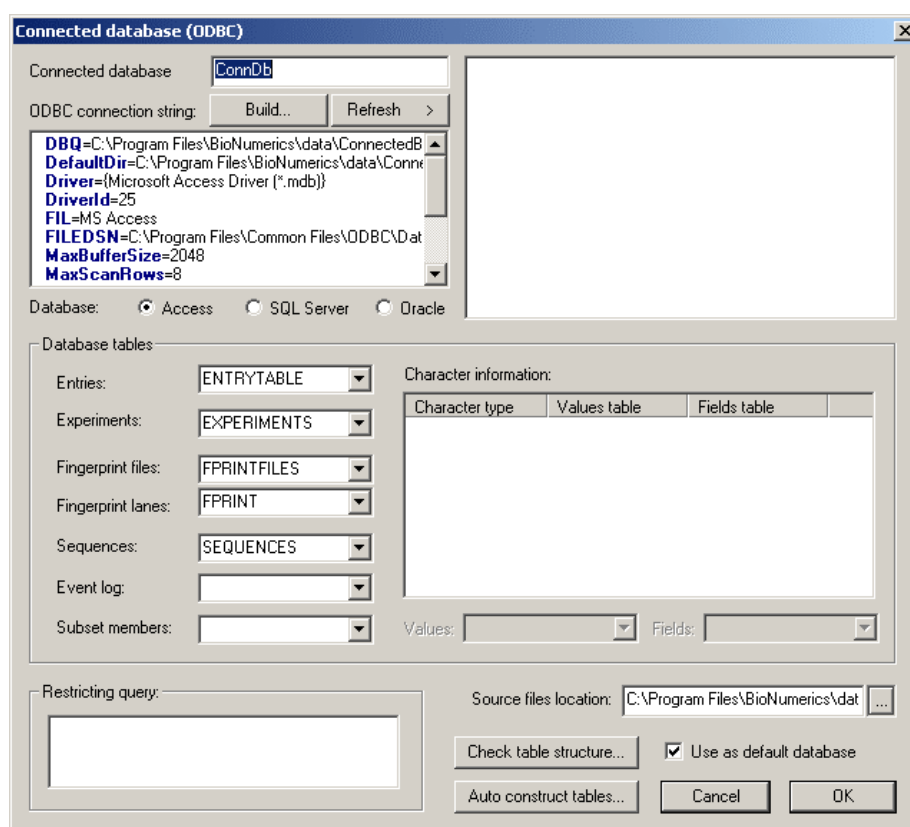


Figure 143. Connected Database configuration dialog box.

Under *ODBC connection string*, the ODBC connection string is defined. The same string can be found in the connection description file, under the tag [CONNECT].

The **<Build>** button allows a new connection string to be defined. This will call the Windows setup dialog box to create a new ODBC connection (see also 22.1.8).

By pressing the **<Refresh>** button, the connection between Fingerprinting II Informatix and the Connected Database is refreshed. A tree-like table structure view of the database is displayed in the upper right panel.

The database type can be selected under *Database* (*Access*, *SQL Server*, and *Oracle*). This information is written under [DATABASETYPE] in the connection description file.

The second panel in the *Connected databases configuration* dialog box concerns the tables of the connected database. Fingerprinting II Informatix assumes a certain table structure to be able to store its different kinds of information. This table structure is described in 25.1. The default table names are **ENTRYTABLE** for the entries, **EXPERIMENTS** for the experiments, **FPRINTFILES** for the fingerprint files, **FPRINT** for the fingerprint lanes, **SEQUENCES** for the sequences, **EVENTLOG** for the event log file, and **SUBSETMEMBERS** for the subset members. Each table should contain a set of columns with fixed names (25.1). In a database setup where Fingerprinting II Informatix is connected to an existing database system, views can be created with table names that correspond to the required Fingerprinting II Informatix tables, and that have the required Fingerprinting II Informatix columns. To add flexibility, however, it is also possible to select different table names than the default ones. This allows one to create additional views, for example, where certain information is shown or hidden. These views can be saved under a different name, and specific views can be made visible to users with specific permissions.

The right panel relates to character types. Each character type is stored in two separate tables. One table, **<CharacterTypeName>FIELDS**, contains the field (character) names. When Connected Databases are used, characters can be described by more than one information field (see page 76). The name field and the additional fields are stored in columns in this table. The second table, **<CharacterTypeName>**, contains the character values for the entries. Both tables can also be chosen. The default names are **<CharacterTypeName>** and **<CharacterTypeName>FIELDS** (**<CharacterTypeName>** being the name of the character type).

Under *Restricting query*, there is a possibility to enter a query that restricts the number of entries in the database. The use of restricting queries is explained further in 22.8.

In *Source file location*, the path for storing the source files (TIFF images and contig projects) is entered. The path can be a local directory or a network path, for example on a server computer. To change the path, click  to browse through the computer or the network.

With the checkbox *Use as default database*, the database can be specified to be the default Connected Database or not. **Once a database is specified to be the default Connected Database, it cannot be disabled anymore!**

Two buttons, **<Check table structure>** and **<Auto construct tables>**, allow one to check if all required tables and fields are present in the Connected Database, and to insert new tables and fields where necessary, respectively.

WARNING: when pressing **<Auto construct tables>**, Fingerprinting II Informatix will automatically create a new table for every required table that is not yet linked to an existing table in the database. For tables already linked, it will insert all required fields that do not exist yet in the database. In case you want to link Fingerprinting II Informatix to an existing database, this may cause a number of undesired tables and fields to be created, and cause irreversible database changes! Solutions to link Fingerprinting II Informatix to existing databases having different table structures are explained in 22.5.

22.3 Working in a Connected Database

Once a Connected Database is correctly set up, adding, processing and analyzing data is nearly identical to working in a local database. Entries stored in the Connected Database are underlined.

NOTES:

(1) When entry information fields are obtained from a view (query) in the Connected Database, it will not be possible to define new information fields directly from Fingerprinting II Informatix. In that case, you will have to create the field in Oracle, SQL Server or Access, add it to the view, and reload the Fingerprinting II Informatix database (see 22.6.3).

(2) Views with joined columns may be read only and it may not be possible to add new records to the database that are seen through these views (e.g. entries, experiments). It is possible to bypass this in Oracle or SQL Server using triggers.

There are a few differences, however, concerning (1) adding new entries to the database, (2) the default directories for images and contig files, and (3) the way log files are recorded and viewed.

- When adding new entries to the database using the menu command **Database|Add new entries**, the choice is offered to add the entries to the local database or to the Connected Database. When no Connected Database is the default database, you will be able to choose between these two possibilities. Once a Connected Database is specified as the default database, however, it will only be possible to add new entries to the Connected Database.
- In a standard Connected Database setup, images, contig project files, and other source files are stored in a common directory **Sourcefiles** under the local database directory. For example, in case of the newly created database **ConnectedBase**, the default directory for such files is **c:\Program files\Fingerprinting II Informatix\Data\ConnectedBase\Sourcefiles**. Within **Sourcefiles**, there are two subdirectories: **Contig** and **Images**. The **Images** subdirectory contains the TIFF files for Fingerprint types, and are placed in this directory using the command **File|Add experiment file** in the Fingerprinting II Informatix main window. To make a gel TIFF file visible in the **Files** panel in a Connected Database, the file should be present in this directory. Under **Contig**, contig (sequence assembly) projects are saved. The source file directory can be modified as described in 22.2. The path can be a network path, for example on a server computer.
- Log files are stored in a different way in a Connected Database. The log events are stored in a database table called EVENTLOG. Different events are stored under different categories: **Database** concerns all actions affecting the database (adding, changing or removing information fields, adding experiment types, adding entries, changing entry information etc.). Furthermore, there is a category **EXPER_<ExperimentName>** (<ExperimentName> being the name of the experiment), relating to changes made to the experiment type (i.e. normalization settings in case of Fingerprint type, adding, removing, or renaming characters in Character type, etc.). A third category reports on changes made to the data in a certain experiment type. In this category, components have the name of the experiment type.

The *Event log* window (Figure 9) called from the main program offers the possibility to view the log file for a Connected Database or the local database under **Database**. Under **Component**, you can choose to view a specific component, e.g. Database, an experiment type, or data belonging to an experiment type. With **All**, you can view all components together, listed chronologically. The components can only be selected when a Connected Database is selected.

22.4 Linking to an existing database with standard Fingerprinting II Informatix table structure

Any computer running Fingerprinting II Informatix can link up to an existing Fingerprinting II Informatix **Connected Database** at any time. When this Connected Database has its table structure in the standard Fingerprinting II Informatix format (see 25.1), this can be done very easily in the Startup program.

22.4.1 In the Fingerprinting II Informatix Startup program, click **<New>** to create a new database.

22.4.2 Enter a name for the **Connected Database** (this can be a different name on different computers).

22.4.3 In the next step, choose **Yes** to automatically create the required directories, since a local database associated with the Connected Database is required.

22.4.4 In the next step, choose to whether or not create log files, and press **<Finish>**.

A new dialog box pops up, prompting for the type of database: *Local database*, *New Connected database*, or *Connect to an existing connected database* (Figure 141).

22.4.5 Select *Connect to an existing connected database* and press **<Build>** to establish the connection to the database.

22.4.6 The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed. Therefore we refer to the Windows manual or help function to select or create a DSN file (ODBC Data Source) that specifies the ODBC driver, and to set up a connection to the database.

22.4.7 Once the database connection is defined, you can press **<OK>** to quit the database setup.

22.4.8 Press the **<Analyze>** button.

The Fingerprinting II Informatix main window will open. If the Connected Database contains the standard table structure for Fingerprinting II Informatix (see 25.1), no error message is produced and you can start working immediately. Fingerprinting II Informatix will automatically recognize the existing information fields, experiment types, subsets, entries and data. If the table structure is not in standard Fingerprinting II Informatix format however, a dialog box appears, warning for several errors that have occurred while trying to open specific tables in the connected database that weren't found. See 22.5.10 and further to assign the correct tables or views from the database.

22.5 Linking to an existing database with table structure not in Fingerprinting II Informatix format

This paragraph describes the situation where an Oracle, SQL Server or Access database, containing descriptive information on organisms (entries) and/or experiment data is already present and Fingerprinting II Informatix should be hooked up to that database in order to read and write experiment data and information fields.

Before proceeding with the configuration of the database connection, it will be necessary to make the database compatible with the Fingerprinting II Informatix table structure. In a typical case, a number of information fields and/or experiment fields from the Connected Database will need to be linked to

Fingerprinting II Informatix. However, these fields will occur in different tables having different field names. The obvious method in this case is to create views (or, in Access, queries) in the database.

- For those Fingerprinting II Informatix tables for which the Connected Database contains fields to be used, a view (query) should be constructed in the database. Within that view (query), those database fields that contain information to be used by Fingerprinting II Informatix should be linked to the appropriate field.
- Fingerprinting II Informatix tables for which the Connected Database contains no fields can be created automatically by Fingerprinting II Informatix.
- Finally, the database should be configured such that the Fingerprinting II Informatix tables that contain fields already present in the database, be present either as table or as view, with all the recognized field names as outlined in 25.1. The names for the tables or views, however, can be freely chosen.
- Additional tables required by Fingerprinting II Informatix that contain no fields already present in the database can be created automatically by Fingerprinting II Informatix.

NOTE: When view are created in the database, to match the required Fingerprinting II Informatix tables, it is recommended to name the views using the standard Fingerprinting II Informatix names for the required tables. This will allow new users to log on to an existing Connected Database in the easiest way, by just defining the connection in the startup program (22.4). By using different names, new users will have to specify the table/view names manually in the Connected database configuration window (Figure 143) after defining the Connected Database. Using different names for the views is only useful if it is the intention to assign different permissions to different users; in this way, views can be created showing only restricted information, while other views show full information, etc.

22.5.1 In the Fingerprinting II Informatix Startup program, click **<New>** to create a new database.

22.5.2 Enter a name for the new database.

22.5.3 In the next step, choose **Yes** to automatically create the required directories, since a local database associated with the Connected Database is required.

22.5.4 In the next step, choose to whether or not create log files, and press **<Finish>**.

A new dialog box pops up, prompting for the type of database: **Local database**, **New Connected database**, or **Connect to an existing connected database** (Figure 141).

22.5.5 Select **Connect to an existing connected database** and press **<Build>** to establish the connection to the database.

22.5.6 The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed. Therefore we refer to the Windows manual or help function to select or create a DSN file (ODBC Data Source) that specifies the ODBC driver, and to set up a connection to the database.

22.5.7 Once the database connection is specified, you can press **<OK>** to quit the database setup.

22.5.8 Press the **<Analyze>** button.

The Fingerprinting II Informatix main window will open. Since the Connected Database does not contain the standard table structure for Fingerprinting II Informatix (see 25.1), a dialog box now

appears, warning for several errors that have occurred while trying to open specific tables in the connected database that weren't found.

22.5.9 Press <OK> to close the message(s). The Fingerprinting II Informatix main program now opens with a blank database.

In the Fingerprinting II Informatix main program, you can now configure the database connection as described in 22.2:

22.5.10 Select **Database | Connected databases**.

This opens a list of all currently defined Connected Databases for this Fingerprinting II Informatix database (normally just one; see Figure 142).

22.5.11 Select the Connected Database of choice and click <Edit>, or double-click on the name.

This results into the *Connected Database configuration* dialog box (Figure 143). This window shows the default suggested table names for the required database components under **Database tables** (see 22.2). Some, or all, of these tables do not correspond to the tables of the database.

22.5.12 Press the <Refresh> button. The upper right panel now lists the tables and views in the Connected Database, as it exists.

22.5.13 You can expand each table/view to display its fields by clicking on the "+" sign on the tree.

22.5.14 Under **Database tables**, select the corresponding table or view for each component.

22.5.15 When this is finished, check the correspondence by pressing <Check table structure>.

When required, you can further configure the database, leaving the *Connected Database configuration* window open. As soon as the new configuration is done, press <Refresh> and check the table structure again.

22.5.16 Finally, when all links to existing database tables/views are made correctly, you can allow Fingerprinting II Informatix to create additional tables for which there are no fields available in the external database, by pressing <Auto construct tables>. Fingerprinting II Informatix will now only construct tables that are not yet linked, and fields that are not yet present in the connected tables.

NOTE: it will not be possible for Fingerprinting II Informatix to create new fields within a view/query. In that case, you will have to create the field in Oracle, SQL Server or Access, add it to the view, and reload the Fingerprinting II Informatix database.

22.6 Converting a local database to a Connected Database

Fingerprinting II Informatix offers the possibility to convert an entire local database at once to a new Connected Database. This is **an irreversible operation, which causes the local database to be removed once the conversion is done**. It is therefore strongly recommended to make a backup copy of the local database before carrying out a conversion to a Connected Database.

NOTE: It is not recommended to convert a local database into a Connected Database that already contains data, using this tool, since experiment types with the same name would be overwritten.

There are other tools to convert entries and data from a local database into a Connected Database, using scripts.

To convert a local database into a new Connected Database, proceed as follows:

22.6.1 Create a new empty database in Oracle, SQL Server or Access.

22.6.2 Open the local database in the Fingerprinting II Informatix main program.

22.6.3 In the main program, select **Database | Connected databases**.

This opens a list of all currently defined Connected Databases for this Fingerprinting II Informatix database (normally empty at this stage; see Figure 142).

22.6.4 Click **<New ODBC>**.

In the *Connected databases configuration* dialog box that appears, click **<Build>**.

22.6.5 The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed. Therefore we refer to the Windows manual or help function to select or create a DSN file (ODBC Data Source) that specifies the ODBC driver, and to set up a connection to the database.

22.6.6 Make sure the Connected Database is checked as the default database; otherwise, the conversion cannot be executed.

22.6.7 Check the table structure of the database, if it does not contain the required tables and fields, press **<Auto construct tables>** to allow Fingerprinting II Informatix to construct its tables.

22.6.8 Once the connection is defined correctly, press **<OK>** to close the *Connected databases configuration* dialog box.

22.6.9 Close the Connected Databases list window.

22.6.10 In the main program, select **Database | Convert local data to connected database**.

An important warning message is displayed. If you are converting the local database to a NEW Connected Database, and if you have made a backup of the data before starting this conversion, you can safely click **<OK>** to start the conversion.

Depending on the size of the database, the conversion can take seconds to hours. Fingerprint image files take most time to convert. When the conversion is finished successfully, Fingerprinting II Informatix will automatically restart with the Connected Database, and the contents of the local database will be removed.

22.7 Opening and closing database connections

- Connecting to multiple Connected Databases

It is possible to connect to other Connected Databases in addition to the default Connected Database.

22.7.1 In the main program, select **Database | Connected databases**.

This opens a list of all currently defined Connected Databases for this Fingerprinting II Informatix database (normally just one; see Figure 142).

22.7.2 Click **<New ODBC>**.

In the *Connected databases configuration* dialog box that appears, click **<Build>**.

22.7.3 The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed. Therefore we refer to the Windows manual or help function to select or create a DSN file (ODBC Data Source) that specifies the ODBC driver, and to set up a connection to the database.

22.7.4 In the *Connected databases configuration* dialog box, enter a name for the connected database definition file (upper left input field, **Connected database**). This name should be different from the names of any of the existing Connected Databases.

22.7.5 Under **Source files location**, select the directory where the source files can be found for this Connected Database. This directory should in any case be different from the *Source files* directory of the default Connected Database.

22.7.6 Once the connection is defined correctly, press **<OK>** to close the *Connected databases configuration* dialog box.

NOTE: When the two connected databases have the same Source files directory associated, an error message is produced at this time: "Another connected database is already associated with this source files directory." It will not be possible to save this new connection until the source files directories are different.

The new Connected Database is listed in the list window. When you open the main program, the contents of the two databases are seen together.

- Closing or deleting a Connected Database

22.7.7 In the Connected Databases list window (**Database|Connected databases**), select the Connected Database you want to close, and press **<Close>**.

22.7.8 Confirm with **<Yes>**. The database disappears from the list, and the contents of the closed database disappears from the main window.

Closing a Connected Database is temporarily. When it is closed, it will automatically be reopened the next time the Fingerprinting II Informatix analyze program is started up with the same database.

To delete a connection to a database, press **<Delete>** in the Connected Databases list window. The Connected Database will never reappear until you build the connection again.

- Reloading a Connected Database

Suppose you have modified the Connected Database directly in Oracle, SQL Server or Access, you can use the function **<Reload>** in the Connected Databases list window. Any columns that were added, for example as information fields, or any entries or data that was added externally after Fingerprinting II Informatix was started up will be updated in the Fingerprinting II Informatix main window.

Reloading a Connected Database can also be useful in case several persons are working in the database simultaneously. Any entries added by other persons will not be seen in your session until you reload the database.

22.8 Restricting queries

In the *Connected databases configuration* dialog box (Figure 143), it is possible to enter a *restricting query* in the input field **Restricting query**. A restricting query is of the general format **FieldName=String**. **FieldName** is the name of the the restriction is applied to, and **String** is the restricting string. As a result, when the Fingerprinting II Informatix main program is opened with the Connected database, only those entries having **String** filled in the field **FieldName** will be seen in the database.

In addition, when new entries are added to the database, they will automatically have their field **FieldName** filled with **String**.

As an example, suppose that the database DemoBase has been converted to a Connected Database, you can enter a restricting query to visualize only Ambiorix, as follows.

22.8.1 In the *Connected databases configuration* dialog box under Restricting query, type “**genus=Ambiorix**”.

22.8.2 Press <OK> to confirm the changes. The main window now only shows *Ambiorix*.

22.8.3 Add a new entry with *Database|Add new entries*. The new entry is automatically called *Ambiorix* in its **Genus** field.

Restricting queries can be combined by separating them with semicolons. For example, if you want to visualize only *Ambiorix sylvestris* entries, enter “**genus=Ambiorix;species=sylvestris**” as a restricting query string.

The result is a database that only shows *Ambiorix sylvestris*. New entries will automatically be added as *Ambiorix sylvestris*.

NOTE: Never use spaces in a restricting query.

23. Preserving the Fingerprinting II Informatix database integrity

In many cases, Fingerprinting II Informatix will be used to construct large databases of information that has been collected over a long time span. Obviously, the user should pay attention to protect such databases from accidental data losses, e.g. due to hard disk crashes, power interruptions, etc. Although a practical strategy depends to a large extent on the computer hardware configuration, this chapter introduces some basic considerations to protect the data.

23.1 Taking backups of a database

All the data files that belong to a particular Fingerprinting II Informatix database are stored on the hard disk in subdirectories of a single top directory (see also 1.3). If Fingerprinting II Informatix is opened with this database, this directory is indicated in the status bar on the bottom of the main window. Alternatively, one can open the Fingerprinting II Informatix intro screen, select a database, and press the **<Settings>** button. A dialog box appears which shows the top directory of the selected database.

Since all important information concerning a database is stored inside this top directory, one only needs to back up this complete directory (including subdirectories) to have a complete copy of all data. When the database needs to be restored later on, this top directory can be copied back to the right place on the hard disk. Note that backups restored from CD-ROM may be read-only. In this case you will have to specify the files to be write-accessible before you run Fingerprinting II Informatix with the restored database.

It is possible to create a duplicate of a Fingerprinting II Informatix database in a similar way. To this end, copy the entire contents of the database's top directory to a new directory. In the Fingerprinting II Informatix intro screen, select **<New>** to create a new database. When the database creation wizard pops up, fill in a name of the duplicate database and click **<Next>**. In the next tab, click **<Change>** to change the database top directory into the name of the duplicate directory. In addition, specify **<No>** to the question "*Do you want to automatically create the required directories?*" Proceed in the usual way to finish the creation of the database.

23.2 Detecting and correcting faults in a database

During the use of a Fingerprinting II Informatix database, there may occur some faults in the structure, which need to be repaired. For instance, if a power failure or a system crash occurs while Fingerprinting II Informatix is saving a gel file, this file may become corrupted and unreadable. Moreover, due to the relational structure of the database, there may occur some more subtle inconsistencies in the database. For example, two different database entries may end up having the same key. Such a conflict may arise if someone has copied new data files from one database into another without taking care.

Fingerprinting II Informatix comes with a special diagnostics/repair program that is able to detect and solve such problems in a *local* database. Connected Databases (chapter 22) cannot be inspected using

this tool. To start this program, click on the *<Inspect>* button in the intro screen. The program's main window is divided in three parts (Figure 144):

1. Top panel: a list of all the problems the program has detected in the current active database. If the program can solve the problem automatically, a *<Solve>* button appears right from it. The user can click on this button to see a menu of possible solutions. If a solution is clicked, it is applied on the database.
2. Bottom left panel: a list of all experiments present.
3. Bottom right panel: a list of all database keys having a conflict. For each key, this window indicates the number of times it occurs as a database entry list and as each of the experiment types. If duplicates exist, they are marked in red. Double clicking on a key in this list opens a detailed entry window, specifying the place of occurrence of this key in the database and for every experiment. For each occurrence, the key can be changed manually using *Edit|Change key*. In this way, the user can manually intervene, having complete control over the way key conflicts are resolved.

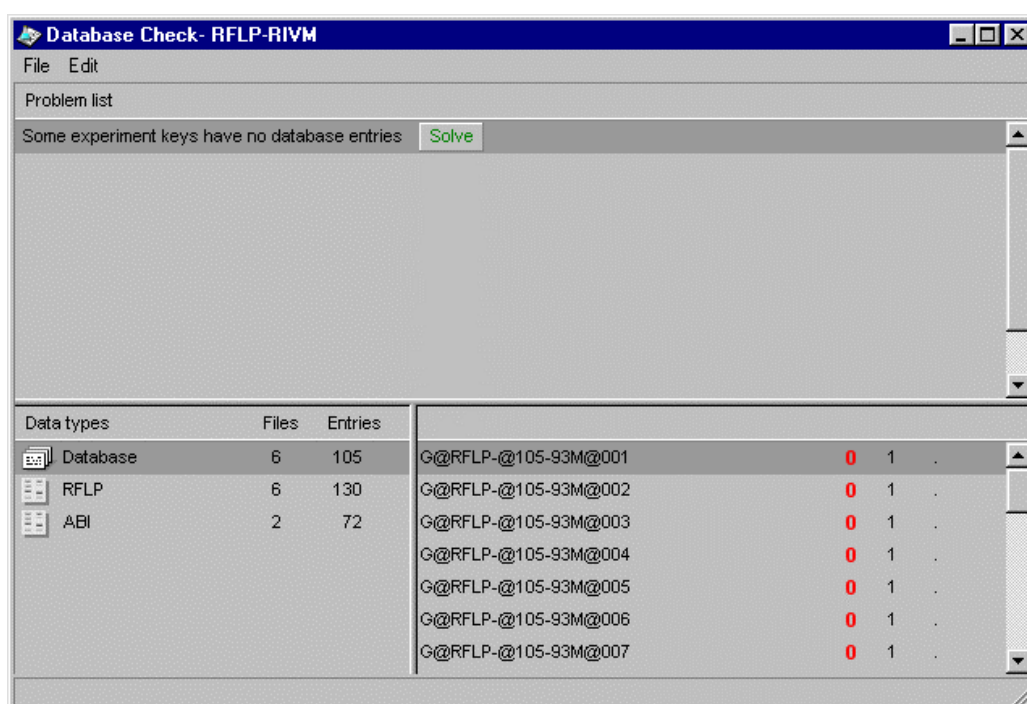


Figure 144. The Fingerprinting II Informatix diagnose program.

Problems can be solved by either selecting one of the standard solutions proposed by the software, or changing the database structure manually. If the solution involves changing keys in the database, the user needs to save the changes explicitly by using the menu item *File|Change changes to disk* afterwards.

The remainder of this chapter contains a list of problems that can be detected, together with a discussion of possible solutions.

23.3 Missing directories

If a directory is missing in the database directory structure, the program reports it. The proposed solution is to create the missing directory.

23.4 Corrupted files

The diagnose program checks the following files for errors:

- Fingerprint, character, sequence & matrix types configuration files (.CNF)
- Fingerprint, character, sequence & matrix data files (.DEF)
- Database information files (.DBS)
- Comparison files (.CMP)
- Identification libraries and unit files (.LIB & .CMP)

If a corrupted file is detected, the only automatic solution provided by the software is to delete it.

23.5 Empty files

It may happen that, if a number of database entries have been removed, an empty file exists on the hard disk (e.g. a database file or a fingerprint, character or sequence data file). Although this situation does not harm the usage of the database, the software detects this and proposes to remove the empty file(s).

23.6 Database entries with identical keys

In certain circumstances, a database might end up with two database entries having the same key. For example, this might occur if one copies database files from one database to another, without taking care of the key names. In such a situation, the software is unable to distinguish between both entries. Two automatic solutions are proposed:

- Remove the duplicate keys (i.e. replace with empty strings).
- Automatically rename the duplicate keys in order to force them to be unique.

23.7 Database entries without keys

If a database entry ends up having no key (i.e. an empty string), the software is not able anymore to link any experiments to it or to use it in an analysis. The proposed automatic solution is to create a new, unique key for such a database entry.

23.8 Experiments with identical keys

This problem is similar to the previous one. It occurs when two different “experiments” of the same experiment type, (e.g. fingerprint lanes of gels belonging to the same type), have exactly the same key. If this happens, there is a conflict because both will be linked to the same database entry. Again, two automatic solutions are possible:

1. Remove the duplicate keys. In this case, the experiments that have duplicate keys will become unlinked, i.e. not associated with any database entry. Afterwards, the user can re-link them to the appropriate entries.

2. Automatically rename the duplicate keys into unique strings. Note that, if this option is chosen, a new conflict will arise because this will create experiments with keys that do not correspond to any database entry. However, this can be solved by automatically creating new database entries for these experiments (see next paragraph).

23.9 Experiment keys without database entries

It might happen that an experiment (e.g. a fingerprint lane of a gel) has a “phantom link”, i.e. it has a key that does not correspond to any database entry. This situation may occur if one removes a database entry without taking care of the experiments that are linked to it. The software can perform two automatic actions:

1. Remove the key from that experiment. This means that, at this moment, the experiment becomes unlinked (not linked to any database entry). Afterwards, a new link can be created in Fingerprinting II Informatix.
2. Add a new database entry for this key. This solution creates a new database entry with a key that corresponds to the key of the experiment.

24. Reading data files using macros

The *Macro* language is available in GelCompar/BioNumerics since version 1.10. It is now replaced by the much more powerful *Script* language discussed in the Script Manual. For reasons of compatibility only, the macro language is still available in Fingerprinting II Informatix, and it is described in this chapter. It is recommended for new users of Fingerprinting II Informatix to learn the general script language rather than the macro language, which is very limited in possibilities, only intended for the import of text files, and not supported anymore.

24.1 Importing experiment data from text files

Fingerprinting II Informatix is equipped with import routines that allow you to read and parse the information from virtually every text file format, and import it into the database.

The heart of these import routines consists in a generic macro language. For each file format that needs to be imported, the user can create a custom-tailored macro that scans the file, extracts the appropriate information, and adds it to the database. A limitation to the import functions using macros is that files should not contain more than 300 entries. If larger files are to be imported, the files should first be saved as partial files containing no more than 300 entries each.

Example

Suppose one wants to import a tab-delineated file that consists of two information fields and a number of character values:

Caesar	Julius	1	0	5	7	1	0	3	...
Livius	Titius	6	0	0	4	2	4	1	...
Aurelius	Marcus	6	8	9	5	1	2	4	...

The two information fields should correspond to the database fields “Genus” and “Species”, whereas the following 20 character values should be imported as the 20 values of the character experiment type “Phenotest”. This example character text file is available on the installation CD-ROM, under `Examples\TXTfiles\Expmacro\Example0.txt`.

24.1.1 Select the experiment type “Phenotest” in the main window’s experiment panel.

24.1.2 Call *File | Import experiment data*.

24.1.3 Select file `Examples\TXTfiles\Expmacro\Example0.txt` on the CD-ROM in the file dialog box.

24.1.4 In the “Import macro” list box, select *<Create new data import macro>* in order to create a new macro.

24.1.5 Press *<Open>*.

The macro editor window is created. This window contains 4 panels (see Figure 145):

- Upper left: source code panel
- Upper right: data import panel
- Lower left: import files panel
- Lower right: debug panel

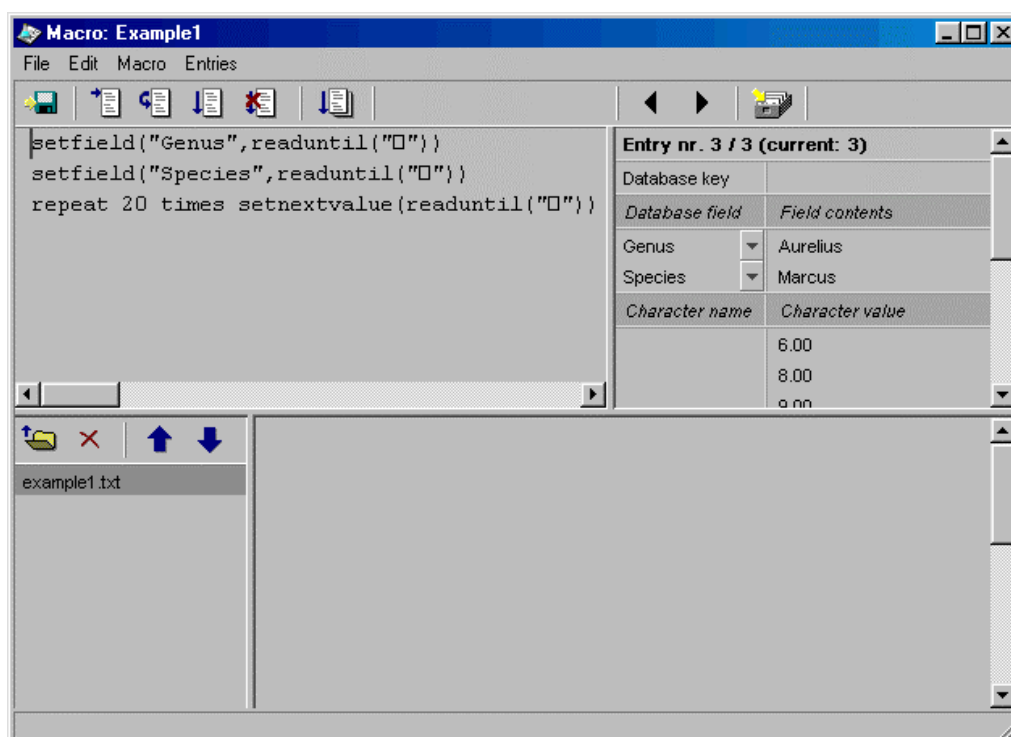


Figure 145. The Fingerprinting II Informatix import macro editor.

Enter the following macro source text in the source code panel:

```
setfield("Genus",readuntil("»"))
setfield("Species",readuntil("»"))
repeat 7 times setnextvalue(readuntil("»"))
```

where the symbol » stands for a tab and is shown in the editor as □. It is entered from the keyboard by pressing the TAB key.

24.1.6 Save the macro as **Example0** using *File | Save macro*.

24.2 Running the macro automatically

24.2.1 Select *Macro | Run* or  (F5 key) to let the macro automatically process the entire import file.

Upon completion, the data import panel shows the result of processing the file. You can scroll through the different entries using the  and  buttons arrow buttons. For each entry, the corresponding database fields and character values, as read from the file, are shown.

24.2.2 Select *Entries | Import* or  to add the imported entries to the database.

This option creates a dialog box that prompts for the filename where the imported entries will be stored in.

24.3 Running the macro step by step (debugging)

You can run the macro step by step, and inspect how the data file is processed (debug mode).

24.3.1 Enter the debug mode by selecting *Macro | compile & start* or .

The debug panel then shows three lines of the import file: the current line, the previous line and the next line. The unprocessed text is marked with a light blue background. If the macro uses variables, their content is also listed in the debug window.

24.3.2 Process next statement in the macro: *Macro | Execute one step* or  (F11 key)

24.3.3 Continue with automatic execution with *Macro | Run*.

24.3.4 Stop the execution with *Macro | Stop* or .

At any time, the line containing last executed statement in the macro is marked with a pink background.

24.4 Selection of data import files

Multiple files can be selected in the files import dialog box. In addition, data files can be added using *File | Include additional files*.

The macro is automatically started and executed on all files in the list using *Macro | Run all files* or . The files are processed in the order they appear in the list. They can be moved up or down using the arrow buttons or *File | Move file up* () and *File | Move file down* ()

24.5 Changing the names of the database fields

Before the data is actually imported, it is possible to change the names of the database fields, created by the macro, in order to match the database field names of the existing Fingerprinting II Informatix database. To this end, click on the arrow down button, which appears right from each database field in the data import panel. A menu appears, offering a number of options:

- *Do not use this database field* (it will not be imported into Fingerprinting II Informatix);
- *Use as key*; i.e. use this database fields as the database key;
- *<Field name>*: Rename this database field to one of the existing unused database fields in the Fingerprinting II Informatix database.

24.6 The macro language

If, while running the macro, the last statement of the macro has been executed while the end of the import data file has not yet been reached, the macro is restarted from the beginning. In this way, the macro is each time re-executed till the end of the data file is reached.

Variables

If necessary, a data import macro can use variables to hold and process data. There are three possible data types:

String: holds a character string of variable length (with a maximum of 8000 characters)

Integer: holds an integral value (e.g. 0 , 1 , -5 , 1256 , ...)

Float: holds a floating point value (e.g. 1.12554 , 5.25 , 7 , -56.99 , ...)

all definitions of variables should occur before the first executable statement in the macro.

Some examples:

string str1	defines a character string called str1;
integer i,j,k	defines three integral values: i, j and k;
float r,alpha	defines two floating point numbers: r and alpha.

In addition, one can also use **arrays of variables** in a macro. It is not necessary to specify the size of the arrays in advance, since they are resized automatically. Some examples:

A one-dimensional array of strings:

```
string str[]
```

Inside the macro, the elements can be addressed, e.g. the 10th element of this array can be accessed using **str[10]**.

A two-dimensional table of integers:

```
Integer table[][]
```

The element (5,15) of this table can be accessed using the expression **table[5][15]**.

Mathematical expressions

Inside the macro, any mathematical expression can be used that executes additions (+), subtractions (-), multiplications (*), divisions (/) and powers (^). In addition, brackets can be used to change the priority of execution. Inside the expression, integer and floating variables can be used. The result can be assigned to an integer or floating variable (=), or used as an argument for a function.

Examples:

```
i=j+1
j=10*i-5*(j+k)-10
r=(alpha-5)/2.77+3.1415927*(1+r^2)-7/(alpha^0.5)
```

Logical expressions

A logical expression either returns 1 (true) or 0 (false). The following comparisons can be used to create logical statements:

= (equal), < (smaller), > (bigger), <= (smaller than or equal), >= (bigger than or equal), <> (different).

These comparisons can be performed on integers, floatings and strings.

In addition, logical operators can be used to manipulate logical values:

not (*a*) : inverts the logical value of *a*;
a **and** *b*: returns true only if *a* and *b* are true;
a **or** *b*: returns true if either *a* or *b* are true.

Examples:

i >5	true if <i>i</i> is bigger than 5;
i >=5 and i <=10	true if <i>i</i> is between 5 and 10;
str <"example"	true if the string str comes lexicographically before "example";
not (i >0)	true if <i>i</i> is smaller than or equal to 0.

String manipulations

The following string manipulation instructions are available:

<i>a</i> + <i>b</i> :	concatenation of two strings;
length (<i>a</i>) :	returns the length of the string (=number of characters);
substring (<i>a</i> , <i>i</i> , <i>j</i>) :	returns a substring of <i>a</i> that contains the range of characters from position <i>i</i> to <i>j</i> ;
find (<i>a</i> , <i>b</i>) :	returns the position of the first occurrence of the string <i>b</i> inside the string <i>a</i> (zero if absent);
val (<i>a</i>) :	converts the string into a number or a floating point;
str (<i>r</i>) :	converts an integral number or a floating point into a string;
filename :	returns a string containing the name of the file currently being processed.

Examples:

"test"+"string"	returns "teststring"
length("string")	returns 6
substring("teststring",3,5)	returns "sts"
find("teststring","str")	returns 5
val("1.23")	returns value 1.23
str(7)	returns string "7"

File reading instructions

There are two instructions that read text from the data file. Each of them returns the result as a string.

read (<i>x</i>) :	reads a fixed number of characters from the current line in the file. The number of characters is specified between brackets.
----------------------------	---

readuntil("c") : reads characters from the current line in the file until a given character is reached. This character is specified between brackets.

readwhile("str") : reads characters from the current line while they are contained in a given set of characters. These characters are specified between brackets.

Examples:

str=read(10) reads 10 characters in copies the result in str;
str=readuntil(",") reads until a comma is found and copies the result in str;
readwhile(" ") removes spaces from the line until another character is encountered.

NOTE: you may also specify more than one character using readuntil. The function then stops reading as soon as any of these characters is found. If you specify an empty string, the complete line is read.

A second set of functions allow one to reveal and change the current position in the file.

givecolumn: returns the column of the current cursor position in the file;
setcolumn(x): changes the current cursor position in the file and places it at a given column. The column number is given between brackets.
endoffline: returns TRUE if the end of the current line is reached;
finished: returns TRUE if the end of the current file is reached;
nextline: skips the remainder of the current line and moves to the next line;

Example:

Setcolumn(givecolumn+5) skips 5 characters in the current line.

Controlling the macro flow

repeat n times statement repeats *n* times *statement*;
if condition then statement executes *statement* only if *condition* is true;
while condition do statement Repeats the execution of *statement* as long as *condition* is true.

condition should be a logical expression that can be true or false (see **Logical expressions**).

Examples:

repeat 10 times nextline skips the 10 following lines in the data file;
if i<0 then i=0 puts *i* equal to zero if it has a negative value;
while not(finished) do nextline skips lines in the data files till the end of the file is reached.

It is possible to group multiple statements after one of these expressions using curly brackets { }. In this case, the whole block will be considered as one single statement. This block of statements may be separated over several lines.

Example:

repeat 3 times {
 str1=read(10)

```

    str2=read(10)
}

```

This macro reads 3 times two strings of 10 characters

In addition, there are three functions that interrupt the macro execution:

stop: stops the macro in a normal way;
error(*str*) : stops the macro and shows an error message, given by *str*, on the screen;
break: interrupts the program and starts the debug mode at the current statement (only for debugging purposes).

Example:

```
Error("This data file has an invalid format")
```

Filling in the information for the imported database entries

During one single import action, up to 300 database entries can be imported from a data file. When the macro is running, only one database entry is accessible at a time: the so-called current entry. When the execution starts, a first entry is created and the current entry is set to this one. Each time the macro restarts during its execution (because the end of the macro is reached while there are still some unprocessed lines in the data file), a new entry is automatically created, and the current entry is assigned to this new one. In addition, two functions help to manually reposition the current entry:

curentry: returns the number of the current entry (ranging from 1 to 300);
setcurentry(*nr*) : sets the current entry to the number *nr* (and creates it if it does not exist already).

Example:

```
setcurentry(curentry+1)    sets the next entry as current entry.
```

Reading the database information fields

Two functions are used to fill in the database information fields of the current entry.

setfield(*name*,*content*) : fills the database field specified by the string *name* with the string *content*;
setkey(*key*) : fills the database key with the string *key*.

Examples:

```
setfield("genus",readuntil(", "))    reads a string till the next comma and assigns it to
                                     the database field "genus" of the current entry;
setkey("key1")                       set the database key of the current entry equal to
                                     "key1".
```

Reading the experiment data for fingerprint experiment types

If, in the main window, *File|Import experiment data* is called for a fingerprint type, the software offers two choices:

1. Import of densitometric curves: the import file contains curves, specified as a number of densitometric values;
2. Import of tables of peaks: the import file contains peak positions and peak amplitudes.

- Densitometric curves:

If this option is chosen, the macro should read for each entry a number of densitometric values from the data file. This can be done using the command

setnextvalue (*value*) : takes *value* as the value of the next densitometric point.

Example:

```
while not(finished) do
  setnextvalue(readuntil(""))    reads a densitometric value from each line of the file.
```

- Tables of peaks:

If this option is chosen, the macro should read peak positions and amplitudes. The peak positions are interpreted as fingerprint metrics (e.g. fragment lengths, molecular sizes, etc.), as specified by the metric of the active reference system, see 7.3.36). If necessary, one can create a new reference system in order to be able to read these data (see 7.6).

The following command is used:

setvalue (*metrics*, *amplit*) : adds a new peak with metrics value *metrics* and amplitude *amplit*

Example:

```
while not(finished) do
  setvalue(readuntil(", "), readuntil(""))    reads a peak from each line of the file;
                                              the position and amplitude are
                                              separated by a comma
```

If tables of peaks are imported into a Fingerprinting II Informatix database, the software automatically creates a synthetic densitometric curve, using Gaussian curves for the peak shapes. When the data are imported using *Entries | Import*, a dialog box prompts for:

- The resolution (in points) of the corresponding densitometric curves;
- The width (in % of the total length) of the Gaussian curves used to model the peaks;
- The maximum amplitude of the peaks.

Reading the experiment data for character types

Character data can be read in two ways:

- Unnamed characters

The name of the character is not specified. All characters are simply read in a sequential way, starting with the first character and ending with the last one. This approach is only applicable to closed

character sets (see 7.6). All characters must be read, and in the same ordering as they occur in the experiment type.

setnextvalue (*value*) : takes *value* as the value of the next character.

Example:

repeat 20 times setnextvalue(readuntil("□")) reads 20 character values, separated by a tab.

- Named characters

For each character that is read, its name is specified. This approach must be followed for open character sets (see 7.8). The sequential ordering in which the characters are read is not necessarily the same as the ordering in the experiment type. Some characters may be left empty.

setvalue (*name*, *value*) : assigns *value* to the character with the name specified by the string *name*.

Example:

setvalue("penicillin", readuntil("")) reads value of the character "penicillin" from the current line.

Reading the experiment data for sequence files

A sequence can be read using the command

addsequence (*seq*) : adds the sequence contained in the string *seq* to the current sequence.

Example:

**while not(finished) do
addsequence(readuntil(""))** treats all characters in the complete file as sequence data.

Importing the data into the Fingerprinting II Informatix database

For each entry that is imported into a Fingerprinting II Informatix database, there are 5 different possible situations:

1. No key is specified and no database field is specified
⇒ No database entry is created. The experiment data is imported, but is not linked to any database entry.
2. No key is specified and one or more database fields are specified
⇒ A new database entry is created, using an automatically created key. The database fields are filled in and the experiment data is linked to this new entry.
3. A key is specified which is not yet used in the database
⇒ A new database entry is created, using the specified key. The database fields are filled in and the experiment data is linked to this new entry.
4. A key is specified which is already used in the database, but the database entry with that key doesn't have the experiment yet

- ⇒ The experiment data is linked to this existing database entry. The existing database fields are unaltered.
5. A key is specified which is already used in the database, and the database entry with that key already has the experiment
- ⇒ A new database entry is created, using the specified key with the extension /#1. The database fields are filled in and the experiment data is linked to this new entry. If /#1 already existed, a new entry will be created with the extension /#2, etc.

NOTE: The solution in case 5. is very useful in case you want to import experiments that are run in duplicate for one or more organisms. Rather than overwriting the first entry or disregarding duplicate entries, Fingerprinting II Informatix automatically considers them as duplicates and assigns an extension /#x to such duplicates. In case for a given entry a duplicate already exists (after import of another experiment), Fingerprinting II Informatix will automatically fill such existing duplicates that are still empty for the experiment type that is being imported. Database fields are automatically taken over from the "master" entry, i.e. the entry without extension. If the database fields from the "master" entry are changed, the /#x duplicates are automatically changed accordingly.

Some examples of file formats and macros

Example 1: Sample file from a microplate reader

File CD-ROM:\Examples\TXTfiles\Expmacro\Example1.txt

Tuesday 16-11-1993
NUMBER OF MEASUREMENTS: 96
ID: 12542

	1	2	3	4	5	6	7	8	9	10	11	12
A	0.059	0.098	2.275	1.846	1.221	0.820	0.039	2.080	0.068	0.156	0.000	1.943
B	0.168	2.361	0.256	2.680	0.005	1.597	0.098	0.186	0.127	2.480	2.002	2.119
C	0.078	1.914	1.680	0.195	0.177	1.924	1.924	2.031	0.293	0.166	0.830	1.709
D	1.357	1.113	1.064	0.410	0.088	0.156	2.344	0.137	0.059	0.449	0.098	0.059
E	0.078	0.117	0.313	0.117	0.039	1.592	0.127	1.982	0.098	0.078	0.088	2.363
F	1.514	0.547	0.137	0.830	0.536	1.084	1.328	2.207	1.357	1.807	0.908	0.693
G	0.957	0.107	0.098	0.234	0.049	1.696	0.537	0.342	1.584	0.908	0.117	0.049
H	0.166	2.324	1.572	1.465	0.039	0.059	0.039	0.117	2.393	1.445	1.490	0.334

Tuesday 16-11-1993
NUMBER OF MEASUREMENTS: 96
ID: 12543

	1	2	3	4	5	6	7	8	9	10	11	12
A	0.068	2.061	0.156	2.080	0.605	1.797	0.098	0.186	0.127	2.480	2.002	2.119
B	0.059	0.098	2.275	1.846	1.221	0.820	0.039	2.080	0.068	0.156	0.000	1.943
C	1.357	1.113	1.064	0.410	0.088	0.156	2.344	0.137	0.059	0.449	0.098	0.059
D	1.514	0.547	0.137	0.830	0.596	1.084	1.328	2.207	1.357	1.807	0.908	0.693
F	0.078	1.914	1.680	0.195	0.107	1.924	1.224	2.231	0.293	0.166	0.830	1.709
G	0.957	0.107	0.098	0.234	0.049	1.699	0.137	0.342	1.484	0.908	0.117	0.049
H	0.078	0.557	0.513	0.617	0.139	1.392	0.527	1.982	0.098	0.078	0.088	2.363
I	0.166	2.324	1.572	2.465	0.039	0.259	0.139	0.517	1.123	1.445	2.490	2.334

Import macro:

```
read(8)
setfield("Date",readuntil(""))
nextline;nextline
```

```

read(4)
setfield("ID",readuntil(""))
repeat 3 times nextline

repeat 8 times {
  read(4)
  repeat 12 times setnextvalue(readuntil(" "))
}

```

Example 2: Line-delineated file with 6 info fields and 90 binary characters

File CD-ROM:\Examples\TXTfiles\Expmacro\Example2.txt

```

Xanthomonas
campestris
citri A
LMG 680
A
24
001100010001011111000101101000110011100000001100000001110111100101000000001000000000001111
Xanthomonas
campestris
citri A
LMG 8657
A
24
00110001000101111100110110100011001110000001100110001110111101111000001001110010000001000
Xanthomonas
campestris
citri B
LMG 9182
A
24
00000001000101001100000100000011001101000000110000000111011110010000000100100000000000000

```

Import macro:

```

setfield("Genus",readuntil(""))
setfield("Species",readuntil(""))
setfield("Subspecies",readuntil(""))
setfield("Catalogue #",readuntil(""))
setfield("Category",readuntil(""))
setfield("Number",readuntil(""))
repeat 90 times setnextvalue(read(1))

```

Example 3: Tab-delineated file with 4 info fields and 8 characters

File CD-ROM:\Examples\TXTfiles\Expmacro\Example3.txt

Xanthomonas	campestris	hyacinthi	LMG739	1	5	8	4	3	2	4	1
Xanthomonas	campestris	campestris	LMG568	3	5	6	2	3	1	4	3
Xanthomonas	campestris	phaseoli	LMG803	2	4	7	4	1	2	3	2
Xanthomonas	campestris	hyacinthi	LMG946	2	3	6	2	1	0	1	0

Import macro:

```

setfield("Genus",readuntil("□"))

```

```

setfield("Species",readuntil("□"))
setfield("Subspecies",readuntil("□"))
setfield("Catalogue",readuntil("□"))
repeat 8 times setnextvalue(readuntil("□"))

```

The □ sign in the macro stands for a tab character and can be entered using the tab key on the keyboard.

Example 4: Band matching table.

File CD-ROM:\Examples\TXTfiles\Expmacro\Example4.txt

Each in the file is a band with a fragment length specified in the first column. Each fingerprint corresponds to a column of "+" and "-". "+" means that the band is present on this fingerprint, "-" means that it is absent

```

523.7  +---++++-----+---++-++-++-+-
405.9  -----+-----+-----+-----
324.6  -++-++-++-++-++-++-++-++-++-
202.0  +++++-++++++-++-++-++++++-
172.6  ---+-----++-----+-----
130.4  ---+-----+---+---+---+---+
90.5   +++++++-++-++-++-++-++-++-
79.6   +-----+-----+-----+-----
61.0   ++++-----+-----+---+---+
55.9   +---+---+---+---+---+---+---

```

Import macro:

```

string fragment
while not(finished) do {
  fragment=read(5)
  read(2)
  setcurentury(1)
  while not(endofline) do {
    if read(1)="+" then setvalue(fragment,1)
    if not(endofline) then setcurentury(curentury+1)
  }
}

```

Example 5: A file containing densitometric curves with a variable resolution

File CD-ROM:\Examples\TXTfiles\Expmacro\Example5.txt

```

ID: AC125
NAME: sample1
RESOLUTION: 30
1 2 2 2 3 8 9 15 50 30 15 6 0 1 2 2 5 15 20 17 6 3 5 7
9 12 30 70 50 30
ID: AC126
NAME: sample2
RESOLUTION: 35
1 2 2 2 3 8 9 15 50 30 15 6 0 1 2 2 5 15 20 17 6 3 5 7
9 12 30 70 50 30 25 22 10 5 2

```

Import macro:

```
integer resolution

read(4);setfield("ID",readuntil(""))
read(6);setfield("Name",readuntil(""))
read(12);resolution=val(readuntil(""))
repeat resolution times setnextvalue(readuntil(" "))
```

Example 6: An ABI Genescan file containing band sizes and areas

File CD-ROM:\Examples\TXTfiles\Genescan.txt

The file can be found on the CD-ROM as **Examples\TXTfiles\Genescan.txt**. Below is only an extract of the whole file (5 patterns).

16B,1	30.90	50.32	120	557	309
16B,2	35.30	69.74	363	1946	353
16B,3	36.80	76.35	169	1108	368
16B,4	50.50	134.62	170	2053	505
16B,5	55.30	154.84	955	8872	553
16B,6	64.40	192.97	100	538	644
16B,7	67.80	207.13	103	953	678
16B,8	69.70	215.16	794	6394	697
16B,9	78.10	251.08	1960	13690	781
16B,10	81.60	266.33	166	1317	816
16B,11	97.30	339.15	128	771	973
16B,12	101.60	360.02	131	478	1016
16B,13	109.00	397.88	233	2109	1090
16B,14	121.00	464.69	137	1316	1210
16B,15	136.50	561.48	314	2020	1365
16B,16	139.30	580.42	269	2086	1393
16B,17	152.70	679.60	180	1611	1527
16B,18	161.50	753.53	151	925	1615
16B,19	162.10	758.86	181	1181	1621
16B,20	163.30	769.64	192	1580	1633
16B,21	168.80	821.16	207	1467	1688
17B,1	33.00	60.47	228	929	330
17B,2	34.60	67.53	201	815	346
17B,3	43.30	106.02	113	855	433
17B,4	52.90	146.14	381	1908	529
17B,5	88.20	298.95	131	690	882
17B,6	89.00	302.68	1425	7821	890
17B,7	155.40	709.46	304	1800	1554
17B,8	158.50	736.00	182	966	1585
17B,9	165.10	796.02	121	713	1651

Import macro:

```
string  keystr,curstr,mw
integer height

keystr=readuntil(",")
curstr=keystr
setcolumn(1)
setfield("sample",keystr)
while curstr=keystr do {
  curstr=readuntil(",")
  if curstr=keystr then {
    readuntil("□");readuntil("□")
    mw=readuntil("□")
    if mw<>"" then {
      height=readuntil("□")
      setvalue(mw,height)
    }
    nextline
  }
}
```

```
setcolumn(1)
```

```
(□ = tab)
```

Example 7: A file containing aligned sequence data.

File CD-ROM:\Examples\TXTfiles\Expmacro\Example7.txt

The spaces and the numbering need to be removed. The dashes “-” mean gaps and should be converted to spaces in order to match the Fingerprinting II Informatix format.

```
1  GATACGACGC CTG----- -----GCGAT CGATCGATCG
41 ATCGATCGAT ATCGATCGAT CGATCGATCG ATCGATCGAT
81 CTAGCTCGCT CGCTAGCTCG CGATC----- -----
121 -----CGCT CGCTCGCTCT CGCTCGCGCG CGCTCGCGCG
```

Import macro:

```
string sq
while not(finished) do {
  readwhile(" 0123456789")
  sq=readuntil(" 0123456789")
  while find(sq,"-")>0 do
    sq=substring(sq,1,find(sq,"-")-1)+" "+substring(sq,find(sq,"-")+1,1000)
  addsequence(sq)
}
```

Example 8: Report of a fatty acid methyl ester (FAME) analysis of bacterial strains using the MIS system (MIDI).

File CD-ROM:\Examples\TXTfiles\Fame.txt

The spaces and the numbering need to be removed. The dashes “-” mean gaps and should be converted to spaces in order to match the Fingerprinting II Informatix format.

DATA2:F90705633 06-JAN-95 16:29:38

 ID: 4632 XANT-ALBIL (LMG 482) Date of run: 06-JUL-90 03:16:41
 Bottle: 18 SAMPLE [AEROBE] Date edited: 05-OCT-93 18:09:08

RT	Area	Ar/Ht	Respon	ECL	Name	%	Comment 1	Comment 2
1.733	31287000	0.062	. . .	6.997	SOLVENT PEAK	. . .	< min rt	
3.131	1399	0.033	1.210	9.999	10:0	0.89	ECL deviates -0.001	Reference 0.002
3.522	9527	0.036	1.173	10.606	11:0 ISO	5.85	ECL deviates 0.001	Reference 0.004
4.724	5540	0.041	1.093	12.086	11:0 ISO 3OH	3.17	ECL deviates -0.004	
6.291	2791	0.050	1.030	13.449	12:0 3OH	1.50	ECL deviates -0.006	
7.159	4777	0.051	1.004	14.106	13:0 ISO 3OH	2.51	ECL deviates -0.004	
7.918	3182	0.052	0.985	14.621	15:0 ISO	1.64	ECL deviates 0.000	Reference 0.000
9.360	2525	0.061	0.955	15.550	16:0 N alcohol . . .	1.26	ECL deviates 0.001	
9.482	10702	0.055	0.952	15.626	16:0 ISO	5.34	ECL deviates -0.000	Reference -0.001
9.788	26845	0.057	0.947	15.816	16:1 w7c	13.30	ECL deviates -0.001	
10.082	30425	0.055	0.942	15.999	16:0	14.99	ECL deviates -0.001	Reference -0.002
10.415	1486	0.056	. . .	16.197		. . .		
10.787	48496	0.058	0.930	16.417	ISO 17:1 w9c	23.61	ECL deviates 0.001	
11.144	35119	0.057	0.925	16.629	17:0 ISO	17.00	ECL deviates 0.000	Reference -0.001
11.301	8521	0.057	0.923	16.723	17:0 ANTEISO	4.11	ECL deviates 0.001	Reference -0.001
11.418	1096	0.056	0.921	16.792	17:1 w8c	0.53	ECL deviates 0.000	
12.471	1525	0.068	0.907	17.408	17:0 10 methyl . . .	0.72	ECL deviates -0.002	
13.185	6278	0.068	0.898	17.822	Sum In Feature 7 . .	2.95	ECL deviates -0.000	18:1 w7c/w9t/w12t
13.767	1317	0.067	0.892	18.159	17:0 ISO 3OH	0.61	ECL deviates -0.005	
*****	6278	. . .	. . .	. . .	SUMMED FEATURE 7 . .	2.95	18:1 w7c/w9t/w12t	18:1 w9c/w12t/w7c
*****		. . .	. . .	. . .		. . .	18:1 w12t/w9t/w7c	

Solvent Ar	Total Area	Named Area	% Named	Total Amnt	Nbr Ref	ECL Deviation	Ref ECL Shift
31287000	201551	200065	99.26	191066	7	0.002	0.002

Import macro:

```
string st
string RT,nm,cn
integer i
```

```
st=read(3)
while st<>"ID:" do {
  nextline; st=read(3)
  if finished then stop
}
setcolumn(20)
st="";i=0
while st<>" " do {
  st=readuntil(" -")
  if st="XANT" then st="Xanthomonas"
  if st="ALBIL" then st="albilineans"
  if st="ARBOR" then st="arboricola"
  if st="JUCLA" then st="juglandis"
  if st="CORYL" then st="corylina"
  if st="POPUL" then st="populi"
  setcolumn(givecolumn-1)
  i=i+1;setfield("Field"+str(i),st)
  st=read(1)
}
readuntil("(")
setfield("Code",readuntil("-"))
repeat 5 times nextline
```

(abbreviated names are replaced)

```
RT=""
while (RT<>" " and (RT<>"-----")) do {
  RT=read(7)
  setcolumn(43);nm=read(20)
  setcolumn(64);cn=read(6)
  while (substring(nm,length(nm),length(nm))=".") or
  (substring(nm,length(nm),length(nm))=" ") do
    nm=substring(nm,1,length(nm)-1)
```

(seven spaces)

```
if (val(RT)>0) and (length(nm)>0) and (val(cn)>0) then
    setvalue(nm,cn)
nextline
}
```

25. Appendix

25.1 Connected Database table structure

25.1.1 Introduction

In the description below, the structures of the tables required by Fingerprinting II Informatix in a Connected Database are given (see chapter 22). The tables are indicated with their default names. As pointed out in 22.5, however, it is possible to use different names for these tables or views in an actual database, which are recorded in the connected database configuration file (.xdb). The names of the columns within the tables, however, are fixed.

The object “**CLOB**” means a large text field. This may be described differently depending on the database use (e.g., the Access equivalent is “**memo**”).

NULL values should be allowed for all fields.

25.1.2 Table ENTRYTABLE

This table contains a record for every entry in the database.

- KEY (VARCHAR(80))
The unique identifier for every entry in the database (e.g. isolate number).

Other fields: additional database information fields.

25.1.3 Table EXPERIMENTS

This table contains a record for every experiment type in the database.

- EXPERIMENT (VARCHAR(80))
Holds the name of the experiment (should be unique through the whole database).
- TYPE (VARCHAR(80))
Can be “Fingerprint”, “Character”, or “Sequence”.
- SETTINGS (CLOB)
XML string that holds the processing, visualization and analysis settings of the experiment type.
- TABLES (VARCHAR(160))
Used for character experiments only: holds the name of the tables that hold character values and additional character fields (separated by a comma).

25.1.4 Table FPRINTFILES

This table contains a record for every “batch” of fingerprints that is entered in the database. A batch may correspond to fingerprints that should be normalized simultaneously: e.g. they were run on the same electrophoresis gel, or run in the same batch on a sequencer, etc.

- FILENAME (VARCHAR(80))
The name of the batch (should be unique for every batch). In case of scanned electrophoresis gels, this corresponds to the name of the TIFF image file.
- EXPERIMENT (VARCHAR(80))
Name of the experiment type to which this fingerprint batch belongs.
- LOCKED (VARCHAR(10))
Whether or not this batch is locked (Yes or No).
- INLINELINK (VARCHAR(80))
If this batch is linked to another batch (for normalization purposes), this specifies the name of the batch that contains normalization info.
- BOUNDINGBOX (VARCHAR(200))
Specifies the bounding box of the lanes on a 2D fingerprint image.
- SETTINGS (VARCHAR(250))
Data processing settings.
- TONECURVE (VARCHAR(200))
Specifies how bitmap pixel values are mapped to grey shades on the screen.
- REFSYSTEM (CLOB)
Specifies the reference system that is used to normalize the batch.
- MARKERS (VARCHAR(200))
Holds marker points that may be used to align linked fingerprint images to each other.

25.1.5 Table FPRINT

This table contains a record for every fingerprint that is entered in the database.

- KEY (VARCHAR(80))
The unique identification key of the sample to which this fingerprint belongs.
- EXPERIMENT (VARCHAR(80))
The name of the experiment type to which this fingerprint belongs.
- FILENAME (VARCHAR(80))
The name of the batch to which this fingerprint belongs.
- FILEIDX (NUMBER)
The number of the fingerprint inside the fingerprint file.
- SPLINE (VARCHAR(200))
Holds the exact positioning and size of the gelstrip on the image.

- CURVESPLINE (VARCHAR(200))
Describes what part of the gelstrip is used for calculation of the densitometric curve.
- GELSTRIPINFO (VARCHAR(50))
Contains resolution information about the gelstrip image info.
- GELSTRIP (CLOB)
This field holds the bitmap values of the gelstrip.
- DENSCURVEINFO (VARCHAR(50))
Holds the resolution of the densitometric curve.
- DENSCURVE (CLOB)
Holds the densitometric curve data.
- BANDS (CLOB)
Holds information about the bands assigned on the fingerprint.
- BANDCONC (CLOB)
Holds information about 2D concentration estimates.
- BANDCONCINFO (CLOB)
Holds information about 2D concentration estimates.
- REFPOS (VARCHAR(250))
Contains the reference positions assigned to this fingerprint.
- MAPFORWARD (CLOB)
Contains a forward normalization vector.
- MAPBACK (CLOB)
Contains the reverse normalization vector.
- REFSYSTEM (CLOB)
Holds the reference system of the fingerprint.
- ONECURVE (VARCHAR(250))
Contains the tone curve.

25.1.6 Character Values table

Each character type has its own table holding character value information for the database entries. The default name of this table is the name of the character type, although it is possible to specify any table name (the exact name is contained in the TABLES column of the EXPERIMENTS table). Each record in the table corresponds to a single character value belonging to a single entry in the database.

- KEY (VARCHAR(80))
Key of the entry this character value belongs to.
- CHARACTER (VARCHAR(80))
Key of the character.
- VALUE (FLOAT)
Numerical value.

25.1.7 Character Fields table

Contains information about the additional information fields that can be stored together with characters in a character type. The default table name is the name of the character type, padded with "FIELDS", but it is possible to specify any other name (the exact name is contained in the TABLES column of the EXPERIMENTS table). Every record in this table corresponds to a single field for a single character

- CHARACTER (VARCHAR(80))
Name of the character this information field belongs to.
- FIELD (VARCHAR(80))
Name of the field.
- CONTENT (VARCHAR(150))
Content of the field.

25.1.8 Table SEQUENCES

This table holds the sequence information stored in the database.

- KEY (VARCHAR(80))
Key of the database entry this sequence belongs to.
- EXPERIMENT (CHARCHAR(80))
Experiment type of the sequence.
- SEQUENCE (CLOB)
Sequence data.
- CONTIGFILE (VARCHAR(80))
File name of the contig file that is associated to this sequence (if any). This corresponds to a file in the "contig" subdirectory of the "source files" directory.

25.1.9 Table MATRIXVALS

Holds pairwise similarity values. Each record in this table represents a single similarity value between two database entries.

- EXPERIMENT (VARCHAR(80)).
Name of the experiment type this similarity value belongs to.
- KEY1 (VARCHAR(80))
Key of the first database entry.
- KEY2 (VARCHAR(80))
Key of the second database entry.
- VALUE (FLOAT)
Similarity value.

25.1.10 Table SUBSETMEMBERS

This table contains information about the subsets that were defined in the database. Each record specifies the membership of a single entry to a single subset.

- KEY (VARCHAR(80))
The key of the database entry.
- SUBSET (VARCHAR(80))
The name of the subset to which this key belongs.

25.1.11 Table EVENTLOG

This table maintains a history list of events that were generated during the manipulation of the database.

- DATETIME (VARCHAR(80))
Recording date and time of the event.
- LOGIN (VARCHAR(50))
Windows login at the moment the event was generated.
- TYPE (VARCHAR(10))
Event type.
- SUBJECT (VARCHAR(50))
Database component for which this event was generated.
- DESCRIPTION (VARCHAR(500))
Description of the event.

25.1.12 Indices in the database

In order to obtain sufficient speed for larger databases, it is absolutely necessary that a number of indices are present. This section contains a list of advised indices. However, depending on the purpose of the database (emphasis on read or write, database size...), it may be preferable to modify, add or remove indices. For larger databases where speed becomes critical, it is strongly advised to use the tuning tools provided with the database in order to optimize the various settings and indices.

- ENTRYTABLE:
 - KEY (may be defined as primary key).
- EXERIMENTS:
 - EXPERIMENT (may be defined as primary key). This usually won't attribute to the performance, since the number of records in this table is usually very limited.
- FPRINTFILES:
 - FILENAME (may be defined as primary key).
- FPRINT:

- KEY. It should not be unique or primary key, since some lanes on a gel image may not be added to the database and will have an empty key (e.g. reference lanes).
 - FILENAME. Note that this field should not be required, because some databases may contain fingerprints that are not associated with any batch (file).
 - FILENAME,FILEIDX.
- Character values table:
 - CHARACTER.
 - KEY.
- Character fields table:
 - CHARACTER,FIELD.
- SEQUENCES
 - KEY.
- MATRIXVALS
 - EXPERIMENT,KEY1,KEY2.
- SUBSETMEMBERS
 - KEY.
 - SUBSET.

25.2 Regular expressions

A "regular expression" is a pattern that describes a set of strings. Regular expressions are constructed analogously to arithmetic expressions, by using various operators to combine smaller expressions. ``grep'` understands two different versions of regular expression syntax: "basic" and "extended". In GNU ``grep'`, there is no difference in available functionality using either syntax. In other implementations, basic regular expressions are less powerful. The following description applies to extended regular expressions; differences for basic regular expressions are summarized afterwards.

The fundamental building blocks are the regular expressions that match a single character. Most characters, including all letters and digits, are regular expressions that match themselves. Any metacharacter with special meaning may be quoted by preceding it with a backslash. A list of characters enclosed by ``['` and `']'` matches any single character in that list; if the first character of the list is the caret ``^'`, then it matches any character **not** in the list. For example, the regular expression ``[0123456789]'` matches any single digit. A range of ASCII characters may be specified by giving the first and last characters, separated by a hyphen.

Finally, certain named classes of characters are predefined, as follows. Their interpretation depends on the ``LC_CTYPE'` locale; the interpretation below is that of the POSIX locale, which is the default if no ``LC_CTYPE'` locale is specified.

``[:alnum:]'`

Any of ``[:digit:]'` or ``[:alpha:]'`

``[:alpha:]'`

Any letter:

``a b c d e f g h i j k l m n o p q r s t u v w x y z'`,

``ABCDEFGHIJKLMNOPQRSTUVWXYZ'.`

``[:blank:]'`

Space or tab.

``[:cntrl:]'`

Any character with octal codes 000 through 037, or ``DEL'` (octal code 177).

``[:digit:]'`

Any one of ``0 1 2 3 4 5 6 7 8 9'`.

``[:graph:]'`

Anything that is not a ``[:alnum:]'` or ``[:punct:]'`.

``[:lower:]'`

Any one of ``a b c d e f g h i j k l m n o p q r s t u v w x y z'`.

``[:print:]'`

Any character from the ``[:space:]'` class, and any character that is *\*not\** in the ``[:graph:]'` class.

``[:punct:]'`

Any one of ``! " # $ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ ` { | } ~'`.

``[:space:]'`

Any one of ``CR FF HT NL VT SPACE'`.

``[:upper:]'`

Any one of ``A B C D E F G H I J K L M N O P Q R S T U V W X Y Z'`.

``[:xdigit:]'`

Any one of ``a b c d e f A B C D E F 0 1 2 3 4 5 6 7 8 9'`.

For example, ``[[[:alnum:]]'` means ``[0-9A-Za-z]'`, except the latter form is dependent upon the ASCII character encoding, whereas the former is portable. (Note that the brackets in these class names are part of the symbolic names, and must be included in addition to the brackets delimiting the bracket list.) Most metacharacters lose their special meaning inside lists. To include a literal ``]`, place it first in the list. Similarly, to include a literal ``^'`, place it anywhere but first. Finally, to include a literal ``-'`, place it last.

The period ``.'` matches any single character. The symbol ``\w'` is a synonym for ``[[[:alnum:]]'` and ``\W'` is a synonym for ``[^[:alnum]]'`.

The caret ``^'` and the dollar sign ``$'` are metacharacters that respectively match the empty string at the beginning and end of a line. The symbols ``\<'` and ``\>'` respectively match the empty string at the beginning and end of a word. The symbol ``\b'` matches the empty string at the edge of a word, and ``\B'` matches the empty string provided it's not at the edge of a word.

A regular expression may be followed by one of several repetition operators:

``?'`

The preceding item is optional and will be matched at most once.

``*'`

The preceding item will be matched zero or more times.

``+'`

The preceding item will be matched one or more times.

``[N]'`

The preceding item is matched exactly N times.

``[N,]'`

The preceding item is matched n or more times.

``[N,M]'`

The preceding item is matched at least N times, but not more than M times.

Two regular expressions may be concatenated; the resulting regular expression matches any string formed by concatenating two substrings that respectively match the concatenated subexpressions.

Two regular expressions may be joined by the infix operator ``|'`; the resulting regular expression matches any string matching either subexpression.

Repetition takes precedence over concatenation, which in turn takes precedence over alternation. A whole subexpression may be enclosed in parentheses to override these precedence rules.

The backreference ``\N'`, where N is a single digit, matches the substring previously matched by the Nth parenthesized subexpression of the regular expression.

Alphabetical list of menu commands and keywords

—A—

ABI, importing sequencer files, 66
Abiconv (program), 65
Account for covariance structure (MANOVA), 202
Add (Netkey), 22
Add array of characters, 76
Add color, 75
Add new entries, 33
Add new experiment file, 78, 94
Advanced query tool, 109
Align external branch, 182
Align internal branch, 182
Align|Align incremental, 101
Align|Break alignment, 100
Align|Invert and complement group, 101
Alignment parameters (contigs), 99
Alignment|Align all sequences, 100
Alignment|Set parameters, 99
amino acid sequences, 93
Analysis, 25
Analyze, 9, 31, 32, 107
Area sensitive (coefficient), 143
Arithmetic average, 47
Arrange by similarity, 206
Auto clean sequences (contigs), 100
Auto construct tables, 237
Auto create (.mdb), 235
Auto insert notification, 17
Autocorrection parameters (contigs), 99
Average (K-means), 153
Average similarities (jackknife), 155
Average thickness, 66
Averaging thickness (curves), 47

—B—

Background subtraction, 38, 47
Background subtraction (2-D image), 42
Background subtraction (BNIMA), 82
Ball size, 82
Band class filters, 132
Band classes|Add new band class, 129, 130
Band classes|Assign band to class, 129
Band classes|Auto assign bands to class, 129, 130
Band classes|Center class position, 130
Band classes|Remove band class, 129, 130
Band classes|Remove band from class, 130
Band finding (settings), 53
Band height, 135
Band matching, 123
Band search filters, 53
Band search, shoulder sensitivity, 55
Band surface, 135
Bandmatching, 120
Bandmatching|Auto assign all bands to all classes, 136
Bandmatching|Band class filter, 132
Bandmatching|Comparative Quantification settings, 135
Bandmatching|Perform band matching, 126, 136, 137
Bandmatching|Polymorphic bands only (for selection list), 137

Bandmatching|Polymorphic classes only (for selection list), 131
Bandmatching|Search band classes, 131
Bands, 39, 55, 166
Bands (assigning), 53
Bands|Add new band, 56
Bands|Auto search bands, 55
Bands|Delete selected band(s), 57
Bands|Mark band(s) as certain, 57
Bands|Mark bands as uncertain, 57
BioNumerics Server, 219
Bitmap export, 158
BNIMA, 82
Bootstrap analysis, 151
Build (connected databases), 237
Bundles, 215
Bypass normalization, 60

—C—

Calculate quality factors, 209
Calculate relative character importance (MANOVA), 202
Calculate|Experiment correlations, 160
Calculate|Similarity plot, 161
Calibration curve, 72
Canberra metric, 170
Case sensitive, 109
Case sensitive (client/server), 223
Categorical coefficient, 162, 170
Cells|Add disk to mask, 83
Cells|Add pixels to mask, 84
Cells|Add selected, 83
Cells|Edit color scale, 84
Cells|Remove pixels from mask, 84
Change access (Netkey), 23
Change entry key, 33
Change towards end of fingerprint, 126, 144
Changing fingerprint type, 65
Character file, new, 78
Character import, from TIFF, 80
Character types, 9, 17, 37, 74, 77, 105
Character value (query), 110
Character|Change character range value, 75
Characters|Add new character, 74, 79
Characters|Order characters by component, 197, 199
Characters|Use character for comparisons, 76
Check table structure, 237
Client functions of BioNumerics, 219
Cluster analysis, 139
Cluster cutoff method, 151
Clustering, 120, 172, 174
Clustering of characters, 169
Clustering of fingerprints, 165
Clustering of sequences, 171
Clustering, global alignment, 180
Clustering|Bootstrap analysis, 151
Clustering|Calculate cluster analysis, 143, 148, 151, 162, 166, 191
Clustering|Calculate cophenetic correlations, 150
Clustering|Calculate error flags, 150
Clustering|Collapse/expand branch, 148
Clustering|Congruence of experiments, 159

Clustering|Maximum likelihood cluster analysis, 189
Clustering|Maximum parsimony cluster analysis, 187
Clustering|Reroot tree, 148
Clustering|Select root, 148
Clustering|Swap branches, 148
Clustering|Tolerance & optimization analysis, 166, 167
Color cale (BNIMA), 82
Comparative quantification, 123
Comparison|Compare two entries, 117
Comparison|Create new comparison, 118
Complete linkage, 144
Component type, 195
Composite data set, 140
Composite|Calculate consensus matrix, 164
Composite|Discriminative characters, 137, 140
Composite|Export character table, 134, 135, 140
Composite|Show quantification (colors), 135, 138
Composite|Show quantification (values), 135
Composite|Sort by character, 138, 140
Concentration, 135
Congruence between techniques, 159
Connect to an existing connected database, 234
Connected database, 27
Connected Databases, 233
Consensus match, 175
Consensus sequence, 171
Consider absent values as zero, 74
Conversion to binary, 169
Cophenetic correlation, 150
Copy from character (color), 75
Copy to all characters (color), 76
Copy to character (color), 75
Copy to clipboard (log file), 29
Correct for internal weights, 162
Correction parameters (sequence clustering), 174
Correlation type, 160
Cosine coefficient, 143, 170
Cost table (parsimony), 188
Create a new, empty connected database, 234
Create character (ODBC), 231
Create from database field, 149
Create keys from tag, 94
Create new fingerprint type, 39
Cup type, 76
Curves, 39, 68
Curves|Spectral analysis, 48

—D—

Database directory, 27
Database field (query), 110
Database field range (query), 110
Database fields, 94
Database sharing tools, 215
Database|Add all lanes to database, 65
Database|Add lane to database, 65
Database|Add new entries, 33, 62, 64, 238
Database|Add new information field, 33
Database|Change entry key, 33
Database|Change fingerprint type of lane, 65
Database|Configure external database link, 229
Database|Connected databases, 235
Database|Copy from external database, 230
Database|Download field from external database, 230
Database|Download selection list, 223

Database|Identify selection list, 225
Database|Link lane, 64
Database|Remove all links, 65
Database|Remove entry, 33
Database|Remove information field, 34
Database|Remove link, 65
Database|Remove unlinked entries, 33
Database|Rename information field, 34
Database|Search, 222
Database|Select list from external database, 230
Database|Send bundle to server, 224
Database|Upload comparison settings, 227
Databases, 11
Degree (congruence of techniques), 161
Delete (Netkey), 23
Delete database, 27
Demobase, 17, 31, 32
Densitometric curves, 46, 47, 66
Densitometric values (BNIMA), 82
Details (bundle), 218
Dice, 143, 144, 167
Different bands (coefficient), 143
Dimensioning|Multi-dimensional scaling, 191
Dimensioning|Principal Components Analysis, 193, 198
Dimensioning|Self organizing map, 199
Discard unknown bases, 173, 174, 181
Disconnect (Netkey), 23
Discriminants (with variance), 198
Discriminants (without variance), 198
Divide by variance, 198
Divide by variance (PCA), 194
DNS Configuration, 20
DNS host name, 20
Do not create keys, 94
Duplicate keys, 64
Dynamical preview, 40, 66

—E—

Edit database fields, 200
Edit image (BNIMA), 81
Edit|Arrange entries by database field, 35, 120
Edit|Arrange entries by database field (numerical), 35
Edit|Arrange entries by similarity, 206
Edit|Auto trim sequences, 98
Edit|Bring selected entries to top, 108, 137
Edit|Change brightness & contrast, 40, 44, 66
Edit|Change key, 246
Edit|Clear selection list, 108, 109, 116
Edit|Copy selection, 116, 121, 147, 208
Edit|Cut left from cursor, 98
Edit|Cut right from cursor, 98
Edit|Cut selection, 116, 120, 121, 131, 147, 148, 182, 187
Edit|Delete base, 100
Edit|Delete current (subset), 117
Edit|Delete selection, 117
Edit|Edit tone curve, 44
Edit|Load default settings, 59
Edit|Paste selection, 116, 120, 121, 131, 147, 148, 182, 205, 208
Edit|Previous page, 157
Edit|Redo, 39
Edit|Reload sequence, 98
Edit|Rename current (subset), 117
Edit|Rescale curves, 48, 57

Edit|Save as default settings, 59
 Edit|Search entries, 108
 Edit|Set database field length, 35
 Edit|Settings, 42, 46, 47, 53
 Edit|Settings (BNIMA), 82, 84, 86
 Edit|Settings (fingerprints), 48
 Edit|Show value scale (BNIMA), 82
 Edit|Toggle overwrite/insert mode, 98
 Edit|Undo, 39
 Edit|Zoom in, 40, 157
 Edit|Zoom out, 40, 157
 EMBL format, 94
 Enable log files, 29
 Enable the use of log files, 27
 Enhanced metafile export, 158
 Entries|Add new entries, 79, 94
 Error flags, 150
 Estimate errors, 190
 Euclidean distance, 170
 Experiment, 31, 60, 62, 106, 125
 Experiment card, 76, 103, 104, 105
 Experiment presence (query), 110
 Experiment|Comparison settings, 142
 Experiment|Correct for internal weights, 141, 162
 Experiment|Train neural network, 212
 Experiment|Use for identification, 207
 Experiment|Use in composite data set, 125, 141
 Experiments|Create new character type, 73
 Experiments|Create new composite data set, 125, 141
 Experiments|Create new fingerprint type, 37
 Experiments|Create new matrix type, 101
 Experiments|Create new sequence type, 92
 Experiments|Edit experiment type, 59, 74
 Export band metrics, 104
 Export normalized band positions, 104
 Export normalized curve, 104

—F—

Fields|Add new field, 76
 Fields|Remove field, 76
 Fields|Rename field, 76
 Fields|Set field content, 76
 Fields|Use as default field, 76
 File|Add experiment file, 238
 File|Add new library unit, 208
 File|Calculation priority settings, 146
 File|Clear log file, 29
 File|Copy correspondence plot to clipboard, 204
 File|Copy discriminants to clipboard, 204
 File|Copy image to clipboard, 189, 193
 File|Copy image to clipboard (characters), 197
 File|Copy image to clipboard (entries), 197
 File|Copy page to clipboard, 158
 File|Create new bundle, 215
 File|Delete experiment file, 38, 80, 95
 File|Edit character data, 106
 File|Edit fingerprint data, 106
 File|Edit library unit, 208
 File|Edit sequence data, 106
 File|Exit, 32
 File|Export bands (comparison), 119
 File|Export character coordinates, 197
 File|Export database fields, 152, 188, 192, 196, 206
 File|Export densitometric curves (comparison), 119

File|Export entry coordinates, 197
 File|Export report to file, 210
 File|export sequences, 184
 File|Export similarity matrix, 152
 File|Import experiment data, 71
 File|Import experiment file, 37, 77, 93, 102
 File|Import from external database, 231
 File|Import sequences from clipboard, 101
 File|Import sequences from file, 101
 File|Link to reference gel, 67
 File|Load configuration, 87
 File|Load image (BNIMA), 81
 File|Lock, 28
 File|Open additional database, 229
 File|Open bundle, 216
 File|Open database, 227
 File|Open experiment file (data), 38, 66, 67, 78, 94
 File|Open experiment file (entries), 61, 64, 79, 95
 File|Open project, 101
 File|Open reference gel, 67, 68
 File|Print all pages, 158
 File|Print correspondence plot, 204
 File|Print database fields, 206
 File|Print discriminants, 204
 File|Print image, 189, 193
 File|Print image (characters), 197
 File|Print image (entries), 197
 File|Print preview, 157
 File|Print report, 210
 File|Print this page, 158
 File|Printer setup, 158
 File|Save configuration as, 87
 File|Save file as, 66
 File|Save project, 101
 File|Tools|Horizontal mirror of TIFF image, 39
 File|Tools|Vertical mirror of TIFF image, 39
 File|Update linked information, 67, 68
 File|Upload file to server, 224
 File|View log file, 29
 Files, 66
 Filter programs, 25
 Filtering, 47
 Filters, 66
 Fingerprint bands (query), 110
 Fingerprint data editor, 39, 40, 46, 50, 55
 Fingerprint types, 9, 17, 37, 38, 59
 Force through 100%, 161
 Foreground, 146
 Furthest neighbor (K-means), 153
 Fuzzy logic (coefficient), 143

—G—

Gap penalty, 173, 174, 181
 GelCompar version 4.x, import from, 73
 Gelstrip thickness, 66
 GeneBuilder, 95
 GeneMaths, 170
 Genescan tables, importing, 69
 Genus, 149, 205
 Global alignment, 172
 Gower, 170
 Grazy zone (bands), 54
 Grid definition, 82
 Grid|Add new, 83

Grid|Delete, 83
Grid|Delete selected, 83
 Group separation statistics, 154
 Group violation, 154
 Group violations, 156
Group|Create from database field, 149
Group|Partitioning of groups, 153
 Groups, 149
 Groups|Assign selected to, 149
 Groups|Assign selected to|None, 153
 Groups|Create from database field, 166
 Groups|Group separations, 154
 Groups|Multivariate Analysis of Variance, 202
 Groups|Partitioning of groups, 154

—H—

Hidden nodes, 212
 Home directory, 11, 25, 26
 Homedir, 20
 HTML, 220
 Hue only (BNIMA), 82

—I—

ID code, 27, 28
 Identification, 205, 207
 Identification against database entries, 205
 Identification|Create new library, 207
 Identification|Identify selected entries, 209
 Idle time background, 146
 Image type (BNIMA), 82
 Importing using ODBC, 229
 Info, 20
 Inspect, 25
 Install BioNumerics, 17
 Install Netkey server program, 20
 Internal reference markers, 67
 IP address, 20, 21, 22

—J—

Jaccard, 143
 Jackknife, 154, 155
 Jeffrey's X, 143
 Jukes and Cantor, 174, 181

—K—

Kendall's tau, 160
 Kimura 2 parameter, 181
 K-means partitioning, 153
 Kohonen map, 199

—L—

Lane|Move down, 66
Lane|Move up, 66
Lane|Remove, 66
Lanes|Add marker point, 68
Lanes|Add new lane, 43
Lanes|Auto search lanes, 41, 66
Lanes|Copy geometry, 68
Lanes|Delete selected lane, 43
Lanes|Paste geometry, 68

Layout, 191
Layout|Compress (X dir), 128
Layout|Create rooted tree, 189
Layout|Enlarge image size, 158
Layout|Optimize branch spread, 189
Layout|Preserve aspect ratio, 196
Layout|Reduce image size, 158
Layout|Rescale curves, 120
Layout|Show 3D plot, 196
Layout|Show bands, 119
Layout|Show branch lengths, 188, 190
Layout|Show construction lines, 193
Layout|Show curves as images, 119
Layout|Show dendrogram, 172, 193
Layout|Show densitometric curves, 119
Layout|Show distances, 148
Layout|Show group colors, 188, 193, 196
Layout|Show image, 126, 172
Layout|Show keys, 192, 196
Layout|Show keys or group numbers, 188
Layout|Show matrix, 152, 172
Layout|Show metric scale, 119, 128
Layout|Show rendered image, 193
Layout|Show similarity matrix, 158
Layout|Show similarity values, 152
Layout|Show space between gelstrips, 119, 159
Layout|Similarity shades, 152
Layout|Stretch (X dir), 128
Layout|Use as X axis, 196
Layout|Use as Y axis, 196
Layout|Use colors, 158
Layout|Use group numbers as key, 196
Layout|Use group numbers as keys, 188, 192
Layout|Zoom in, 119, 127
Layout|Zoom out, 119, 127
 Least square filtering, 48
 Library, 207
 Local database, 27, 245
 Local database, converting to connected database, 241
 Log files, 28
 Logarithmic Dependence, 63
 Login name, 221

—M—

MANOVA, 201
 Match against selection only (Jackknife), 155
 Matrix types, 9, 18, 37, 101
 Maximal similarities (jackknife), 155
 Maximum likelihood, 187
 Maximum number of gaps, 173, 174, 175
 Maximum parsimony, 187
 Maximum similarity used, 160
 Maximum value, 66
 Maximum value (grayscale), 41
 MDS, 191
 Median filter, 47, 48
 Metric|Assign unit, 63
 Metrics range of fingerprint, 72
 Metrics|Add marker, 63
 Metrics|Copy markers from reference system, 63, 71
 Metrics|Cubic spline fit, 63
 Microplate (BNIMA), 81
 Minimal area, 55
 Minimal profiling, 54, 55

Minimum match sequence, 173, 174, 175
Minimum similarity used, 160
Minimum value (grayscale), 40
Mode filter, 47
Molecular sizes (defining), 63
Monotonous fit, 161
Multi-dimensional scaling, 191
Multiple alignment, 171, 172
Multivariate analysis of variance, 201
Mutation rate, 190

—N—

Navigator, 40
Nearest neighbor, 154
Nearest neighbor (K-means), 153
Negative search, 108, 109, 205
Negative search (client/server), 223
Neighbor Joining, 144, 148, 171, 174, 181
Neighbor match, 175
Netkey, 20
Network, 20
Network settings, 20, 24
Neural network, 211
New character type, 73
New database (creating), 26
New fingerprint type, 37
New matrix type, 101
New ODBC, 243
New sequence type, 92
Normal priority background, 147
Normalization, 39, 60, 66
Normalization|Auto assign (bands), 52, 67
Normalization|Auto assign using standard (bands), 52
Normalization|Auto assign using standard (curves), 52
Normalization|Delete all assignments, 52
Normalization|Show distortion bars, 53
Normalization|Show normalized view, 50, 52
Normalization|Update normalization, 53
Normalized view, 67
nucleic acid sequences, 93
Number of bootstrap simulations, 188
Number of columns (character type), 74
Number of groups, 154
Number of nodes, 66
Number of rows (character type), 74
Numerical values, 73

—O—

Ochiai, 143
ODBC connection string, 237
ODBC, import, 77
One dimension, 135
Open entry, 34
Open gap penalty, 173, 174
Optimization, 126, 144, 167
Optimization parameters (contigs), 99
Optimization, find best, 166
Optimize positions (MDS), 191
Optimize topology, 188

—P—

Pairwise alignment, 172

Pairwise alignment settings, 172
Parsimony, 187
Password, 221
Paste data from clipboard, 87
PCA, 120, 191, 193
Pearson correlation, 143, 170
Pheno, 74
Plate (characters), 76
Plot|Use discriminant as X axis, 204
Plot|Use discriminant as Y axis, 204
Polymorphism analysis, 123
Polynomial degree (BNIMA), 86
Port number, 20, 221
Position tolerance, 126, 144, 167
Position tolerance, find best, 166
Preview (band search), 55
Principal Components Analysis, 193
Properties, 17, 20

—Q—

Quantification units, 53
Quantification, comparative, 123
Quantification|Add cells to character set, 86
Quantification|Assign value, 58
Quantification|Band quantification, 58
Quantification|Calculate concentrations, 58
Quantification|Define calibration point, 86
Quantification|Export to clipboard (BNIMA), 87
Quantification|Search all surfaces, 58
Quantification|Search surface of band, 58
Queries, 109

—R—

Rainbow palette, 41
Raw data, 42
Reference lane, 67
Reference system, 51
Reference|Use as reference lane, 50
References|Add external reference position, 50
References|Add internal reference position, 52
References|Copy normalization, 68
References|Paste normalization, 68
References|Use all lanes as reference lanes, 67
Refresh (connected databases), 237
Registry, 11
Regression (congruence of techniques), 161
Regression curve, 63, 69
Relative band surface, 135
Relative to max. value (bands), 55
Relative usage (Netkey), 23
Relative volume, 135
Rename (bundles), 217, 218
Replace list (client/server), 222
Represent as List, 76
Represent as Plate, 76
Resolution of normalized tracks, 60
Restricting queries, 244
Restricting query, 237

—S—

Scripts|Browse Internet, 77
Search functions, 108

Search in list, 108, 205
Search in list (client/server), 222
 Security driver, 19
 Security key, 19
Select branch into list, 147
 Self organizing map, 199
Send message (Netkey), 23
Send message to all users (Netkey), 23
 Sequence types, 9, 18, 37, 93, 106
Sequence|Align external branch, 182
Sequence|Align internal branch, 182
Sequence|Calculate global cluster analysis, 180, 182
Sequence|Change saved sequence, 179
Sequence|Consensus blocks, 176
Sequence|Consensus difference, 176
Sequence|Create consensus of branch, 176, 184
Sequence|Create locked group, 179
Sequence|Edit, 94, 95
Sequence|Find sequence pattern, 180
Sequence|Lock / unlock dendrogram branch, 179
Sequence|Multiple alignment, 174
Sequence|Neighbor blocks, 175
Sequence|Paste from clipboard, 95
Sequence|Reload sequence from database, 179
Sequence|Show global cluster analysis, 182
Sequence|Unlock group, 179
 Server, 219
Server computer name, 20
Server computer name or IP address, 221
Settings, 17, 25, 27, 28, 29
Settings (Netkey), 23
Settings|Binary conversion settings, 76
Settings|Brightness & contrast, 60
Settings|Comparative quantification, 60
Settings|Edit reference system, 63, 71
Settings|Exclude active region, 183
Settings|General settings, 60, 76, 88
Settings|Include active region, 183
Settings|New reference system (curve), 72
Settings|New reference system (positions), 70
Settings|Set as active reference system, 71, 72
Settings|Statistics, 155
Shoulder sensitivity, 55
 Shoulder sensitivity, band search, 55
Show bands, 119, 135
Show dendrogram, 149, 150, 159, 182
Show matrix, 159, 191
Show quantification (colors), 162
Show|Detailed report, 210
Show|Identification comparison, 210
Show|Maximum similarities, 209
Similarity calculation, 172, 173
Similarity coefficient, 142
 Single linkage, 144
 Solving database problems, 245
 SOM, 199
Source file location, 237, 238, 243
Spot removal (2-D image), 42
 Standard, 61
 Standard deviation, 150
Standardized characters, 142, 143
Start service (Netkey), 21

Startup program, 25
Statistics (Netkey), 23
Status (Netkey), 24
Stop Service, 23
 Strips, 39, 68
Strips|Increase number of nodes, 43
Strips|Make larger, 43
Strips|Make smaller, 43
Subsequence (query), 110
 Subsequence search, 180
 Subsets, 115
Subtract average (PCA), 194

—T—

Tables of peaks, 71
Take from experiments, 142, 162
 TCP/IP, 20
 TCP/IP (client server), 219
Thickness (image strips), 42
Tie handling, 155
 Tolerance & optimization statistics, 166
 Tone curve, 44
Two dimensions (quantification), 135

—U—

Uncertain bands, 54, 126, 145
Unit gap penalty, 173, 174
 UPGMA, 143, 144, 149, 154, 174
Use active zones only, 181
Use as default database, 237
Use binary character table, 142
Use character table value, 142
Use conversion cost, 173
Use fast algorithm, 173, 175
Use quantitative values (PCA), 194
Use square root, 143, 170
Use the local database, 234
Use variance (MANOVA), 202

—V—

Validation samples, 212
View calibration curve (BNIMA), 87
View|Customize colors, 98
View|Decrease number of curves, 98
View|Go to next gap, 100
View|Go to next mismatch, 100
View|Go to next problem, 100
View|Go to previous gap, 100
View|Go to previous mismatch, 100
View|Go to previous problem, 100
View|Increase number of curves, 98
View|Toggle nnormalization, 98
 Volume, 135

—W—

Ward, 144, 174
 Webserver, 220



**Bio-Rad
Laboratories, Inc.**

**Life Science
Group**

Web site www.bio-rad.com **Bio-Rad Laboratories Main Office** 2000 Alfred Nobel Drive, Hercules, CA 94547, Ph. (510) 741-1000, Fx. (510) 741-5800
Also in: **Australia** Ph. 02 9914 2800, Fx. 02 9914 2889 **Austria** Ph. (01) 877 89 01, Fx. (01) 876 56 29 **Belgium** Ph. 09-385 55 11, Fx. 09-385 65 54
Brazil Ph. 55 21 507 6191 **Canada** Ph. (905) 712-2771, Fx. (905) 712-2990 **China** Ph. (86-21) 63052255, Fx. (86-21) 53964775
Czech Republic Ph. (420) 2-4141 0532 Fx. (420) 2-4143 1642 **Denmark** Ph. 45 44 52-1000, Fx. 45 4452 1001
Finland Ph. 358 (0)9 804 2200, Fx. 358 (0)9 804 1100 **France** Ph. 01 47 95 69 65, Fx. 01 47 41 9133 **Germany** Ph. 089 318 84-177, Fx. 089 318 84-123
Hong Kong Ph. 852-2789-3300, Fx. 852-2789-1257 **India** Ph. (91-124)-6398112/113/114, 6450092/93, Fx. (91-124)-6398115/6450095
Israel Ph. 03 951 4127, Fx. 03 951 4129 **Italy** Ph. 39 02 216091, Fx. 39 02 21609399 **Japan** Ph. 03-5811-6270, Fx. 03-5811-6272
Korea Ph. 82-2-3473-4460, Fx. 82-2-3472-7003 **Latin America** Ph. 305-894-5950, Fx. 305-894-5960 **Mexico** Ph. 52 5 534 2552 to 54, Fx. 52 5 524 5971
The Netherlands Ph. 0318-540666, Fx. 0318-542216 **New Zealand** Ph. 64-9-4152280, Fx. 64-9-443 3097 **Norway** Ph. 47-23-38-41-30, Fx. 47-23-38-41-39
Poland Ph. (48) 22-8126 672, Fx. (48) 22-8126 682 **Portugal** Ph. 351-21-472-7700, Fx. 351-21-472-7777 **Russia** Ph. 7 095 721 1404, Fx. 7 095 721 1412
Singapore Ph. 65-2729877, Fx. 65-2734835 **South Africa** 00 27 11 4428508, Fx. 00 27 11 4428525 **Spain** Ph. 34 91 590 5200, Fx. 34 91 590 5211
Sweden Ph. 46 (0)8-55 51 27 00, Fx. 46 (0)8-55 51 27 80 **Switzerland** Ph. 061 717-9555, Fx. 061 717-9550 **United Kingdom** Ph. 0800-181134, Fx. 01442-259118