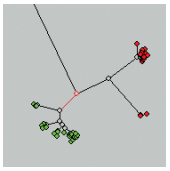
Bioinformatics Software

FPQuest™ Software

Instruction Manual | Version 5



BIO-RAD

NOTES

SUPPORT 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 FPQuest, or suggestions for improvement, refinement or extension of the software to your specific applications:

Bio-Rad Laboratories, Inc.
Life Science Group
2000 Alfred Nobel Drive
Hercules, CA 94547
Technical Support Phone: (800) 424-6723 and (510) 741-6910
FAX: 510-741-5802
E-MAIL: LSG.TechServ.US@Bio-Rad.com (US)
LSG.TechServ.Intl@Bio-Rad.com (International)
URL: www.consult.bio-rad.com

LIMITATIONS ON USE

The FPQuest 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.

Table of Contents

1. INTRODUCTION

1.1 The concepts of FPQuest

1.1.1 The programs

The FPQuest 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. All import and analysis functions are done in the Analyze program. This includes processing of gel files starting from TIFF images, including lane finding and normalization. Independent plugins allow the import of data in different file formats.

1.1.2 The database and the experiments

The logical flow of processing raw experiment files is represented in Figure 1-1. The basis of FPQuest is a relational *database* consisting of *entries*. The entries correspond to the individual organisms or samples under study: animals, plants, fungi, bacterial or viral strains, organic samples, tissue samples,.... 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. The organization and functions of FPQuest databases are discussed in Chapter 2. Each entry in a database may be characterized by one or more *experiments* that can be linked easily to the entry. What we call experiments in FPQuest are in fact the fingerprinting experiments performed to estimate the relationship between the organisms. The FPQuest software is able to process and analyze any densitometric record seen as a profile of peaks or bands. Examples are of course gel and capillary electrophoresis patterns, but also gas chromatography or HPLC profiles, spectrophotometric curves, etc. Fingerprint types can be derived from TIFF or bitmap files as well, which are two-dimensional bitmaps. The condition is that one must be able to translate the patterns into densitometric curves. Chapter 3 of this manual deals with setting up experiments in FPQuest.

Essentially, adding a single organism (entry) with its associated experiments to the database constitutes several steps (see Figure 1-1).

1.1.3 Multi-database setup

FPQuest 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, XP and Vista): The FPQuest users are associated with the Windows NT login users. Each Windows NT user can specify his/her FPQuest databases directory, and FPQuest saves this information in the user's system registry. For example, suppose that a user *X* logs in on a Windows NT machine with FPQuest installed. This user can create a directory, and specify this directory as the **home directory** in the Startup program. FPQuest will save this information in this user's system registry, so that each time the user logs in, FPQuest will automatically consider the same directory as the home directory. In this way, each Windows NT user can define his/her own FPQuest home directory, without interfering with other users. Within this home directory, the user can specify as many ***Databases*** as desired. FPQuest allows two types of databases to be created: a local file-based database using a dedicated database mechanism developed by Bio-Rad, and a DBMS based type which relies on an external ODBC compatible relational database management engine (see Section 2.3 for more information). In the first type, protection of the FPQuest databases depends on the protection of the specified directory by the Windows NT user. If a user protects the directory containing the FPQuest databases, other users will not be able to change or to read the databases, depending on whether the directory is write or read protected. In the second type, protection also relies on the protection and security measures provided by the DBMS (see 2.3.10).

1.1.4 Home directory and databases

As explained in the previous paragraph, FPQuest recognizes its databases by means of a **home directory**. This home directory can be different per Windows login, and can even be on a different computer in the network.

By default, FPQuest will install its databases under the home directory. However, a database can also be located in a different directory, and even on a different computer in the network. What is important is that in the home directory, a **Database descriptor file** is present for each database. These files have the name of the databases with the extension "**.dbs**". The ***Database*.dbs** file is a pure text file which can be edited in Notepad or any other text editor.

The line after [BACKCOL] contains the RGB values for the window background color, and the line after [SAVELOGFILE] indicates whether log files are saved or not.

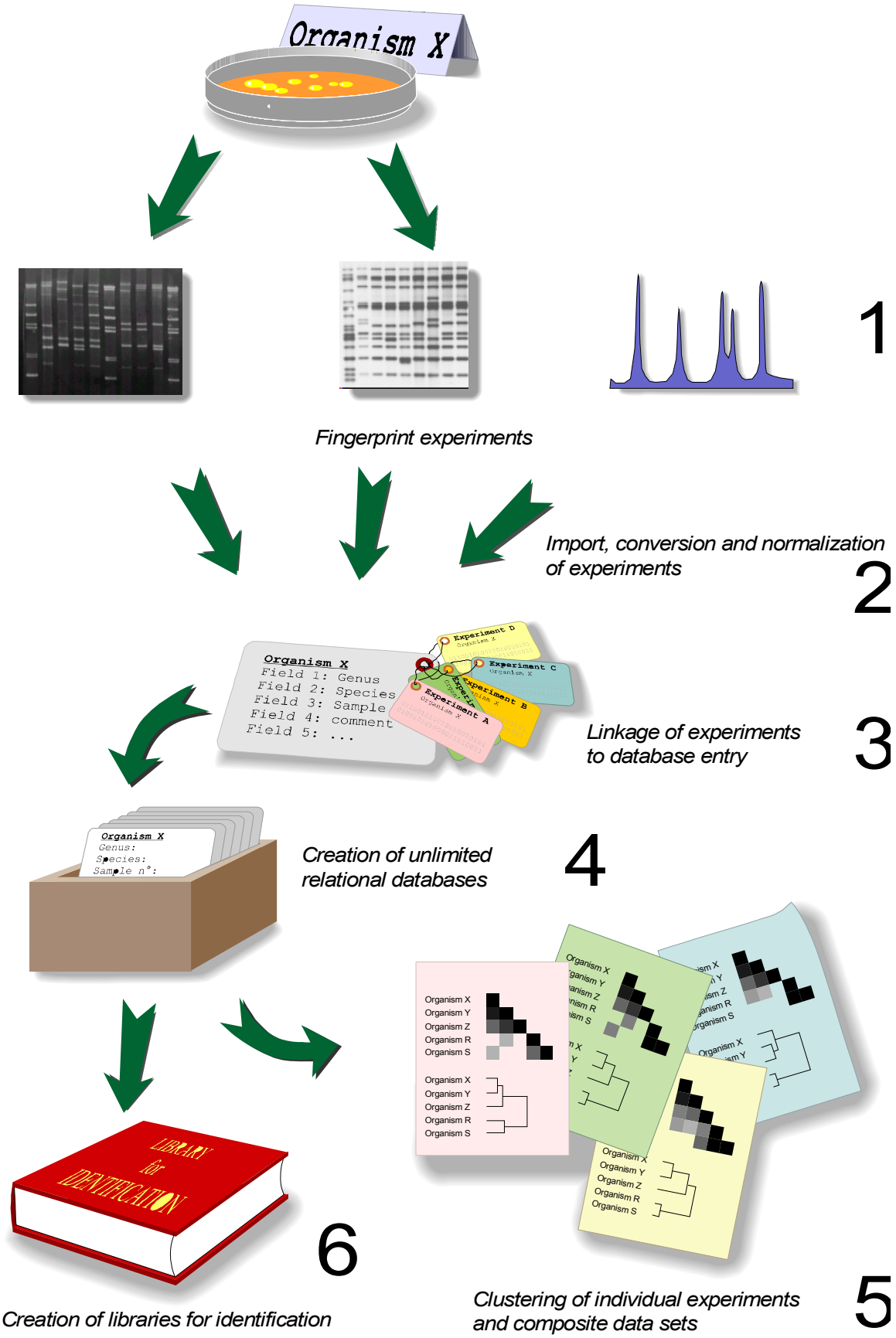


Figure 1-1. flow chart of conversion and import of TIFF and densitometric files.

For databases created in a FPQuest version prior to version 5.0, the line after the tag [DIR] indicates the full path where the database is located.

```
[DIR]
C:\Program Files\InfoQuestFP\data\DemoBase
[BACKCOL]
150 171 172
[SAVELOGFILE]
0
```

In databases created in version 5.0, the full path is replaced by a [HOMEDIR] tag. This tag points to the home directory as defined in the Startup screen. Because the database paths in version 5.0 are stored relatively with respect to the home directory, databases can easily be copied to other locations or computers. After copying the database(s) (and their .dbs files) to another location, you only need to change the home directory (see 1.5.2.1).

```
[DIR]
[HOMEDIR]\DemoBase
[BACKCOL]
150 171 172
[SAVELOGFILE]
0
```

NOTES:

(1) If a database, created in a FPQuest version prior to version 5.0, is moved from one computer to another, you need to edit the .dbs file and enter the correct path. The correct path for a database can also be entered from the Startup screen by pressing  *and selecting* ***Database settings*** *and* ***<Change directory>***.

(2) If you are working in FPQuest version 5.0 with databases created in a prior version, it may be useful to replace the paths in the .dbs files by a [HOMEDIR] tag. A script is available to store the paths relatively with respect to the home directory. Contact Bio-Rad to obtain this script.

(3) In case a database has been physically removed (or moved) from a computer, the ****Database*.dbs*** *file may still be present in the home directory, which causes the FPQuest Startup program to list the database. When attempts are made to open or edit such a removed database, FPQuest will produce an error. The only remedy is to delete the* ****Database*.dbs*** *file.*

1.1.5 Modules and features

The FPQuest software consists of an **application module** for the analysis of *Fingerprint types* FP (referred to as the *Basic Software*) and five **analysis modules**:

- The *Comparison and cluster analysis* module CL allows the user to create comparisons (see Chapter 4) and groups all functionality regarding cluster analysis.
- The *Identification* module ID allows the user to identify unknown entries using the database or identification libraries (see Chapter 5).
- The *Dimensioning and Statistics* module DI offers several non-hierarchical grouping methods, such as Principal Component Analysis and Multidimensional Scaling (see Section 4.8). In addition, it comprises powerful statistics tools (see Section 4.9).
- The *Comparative quantification* CQ module allows the user to perform a quantification of bands based on two-dimensional TIFF images, e.g. for use in band matching tables.
- The *Database sharing tools* module DS allows researchers to exchange data with other institutions using bundles and XML files (see Section 2.3).

For each section in this manual where specific features are described, the required modules will be indicated in the section title.

The specific FPQuest package you are working with might not include all modules. To check which modules are present, proceed as follows:

1.1.5.1 In the *FPQuest main* window (see Figure 1-15), select ***File > About FPQuest***.

A window pops up, showing the version of the software, the package serial number, and a list with modules (see Figure 1-2). A module is present in the installed FPQuest package when the module name is preceded by a hyphen.

1.1.6 The FPQuest script language

FPQuest is a comprehensive software package which has many data import, export and analysis functions already included in the software. Additional functionality - often related to specific applications - is bundled

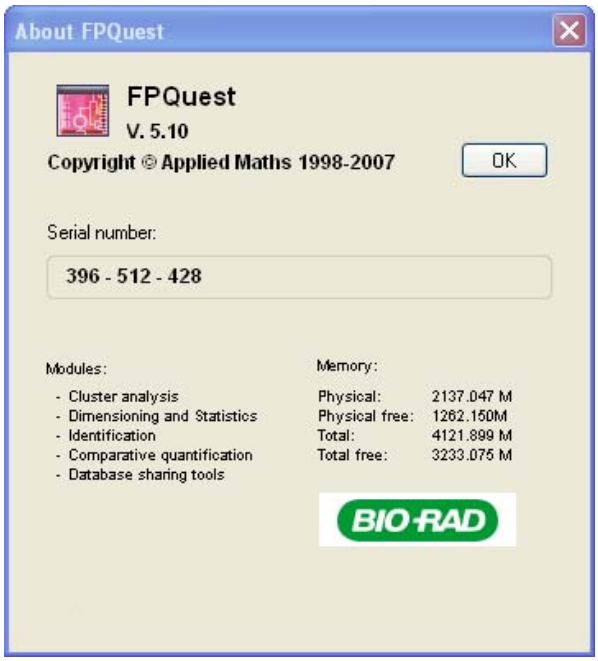


Figure 1-2. Window containing information about the installed FPQuest package.

into convenient plugins (see 1.5.3). However, ultimate flexibility is offered by FPQuest's own script language.

The FPQuest script language is a programming language, containing a large array of specialized functions, that allows the user to create scripts (i.e. small programs) for automation of specific tasks e.g. the import of own data formats.

A script editor can be opened from the *FPQuest main* window by selecting ***Scripts > Edit script***. For a description of the *Script Editor* window and an explanation of the numerous script functions available, we refer to the separate script manual.

NOTE: The functionality to edit scripts is available in any FPQuest configuration, but depending on the software configuration (modules present or not, see 1.1.5), some script functions might be disabled.

A number of general scripts are available on the website of Bio-Rad. These scripts can be launched from the *FPQuest main* window, using ***Scripts > Browse internet*** or by pressing the  button. In the browser window that appears, click on a category to display the relevant scripts. When leaving the  checkbox in the browser toolbar unchecked, a script can be executed directly over the internet by clicking on its name (recognized by the preceding  icon). When the checkbox is checked (), you will be prompted for a destination folder. Scripts can be saved in any folder, but two locations are predestined: Saving the script in **C:\Program files\FPQuest\Scripts** makes the script available as a menu item in the ***Scripts*** menu of any FPQuest database. To make a script only appear as a menu item in a specific database, save the script in the corresponding **[HOMEDIR]*dbname*\Scripts** folder.

NOTE: Using **Scripts > Run script from file**, *a script can be executed from any location.*

1.2 About this guide

1.2.1 Conventions

In the sections that follow, all menu commands and button text is typed in ***bold-italic***. Submenus are separated from parent menus by a "greater than" sign (>). Button text is always given between < and > signs.

For example, the following menu command (Figure 1-3) will be indicated as ***Edit > Arrange entries by field***.

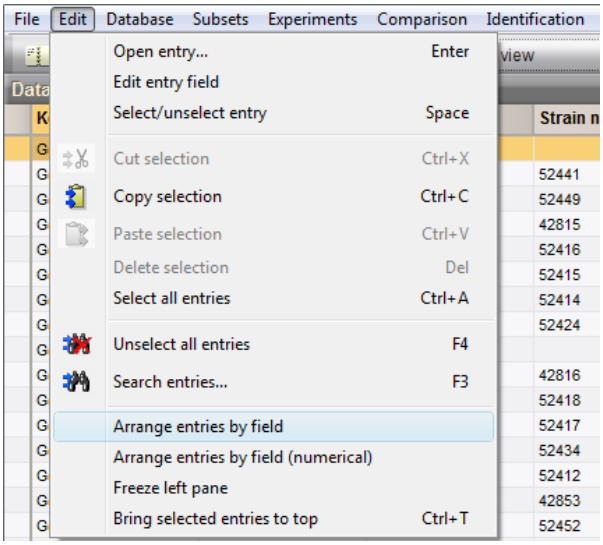


Figure 1-3. Illustration of the menu command *Edit > Arrange entries by field*.

In Figure 1-4, the following buttons are indicated as ***Inverted values*** (check box), ***<OK>***, ***<Cancel>***, ***<Apply>*** (disabled).

Each window and dialog box described in this 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 in Figure 1-4 will be called the *Fingerprint conversion settings* dialog box.

Descriptive text, such as explaining the layout of windows, describing the function of available menu items and buttons, or providing background information on the use of different algorithms, etc., is always displayed in normal text layout (such as the present paragraph), without preceding paragraph number. Tutorial text, which guides the user through the program by applying the available analysis functions on example data, are always preceded by a paragraph

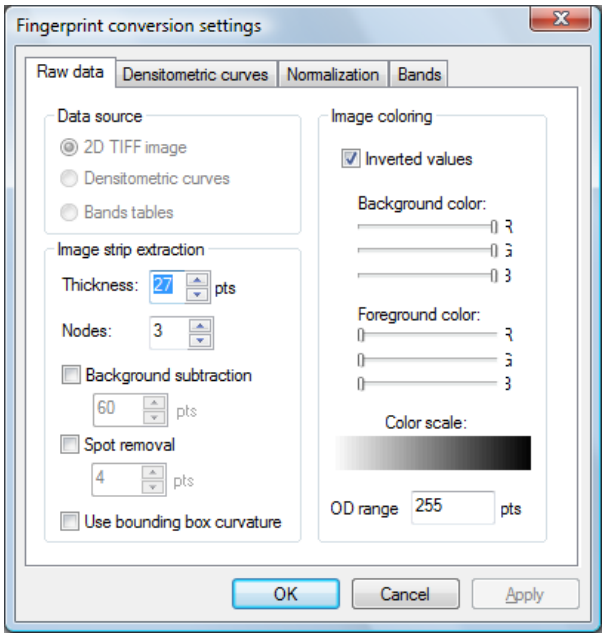


Figure 1-4. Illustration of buttons and check boxes in the *Character type settings* dialog box.

number for easy reference. This is illustrated in the following example:

1.2.1.1 In the Startup screen, double-click on **DemoBase** to open the database for analysis. Alternatively, you can select **DemoBase** from the list and then press the  button.

Names of databases (e.g. **DemoBase**) and experiment types (e.g. **RFLP1**) in FPQuest are typed in **bold** face.

1.2.2 Toolbars

In almost every window in the FPQuest Analyze program, there is a toolbar containing buttons for the most common functions available in the window. Placing the mouse pointer on a button for one second invokes a tool tip to appear, explaining the meaning of the button.

1.2.3 Floating menus

In almost every window in the FPQuest 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 1-5).

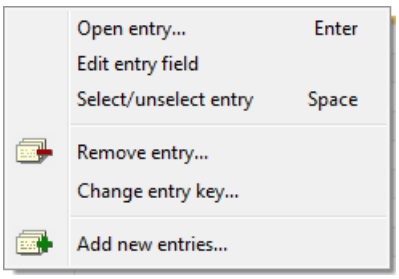


Figure 1-5. Floating menu appearing after right-clicking on a database entry.

The floating menus make the use of FPQuest 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.

1.3 Installing the software as a standalone license

1.3.1 The FPQuest Setup program

The FPQuest software is delivered on CD-ROM or can be downloaded from the Bio-Rad website (www.bio-rad.com/softwaredownloads).

1.3.1.1 Insert the protection key (dongle) in the parallel or USB port of the computer.

1.3.1.2 If you insert the CD-ROM in the drive, the Setup program will automatically load if the *Auto insert notification* of the CD drive is enabled. If not, or if you have downloaded the setup files from the website, run **Setup.exe**.

1.3.1.3 On the Setup intro screen (Figure 1-6), click ***Install***.

The *Installation* dialog box (Figure 1-7) allows you to change the ***Installation directory***, to specify the ***Database home directory, Install the Sample database***, and to Install the ***Netkey Server program***.

- **Installation directory:** By default, the software installs itself in a subdirectory FPQuest of the Program files directory. To change the installation path, click the ***<Browse>*** button and navigate to another path.

- **Database home directory:** The program offers two default options for the location of the home directory: ***In Common Documents*** and ***In My Documents***. The option ***In Common Documents*** makes the FPQuest databases available for all users on that computer. The option ***In My Documents*** makes the databases only accessible to the current user. The third option,

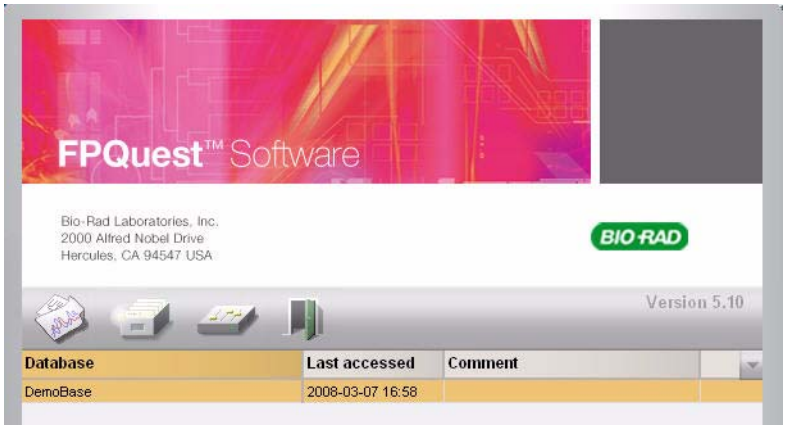


Figure 1-6. The FPQuest Setup screen.

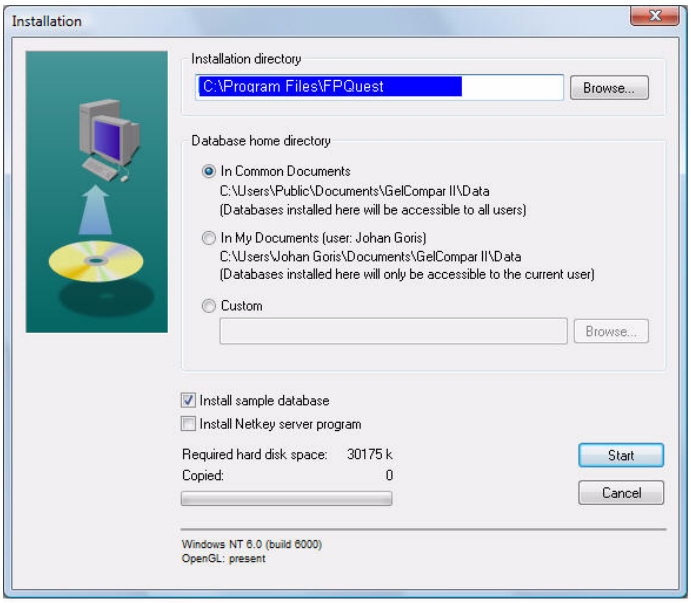


Figure 1-7. The *Installation* dialog box.

Custom folder, allows the user to specify any desired directory. You can even specify a directory on a network drive, on condition that this drive is permanently available.

- **Install sample database:** This option installs the sample database **DemoBase** with the software.

- **Install Netkey server program:** see 1.4.2.

1.3.1.4 If this is the first installation of FPQuest, you should allow the program to install the sample database. Do not check the ***Install Netkey server program*** checkbox if you are installing a standalone version of the software.

When file copying is completed, the Setup program prompts to create a shortcut for FPQuest on the desktop (recommended).

Upon completion, the installation program prompts you to confirm the installation of the drivers for the protection key.

1.3.1.5 If you are not installing a network license (see Section 1.4 for the installation and features of FPQuest network licenses), press ***<OK>*** to allow the installation of the dongle drivers.

1.3.1.6 If you allowed the setup program to create a shortcut on the desktop, double-click on the FPQuest icon to start the software. If not, open the Start menu and select ***FPQuest*** under ***All programs***.

When FPQuest is started for the first time, it will prompt you to enter the *License String*. The License String is required to activate your license and has to be entered only one time after a new installation and again after the installation of an upgrade. It is stored in the Windows registry.

The License String is provided on the jewel case of the CD-ROM, or in case of an upgrade or an Internet license, you may have obtained it electronically.

Enter the 6 x 4 characters License String in the input fields and press ***<OK>***. The *FPQuest Startup* program now appears (see 1.5.1).

1.3.2 Example database

One database, **DemoBase**, is installed 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 fingerprint types:

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

- **AFLP:** Amplified Fragment Length Polymorphism profiles (AFLP), run on an ABI PRISM 310 Genetic Analyser (Applied BioSystems).

The **DemoBase** database which is installed with the software, is a *local* database (see 2.1.1 for more information on local and connected databases). A number of more advanced features in FPQuest are only available for connected databases. Therefore, the data contained in **DemoBase** is also provided as a connected database. Since the ODBC connection is dependent on the computer configuration, the data are provided as a **DemoBase_SQL.mdb** file on the installation CD-ROM. Proceed as follows to install this database:

1.3.2.1 In the FPQuest Startup program (see 1.5.1), press the  button to create a new database.

1.3.2.2 The *New database* wizard (see 1.5.2 for a detailed description) prompts for a database name, enter **DemoBase_SQL**.

1.3.2.3 Leave all settings at their defaults (press the ***<Next>*** button twice) and press ***<Finish>*** to complete the setup of the new database.

1.3.2.4 Leave the default option ***New connected database (automatically created)*** enabled and press ***<Proceed>***.

1.3.2.5 Press ***<Proceed>*** in the *Plugin installation* toolbox, without installing any plugin (see 1.5.3 for more information on plugins).

The FPQuest home directory (as defined in the Startup program, see 1.5.2) should now contain a database folder called **DemoBase_SQL**.

1.3.2.6 In Windows explorer, simply replace the automatically generated **DemoBase_SQL.mdb** file from the **DemoBase_SQL** folder with the **DemoBase_SQL.mdb** file from the installation CD-ROM.

When the connected database **DemoBase_SQL** is now opened from the FPQuest Startup screen, it will contain the same example data as available in the local database **DemoBase**. Any tutorial paragraph in this manual that uses **DemoBase** is also applicable for **DemoBase_SQL**.

Additional example data are available on the installation CD-ROM, in the directory **Sample and Tutorial data**. Alternatively, the same data can be downloaded from the Bio-Rad website (www.bio-rad.com/softwaredownloads). Navigate to the download page and click on "Sample data" in the left menu to display a list with available data. Click on a list item to download the .zip file and unzip the sample data to a destination folder of your choice.

1.4 Installing the FPQuest network software

1.4.1 Introduction

The FPQuest network software is compatible with any TCP/IP supporting network in combination with Windows 2000, Windows NT 4.0, Windows XP and Windows Vista. The communication is based on TCP/IP sockets provided by Windows.

What we call a *server* is the computer that manages the network licenses. The *server* may be any Windows 2000, XP, NT 4.0, or Vista computer in the network. On the server computer, a *server* program called **Netkey** manages the network licenses. The *clients,* running FPQuest, 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 FPQuest 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 FPQuest 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!***

1.4.2 Setup

The License server computer has the *security key* inserted in the LPT1 or USB port and runs the **Netkey** *server program*, which manages the network licenses. All computers connected to the network can have the FPQuest *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, XP Professional or Vista is to be preferred, only for reasons of stability of the operating system.

Once the License server is located, install FPQuest on the License server. When installing FPQuest on the server, you should check the option ***Install Netkey server program*** in the *Startup* wizard (see Figure 1-7)

After installation of the *server program* on the *server* computer, install FPQuest as a standalone license on one or more client computers in the Network (see Section 1.3). Start perhaps with installing FPQuest on just one client computer and take note of all steps needed to configure the network software.

Do not forget to plug the security key (dongle) into the parallel or USB port of the License server computer!

After installation on the server, the following programs are installed under ***Program files > FPQuest***:

- FPQuest
- Netkey

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

NOTE: With Windows Vista as operating system, the Netkey program should be ran as administrator in order to run a service. To do so, select ***NetKey.exe*** *in*

Windows explorer, right-click on it and select "Run as administrator" from the drop-down menu.

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 FPQuest 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 FPQuest 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 FPQuest Startup program by pressing  and selecting ***License settings***. In the *License 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, FPQuest has created a settings file NETKEY.INI, which needs to be completed for the network. Run the FPQuest Startup program on the server, click on  and select ***License settings***. 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 FPQuest to find the server in the network.

The **SERVERPORT** is the TCP port that is used by the Netkey server and the clients to communicate with each other, and thus should be the same on all computers. In normal circumstances, you can leave **SERVERPORT** unchanged. However, in case there is a firewall between the Netkey server and the clients, or in case the client and/or server computer has a software firewall installed, you will have to open the specified TCP port. Any other port number can be specified, as long as the port number is correctly indicated in the SERVERPORT line, both on the Netkey server computer and on the clients.

Start the **Netkey** configuration program on the server computer. The following window appears (Figure 1-8).

Initially, the panel 'Registered computers' is empty and the panel 'Current connected users' does not appear, but a message “Unable to connect to the NetKey service” appears. This is because the service has not been started up yet.

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 bottom panel now lists the currently connected users (empty; see Figure 1-8). The Netkey service is now ready to distribute licenses.

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

•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 FPQuest 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

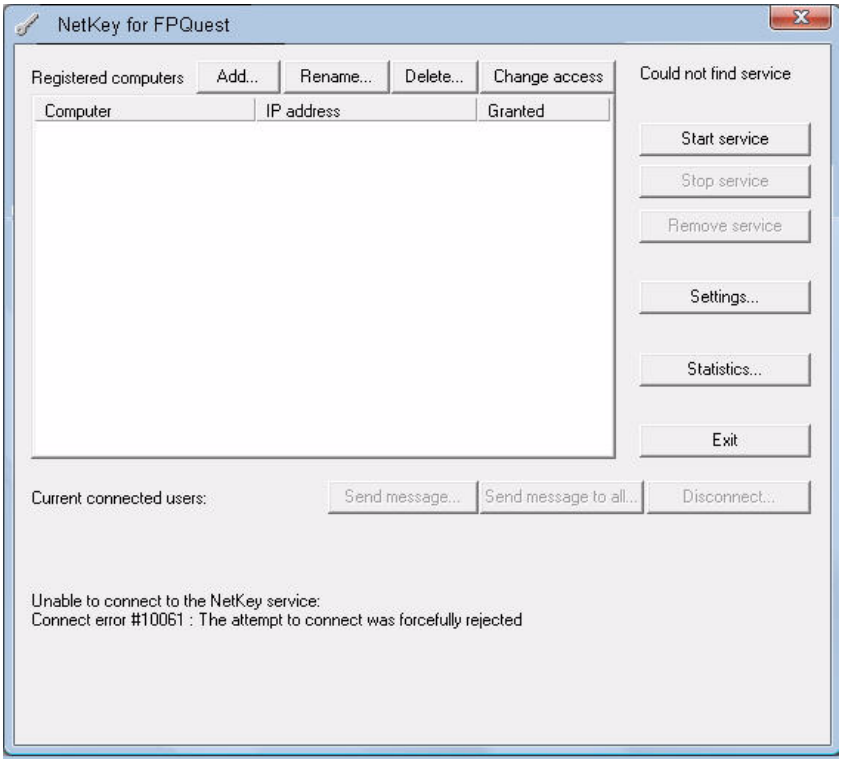


Figure 1-8. The Netkey configuration program, initial view.

(permanent). From this point on, the client has access to the FPQuest network software.

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

•Running FPQuest

On the client computer, open a database in the Startup program. FPQuest 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 FPQuest is now listed in the lower panel, showing its IP address, DNS host name, total usage time (elapsed) and idle time (1.4.3) (Figure 1-9).

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.

1.4.3 Advanced features of the Netkey server program

The Netkey server program is a Windows 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 loose his license in favor of the first in the waiting list. Specify a maximum idle time with ***<Settings>***, and enter the

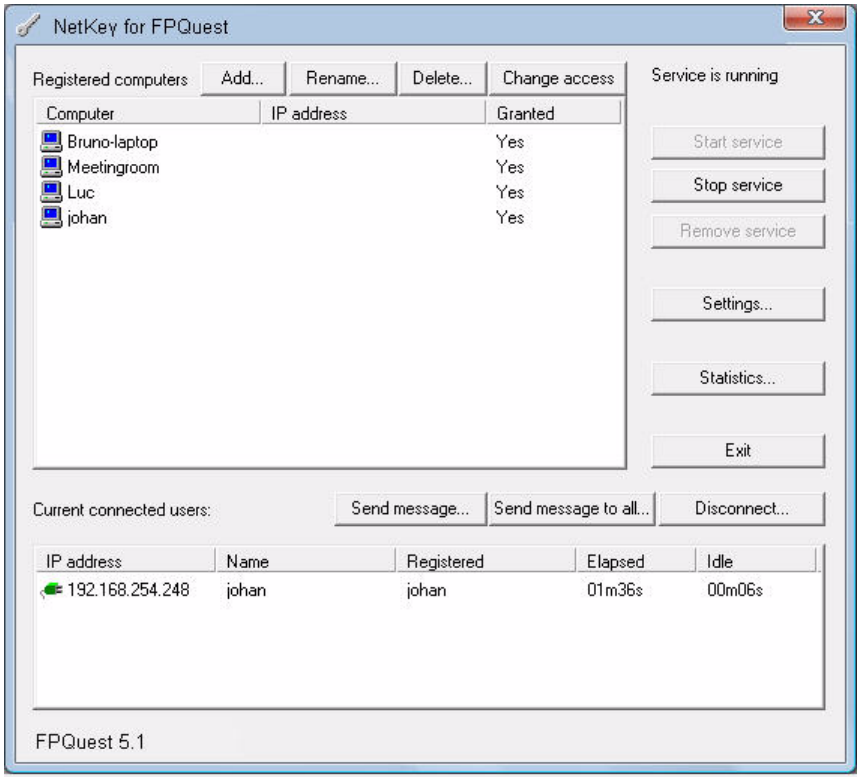


Figure 1-9. The Netkey configuration program, listing all computers that are granted access (top) and licenses in use (bottom).

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 loose 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 press ***<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 FPQuest 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.

1.4.4 Features of the client program (FPQuest)

•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 FPQuest software pops up as soon as the client's license becomes available.

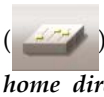
The user can request an overview of the computers currently using a FPQuest license by pressing  in the FPQuest Startup program, selecting ***License settings*** and then clicking **<*Status*>**. It shows for each connected computer the IP address, the computer name, the total usage time and the idle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

•Disconnection by server or license loss

If the client is disconnected by the server or looses 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. FPQuest 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.

1.5 Starting and setting up FPQuest

1.5.1 The FPQuest Startup program

1.5.1.1 Double-click the FPQuest icon on the desktop to run the **Startup program.** This program shows the *Startup screen* (see Figure 1-10). It allows you to run the FPQuest main application with  , to create new databases (), and customize various settings () such as the home directory (with ***Change home directory***), the directory of a selected database (with ***Database settings***) and the license and network settings (with ***License settings***).

1.5.1.2 Use the  button when you are finished running the FPQuest applications.

1.5.2 Creating a database

In order to facilitate the use of FPQuest in different research projects, it is possible to set up *Databases*. The principles of a database are explained in Section 2.1. The FPQuest Startup program will look for all databases in one *home directory*, specified by the user. Note that, in Windows 2000, Windows XP, and Windows Vista, each Windows user may specify a different home directory. The FPQuest home directory is saved with the system registry of the user.

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

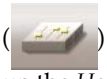
1.5.2.1 In the Startup screen, press the settings button () and select ***Change home directory***. This pops up the *Home directory* dialog box (Figure 1-11).

Figure 1-10. The FPQuest Startup screen.

The program offers two default options for the location of the home directory: ***In Common Documents*** and ***In My Documents***. The option ***In Common Documents*** makes the FPQuest databases available for all Windows users on that computer. The option ***In My Documents*** makes the databases only accessible to the user currently logged on. The third option, ***Custom folder***, allows the user to specify any desired directory. You can even specify a directory on a network drive, on condition that this drive is permanently available with write-access.

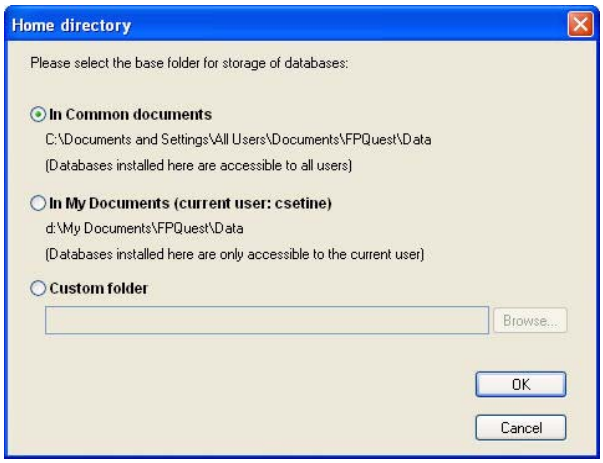


Figure 1-11. The *Home directory* dialog box.

1.5.2.2 Press **<*OK*>** to select the new home directory. The program updates the list of available databases in the new directory.

Create a new database as follows:

1.5.2.3 Press the  button to enter the *New database* wizard.

1.5.2.4 Enter a name for the database, e.g. **Example**, and press **<*Next*>**.

NOTES:

(1) The program automatically creates a folder within the current home directory for storage of the database-specific files and folders (in this example: **[HOMEDIR]\Example***). If you want to change this (not recommended), press* **<Browse>***. 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.*

(2) If you do not want the program to automatically create subdirectories, click **No** *to this question (not recommended, unless the subdirectories already exist). In that case, you will have to create the subdirectories manually (see Figure 2-2).*

1.5.2.5 Press **<*Next*>** again.

1.5.2.6 You are now asked whether or not you want to create log files. If you enable FPQuest 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. For more information on log files, see 2.1.6.

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

1.5.2.8 A new dialog box pops up, prompting for the type of database (see Figure 1-12).

FPQuest offers two alternative database solutions to store its databases: the program's own built-in database (= **local database**) or an external SQL and ODBC compatible database engine. The latter solution is called a **connected database**. Because of the multi-user access and the extended range of features only available in connected databases, the creation of a connected database is the default option in FPQuest version 5.0. The use of connected databases and the different options (see Figure 1-12) are discussed in Section 2.1.

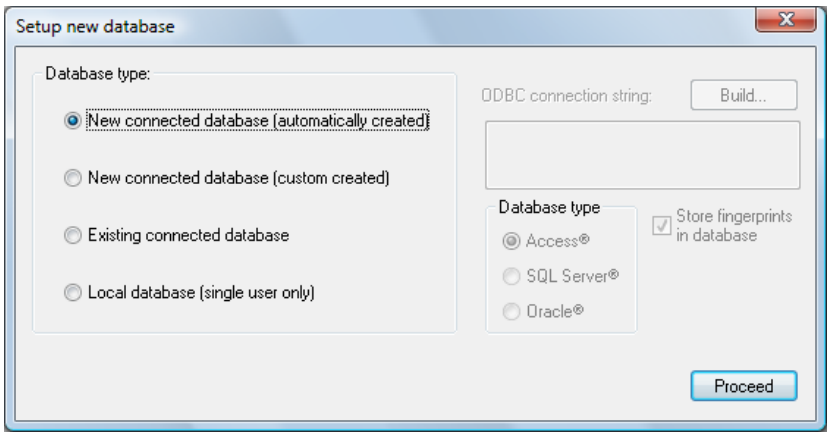


Figure 1-12. *Database selection* dialog box.

1.5.2.9 For this example, leave the default option ***New connected database (automatically created)*** enabled and press ***<Proceed>***.

1.5.2.10 The *Plugin installation* toolbox appears. For more information on the installation of plugins, see 1.5.3.

1.5.2.11 Press ***<Proceed>*** to start working with the newly created database without installing any plugins.

1.5.3 Installing plugin tools in a new database

When a new database is opened for the first time (1.5.2.10), FPQuest will provide the opportunity to install *Plugin tools*. Plugins are tools written in the FPQuest script language, available as binary encoded packages. The plugins offer extra functionality, often to import or export various types of data. Plugins can also provide extra functions related to dendrogram analysis, statistics, database management, etc.

1.5.3.1 Open the database **DemoBase** by selecting **DemoBase** in the Startup program, and click on the  button. Simply double-clicking on the database name does the same.

If this is the first time the database **DemoBase** is opened, the *Plugin installation* toolbox will appear, as illustrated in Figure 1-13. If not, the database will open without showing this toolbox. In that event, you can still open the *Plugin installation* toolbox from the FPQuest *FPQuest main* window (see Figure 1-15) with ***File > Install / Remove plugins*** or by pressing the  button.

The listbox (left) shows the plugin tools that are present in the installation folder of FPQuest.

1.5.3.2 With ***<Check for updates>*** the latest versions and/or new plugins are downloaded from the Bio-Rad website and shown in the listbox.

WARNING: Pressing the ***<Check for updates>*** button will update ANY plugin for which a newer version is available.

1.5.3.3 When a particular plugin is selected, a short description appears in the right panel along with a version number.

1.5.3.4 A selected plugin can be installed with the ***<Install>*** button.

1.5.3.5 Once a plugin is installed, it is marked with a green V-sign. It can be removed again with the ***<Remove>*** button.

1.5.3.6 If a manual exists for the plugin, a PDF manual is opened in Adobe Acrobat Reader with ***<Manual>***.

1.5.3.7 If you have finished installing plugins, you can proceed to the *FPQuest main* window with ***<Proceed>***.

When installed, a plugin installs itself in a menu of the software, and is characterized by a plug icon left from the menu item. For example, if "**Fingerprint processing reports**" is installed, a new item ***Print TIFF image*** becomes available in the ***File*** menu of the *Fingerprint data* window (see Figure 1-14).

At the time of writing of this manual, the following plugins were available:

- **Database tools:** Offers additional search functions (fuzzy search, find and replace) and database layout tools.
- **Dendrogram tools (CL):** Contains tools for working with dendrograms in the *Comparison* window.

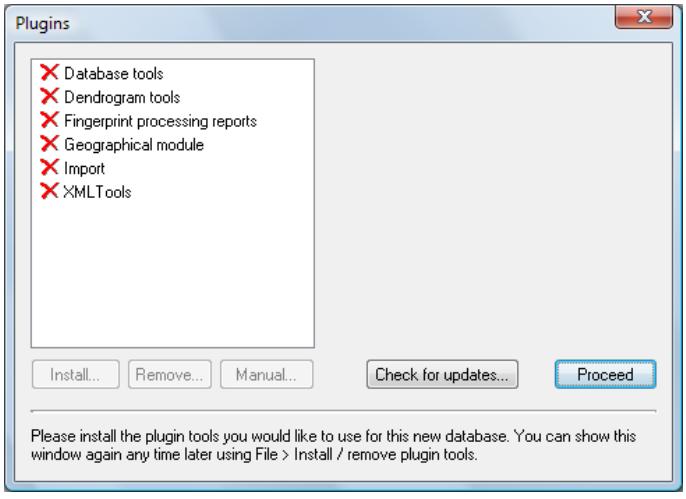


Figure 1-13. The *Plugin installation* toolbox.

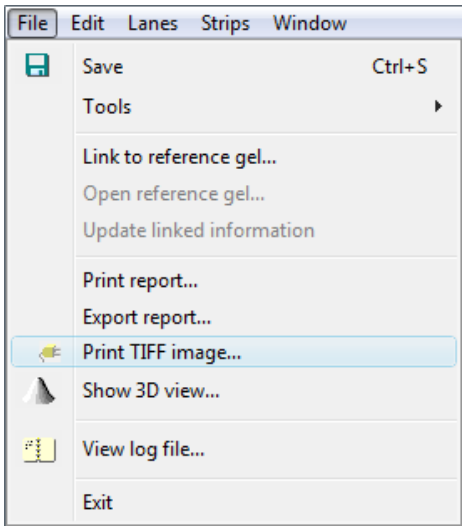


Figure 1-14. Plugin menu commands are characterized by a yellow plug icon left from the menu item.

- **Fingerprint processing reports (FP):** Contains tools for exporting data from the *Fingerprint processing* window.

- **FPQuest Help plugin:** Contains a compiled help function for working with FPQuest, opens a PDF version of this manual and offers an easy link to the online FAQ database and the support question webform.

- **Geographical plugin (DS):** Plots database entries on a geographical map, e.g. for epidemiological investigations.

- **Import plugin:** Offers convenient import routines for the import of sequencer fingerprint files and GeneMapper (Applied BioSystems) peak files.

- **XML Tools plugin (DS):** Allows the import and export of database information as XML files.

IMPORTANT: Always check the *Plugin installation* toolbox after pressing ***<Check for updates>*** to find the latest versions and the most complete list of available plugins.

Some plugins depend on the functionality offered by specific modules (see 1.1.5), e.g. the XML Tools plugin requires the *Database sharing tools* module to be present. If a required modules is missing, the plugin cannot be installed and an error message is generated.

1.6 The FPQuest user interface

1.6.1 Introduction to the FPQuest user interface

FPQuest is a very comprehensive software package. For every specific task in FPQuest, a **window** is available that groups the relevant functionality for that specific task. From an active window, dialog boxes and subwindows can be launched. Each window at its turn consists of several **panels** containing specific information. The FPQuest user interface is very flexible and can be customized by the user at three different levels:

1. The behaviour and general appearance of windows can be set.

2. For each separate window, the toolbars and panels that are displayed can be chosen, as well as the size and the location of panels.

3. In grid panels, columns can be displayed or hidden and the relative position of rows and columns can be chosen.

The FPQuest user interface will be explained using the example database provided with the software, called **DemoBase** (see 1.3.2 for a description of the data available in this database).

NOTE: In comparison with previous versions, the user interface of FPQuest 5.0 is completely renewed. The major changes include: direct editable information fields, control over how the windows stack appears, preset color schemes, font types, dockable panels, toolbars that can be displayed or hidden, and zoom sliders. Even for the experienced FPQuest user, it might be useful to read this chapter to take full advantage of these new features.

1.6.2 The *FPQuest main* window

1.6.2.1 Open the database **DemoBase** by selecting **DemoBase** in the Startup program, and click on the  button. Simply double-clicking on the database name does the same.

If the *Plugin installation* toolbox appears instead of the database (Figure 1-13), it means that you are opening this database for the first time. See 1.5.3 for further explanation on the installation of plugin tools.

The *FPQuest main* window in default configuration (Figure 1-15) consists of a menu, a toolbar for quick access to the most important functions, a status bar, and the following eight panels:

- The *Database entries* panel, listing all the available entries in the database, with their information fields and their unique keys (see 1.1.2). A local FPQuest database can contain up to 200,000 entries. A connected database can contain many more entries, but only 200,000 can be displayed in one view.

- The *Experiments* 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 (colored dot) or not. Clicking on a colored dot causes the *Experiment card* for that experiment to be popped up (see Figure 3-40).

- The *Files* panel, showing the available data files for the experiment type selected in the *Experiments* panel, with their date of creation, the date when the files were last modified and their location (Local or Shared for local database or connected database, respectively).

- The *Comparisons* panel, listing all comparisons that are saved, with their date of creation, the date when the comparisons were last modified and their location (Local or Shared for local database or connected database, respectively).

- The *Libraries* panel, which shows the available identification libraries and their location (Local or Shared for local database or connected database, respectively).

NOTES:

(1) In default configuration, the Libraries *panel appears as tabbed view together with the* Comparisons *panel, with the* Comparisons *panel displayed.*

(2) Unless otherwise stated, all screenshots in this manual are taken using default settings. If the FPQuest main *window is displayed differently on your screen than in the screen shot in Figure 1-15, then your current settings might be different from the default*

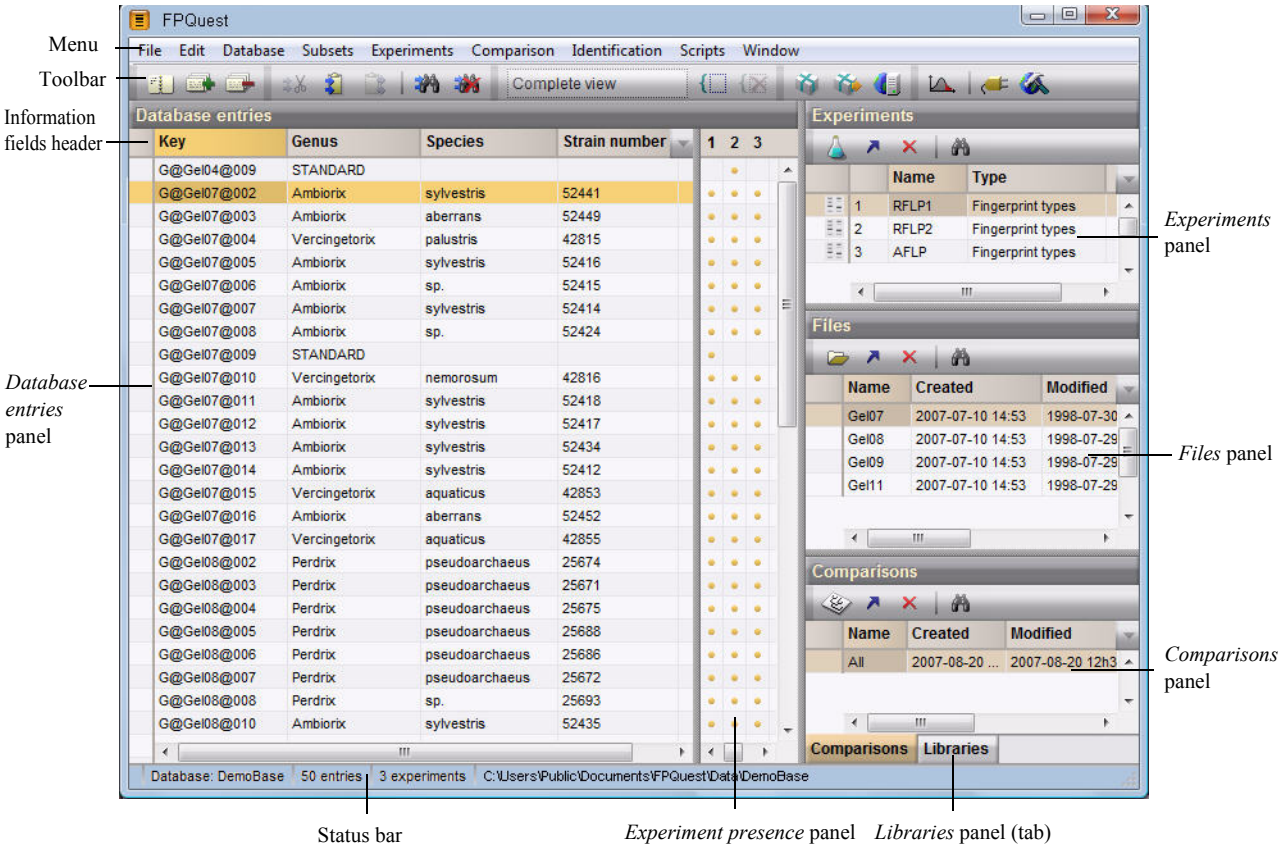


Figure 1-15. The *FPQuest main* window.

settings. How to return to the default settings is described in the next paragraphs.

1.6.3 General appearance of FPQuest windows

FPQuest offers the choice between eight preset color schemes, allows the adjustment of brightness and visual effects and lets the user select the font type. These general appearance settings have an effect on **all** FPQuest windows.

1.6.3.1 In the *FPQuest main* window, select ***File > Preferences***. The *Preferences* dialog box appears (Figure 1-16). From the list on the left side of the dialog box, select ***Windows appearance***.

From the pull-down list next to ***Color scheme***, a selection can be made from eight preset color schemes. The ***Brightness*** can be adjusted by entering a percentage. According to your own preferences, ***Glossy effects*** and ***Use alternating line colors*** can be either checked or unchecked.

1.6.3.2 Select any color scheme of your choice, try entering a different brightness percentage and/or uncheck any of the visual effects to notice their effect. Press **<*OK*>** to display the modified appearance of the *FPQuest main* window and other FPQuest windows. It might be necessary to restart FPQuest to enable the applied changes.

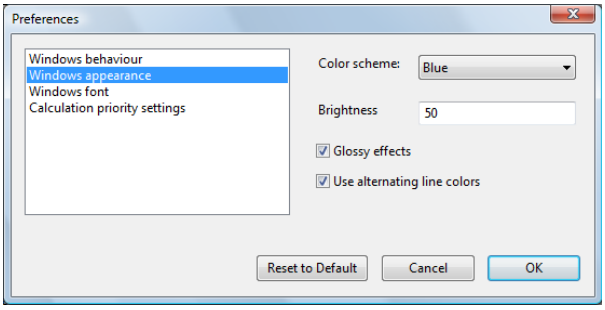


Figure 1-16. The *Preferences* dialog box, *Windows appearance* settings.

1.6.3.3 Call the *Preferences* dialog box again with ***File > Preferences*** and select ***Windows appearance*** from the list. Pressing **<*Reset to Default*>** will restore the default appearance settings: Orange and Gray color scheme, brightness 50% and both ***Glossy effects*** and ***Use alternating line colors*** enabled. Press **<*OK*>** to have the FPQuest windows displayed with default settings again. It might be necessary to restart FPQuest to enable the applied changes.

1.6.3.4 In the same *Preferences* dialog box, clicking on ***Windows font*** enables you to set the type and size of the font used in all panels. Pressing **<*Reset to Default*>** will restore the default font settings: font Arial, size 11.

NOTE: All preferences are saved at a database level, allowing the user to specify different preference settings for different databases.

1.6.4 Display of panels

FPQuest windows can be customized up to a high degree by the user. All panels can be resized to make optimal use of the display by dragging the horizontal separators between the panels up or down or by dragging the vertical separators left or right.

Two types of panels are available in FPQuest windows: fixed panels and dockable panels. The displayed information in **fixed** panels is indispensable in any type of analysis. Therefore, these panels are always displayed in their corresponding window. Depending on the nature of the experiment, the type of analysis performed and user preferences, the information displayed in **dockable** panels may not always be essential. Therefore, FPQuest offers the possibility to either show or hide dockable panels. If shown, the way dockable panels are displayed on the screen can be further customized. These features allows the user to hide infrequently used panels that would otherwise clutter the workspace. For example, if you do not have the *Identification* module, the *Libraries* panel in the *FPQuest main* window can be hidden for the sake of clarity. Several FPQuest windows contain dockable panels, which all behave identically. The principles are illustrated here for the *FPQuest main* window.

1.6.4.1 In the *FPQuest main* window, click on the *Libraries* tab to display the *Libraries* panel.

1.6.4.2 Select ***Window > Show / Hide panels*** in the *FPQuest main* window. This displays a submenu, listing all available panels. Panel names for which a check mark is present left of the menu item are shown in the *FPQuest main* window.

1.6.4.3 Click on ***Libraries*** in the submenu. The *Libraries* panel is now hidden from the *FPQuest main* window.

Furthermore, dockable panels can be placed on the screen in one of two modes: floating or docked. **Floating** allows the window to be placed anywhere on the screen, similar to a normal window of a base size (not maximized). The **docked** mode automatically places the panel in one of five locations: top, bottom, left, right, or stacked onto another panel (tabbed view). The position of a panel is controlled with a **docking guide**.

1.6.4.4 Click in the header of the *Files* panel and - while keeping the mouse button pressed - drag it upwards in the window. As soon as the panel leaves its original position, a docking guide appears in the center of the *Experiments* panel. Release the mouse button on any place next to the docking guide to leave the panel floating in the window.

A floating window can be repositioned to any place on your monitor.

1.6.4.5 Click in the header of the *Files* panel and drag it towards the *Experiments* panel again. Drop the floating panel on the top part of the docking guide that appears (see Figure 1-17). This action will make the *Files* panel appear above the *Experiments* panel in the *FPQuest main* window.

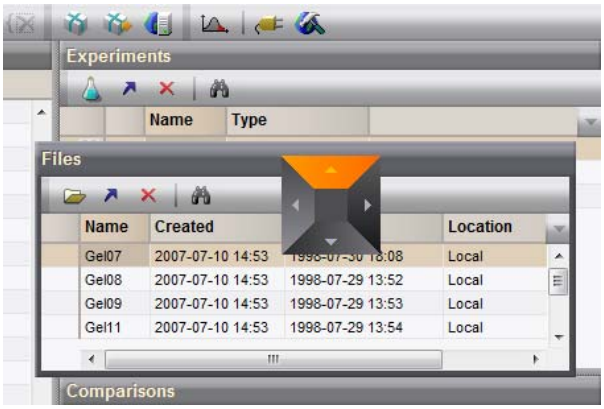


Figure 1-17. Docking the *Files* panel above the *Experiments* panel using the docking guide.

1.6.4.6 Click in the *Files* panel header and drag it towards the *Experiments* panel again. This time, drop the *Files* panel on the center of the docking guide (see Figure 1-18). As a result, the *Files* panel is now displayed as a tabbed view with the *Experiments* panel (see Figure 1-19).

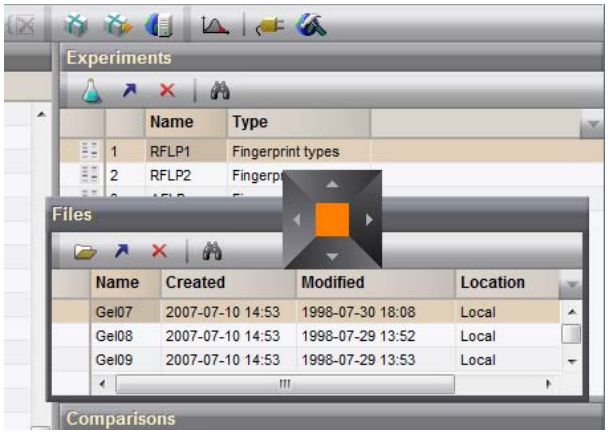


Figure 1-18. Docking the *Files* panel as a tabbed view with the *Experiments* panel.

NOTE: To re-locate a panel that is presently displayed as a tabbed view with other panels, click on the panel tab instead of the panel header to drag the panel to its new position.

After making some changes to the window configuration, it is always possible to return to the default configuration for the active window. This might be useful e.g. to make comparison with screenshots shown in this manual easier. If you intend to revert to the user-defined

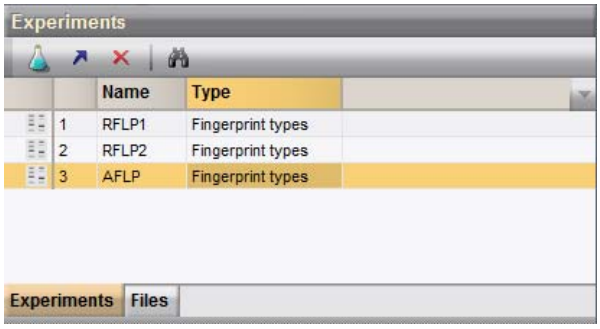


Figure 1-19. Result of the action depicted in Figure 1-18: tabbed view of the *Experiments* and *Files* panel.

configuration afterwards, then you can save the user-defined configuration first and recall it afterwards.

1.6.4.7 Select ***Window > Save current configuration*** to store the configuration that you have just defined.

1.6.4.8 To restore the default configuration, select ***Window > Restore default configuration***. The window now appears back in its original configuration.

1.6.4.9 Recall the user-defined configuration with ***Window > Recall saved configuration***. Notice that the changes you made to the window configuration are introduced again.

In case you do not wish to save the introduced configuration changes, steps 1.6.4.7 to 1.6.4.9 can be skipped:

1.6.4.10 Select ***Window > Restore default configuration*** to restore the default configuration of the *FPQuest main* window again.

Any window configuration can be protected from accidental changes via ***Window > Lock configuration***. A check mark is present in the menu left of ***Lock configuration*** if the configuration of the active window is locked. Configuration changes will be enabled if ***Window > Lock configuration*** is selected again.

1.6.5 Configuring toolbars

In addition to the pull-down menu's that are available for executing commands (see Figure 1-20 for an example), FPQuest also displays toolbars for frequently used commands. Toolbars consist of icons that are arranged in groups, according to their function. An example is the database toolbar,  which is located by default in the header of the *FPQuest main* window. Many panels have their own toolbar, grouping panel-specific commands. Toolbars can either be displayed or hidden, a feature which allows the user to hide infrequently used toolbars.

NOTE: When using Microsoft Windows Vista as operating system, the corresponding toolbar icons

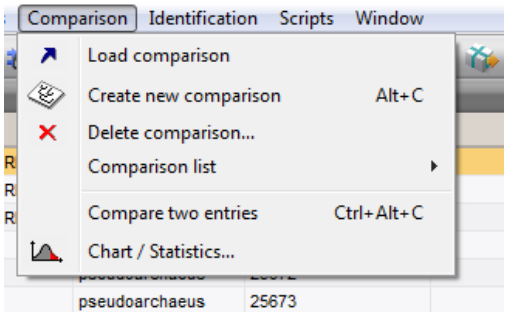


Figure 1-20. The pull-down menu *Comparison*.

appear left of the menu items as a visual aid (see Figure 1-20). With earlier operating systems, toolbar icons are not displayed in the pull-down menus.

1.6.5.1 In the *FPQuest main* window, select ***Window > Show / Hide toolbar*** and, for example, click on ***Database*** to hide the Database toolbar.

When the toolbar is displayed, a check mark is present next to the corresponding menu item. Toolbars specific to certain panels are listed under the corresponding submenu's.

1.6.5.2 Select ***Window > Show / Hide toolbar > Files panel*** and click on ***File tools*** to hide the toolbar specific for the *Files* panel.

The position of a toolbar within a window or panel can also be altered.

1.6.5.3 In the header of the *FPQuest main* window, click on the dark gray area in a toolbar, left of a set of icons. The mouse pointer will take the shape of a hand on top of two arrows. Drag the toolbar left or right to change the order in which the toolbars appear.

1.6.5.4 In the header of the *FPQuest main* window, drag another toolbar slightly downwards to make the toolbars appear in two rows.

1.6.5.5 Click on a toolbar again and drag it to the left (or right or bottom) part of the window to dock it on the left (or right or bottom) part of the window.

The position of panel-specific toolbars can be customized much in the same way as general toolbars, with the restriction that they cannot be positioned outside their corresponding panel.

Individual buttons can be hidden from their toolbars.

1.6.5.6 Right-click on any toolbar. A floating menu appears, listing all buttons of the toolbar (see Figure 1-21). By default, all button names are checked in the menu and the corresponding buttons will appear in the toolbar.

1.6.5.7 Select the button that you want to hide from the floating menu.

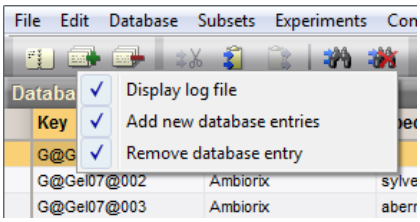


Figure 1-21. Floating menu for the Database toolbar, listing all available buttons.

The toolbar button can be displayed again by repeating the above actions.

As for the display of panels, the configuration of toolbars can be restored to default using ***Windows > Restore default configuration***. In case you want to save the current configuration, follow steps 1.6.4.7 to 1.6.4.9.

1.6.5.8 Select ***Window > Restore default configuration*** to restore the default configuration of the *FPQuest main* window again.

1.6.6 Grid panels

Grid panels contain data in tabular format, i.e. organized in columns and rows. In the *FPQuest main* window, examples are the *Database entries*, *Experiments*, *Files*, and *Comparisons* panels. All grid panels can be customized up to a high degree by the user. The width of columns can be changed by moving the separator line in the column heading left or right. Other column properties can be accessed via the column properties button , located on the right hand side in the information fields header. The column properties of the *Files* panel are illustrated in Figure 1-22. This pull-down menu contains a list of information fields that are available in the grid panel and which can either be displayed or hidden (check mark resp. present or absent).

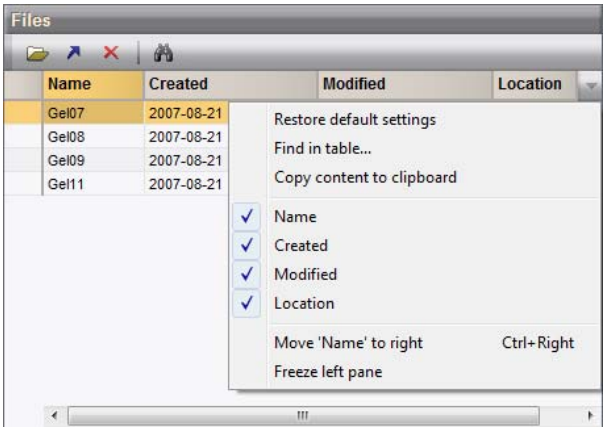


Figure 1-22. Column properties of the *Files* panel.

1.6.6.1 Click on the column properties button ( ) of the *Files* panel (see Figure 1-22) and click on ***Created*** to hide the column displaying the date of creation.

The relative position of a selected column within the panel can be changed using the menu items ***Move 'column_name' to left*** and ***Move 'column_name' to right***.

1.6.6.2 In the *Files* panel, select the column 'Location' and click on the column properties button ( ). Click ***Move 'Location' to left*** or ***Move 'Location' to right*** to shift the 'Location' column to the left or right, respectively. The shortcut keys CTRL+left arrow and CTRL+right arrow can be used for the same purposes.

The option ***Freeze left pane*** allows the user to freeze one or more information fields so that they always remain visible left from the scrollable area.

Similar as for the window configuration (see 1.6.4.10), it is possible to revert to the default column properties settings for the active panel.

1.6.6.3 Click on the column properties button ( ) of the *Files* panel and select ***Restore default settings*** to disable all introduced changes to the column properties of the *Files* panel.

A grid panel can be searched for the occurrence of a text string in any displayed information field:

1.6.6.4 Click on the column properties button ( ) of the *Database entries* panel and select ***Find in table.*** This pops up the *Find in table* dialog box (see Figure 1-23).

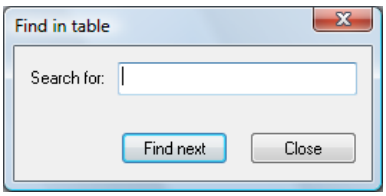


Figure 1-23. The *Find in table* dialog box.

1.6.6.5 Enter a search string, e.g. "verc" and press ***<Find next>*** repeatedly to find all occurrences of the entered search string in the displayed information fields (in this case, all Vercingetorix in the 'Genus' field).

The information contained in any grid panel can be exported to the clipboard for use in other programs:

1.6.6.6 Click on the column properties button ( ) of the *Database entries* panel and select ***Copy content to clipboard***.

1.6.6.7 Paste the content of the clipboard in e.g. Notepad to view the information that was copied from the *Database entries* panel.

In grid panels, rows can be sorted according to a certain field by right-clicking on the field (column) header and selecting ***Arrange ... by field***.

A number of predefined information fields are automatically created when creating a new database.

1.6.6.8 Select the column properties button ( ) of the *Database entries* panel.

The predefined information fields listed in the pull-down menu (see Figure 1-24) can either be displayed or hidden (check mark resp. present or absent).

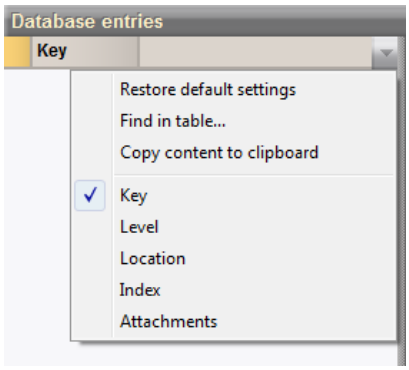


Figure 1-24. Automatically created information fields in the *Database entries* panel.

NOTE: For a local database, the default information fields Created and Modified follow standard Microsoft Windows behaviour. Therefore, if a file is copied from a different location, the moment of copying is taken as Created date and will be more recent than the date displayed in Modified.

In addition to these default information fields, extra information fields can be added with ***Add new information field*** or removed with ***Remove information field***. The menu commands can be accessed in each panel by right-clicking in their information field headers.

Information in non-default information fields can be edited by clicking twice (not double-click) on an item, or by pressing CTRL+Enter on the keyboard. The information then appears selected blue against a bright colored background and can be modified. This is illustrated in Figure 1-25, where information about gel staining is added to the *Files* panel.

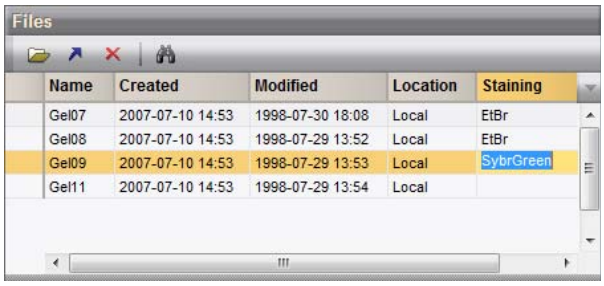


Figure 1-25. Editing information within non-default information fields, illustrated for the *Files* panel. Clicking twice on an information fields enables direct editing.

The *Database entries* panel behaves just as other grid panels, with a few peculiarities. Double-clicking on a database entry or pressing Enter opens its *Entry edit* window (see Figure 1-26). More information on this window is provided in 2.2.3.

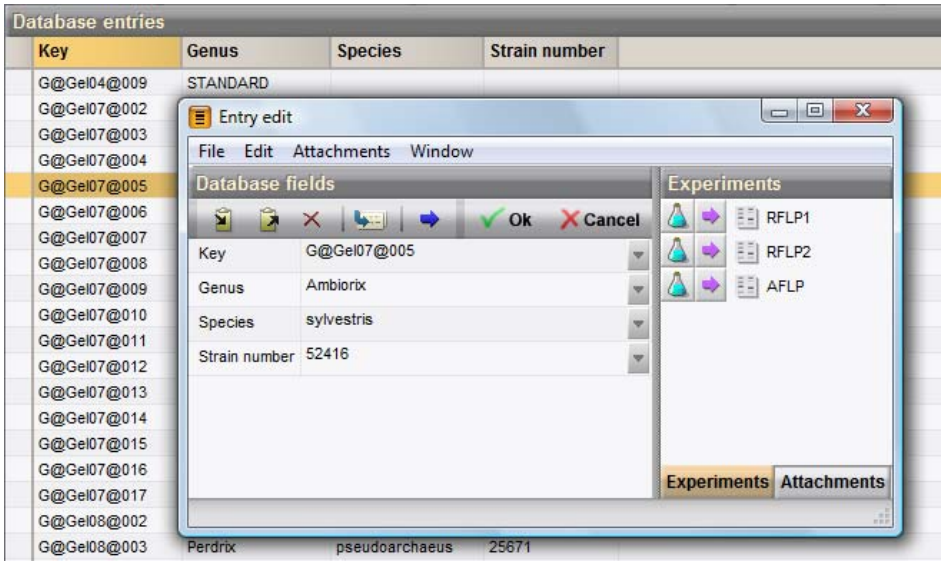


Figure 1-26. Using the *Entry edit* window to modify information fields.

If desired, the direct field editing behaviour can be modified.

1.6.6.9 Via ***File > Preferences > Windows behaviour***, check or uncheck ***Single click field editing*** to enable or disable information field editing after a single mouse click.

> *NOTE: When* **Single click field editing** *is enabled, the* Entry edit *window (see Figure 1-26) is opened by double-clicking in the margin or on the Key field of the database entry.*

For information fields in the *Database entries* panel, *properties* can be set (see 2.2.5 for more information). One of these properties includes a different background color for each field state. The extend of the background color can be set in the *Preferences* dialog box.

1.6.6.10 Select ***File > Preferences*** to call the *Preferences* dialog box and click on ***Windows behaviour*** in the list on the left hand side.

1.6.6.11 Uncheck ***Use color background for complete field*** (checked by default) to limit the background color to a small rectangle, preceding the information field content.

In the panels *Experiments*, *Files*, *Comparisons* and *Libraries* from the *FPQuest main* window, pressing the  button calls the *Field query* dialog box (Figure 1-27). This allows the list of available experiments, files, comparisons or libraries to be searched for name and any user-defined information field (if present). Items that match the search criteria are marked with a small colored triangle in the left column. When the option ***Bring selected to top*** is checked, the selected items appear on top of the list. Further options are available to replace currently selected items with new items, add the newly found items to the list, to search within the selection and to use regular expressions (see Section 6.2 on how to use regular expressions) in the search string.

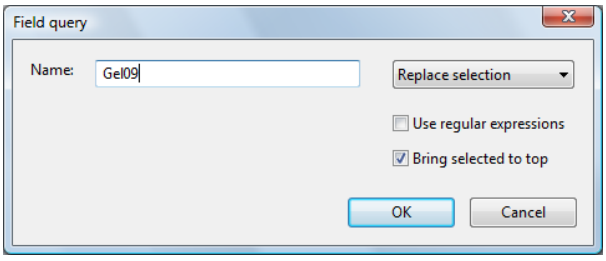


Figure 1-27. *Field query* dialog box for the *Files* panel.

1.6.7 Zoom sliders

In many FPQuest windows, panels containing graphical information can be zoomed in or out to make optimal use of the display. Zooming in or out can be done via the  and  buttons in the toolbar or via the corresponding menu commands. Shortcut keys for these actions are CTRL+PageUp and CTRL+PageDown, respectively.

In addition, graphical panels or windows in FPQuest are equipped with *zoom sliders* in the shape of a narrow vertical or horizontal pane, featuring a colored bar (see Figure 1-28 for an example). Increasing the bar size, by

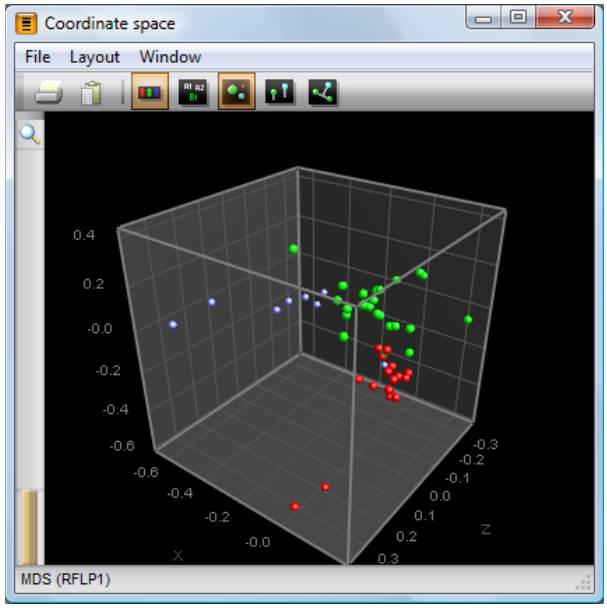


Figure 1-28. Zoom slider (left), illustrated for the *Coordinate space* window.

dragging it with the mouse, zooms in on the image. Decreasing the bar size with the mouse zooms out on the image. The zoom slider can also be operated by hovering over it and using the scroll wheel of the mouse. Alternatively, press the CTRL or SHIFT (in case more than one zoom slider is present) key on the keyboard and use the scroll wheel. Image proportions are maintained when (as in Figure 1-28) the  icon is displayed in the zoom slider. When the  and  icons are shown in the zoom sliders, horizontal and vertical zooming can be performed separately. The gray line in the zoom slider bar corresponds to the original image size (x 1.00). Similar to toolbars, the position of the zoom sliders (left, right, top or bottom) can be changed by clicking on the area above the zoom icons and dragging the zoom sliders in position with the mouse.

1.6.8 Behaviour of FPQuest windows

1.6.8.1 Open the database **DemoBase** by selecting **DemoBase** in the Startup program, and click on the

 button. Simply double-clicking on the database name does the same.

Via ***File > Preferences > Windows behaviour***, the behaviour of the FPQuest windows stack can be controlled. The windows stack can be set either ***Fixed*** or ***Flexible***. In **fixed** mode, the various FPQuest windows are stacked in a fixed order. For example, a *Comparison* window always appears on top of the *FPQuest main* window and in order to view the complete *FPQuest main* window, the *Comparison* window needs to be closed or minimized. Furthermore, in the Taskbar of your operating system, only one tab will appear for the FPQuest software. In **flexible** mode, any type of FPQuest window can be on top of another, regardless of its "rank". For each FPQuest window, a separate tab becomes available in the Taskbar of your operating system.

1.6.9 Navigator pane

For both fixed and flexible mode of the windows stack, a Navigator pane is available (Figure 1-29). The Navigator pane displays all open BioNumerics windows in a tree-like hierachical structure to facilitate navigation between windows. The active window is shown in orange type, inactive windows are shown in white type. Under default settings, the Navigator pane is enabled and appears when moving the mouse to the far right side on the screen.

The position of the Navigator pane on the computer display (top, bottom, left or right) can be modified:

1.6.9.1 Click on the structured part in the Navigator pane and drag it to the desired position with the mouse.

Other display properties of the Navigator pane can be set by right-clicking in the structured part and selecting them from the drop-down menu:

1.6.9.2 Right-click on the structured part of the Navigator pane and uncheck ***Always on top*** if you want the Navigator pane to appear *stacked* with other open windows instead of *on top* of all open windows.

1.6.9.3 Uncheck ***Auto hide*** if you want the Navigator pane to be permanently displayed.

1.6.9.4 Select ***Disable*** from the floating menu and press ***<OK>*** in the confirmation dialog box that appears to disable the Navigator pane.

The Navigator can be enabled again from the *BioNumerics main* window:

Select ***File > Preferences > Windows behaviour*** and check ***Show navigator***.

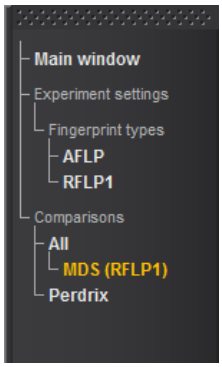


Figure 1-29. Navigator pane of a FPQuest session in which two *Experiment type* windows and two *Comparison* windows are open. For one comparison, an MDS is also available. The MDS window is currently active.

2. DATABASE

2.1 Introduction

2.1.1 Local and connected databases

FPQuest offers two possibilities to store its data: 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, Microsoft SQL Server, Oracle, PostgreSQL, MySQL, and DB2. Others may work as well but are not guaranteed to be fully compatible in a standard setup.

The *local database* is a generic file-based databasing environment with limited possibilities but simple to handle. It was the first database solution available in FPQuest, and is still maintained for compatibility reasons. However, we recommend to use the connected database option to create new databases. As connected databases rely on market standard database software, they offer a number of security and sharing/exchange options and are much more extensible than local databases. In addition, connected databases offer a richer database structure within FPQuest, e.g. by providing fingerprint lane information fields. Connected databases are the default option in FPQuest version 5.0 and further.

2.1.2 Elementary structure of a database

The core unit of a FPQuest database is the *entry*. Entries represent the biological entities for which data is sampled, digitized and imported, to be further compared and analyzed. Each entry is identified by a unique *key*, through which various pieces of information are linked to the appropriate entries: information fields, attachments, fingerprints, etc. (see Figure 2-1).

2.1.3 Location of a database

A FPQuest database can be located on the local computer or anywhere on the network, as long as FPQuest has sufficient privileges to write to the database and its associated folders. FPQuest recognizes and inventories the available databases by looking in the *home directory*, a folder that can be specified by the user, and which contains a descriptive text file for each database. The files have the extension ".dbs" and basically contain a tag [DIR] under which the full database path or network location is written. If the databases are subfolders of the home directory (the default setting), the full path is relativized as **`[HOMEDIR]\*dbname*`** (*dbname* is the name of the database folder). This notation is only used by FPQuest version 5.0 onwards, and is not compatible with earlier versions. The relative reference has the advantage that the home directory with all databases as subfolders can physically be moved to another drive or computer without having to change the database path in the .dbs files. The different steps in

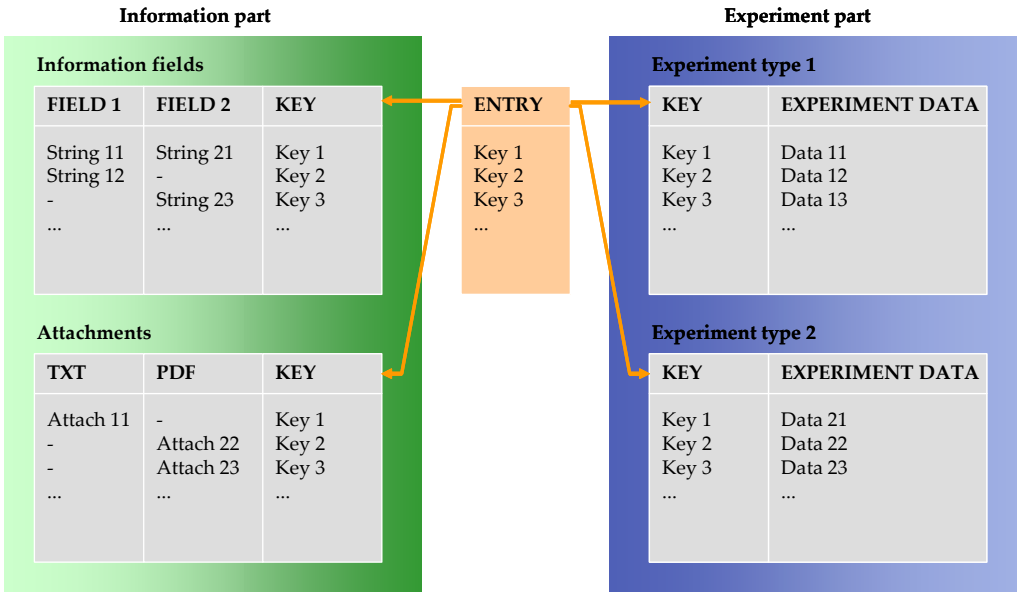
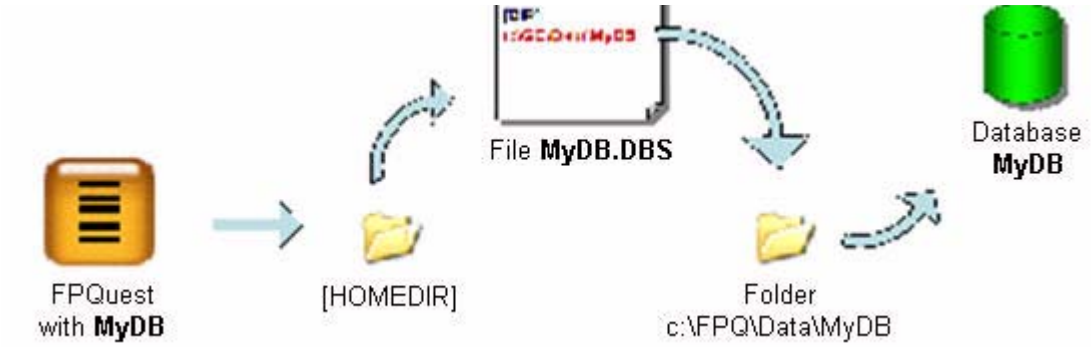


Figure 2-1. Linking various sorts of information to database entries through unique keys.



Steps in opening a database "MyDB": (1) FPQuest looks in the home directory for file MyDB.dbs (2) File MyDB.dbs is opened to obtain the database path; (3) the database is opened in the database path found.

opening a database are schematically represented in Figure .

A FPQuest database requires a number of subfolders to be present in the database folder (see Figure 2-2).

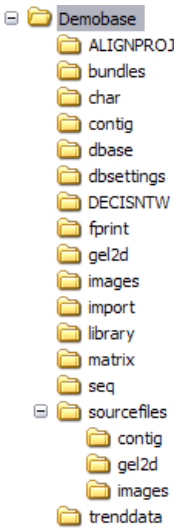


Figure 2-2. FPQuest database folder with its subfolders.

NOTE: A number of automatically created database folders deal with experiment types that cannot be analysed with FPQuest. These folders should however not be removed, since they ensure compatibility with the BioNumerics software package for data exchange or in case of software migration.

These folders are created automatically by the software when the database is set up, or when the software is launched. In a local database setup, they may contain a number of files that store the experiment data, information fields, experiment and analysis settings, etc. For backup purposes, the entire database folder with all its subfolders should be backed up (see Section 2.6). In a connected database, most, but not all of these folders are empty. Window and viewing settings, for example, are stored locally. Optionally, imported files such as gel TIFF images or sequencer trace files, can also be stored locally in a connected database setup, but the default setting is to store all information, including import files, inside the connected database. Figure 2-3 illustrates a typical connected database setup.

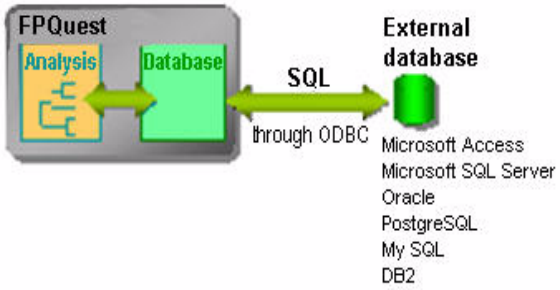


Figure 2-3. Connected database setup: all data is passed on to the SQL database through ODBC; FPQuest's local database is empty.

2.1.4 Setting up a new database

The creation of a new database is described in 1.5.2. This paragraph also describes how to locate and change the home directory. A number of settings can be changed both for local and connected databases:

2.1.4.1 In the Startup program, select a database and press the settings button ( ). From the menu that is displayed, select ***Database settings***. The *Database settings* dialog box appears (Figure 2-4).

Here you can change the database directory with ***<Change directory>***. This option will overwrite the [DIR] tag in the ***Database*.dbs** file (Figure).

With ***Enable log files***, it is possible to log all events that alter database information (see 2.1.6). In a connected database, logging events are written to a table, whereas in a local database, log files are created.

With ***<ID code>***, you can install an ID code to protect important settings in a local database (see 2.1.5). This function has no meaning in a connected database, where

other, more advanced protection mechanisms are available (see 2.3.10).

2.1.4.2 Press **<*OK*>** or **<*Cancel*>** to exit the *Database settings* dialog box.

Figure 2-4. The *Database settings* dialog box in the Startup program.

2.1.4.3 To remove a database, select it from the list, press the settings button ( ) again and select ***Delete database***. The program will ask for confirmation before deleting the database. When the database contains non-empty subfolders, which is usually the case, its folder structure will not physically be removed from the volume. However, it will be removed from the database list in the Startup program.

2.1.5 Protecting a database

The protection tools discussed in the present section are only meant to reduce the chances of incidental erroneous changes or damage to important database or experiment settings. **They are by no means secure enough to prevent others to making changes to the database!** Other, more advanced and secure protection mechanisms are available for connected databases (see 2.3.10).

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 fingerprint types: data of existing entries cannot be changed, however, new information can be added to the fingerprint data file.
- Libraries for identification (locally stored libraries only): 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.

2.1.5.1 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 *FPQuest main* window, and a key icon also appears in the *Fingerprint file information* panel of the file's edit window.

To protect a local database against modification by others or misuse, FPQuest 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. For connected databases, we recommend to use the protection mechanisms provided by the database management software (DBMS). For instructions on how to protect Access databases, see 2.3.10. For other databases, we refer to the specific DBMS documentation.

2.1.5.2 In order to set an ID code for the local database, run the Startup program and press the settings button ( ). From the menu that is displayed, select ***Database settings***.

2.1.5.3 In the *Database settings* dialog box (Figure 2-4), 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.

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

> *NOTE: If you forget the ID code, you will have to contact Bio-Rad.*

2.1.6 Log files

In certified environments and laboratories where conscientious recording of manipulations is important, the *log files* in FPQuest are a useful tool. For every FPQuest 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. There are a number of differences between the logging in a local database and a connected database. Logging is more complete and is centrally maintained in a connected database. In a local database, separate log files are maintained for different data types.

Log files are recorded for the following data types:

- 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 of the experiment types: for every change made, the kind of change is indicated. All settings are

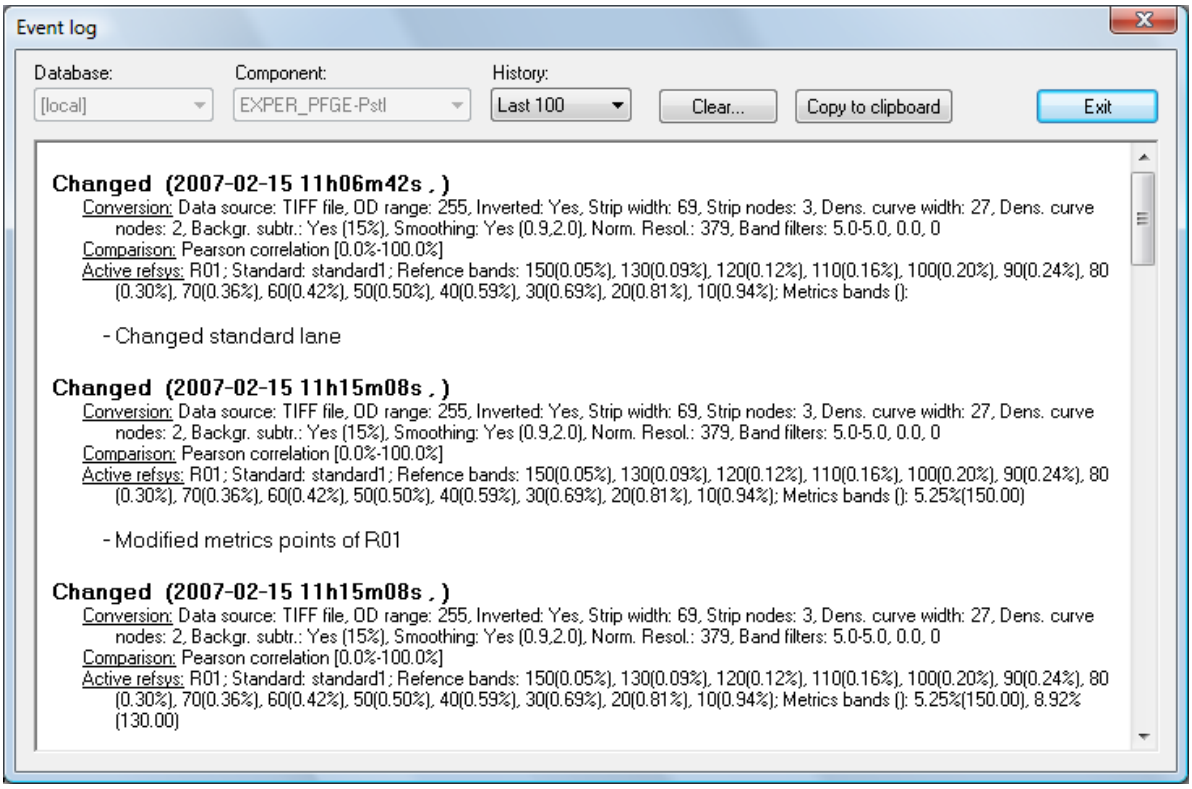


Figure 2-5. Event log viewer for a database component (local database).

recorded in the log file, so that the user may restore the previous settings based upon the log file, if enabled.

- The data for the experiment types: if data for entries are changed, the log file lists these 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.

2.1.6.1 In order to enable the creation of log files for the database, press the settings button ( ) in the Startup program and select ***Database settings***.

2.1.6.2 In the *Database settings* dialog box (Figure 2-4), check the ***Enable log files*** checkbox.

2.1.6.3 In the data file windows, experiment file windows, the *FPQuest main* window, or the *Library* window, select ***File > View log file*** or press to display the log file.

In a local database (as opposed to a connected database; see Section 2.3), FPQuest creates a temporary file, DBASE.LOG or **<EXPERIMENTNAME>.LOG**. For data files, it creates a log file **<DATAFILE>.LOG**.

2.1.6.4 The log files are loaded in FPQuest's Event log viewer (Figure 2-5).

2.1.6.5 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 retained in other software that supports RTF.

The items ***Database*** and ***Component*** are only applicable to connected databases (see Section 2.3).

2.2 Database functions

2.2.1 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 FPQuest again with this new database:

2.2.1.1 Close the **DemoBase** database with ***File > Exit***.

2.2.1.2 Back in the Startup program, select the database **Example** and click on the  button. Simply double-clicking on the database name does the same.

Entries can be added to the database 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 on, 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.

2.2.1.3 Select ***Database > Add new entries*** or press  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 is a connected database associated with the database (see Section 2.3), 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.

2.2.1.4 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:

2.2.1.5 Select the entry and ***Database > Change entry key***.

2.2.1.6 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 FPQuest. In the latter case, they can be effectively used as a 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 1.2.3: right-clicking in the database panel of the* FPQuest main *window pops up the menu* **Add new entries** *and* **Change entry key** *(if you click on an entry).*

2.2.1.7 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, and will warn you if there is any experiment information linked to the entry.

2.2.1.8 To remove all selected entries at once, choose ***Database > Remove all selected entries***. See 2.2.7 to 2.2.9 for more information on the selection of entries.

WARNING: There is no undo function for this action and removed entries are irrevocably lost, together with any experiment information linked to them!

2.2.1.9 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.

2.2.2 Creating information fields

A number of predefined information fields are automatically created when creating a new database. All predefined information fields are listed in the pull-down menu's in the information fields header of each panel and can be either displayed or hidden (see 1.6.6).

In addition to these default information fields, extra information fields can be added with ***Add new information field*** or removed with ***Remove information field***. These menu commands can be accessed in each panel by right-clicking in their information toolbars. Information

fields can also be added to the *Database* panel with the corresponding menu commands.

2.2.2.1 Select ***Database > Add new information field***.

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

2.2.2.3 Select ***Database > Add new information field*** again to define the second field, *Species*.

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

2.2.2.5 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.

> *NOTE: Renaming or removing information fields in FPQuest is not possible when using a connected Access database (.mdb and .accdb). To rename an information field in this case, you should open the database with Access (see 2.1.3 on how to locate the database) and rename the corresponding column in the ENTRYTABLE table. When the database is again loaded in FPQuest, both the old and the renamed information field will appear. The old information field (now empty) can then be removed. Removing an information field should also be performed in Access prior to reloading the database in FPQuest.*

An information field in a local or connected database may contain up to 80 characters. In a local database, a maximum of 150 fields can be defined. In a connected database, many more fields can be defined, but only 150 can be displayed at the same time.

2.2.3 Entering information fields

2.2.3.1 By double-clicking, or pressing Enter on a database entry, the *Entry edit* window appears (Figure 2-6). Right-clicking on the entry, and selecting ***Open entry*** also works.

The *Database fields* panel (left panel in default configuration) of the *Entry edit* window shows the information fields and the *Experiments* panel (right panel in default configuration) shows the available experiments for the entry. The tab in the bottom right of the *Experiments* panel gives access to the *Attachments* panel. The latter allows attachments to be added and viewed for the entry (see 2.2.4). The *Entry edit* window can be rescaled to see more and/or longer information fields. The relative size of the panels can also be modified by dragging the separator line between the panels. Since the *Database fields*, *Experiments* and *Attachments* panels are all dockable panels, their position can be further customized as described in 1.6.4.

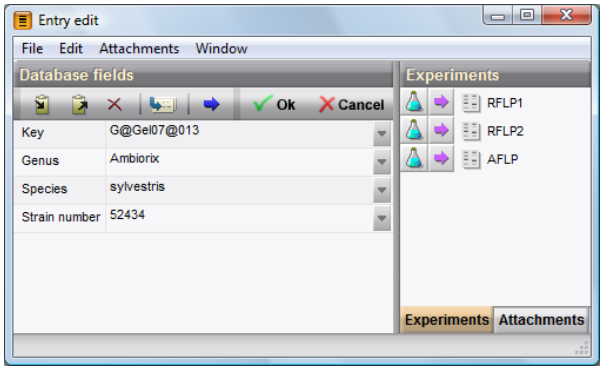


Figure 2-6. The *Entry edit* window.

2.2.3.2 Enter some information in each of the fields (see Figure 2-6).

2.2.3.3 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  .

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

2.2.3.5 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.

2.2.3.6 Drop down a history list by clicking the  button on the right hand side 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 Section 2.3).

2.2.3.7 Using  , you can select or unselect the opened entry in the database (see Figure 2-6), for the construction of comparisons. When the entry is selected, this button shows as  .

2.2.3.8 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.

Alternative to using the *Entry edit* window, information in non-default information fields in the database and in other grid panels can be edited directly by clicking twice on an information field in the database. The information will appear highlighted and can be edited. When field states are defined (see 2.2.5), they now become available as a drop-down list.

*NOTE: Single click field editing can be enabled via **File > Preferences** in the FPQuest main window (see 1.6.6.9).*

2.2.4 Attaching files to database entries

Besides its information fields and the experiments linked to it, a database entry can also have files attached to it. Usually the attachment is a link to a file, except for text attachments, which are physically contained in the database. In addition to the attachment itself, FPQuest also allows a description to be entered for the attachment. The following data types are supported as attachment:

- **Text**: Plain ASCII text attachments of unlimited length. FPQuest contains its own editor (similar to Notepad) to paste or type text strings.
- **Bitmap image**: Images of the following bitmap types are supported: TIFF, JPEG (JPG), GIF, BMP, PNG, and WMF. FPQuest contains its own viewer for image attachments.
- **HTML documents**: HTML and XML documents can be attached as well as URLs. FPQuest contains its own HTML viewer.
- **Word® document**: Documents in Microsoft® Word® format can be attached. The default editor or viewer registered by your Windows system will be opened if you want to edit or view the document.
- **Excel® document**: Documents in Microsoft® Excel® format can be attached. The default editor or viewer registered by your Windows system will be opened if you want to edit or view the document.
- **PDF® document**: Documents in Adobe® PDF® format can be attached. The default editor or viewer registered by your Windows system will be opened if you want to edit or view the document.

2.2.4.1 To create an attachment for an entry, open the *Entry edit* window as described in 2.2.3 and select the *Attachments* tab. The *Attachment* panel contains three buttons, respectively to create a new attachment , to open (view) an attachment and to delete an attachment . The same commands are available from the menu as ***Attachment > Add new***, ***Attachment > Open***, and ***Attachment > Delete***, respectively.

2.2.4.2 Press to create a new attachment. The *Entry attachment* dialog box appears (Figure 2-7).

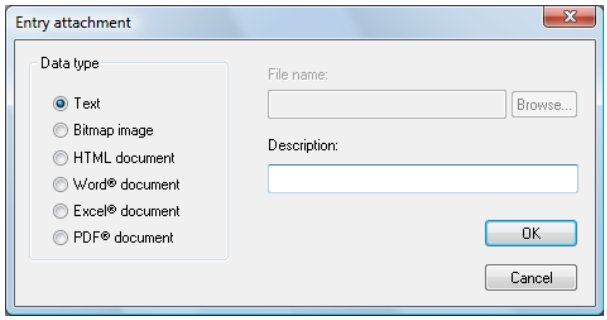


Figure 2-7. The *Entry attachment* dialog box.

2.2.4.3 Under ***Data type***, specify one of the supported data types, as described earlier in this paragraph.

2.2.4.4 All data types except Text link to a file on the computer or the network. You can enter a path and a file name or use the ***<Browse>*** button to browse to a file of the specified type. Text attachments are stored inside the FPQuest database.

2.2.4.5 A ***Description*** input field allows you to enter a description line for the attachment. The description will appear next to the attachment icon in the *Entry edit* window (Figure 2-6) and for text, bitmap, and HTML type attachments, it will also appear in the viewer or editor (text) window when the attachment is opened.

2.2.4.6 To open an attachment, double-click on the attachment icon in the *Entry edit* window.

2.2.4.7 In case of a text attachment, a *Text attachment* editor is opened (Figure 2-8) where one can type or paste a text document of unlimited length. The format should be pure text; any formatting will be lost while pasting texts from other editors. The editor contains a ***Save*** button , an ***Undo*** (shortcut: CTRL+Z) and ***Redo*** (shortcut: CTRL+Y) button, as well as a

Cut (shortcut: CTRL+X), ***Copy*** (shortcut: CTRL+C) and ***Paste*** (shortcut: CTRL+V) button.

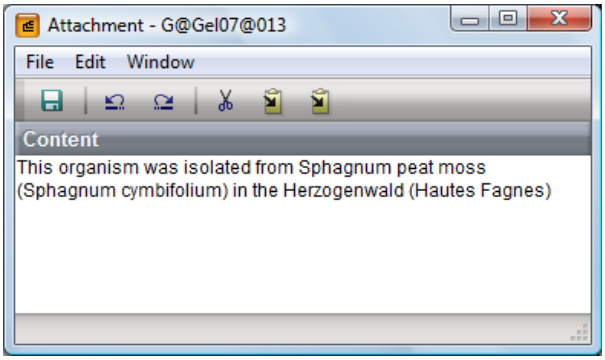


Figure 2-8. The *Text attachment* editor.

2.2.4.8 In case of a bitmap image (TIFF, JPEG [JPG], GIF, BMP, PNG, and WMF), FPQuest's own viewer is opened with the bitmap displayed. The window contains a ***Zoom in*** and a ***Zoom out*** button.

2.2.4.9 In case of HTML and XML attachments, FPQuest's own browser is opened with the HTML or XML file displayed. Note that an HTML document can be a link to a website, in which case the browser will display the website. The browser contains a ***Back*** button to return to the previous page.

2.2.4.10 Word®, Excel® and PDF® attachments are opened in the default programs registered by your Windows system for these file types.

2.2.4.11 To edit the link of an attachment or its description line, use ***Attachments > Edit*** in the menu of the *Entry* editor. The *Entry attachment* dialog box appears as shown in Figure 2-7.

If the predefined information field 'Attachments' is checked in the pull-down menu of the column properties button ( ) in the *Database* panel, the number of attachments is displayed for all entries (see Figure 2-9).

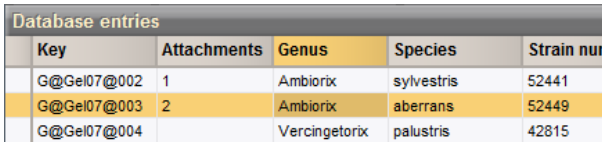


Figure 2-9. Detail of the *Database entries* panel in the *FPQuest main* window, showing two entries with attachments.

2.2.5 Information field properties

In FPQuest, properties can be assigned to a database information field. These properties include a list of *field states*, i.e. possible content that can be contained within the information field. Individual states can each be displayed against a differently colored background, for an improved display in grid panels. Field states also provide an additional display option in an unrooted tree or coordinate space window. Finally, when field states are defined, they become available as a drop-down list, facilitating and harmonizing the input of data via direct field editing.

2.2.5.1 In the *FPQuest main* window with **DemoBase** loaded, click on the header of the information field for which you want to set the properties (e.g. the 'Genus' field).

2.2.5.2 Select ***Database > Information field properties***. The *Information field properties* dialog box appears (see Figure 2-10). The ***Field states*** list is initially empty.

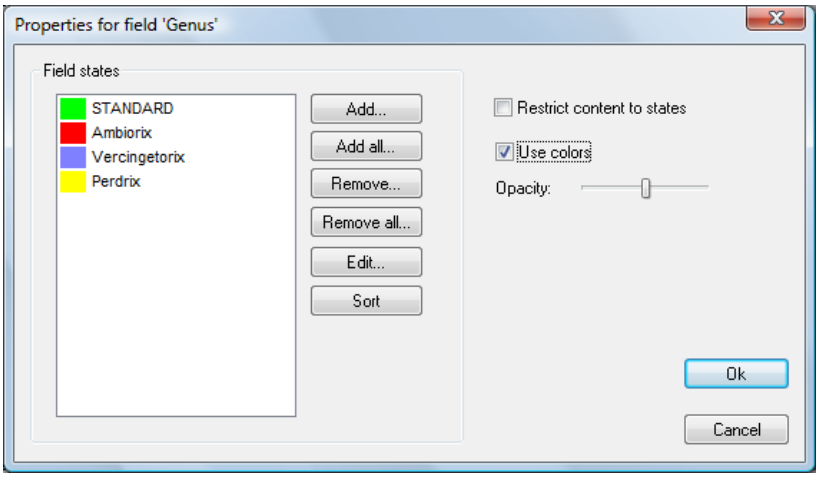


Figure 2-10. The *Field properties* dialog box for the 'Genus' field in DemoBase.

2.2.5.3 Press **<*Add*>** and enter a possible content (a *state*) for the 'Genus' information field, e.g. Ambiorix. If ***Use colors*** is checked (see 2.2.5.9), the same dialog box also allows you to set the ***Background color***.

The state Ambiorix is now listed under ***Field states***. If the information field already contains information (as it is the case with **DemoBase**), this information can be used to automatically create a list of ***Field states***.

2.2.5.4 Press **<*Add all*>** to automatically create all existing states for the field 'Genus'.

2.2.5.5 To edit a field state, select it from the list and press **<*Edit*>**.

2.2.5.6 Likewise, if you want to remove a field state, select it from the list and press **<*Remove*>**.

2.2.5.7 All field states can be removed at once by pressing **<*Remove all*>**. The program will ask for confirmation.

2.2.5.8 Press **<*Sort*>** to have the field states sorted alphabetically.

2.2.5.9 Check ***Use colors*** to display a specific color code for each field state. The *Field properties* dialog box should now look the same as in Figure 2-10.

Color codes will be automatically generated for the first 30 field states, but can be modified if desired using the **<*Edit*>** button. The ***Opacity*** slider sets the applied color intensity.

2.2.5.10 Check ***Restrict content to states*** if the ***Fields states*** list contains all possible states for this information field.

Turning on this feature forces database users to provide consistent information, as no other values than the ones specified in the ***Fields states*** list will be accepted for the information field.

2.2.5.11 Press <OK> in the *Field properties* dialog box.

In the *Database entries* panel of the *FPQuest main* window (and also in the *Information fields* panel of the *Comparison* window), the different states of the 'Genus' field will each appear against their own background color. The extent of the background coloring can be modified using ***File > Preferences*** (see 1.6.6).

2.2.5.12 Click twice on an information field to enable direct editing (see 1.6.6) and to display a button on the right-hand side of the information field.

2.2.5.13 Press the button (or press CTRL+down on the keyboard) to drop down a list from which you can select a field state (see Figure 2-11).

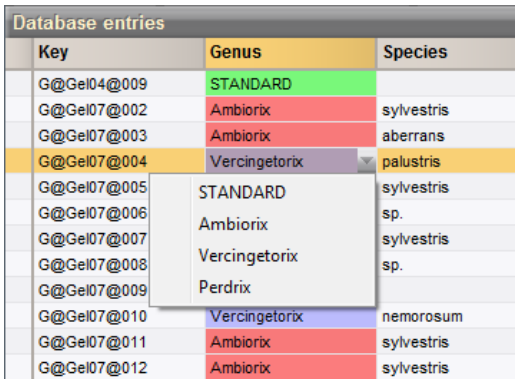


Figure 2-11. Detail of the *Database entries* panel, showing the drop-down list with field states for 'Genus'.

NOTES:

(1) If **Restrict content to states** *(see 2.2.5.10) is checked in the* Field properties *dialog box for the informations field, its content cannot be edited by typing. Only the states available from the drop-down list can be selected as content.*

(2) The field states drop-down list that appears after clicking the button in the Database entries *panel is different from the history drop-down list in the* Entry edit *dialog box (see 2.2.3.6). The latter automatically remembers the 10 last values entered for the field, while the former is not updated when a new state is entered by typing; the field state list needs to be updated via the* Field properties *dialog box.*

2.2.6 Configuring the database layout

Since the *Database entries* panel is a grid panel, all display and customizing features discussed in 1.6.6 are valid for this panel as well. Some features that are particularly useful in the context of database layout will be discussed here in detail.

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

2.2.6.1 Click on one of the database field names in the information fields header of the *Database entries* panel.

2.2.6.2 Select ***Edit > Arrange entries by 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 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 field***), and then numerically using ***Edit > Arrange entries by field (numerical)***. The result will be [**126a**, **126b**, **126c**, **213**].

The user can determine which information fields are displayed and the order in which they are shown.

2.2.6.3 Click on the column properties button in the database information fields header. From the pull-down menu that appears, click on any field name to either display (icon shown in the menu) or hide (no icon shown) the information field in the *Database entries* panel.

This feature can be used to hide fields that are non-informative for the user. For example, if keys are automatically generated, they might not contain useful information for you and can therefore be hidden.

2.2.6.4 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 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' to the left to show just one character.

2.2.6.5 To change the position of an information field, click on the header of the field you want to move and then on the column properties button. Select ***Move 'FieldName' to left*** or ***Move 'FieldName' to right***. Shortcut keys are CTRL+left arrow and CTRL+right arrow, respectively.

2.2.6.6 It is possible to freeze one or more information fields, so that they always remain visible left from the scrollable area. For example, if you want to freeze the 'Key' field, select the field right from the 'Key' in the field header, and select ***Edit > Freeze left pane***. Alternatively, you can select ***Freeze left pane*** from the column properties button. This feature, combined with the possibility to change the order of information fields makes it possible to freeze any subset of fields.

2.2.6.7 The width of the *Database entries* panel as a whole can be changed by dragging the separator lines between the *Database entries* panel, the *Experiment presence* panel and the remaining panels to the left or to the right.

Settings 2.2.6.3 to 2.2.6.7 as well as all window sizes and positions are stored when you exit the software and are specific for each database.

When a new comparison is created or when an existing comparison is opened (see Chapter 4), the same layout as applied in the *Database entries* panel of the *FPQuest main* window (which fields to display/hide, column width and order of information fields) is used for the *Information fields* panel of the *Comparison* window.

2.2.7 Selections of database entries

Selections in FPQuest provide a tool to perform an action on a selected number of database entries, instead of on the database as a whole. As such, selections form the basis for the creation of comparisons (see Chapter 4), enabling a host of analysis tools to be applied on the selected entries. Selections can be cut, copied, pasted or deleted and furthermore allow the user to create subsets within a database (see 2.2.11), define library units for identification (see 5.2.1), select entries to be identified (see 5.2.2) and to share well-defined information with other users via the creation of bundles (see 2.5.2) or XML files (see 2.5.3).

Besides manual selection functions (see 2.2.8), automatic search and select functions are available in FPQuest, with simple (see 2.2.9) and more advanced query functions (see 2.2.10).

2.2.8 Manual selection functions

The manual selection functions will be illustrated using the **DemoBase** database.

2.2.8.1 In the Startup screen, double-click on **DemoBase** to open the database for analysis. Alternatively, you can select **DemoBase** from the list and then press the button.

A single entry can be selected by holding the CTRL key and left-clicking. Selected entries are marked by a colored arrow (Figure 2-12). Selected entries are unselected in the same way.

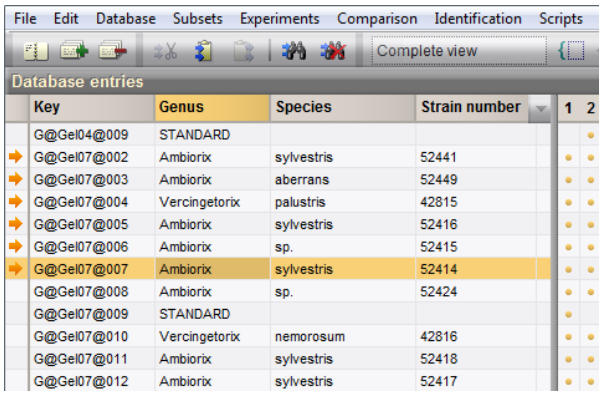


Figure 2-12. *Database entries* panel in the *FPQuest main* window, showing selected entries (orange arrows).

2.2.8.2 Select the first non-standard lane (CTRL + mouse click). The entry is now marked by a colored arrow.

2.2.8.3 In order to select a group of entries, hold the SHIFT key and click on another entry.

2.2.8.4 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.

2.2.8.5 A single entry can be selected or unselected from its *Entry edit* window (Figure 2-6) using the  button. When the entry is selected, this button shows as  .

All the entries from the database or from the current subset (for more information on subsets, see 2.2.11) can be selected using the keyboard shortcut CTRL+A or with ***Edit > Select all*** in the *FPQuest main* window.

2.2.8.6 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*** or use the keyboard shortcut CTRL+T for this utility.

2.2.8.7 Clear all selected entries with ***Edit > Unselect all entries*** (F4 key) or  .

NOTE: A very convenient command in combination with the manual selection functions is **Arrange entries by field**, *which allows the database to be sorted according to the selected information field (see 2.2.6 for a detailed description).*

2.2.9 Automatic search and select functions

In addition to manually selecting entries from the database, entries can be searched and selected automatically using a simple and intuitive search function.

2.2.9.1 Select ***Edit > Search entries*** (F3) or  . This pops up the *Entry search* dialog box (Figure 2-13).

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.

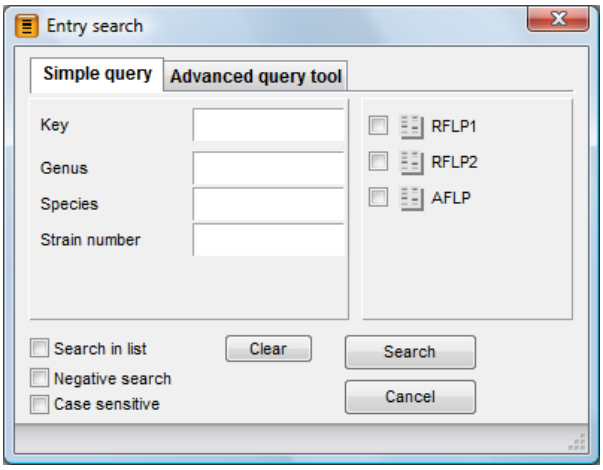


Figure 2-13. *Entry search* dialog box, Simple query tab.

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.

2.2.9.2 As an example, enter ***L*** in the *Species* field.

2.2.9.3 Press **<*Search*>**. All entries having a L in their species name are selected: *Ambiorix sylvestris* and *Vercingetorix palustris*.

2.2.9.4 Call the *Entry search* dialog box again, and press the **<*Clear*>** button.

2.2.9.5 Enter **STANDARD** in the 'Genus' field, and check the ***Negative search*** checkbox.

2.2.9.6 Press **<*Search*>** to select all database entries, except the entries used as standard lanes in the RFLP and AFLP techniques.

2.2.9.7 Clear the selection with the F4 key or click the  button (***Edit > Unselect all entries***).

2.2.10 The advanced query tool

FPQuest contains an advanced query tool that allows searches of any complexity to be made within the database, based on information fields and experiment data.

2.2.10.1 Call the query tool again, by selecting ***Edit > Search entries*** or pressing  .

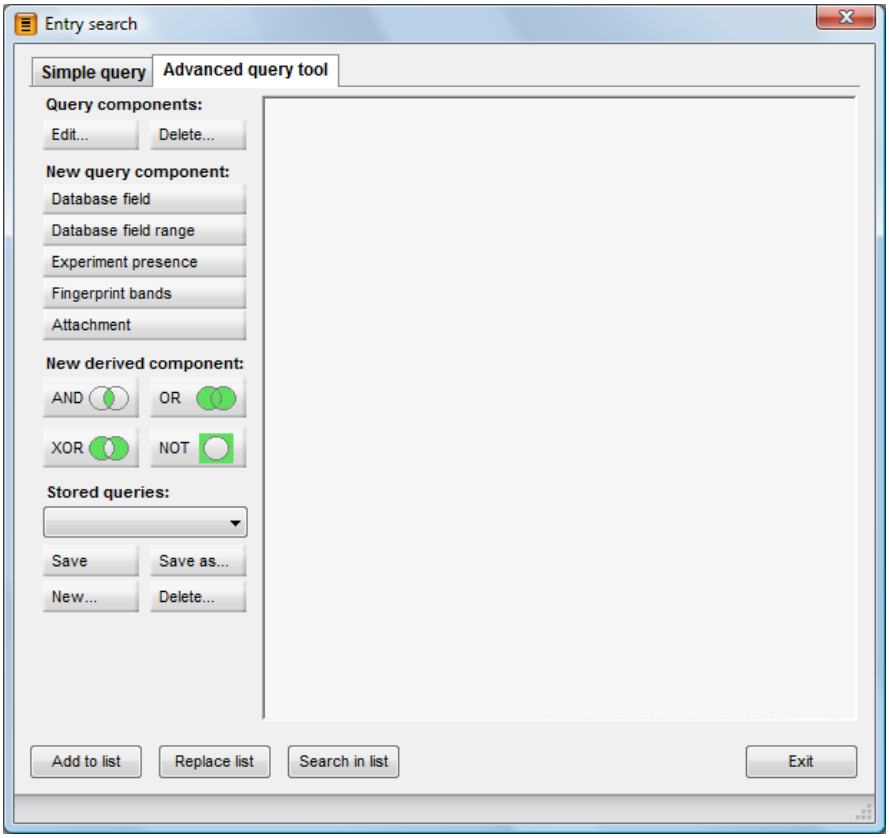


Figure 2-14. The advanced query tool.

The *Entry search* dialog box (Figure 2-13) contains an *Advanced query tool* tab.

2.2.10.2 Press ***<Advanced query tool>***. The normal *Entry search* dialog box changes into the *Advanced query tool* (Figure 2-14).

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***, ***Subsequence***, ***Trend data parameter*** and ***Attachment***.

•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 2-15). Note that the wildcard characters * and ? are not used in the advanced query tool.

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 Section 6.2).

•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

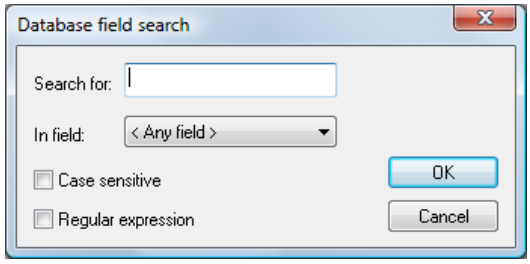


Figure 2-15. *Database field search component* dialog box.

input boxes (Figure 2-16). 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.

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.

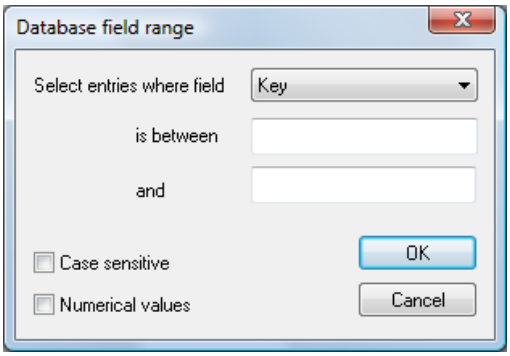


Figure 2-16. ***Database field range*** **dialog box.**

• 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 2-17) allows you to enter a ***Fingerprint experiment***, and specify an ***Intensity filter***, a ***Target range***, and a ***Number of bands present***.

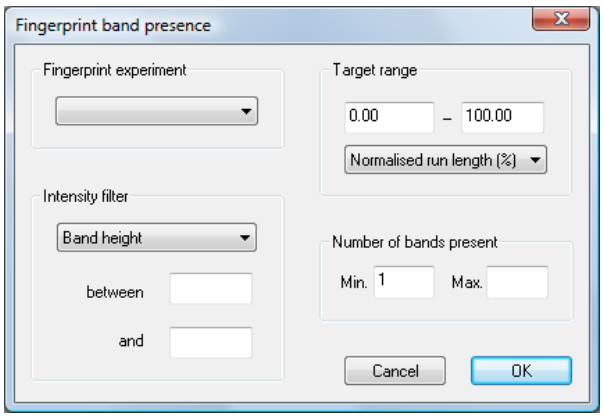


Figure 2-17. ***Fingerprint bands presence*** **dialog box.**

Under ***Intensity filter***, you can choose which intensity parameter to be used: ***Band height***, ***Band surface*** or ***Relative band surface***. When a 2D 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.

• Attachment

With the ***Attachment*** component, one can perform a search in attachments that are linked to database entries (Figure 2-18). With the pick list you can choose the type of attachments to search in. One of the possibilities is **All**, i.e. to search within all attachment types. For all types of attachments it is possible to search in the ***Description*** field, and for text type attachments, it is also possible to search within the ***Text***. The Text option does not apply to the other attachment types.

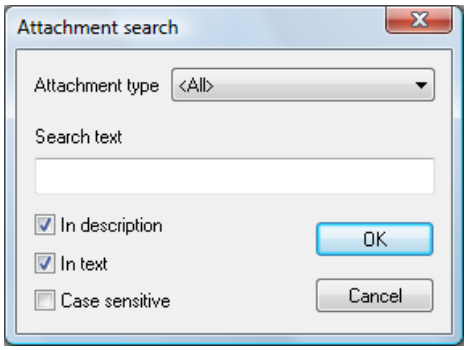


Figure 2-18. The ***Attachment search*** **dialog box.**

• 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 an entry to be selected.

OR, combines two or more components. The condition implied by at least one of the combined components should be fulfilled for an entry to be selected.

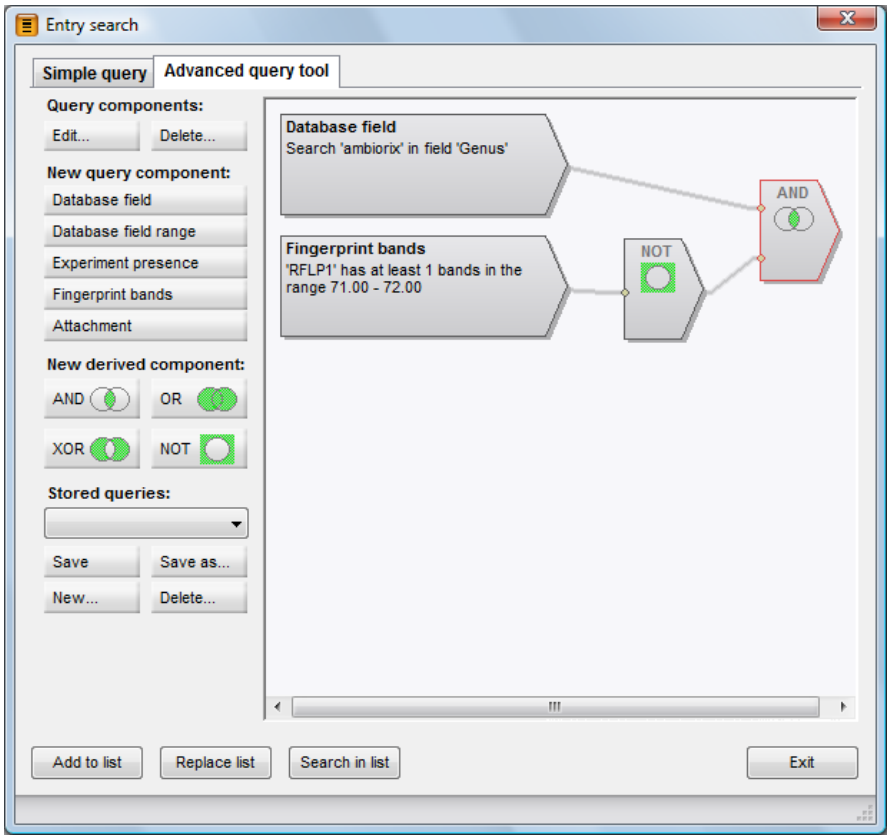


Figure 2-19. Combined query constructed in the *Advanced query tool* (see text for explanation).

XOR **XOR**, combines two or more components. Exactly one condition from the combined components should be fulfilled for an entry 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, sequences, and trend data parameters. As an example, we will select all entries from the genus *Ambiorix* that have no **RFLP1** bands in the range 71-72 base pairs.

2.2.10.3 Press ***<Database field>***. In the box that appears you can choose the genus field or leave **<Any field>** selected.

2.2.10.4 Enter **"ambiorix"** and press ***<OK>***.

A query component now appears in the right panel, stating "**Database field**: Search 'ambiorix' in field 'Genus'".

2.2.10.5 Press ***<Fingerprint bands>***. The *Fingerprint bands presence* dialog box appears (Figure 2-17).

2.2.10.6 Select **RFLP1** from the ***Fingerprint experiment*** pull-down list.

2.2.10.7 Under ***Target range***, enter 71 - 72, and specify **Metric values**.

2.2.10.8 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".

2.2.10.9 Select this **Fingerprint bands** component by clicking on it (highlighted when selected), and press the ***<NOT>*** button NOT.

2.2.10.10 Select the first component by clicking on it.

2.2.10.11 Hold down the CTRL key and click on the **NOT** box resulting from the second component to select it together with the first one.

NOTE: Multiple components/operators can also be selected together by dragging the mouse over the boxes in the right panel.

As both components are now selected, we can combine them with a *logical operator*.

2.2.10.12 Press the ***<AND>*** button AND to combine the created components with AND.

This is now shown graphically in the right panel (Figure 2-19).

2.2.10.13 To view the selected entries, press ***<Add to list>***.

The entries that were found are highlighted with a colored arrow left from them.

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.

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>***.

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 by loading them first and pressing ***<Delete>***.

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.

2.2.11 Subsets

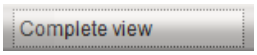
A selection of entries from the database can be saved as a *subset*. Subsets can include a certain target group in a database, for example, a single genus in a database containing many species, or any selection of relevant strains for a certain purpose. Selecting the defined subset displays a view of 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.

2.2.11.1 In database **DemoBase**, make sure no entries are selected using ***Edit > Unselect all entries*** (F4 key) or  .

2.2.11.2 Selecting ***Edit > Search entries*** or press  .

2.2.11.3 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 automatically be placed in the subset.

2.2.11.4 Select ***Subsets > Create new*** or press  . Alternatively, you can click on the subset selector 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.

2.2.11.5 Enter a name for the subset, e.g. the name of the selected genus "**Ambiorix**".

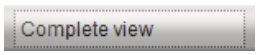
The created subset is now displayed, and the name of the current subset is displayed in the subset selector button 

.

2.2.11.6 Selecting the complete database or another subset, when available, can be done by pressing 

and selecting ***Complete view*** 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.

2.2.11.7 Select subset ***Ambiorix*** from the subset selector button 

.

2.2.11.8 We want to remove all the "sp." entries from this subset. Clear any selected entries by pressing F4 on the keyboard and select the 3 "sp." entries by manual selection or using the search function.

2.2.11.9 Press  or select ***Edit > Cut selection*** to cut the selected entries from the current subset (keyboard shortcut CTRL+X).

2.2.11.10 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*** (keyboard shortcut CTRL+V).

2.2.11.11 If you want to copy entries from one subset to another, without removing them from the first subset, there is also a command ***Edit > Copy selection*** or  (keyboard shortcut CTRL+C).

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 view**.*

2.2.11.12 When you want to remove entries from a subset without overwriting the contents of the clipboard, you can use the command ***Edit > Delete selection*** (keyboard shortcut DEL).

2.2.11.13 The current subset can be renamed using ***Subsets > Rename current***.

2.2.11.14 The current subset can be deleted using ***Subsets > Delete current*** or  .

2.2.12 Opening an additional database

Within FPQuest, an additional FPQuest database can be opened easily using the menu command ***File > Open additional database*** in the *FPQuest main* window. A dialog box appears with the question "Do you want to open this database in a new instance of the software?". If you answer ***<Yes>***, the additional database is opened in its own *FPQuest main* window. If you select ***<No>,*** the additional database is opened in the same *FPQuest main* window. When the additional database contains experiments and/or database fields that are not available in the already open database, FPQuest will automatically create these components in order to be able to display them.

NOTE: Two connected databases cannot be opened simultaneously in the same instance of the software.

2.3 Connected databases

2.3.1 Advantages of a connected database

FPQuest 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, Microsoft SQL Server, Oracle, PostgreSQL, MySQL, and DB2. Others may work as well but are not guaranteed to be fully compatible in a standard setup.

> *NOTE: FPQuest uses **Quoted Identifiers** to pass information to the connected database. Some database systems, for example MySQL, do not use this ANSI standard by default, but optionally. To use the database as a FPQuest connected database, make sure that the use of Quoted Identifiers is enabled in the database 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, FPQuest will allow multiple users to access and modify the database simultaneously. Note, in this respect, that FPQuest takes a "snapshot" of the database when the program is launched. As such, changes to the database made by others while you have a FPQuest session open will not be seen in your current session, until you reload the database during your session (see 2.3.8).

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 PostgreSQL, Oracle or SQL Server will be faster and more efficient in use than FPQuest's own local file-based database system.

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, FPQuest will require a number of tables with specific columns to be available (see Section 6.1). FPQuest 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 FPQuest 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. FPQuest writes and reads the information directly into and from the external database, without storing anything locally. Since every connected database has a local FPQuest database associated with it, the user has the option to store and analyze local entries together with entries in the connected database. The information field 'Location' displays either *Local* or *Shared* for locally or externally stored data, respectively. 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 FPQuest connected database deal with character types, sequence types, 2D gel types, and matrix types (BioNumerics). These tables are also required by FPQuest, in order to assure compatibility with BioNumerics databases and to allow upgrading from FPQuest to BioNumerics.*

There is a function in the FPQuest software that allows a local database to be converted into a connected database at any time (see 2.3.7). 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. However, using the available XML export and import scripts, it is also possible to export the contents of a local database as XML files (see 2.5.3 for more information about the XML Tools plugin) and import them in another connected database. In this way, the local database does not disappear. The same scripts also provide a means to convert connected database entries back into local database entries.

The combined use of local and connected databases is limited to avoid possible conflicts between the two database systems. In particular, the possibility that local and connected experiment types have the same name but different settings, should be avoided. Therefore, a few approved possibilities for working with connected databases are supported:

1. Creating a new database in FPQuest, which is linked to a new connected database. FPQuest is allowed to construct the database layout.

2. Creating a new connected database in FPQuest, by linking to an existing database that has a table structure already in a FPQuest compatible format (e.g. linking to an existing FPQuest connected database).

3. Creating a new connected database, linking to an existing database which is not created using FPQuest.

4. Converting a local database to a new connected database.

These possibilities are described in subsequent paragraphs.

2.3.2 Setting up a new connected database

FPQuest can automatically create a new database in Microsoft Access. When you are using SQL Server, Oracle, or PostgreSQL, however, you will have to create a new blank database before proceeding with the following steps.

2.3.2.1 In the FPQuest Startup program, click the  button to create a new database.

2.3.2.2 Enter **ConnectedBase** as database name.

2.3.2.3 In the next step, choose ***<Yes>*** to automatically create the required directories, since a local database associated with the connected database is required.

2.3.2.4 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 (see Figure 2-20):

- ***New connected database (automatically created)*** is the default setting and creates a new, empty Access database. This is recommended in most cases.
- ***New connected database (custom created)*** should be checked if a DBMS different from Microsoft Access is employed (e.g. SQL Server, PostgreSQL, or Oracle). This option is especially useful when one expects to generate a very large database (>4 gigabyte) or when multi-level database protection tools are required.
- ***Existing connected database*** is to be selected when FPQuest should be linked to an already existing database (see 2.3.5 and 2.3.6 for instructions).
- ***Local database (single user only)*** creates a local file-based FPQuest database. This option is not recommended, since it has limited functionality in comparison with a connected database (see 2.3.1).

2.3.2.5 In most cases, an Access database will be sufficient, so you can leave the default setting ***New connected database (automatically created)*** and press ***<Proceed>***. Next, continue with step 2.3.2.11.

> *NOTE: It is not required to have Microsoft Access installed on your computer to create an Access (.mdb) database in FPQuest. FPQuest uses instead the Microsoft Jet Engine, which comes with the Windows operating system.*

2.3.2.6 If the database engine is SQL Server, PostgreSQL, or Oracle, select ***New connected database (custom created)***.

2.3.2.7 The checkbox ***Store fingerprints in database*** (enabled by default) offers the choice to store fingerprint files (TIFF images and .CRV curve files) in the connected database or in the sourcefiles directory (see 2.3.3). The checkbox can be left checked.

2.3.2.8 To connect FPQuest to the database that you have created in the DBMS, you will need to build a *Connection String* using the ***<Build>*** button.

2.3.2.9 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.

2.3.2.10 Once the database connection is properly configured, you can press ***<OK>*** to quit the database setup.

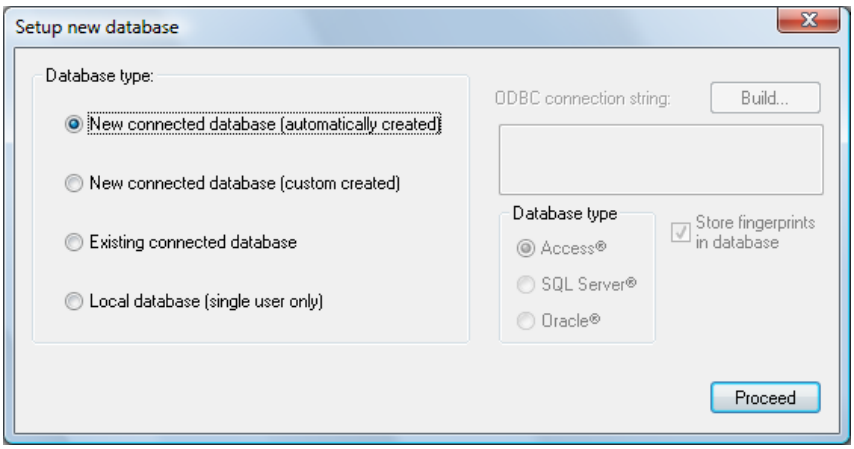


Figure 2-20. ***Database selection*** **dialog box.**

The *Plugin installation* window pops up, from which you can install the available plugins. For more information on the use of plugins, see 1.5.3.

2.3.2.11 Press ***<Proceed>*** in the *Plugin installation* window to open the *FPQuest main* window with the newly created, blank database.

2.3.3 Configuring the connected database link in FPQuest

In the *FPQuest main* window, 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.

2.3.3.1 Select ***Database > Connected databases***.

This opens a list of all currently defined connected databases for this FPQuest database (normally just one; see Figure 2-21).

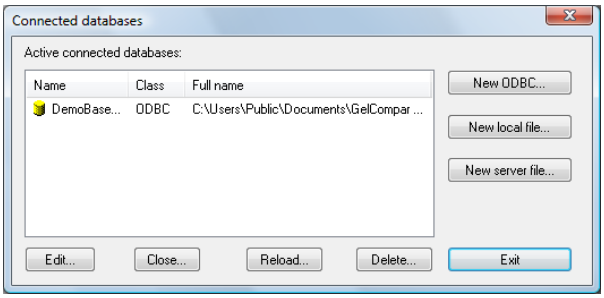


Figure 2-21. *Connected databases list* window.

2.3.3.2 Select the connected database of choice and click ***<Edit>***, or double-click on the name.

This results in the *Connected database configuration* dialog box (Figure 2-22).

The upper left input field (***Connected database***) shows the name of the *connection description file*, which can be found in the local database directory. When FPQuest has created a new connected database in the Startup program, the file is named ***dbname*.xdb** by default. The default directory for the .xdb file is [HOMEDIR]*dbname*. The [HOMEDIR] tag thereby points to the home directory as defined in the Startup program (see 1.5.1). The .xdb file is a text file and can be edited in Notepad or any other text editor.

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].

2.3.3.3 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 2.3.2.9).

2.3.3.4 By pressing the ***<Refresh>*** button, the connection between FPQuest and the connected database is refreshed. A tree-like table structure view of the database is displayed in the upper right panel.

2.3.3.5 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. FPQuest assumes a certain table structure to be able to store its different kinds of information. Each table should contain a set of columns with fixed names. This table structure is described in detail in Section 6.1. In a database setup where FPQuest is connected to an existing database system, views can be created with table names that correspond to the required FPQuest tables, and that have the required FPQuest 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.

Under ***Restricting query***, there is a possibility to enter a query that restricts the number of entries in the database to those that fulfill a specific query. The use of restricting queries is explained further in 2.3.9.

The option ***Prompt at startup*** allows the user to enter or choose a restricting query at startup when the connected database is loaded. The program prompts with a user-friendly graphical query builder similar to the advanced query builder described in 2.2.10. The use of the ***Prompt at startup*** query builder is discussed in 2.3.9.

With the option ***Experiment order statement***, it is possible to define a specific order for the experiments to show up in the *Experiments* panel in the *FPQuest main* window. By default, the experiments are listed alphabetically, which is indicated by the default SQL string "ORDER BY [EXPERIMENT]". [EXPERIMENT] refers to the column EXPERIMENT in table EXPERIMENTS (see 6.1.12), which holds the names of the experiments. This means that the experiments will be sorted by their name. It is possible to add an extra column to this table, with information entered by the user, for example an index number. If this column is specified in the SQL string, the experiments will be ordered by the index.

In ***Source file location***, the path for storing the source files (TIFF images and .CRV curve files) is entered. This is only used when this information is not stored in the database itself (see next paragraph). 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.

When a connected database is automatically created (see 2.3.2.5), fingerprint files (TIFF files, CRV files) are always saved in the connected database. In this case, the option ***Store fingerprint files in database*** is checked and grayed so it cannot be changed by the user (see Figure 2-22). For custom created connected databases (see 2.3.2.6 to 2.3.2.10), the user has the choice whether to store the fingerprint files into the connected databases (default) or in the sourcefiles directory. Contig projects (BioNumerics) are always saved in the connected database. For the trace files from automated sequencers (four-channel sequence chromatogram files), the user has the choice between linking to the original path of the files or storing them in the database, using a checkbox ***Store trace files in database*** (enabled by default). The trace files are stored in column **DATA** of table **SEQTRACEFILES** (see 6.1.17). In case ***Store traces in database*** is not checked, the column **DATA** will hold a link to the original path they were loaded from.

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 automatically insert new tables and fields where necessary, respectively.

WARNING: When pressing ***<Auto construct tables>***, FPQuest 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 yet exist in the database. In case you want to link FPQuest to an existing database, this may cause a number of tables and fields to be created and cause irreversible database changes! Solutions to link FPQuest to existing databases having different table structures are explained in 2.3.6.

2.3.4 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. For entries stored in the connected database, the 'Location' information field displays *Shared*.

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 FPQuest. In that case, you will have to create the field

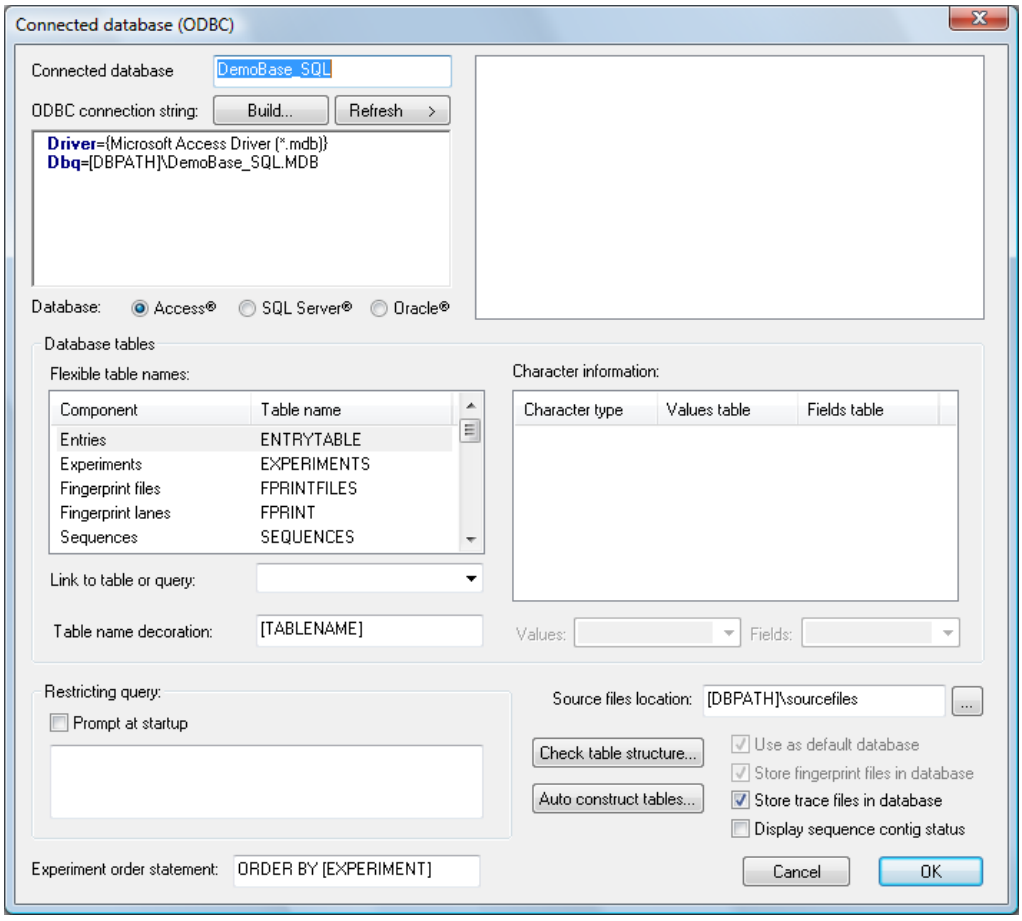


Figure 2-22. The *Connected database configuration* dialog box.

in Oracle, SQL Server, PostgreSQL, or Access, add it to the view, and reload the FPQuest database (see 2.3.7.14).

(2) Certain characters, for example a period, that are allowed in column names in a FPQuest database, may not be allowed in the connected database. We refer to the manual of the database system for more information.

(3) 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 and other source files are stored in the connected database itself. There is also an option to store source files in the common directory **Sourcefiles** under the local database directory. For example, in case of the newly created database **ConnectedBase**, the directory for such files would be **[DBPATH]\sourcefiles**. [DBPATH] hereby refers to the database folder in the FPQuest home directory, as specified under Settings () in the Startup screen.
- Within **Sourcefiles**, there are three subdirectories: **contig, images** and **gel2d**. 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 *FPQuest main* window. When fingerprint files are not stored in the database itself, the TIFF file should be present in this directory to make a gel file visible in the *Files* panel of the *FPQuest main* window. The **contig** and **gel2D** subdirectories contain source files for BioNumerics experiment types (sequences and 2D gels, respectively) that are not available in FPQuest. The source file directory can be modified as described in 2.3.3. 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 (e.g. normalization settings of a fingerprint type, adding or removing experiment types from a composite data set, 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 2-5), called from the *FPQuest main* window via ***File > View log file*** or  , 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 viewed.

2.3.5 Linking to an existing database with standard FPQuest table structure

Any computer running FPQuest can link up to an existing FPQuest connected database at any time. When this connected database has its table structure in the standard FPQuest format (see Section 6.1), this can be done very easily in the Startup program.

2.3.5.1 In the FPQuest Startup program, click the  button to create a new database.

2.3.5.2 Enter a name for the connected database (this can be a different name on different computers).

2.3.5.3 In the next step, choose ***<Yes>*** to automatically create the required directories, since a local database associated with the connected database is required.

2.3.5.4 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: ***New connected database (automatically created)***, ***New connected database (custom created)***, ***Existing connected database***, or ***Local database (single user only)*** (Figure 2-20).

2.3.5.5 Select ***Existing connected database*** and press ***<Build>*** to establish the connection to the database.

2.3.5.6 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.

2.3.5.7 Once the database connection is defined, you can press ***<OK>*** to quit the database setup.

The *Plugin installation* window pops up, from which you can install the available plugins. For more information on the use of plugins, see 1.5.3.

2.3.5.8 Press ***<Proceed>*** in the *Plugin installation* window to open the *FPQuest main* window with the newly created, blank database.

The connected database will now be the default database. If the connected database contains the standard table structure for FPQuest (see Section 6.1), no error message is produced and you can start working immediately. FPQuest will automatically recognize the existing information fields, experiment types, subsets, entries and data. If the table structure is not in standard FPQuest format however, a dialog box appears, warning for several errors that have occurred while trying to open specific tables in the connected database that were not found. See 2.3.6.10 and further to assign the correct tables or views from the database.

2.3.6 Linking to an existing database with table structure not in FPQuest format

This paragraph describes the situation where an Oracle, SQL Server, PostgreSQL, or Access database, containing descriptive information on organisms (entries) and/or experiment data is already present and FPQuest 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 FPQuest 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 FPQuest. 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 FPQuest 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 FPQuest should be linked to the appropriate field.
- FPQuest tables for which the connected database contains no fields can be created automatically by FPQuest.
- Finally, the database should be configured in such a way that the FPQuest 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 Section 6.1. The names for the tables or views, however, can be freely chosen.
- Additional tables required by FPQuest for which there are no fields available in the database can be created automatically by FPQuest.

NOTE: When views are created in the database, to match the required FPQuest tables, it is recommended to name the views using the standard FPQuest 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 (2.3.5). By using different names, new users will have to specify the table/view names manually in the Connected database configuration *window (Figure 2-22) 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.*

2.3.6.1 In the FPQuest Startup program, click the  button to create a new database.

2.3.6.2 Enter a name for the new database.

2.3.6.3 In the next step, choose ***<Yes>*** to automatically create the required directories, since a local database associated with the connected database is required.

2.3.6.4 In the next step, choose whether or not to create log files, and press ***<Finish>***.

A new dialog box pops up, prompting for the type of database: ***New connected database (automatically created)***, ***New connected database (custom created)***, ***Existing connected database***, or ***Local database (single user only)*** (Figure 2-20).

2.3.6.5 Select ***Existing connected database*** and press ***<Build>*** to establish the connection to the database.

2.3.6.6 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.

2.3.6.7 Once the database connection is specified, you can press ***<OK>*** to quit the database setup.

The *Plugin installation* window pops up, from which you can install the available plugins. For more information on the installation of plugins, see 1.5.3.

2.3.6.8 Press ***<Proceed>*** in the *Plugin installation* window to open the *FPQuest main* window with the connected database.

Since the connected database does not contain the standard table structure for FPQuest (see Section 6.1), a dialog box now appears, warning for several errors that have occurred while trying to open specific tables in the connected database that were not found.

2.3.6.9 Press **<*OK*>** to close the message(s). The *FPQuest main* window shows a blank database.

In the *FPQuest main* window, you can now configure the database connection as described in 2.3.3:

2.3.6.10 Select ***Database* > *Connected databases***.

This opens a list of all currently defined connected databases for this FPQuest database (normally just one; see Figure 2-21).

2.3.6.11 Select the connected database of choice and click **<*Edit*>**, or double-click on the name.

This opens the *Connected database configuration* dialog box (Figure 2-22). This dialog box shows the default suggested table names for the required database components under ***Database tables*** (see 2.3.3). Some, or all, of these tables do not correspond to the tables of the database.

2.3.6.12 Press the **<*Refresh*>** button. The upper right panel now lists the tables and views in the connected database, as it exists.

2.3.6.13 You can expand each table/view to display its fields by clicking on the "+" sign on the tree.

2.3.6.14 Under ***Database tables***, select the corresponding table or view for each component.

2.3.6.15 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* dialog box open. As soon as the new configuration is done, press **<*Refresh*>** and check the table structure again.

2.3.6.16 Finally, when all links to existing database tables/views are made correctly, you can allow FPQuest to create additional tables for which there are no fields available in the external database, by pressing **<*Auto construct tables*>**. FPQuest 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 FPQuest 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 FPQuest database.

2.3.7 Converting a local database to a connected database

In order to take full advantage of all features available in FPQuest, the user may want to convert a previously created local database to a connected database. There are two options available for this conversion:

1. Exporting all entries from the local database to XML files and importing these XML files in a new connected database

2. Setting up an ODBC connection and converting the local data to the connected database via a function available in FPQuest.

The first procedure is the safest way of working and is therefore recommended. It does, however, require the *Database sharing tools* module to be present. To check whether you have the *Database sharing tools* module, open any database and select ***File* > *About*** (see 1.1.5).

•Option 1: Using the XML Tools (requires the *Database sharing tools* module DS).

2.3.7.1 Open the local database that you want to convert. In the *FPQuest main* window, select ***File* > *Install/ Remove plugins*** and install the XML Tools plugin (see 1.5.3 on the installation of plugins).

2.3.7.2 In the *Database entries* panel of the *FPQuest main* window, click on the first database entry and, while holding the SHIFT key, click on the last entry to select all database entries. Alternatively, press CTRL+A on the keyboard.

2.3.7.3 Select ***File* > *Export selection as XML***. The *Export data to XML* dialog box appears (see Figure 2-23).

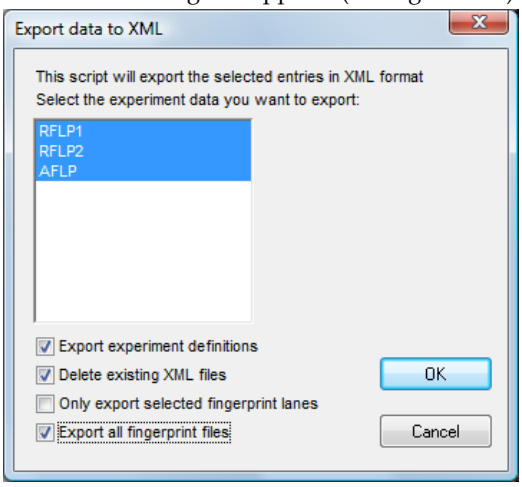


Figure 2-23. The *Export data to XML* dialog box from the XML Tools plugin.

2.3.7.4 Leave all experiment types selected, uncheck ***Only export selected fingerprint lanes*** and check ***Export all fingerprint files***. Next, press **<*OK*>** to start the creation of the XML files.

The complete database information is now exported to XML files. These XML files are stored in the subfolder **Export** of the database folder.

NOTE: This procedure also allows the user to convert only a part of the local database information to a new connected database, by selecting a subset of the database

entries in step 2.3.7.2 and/or selecting a subset of the available experiment types in step 2.3.7.4.

In case the database contains a fingerprint type based on two-dimensional gels, the original TIFF files also need to be exported.

2.3.7.5 With all database entries still selected, select ***File > Export TIFF files for selected entries***. The *Export TIFF files* dialog box pops up (see Figure 2-24).

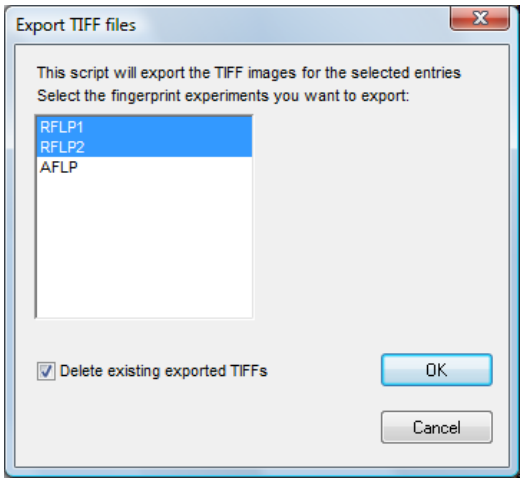


Figure 2-24. The *Export TIFF files* dialog box from the XML Tools plugin.

2.3.7.6 Leave all TIFF files selected and press **<*OK*>**.

2.3.7.7 Close the database.

2.3.7.8 In the Startup screen, create a new, empty connected database as described in 1.5.2.3 to 1.5.2.9. You can leave all settings default.

2.3.7.9 From the *Plugin installation* toolbox that appears, install the XML Tools plugin in the newly created database (see 1.5.3 on the installation of plugins).

2.3.7.10 Select ***File > Import selection as XML*** and browse for the **Export** folder of the exported database. In the *XML import* dialog box (see Figure 2-25), leave all settings default and press **<*OK*>**.

The database information and experiment type information, is now copied from the XML files to the connected database.

In case the exported database contained a fingerprint type based on two-dimensional gels, the TIFF files still need to be imported:

2.3.7.11 Select ***File > Import TIFF files*** and browse for the **Export** folder of the exported database. Select all TIFF files and press **<*Open*>** to import them in the connected database.

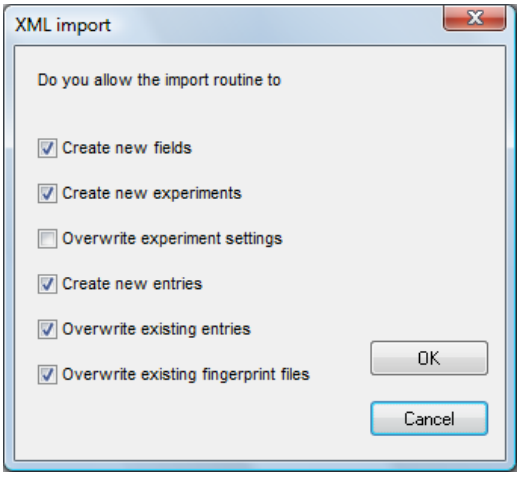


Figure 2-25. The *XML import* dialog box from the XML Tools plugin.

•Option 2: Using the conversion function after setting up an ODBC connection.

FPQuest also offers the possibility to convert an entire local database at once to a new connected database, without the need for the *Database sharing tools* module. 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: This procedure is not recommended to convert a local database into an existing connected database that already contains data, since experiment types with the same name would be overwritten. Converting a local database into a connected database using the XML Tools plugin as described in 2.3.7.1 to 2.3.7.11 is a better option in this case.

To convert a local database into a new connected database, proceed as follows:

2.3.7.12 Create a new empty database in Oracle, SQL Server or Access.

2.3.7.13 Open the local database in the FPQuest main program.

2.3.7.14 In the *FPQuest main* window, select ***Database > Connected databases***.

This opens a list of all currently defined connected databases for this FPQuest database (normally empty at this stage; see Figure 2-21).

2.3.7.15 Click **<*New ODBC*>**.

2.3.7.16 In the *Connected databases configuration* dialog box (see Figure 2-22) that appears, click **<*Build*>**.

2.3.7.17 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.

2.3.7.18 Make sure the connected database is checked as the default database; otherwise, the conversion cannot be executed.

2.3.7.19 Check the table structure of the database, if it does not contain the required tables and fields, press ***<Auto construct tables>*** to allow FPQuest to construct its tables.

2.3.7.20 Once the connection is defined correctly, press ***<OK>*** to close the *Connected databases configuration* dialog box.

2.3.7.21 Close the *Connected databases list* window.

2.3.7.22 In the *FPQuest main* window, 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 (see 2.6.1), 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, FPQuest will automatically restart with the connected database, and the contents of the local database will be removed.

NOTE: If some information fields are not displayed in the Database entries *panel after the conversion, they can be shown by clicking on the column properties button*  *in the database information fields header and selecting them from the pull-down menu.*

2.3.8 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.

2.3.8.1 In the *FPQuest main* window, select ***Database > Connected databases***.

This opens a list of all currently defined connected databases for this FPQuest database (normally just one; see Figure 2-21).

2.3.8.2 Click ***<New ODBC>***.

In the *Connected databases configuration* dialog box that appears, click ***<Build>***.

2.3.8.3 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.

2.3.8.4 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.

2.3.8.5 Under ***Source files location***, select the directory where the source files can be found for this connected database. This directory should always be different from the **Source files** directory of the default connected database.

2.3.8.6 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 *Connected databases list* window. When you open the main program, the contents of the two databases are seen together.

•Closing or deleting a connected database

2.3.8.7 In the *Connected databases list* window (***Database > Connected databases***), select the connected database you want to close, and press ***<Close>***.

2.3.8.8 Confirm with ***<Yes>***. The database disappears from the list, and the contents of the closed database disappears from the *FPQuest main* window.

Closing a connected database is temporary. When it is closed, it will automatically be reopened the next time the FPQuest main 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 were

added externally after FPQuest was started up will be updated in the *FPQuest 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.

2.3.9 Restricting queries

When massive databases are generated, loading the full database into FPQuest might become quite time-consuming and unnecessary for most purposes. To that end, it is possible to load a connected database in FPQuest using a *restricting query* (see also 2.3.3). A restricting query is an SQL query that is used to load only those database entries that comply with the query statement. There are two possibilities of using restricting queries, each serving a more or less different purpose:

1. An automatic query specified in the *Connected databases configuration* dialog box, which will apply each time FPQuest is started up.

2. An interactive one which prompts the user to build a query when FPQuest is started up. Such queries can be saved in query templates and modified or reused at any time. The queries can be built either using a user-friendly graphical query builder, or by typing an SQL query directly, or by combining both.

• Automatic restricting queries

This type of restricting query is particularly useful if you want to work with only one specific group or taxon from the database. For example, if you have a database with a number of species, of which you want to work with only one, you can use the field "Species" to apply a restricting query "Species=...". As a result, only those entries having the specified string in their Species field are loaded. In addition, when new entries are created, they will automatically have the species field filled in. This can save time, help avoid typing errors and restrict users to specific groups of the database.

To specify an automatic restricting query, a restricting query is entered in the input field ***Restricting query*** of the *Connected databases configuration* dialog box (Figure 2-22). A restricting query is of the general format **FieldName=String**. **FieldName** is the name of the field that the restriction is applied to, and **String** is the restricting string. As a result, when the FPQuest 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**.

To try out this feature, you can e.g. install the **DemoBase_SQL** database, as described in 1.3.2. A restricting query to visualize only *Ambiorix* can be entered as follows.:

2.3.9.1 In the *Connected databases configuration* dialog box under Restricting query, type:

```
GENUS=Ambiorix
```

2.3.9.2 Press ***<OK>*** to confirm the changes. The *FPQuest main* window now only shows *Ambiorix*.

2.3.9.3 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 the following as a restricting query:

```
GENUS=Ambiorix;SPECIES=sylvestris
```

The result is a database that only shows *Ambiorix sylvestris*. New entries will automatically be added as *Ambiorix sylvestris*.

NOTES:

(1) Do not use spaces in a restricting query.

(2) If you open a database using a default restricting query, you may not be able to work with gel files, comparisons, libraries, or subsets that contain entries which are not loaded by the restricted query. The program will generate an error message that one or more keys are not present or not loaded in the database. Using the interactive restricting queries, however, missing entries can be loaded during the session if requested (see below).

• Interactive restricting queries

The aim of this type of startup queries is to be able to restrict database loading in a flexible way each time the program is started up. Another source of flexibility in this option is the fact that the software can load additional entries dynamically whenever a gel file, a comparison, a subset, or a library is opened that contains entries which were not loaded by the restricting query used.

2.3.9.4 The interactive queries can be activated by checking ***Prompt at startup*** in the *Connected databases configuration* dialog box (Figure 2-22). As a result, each time the program is started up, an interactive graphical query builder pops up (Figure 2-26).

2.3.9.5 By pressing ***<OK>*** without entering any restricting query, the complete database is loaded.

The interactive restricting query tool is very similar to the *Advanced query tool* described in 2.2.10. It 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***, and ***Subset membership***.

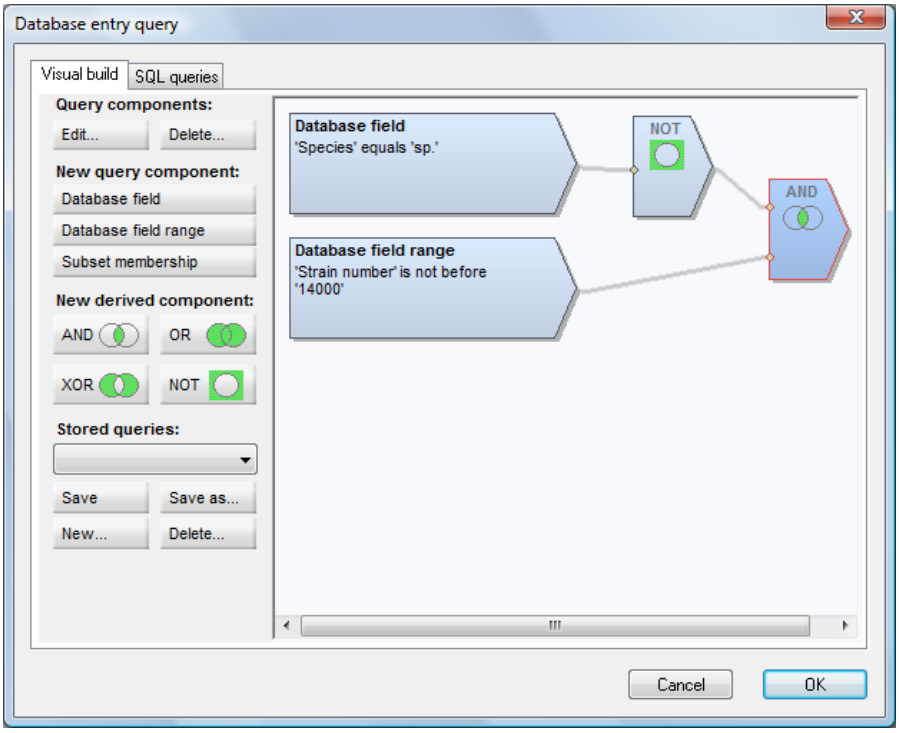


Figure 2-26. The *Interactive query builder*, prompting at startup.

•Database field

Using this component button, you can enter a (sub)string to find in any specific field that exists in the database (Figure 2-27). Note that wildcard characters are not used in this query tool and that the string entered has to match completely with the field contents. The queries are not case sensitive.

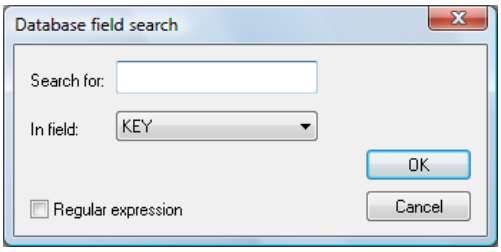


Figure 2-27. *Database field search component* dialog box.

A search string can also be entered as a regular expression (see Section 6.2).

•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 (2.3.7). 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.

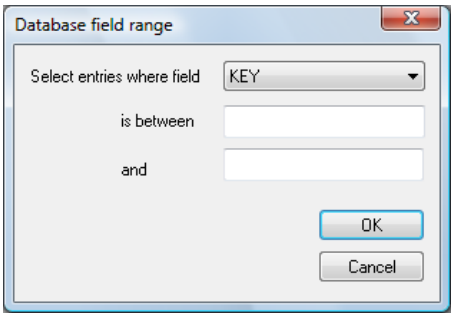


Figure 2-28. *Database field range component* dialog box.

•Subset membership

With this search component, you can specify that only entries belonging to a certain subset should be loaded (Figure 2-29). This option offers additional flexibility as subsets can be composed of any selection of database entries and are not necessarily bound to global query statements.

•Logical operators

NOT, operates on one component. When a component is combined with NOT, the condition of the component will be inverted.

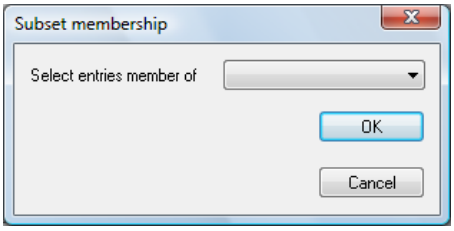


Figure 2-29. ***Subset member component*** **dialog box.**

AND, combines two or more components. All conditions of the combined components should be fulfilled at the same time for an entry to be selected.

OR, combines two or more components. The condition implied by at least one of the combined components should be fulfilled for an entry to be selected.

XOR, combines two or more components. Exactly one condition from the combined components should be fulfilled for an entry to be selected.

NOTES:

(1) The buttons for the logical operators contain a helpful Venn diagram icon that clearly explains the function of the operator.

(2) An example on the use of the logical operators is given in 2.2.10 for the graphical query builder.

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*>**.

2.3.9.6 To view the selected entries, press **<*Add to list*>**.

The entries that were found are highlighted with a colored arrow left from them.

NOTES:

(1) 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.

(2) Multiple components/operators can also be selected together by dragging the mouse over the boxes in the right panel.

2.3.9.7 The second tab of the interactive restricting query builder, ***SQL queries***, contains the actual SQL query statements translated from the active query (Figure 2-30). These SQL statements are passed on to the database to obtain the restricted view.

In principle, the user can compose queries or make changes directly in these fields. This is however not recommended unless you are very familiar with both the SQL language and the FPQuest database table structure. Incorrect SQL query inputs can lead to information partially not being downloaded from the database and might eventually cause the database to become corrupted in case attempts are made to save changes.

When a database is opened with a restricting query, it may occur that an analysis is done which contains entries that are not loaded in the current view. This can happen with gel files, comparisons, subsets, or library units. If such a situation occurs, the program will first generate an error message that one or more keys are not present or not loaded in the database. Next, the program will propose to try to fetch the entries from the database. If you answer **<*Yes*>** the entries will be loaded dynamically from the connected database. In case of a gel file or a subset, this can technically be achieved very quickly. In case of a comparison, however, the operation requires an SQL command to be launched for each additional entry to download. In case of large numbers of additional entries, but also depending on the size of the database and several other factors, this may take considerable time. Therefore, the number of entries to fetch is indicated in the confirmation box (Figure 2-31).

In case the download time to complete a comparison becomes critical, there is a simple workaround by creating or opening a subset and using the feature ***File > Add entries to current subset*** in the *Comparison* window (see also 4.1.7). As soon as this command is executed, the entries from the comparison are added to the current subset and the program automatically retrieves the non-loaded entries. It will prompt to load the entries into the database, and the comparison will be complete at once.

Since library units (see 5.2.1) have physically the same structure as comparisons, the same constraints apply. However, a library unit will never consist of thousands of entries as can be the case with comparisons.

2.3.10 Protecting connected databases with a password

Connected databases can be protected by the use of a password.

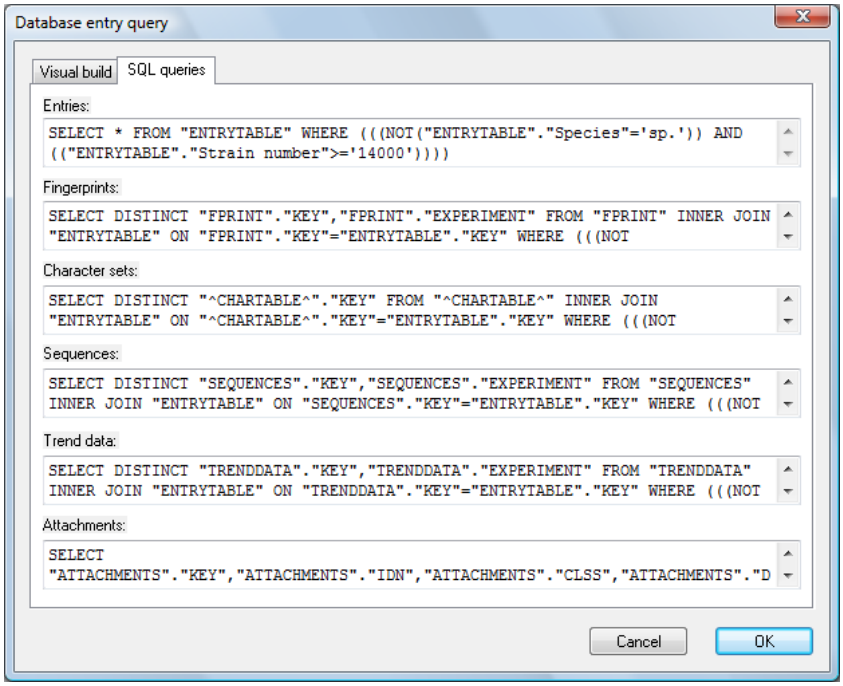


Figure 2-30. SQL query statements translated from a visual query build.

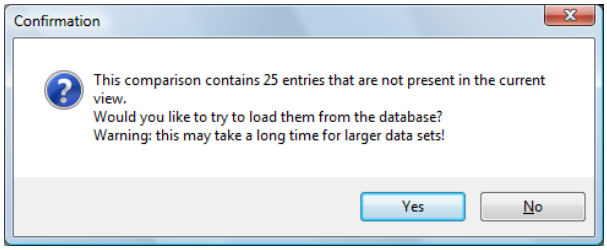


Figure 2-31. Confirmation box to download additional database entries into a comparison.

Access database:

2.3.10.1 Open MS Access and select the 'Open' command in the menu of Access.

2.3.10.2 In the *Open file* dialog box, navigate to the connected database. Click the arrow to the right of the Open button and choose the option '***Open Exclusive***' (see Figure 2-32).

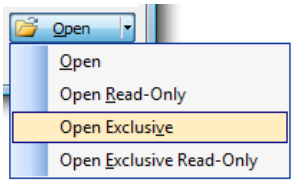


Figure 2-32. Open database for exclusive use.

2.3.10.3 If you are using MS Access 2000 or 2003, go to the **Tools** menu, and select ***Security option > Set database password.*** If MS Access 2007 is installed on your computer, select the *Database Tools* tab and select ***Set Database Password.***

2.3.10.4 A dialog box pops up, asking you to enter and confirm your password (see Figure 2-33).

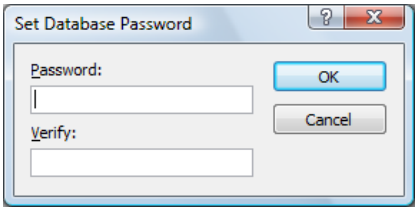


Figure 2-33. *Set Database Password* dialog box.

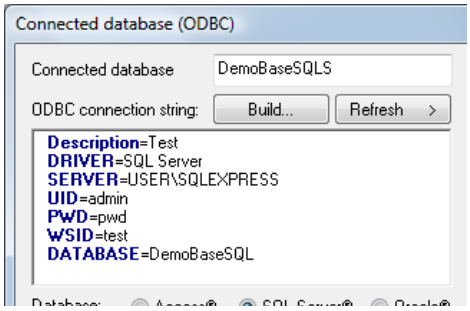
If you close the database in Access and open the database in FPQuest, the program will prompt you for the specified password before loading the database.

Other connected databases:

For all other connected databases (SQL Server, Oracle, …), a username and password are required upon creation of the database. The reason why FPQuest does not prompt for it when loading the database, is because the password is saved in the ODBC string.

2.3.10.5 Open the database and select ***Database > Connected databases***.

The line **"PWD=*password*"** holds the password (see Figure 2-34).



If you want FPQuest to prompt for a username and a password each time you open the database, delete the line **"PWD=*password*"** in the ODBC connection string.

If you want a specific username to be filled in the username box, change the username after **"UID=".**

2.4 Importing data in a FPQuest database

2.4.1 Importing data using the Import plugin

Various options for importing experiment data are described in Chapter 3, which deals with the available experiment types in FPQuest. Many types of data can also be imported using the **Import** plugin, which is installed automatically with the FPQuest software. To activate the Import plugin, select ***File > Install / remove plugins*** in the *FPQuest main* window (see also 1.5.3 on how to install plugins).

The Import plugin allows the following data to be imported:

- **Information field data.** FPQuest database information fields can be imported from text files (tab, comma or semicolon separated) or from other databases (e.g. Access, Excel) via an ODBC link.
- **Fingerprint data from automated sequencers.** Typing techniques for which the electrophoresis step is performed on an automated sequencer (e.g. AFLP, t-RLFP, etc.) can be imported as densitometric curves from the raw chromatogram files. The different file formats from commercially available sequencers (Applied BioSystems, Beckman and Amersham) are supported. See 3.1.13 for more information.
- **Genemapper peak files.** The Genemapper (Applied BioSystems) text files can be imported as fingerprint type. See 3.1.14 for more information.

NOTES:

(1) Some import routines (e.g. automated import of fingerprint files from AB sequencers) are exclusively for data import in connected databases and cannot be used for local databases.

(2) Database information field data and experiment data that are linked to it can be imported directly from another FPQuest database using the XML Tools plugin (see 2.5.3).

For detailed instructions on the use of the Import plugin, we refer to the separate Import plugin manual. A pdf version of this manual becomes available when you click on ***<Manual>*** in the *Plugin installation* toolbox (Figure 1-13).

2.4.2 Importing data via an ODBC link

In a local database, FPQuest 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 FPQuest and an external data source, the user can import data in a completely transparent way into FPQuest. Moreover, the FPQuest local database can be brought up to date using the external data source by performing automatic downloads.

NOTE: The option to configure an external ODBC link is only offered for a local database. However, the use of a connected database is recommended, since there data are stored in a single location (data normalization) and the ODBC link is permanent. This way of working avoids any possible updating conflicts and ensures the data stay up to date. For more information on connected databases, see Section 2.3.

The database records in the external database are mapped into FPQuest entries by making use of the *database key*. The user should specify a unique 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 FPQuest database entry.

2.4.2.1 Setting up the ODBC link

Use the menu item ***Database > ODBC link > Configure external database link*** in the *FPQuest main* window to call the *ODBC configuration* dialog box (see Figure 2-35) This dialog box contains two information fields, which are to be filled in:.

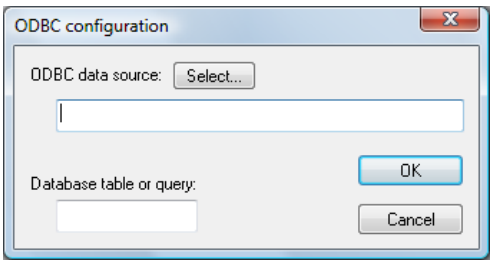


Figure 2-35. The ODBC configuration dialog box.

- **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.

- **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>*** creates the *ODBC database import* dialog box (see Figure 2-36). This dialog box allows the user to specify how each field in the external database should be mapped to a particular field in the FPQuest database. On the left side, the FPQuest fields are listed, while on the right side the external database fields are shown. Initially, all fields are unlinked. You can link two fields by selecting the local FPQuest 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>.***

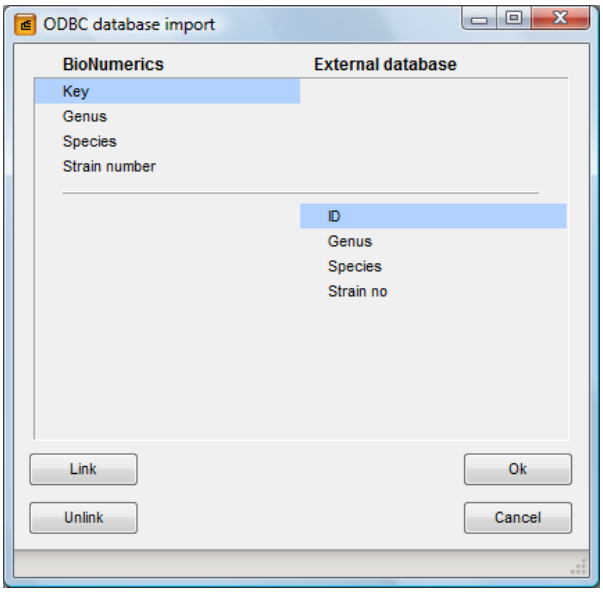


Figure 2-36. The *ODBC database import* dialog box.

Before you will be able to perform any exchange action, you should make sure that the FPQuest '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 FPQuest.

If the necessary links are established between external and local database fields, press ***<OK>*** to validate the ODBC link configuration. At this moment, FPQuest is ready to download information from the external data source.

2.4.2.2 Import of database fields using ODBC

•Update all FPQuest database entries from the external data source

It is possible to automatically update all the information fields from each FPQuest entry, using the data provided by the external database. To this end, select ***Database > ODBC link > Copy from external database*** in the *FPQuest 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 FPQuest software, the program automatically creates new entries in the FPQuest database (after confirmation by the user). In this way, information fields of existing FPQuest database entries 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 FPQuest. To this end, select the empty database field in the *FPQuest main* window (or create a new one), and use the menu command ***Database > ODBC link > 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 FPQuest, 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 visualize the result as a selection list in FPQuest. In the *FPQuest main* window, use the command ***Database > ODBC link > 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 visualized as a selection of the corresponding entries in the FPQuest database (assuming that they are present in the local database).

•Getting a detailed report of the external database record

For each entry in the FPQuest 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.

2.5 Database exchange tools

2.5.1 Solutions for data exchange: bundles and XML files

FPQuest offers two simple and powerful solutions to exchange database information between research sites on a peer-to-peer basis: via *bundles* or *XML files*.

A **bundle** contains selected information (e.g. experiment types, information fields) for a selection of database entries and is the original tool for exchanging FPQuest database information. It is a compact data package contained in a single file, which can be sent to other research sites over the internet. The receiver can open the bundle directly in FPQuest and compare the entries contained in it with the own database. However, the information in a bundle is "as is", and cannot be modified or re-analysed by the receiver.

Exporting FPQuest database information as **XML files** and importing these again in another database is another available exchange tool. Like bundles, selected information can be included for a selection of database entries. When the XML files are imported in a database, the database entries that were contained in the XML files behave just like other database entries.

Which database exchange tool is to be preferred (bundles or XML files), depends on the specific case and will be a trade-off between compactness and flexibility of analysis.

2.5.2 Using bundles in FPQuest

We will illustrate the use of bundles in the **DemoBase** database, by creating a bundle for all entries belonging to the genus *Vercingetorix*.

2.5.2.1 In the *FPQuest main* window with **DemoBase** loaded, select all entries belonging to *Vercingetorix* (see 2.2.8 on how to select database entries).

2.5.2.2 Select ***File > Create new bundle*** or  .

The *Create new bundle* dialog box (Figure 2-37) lists the available database information fields in the left panel and all available experiment types in the right panel.

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.

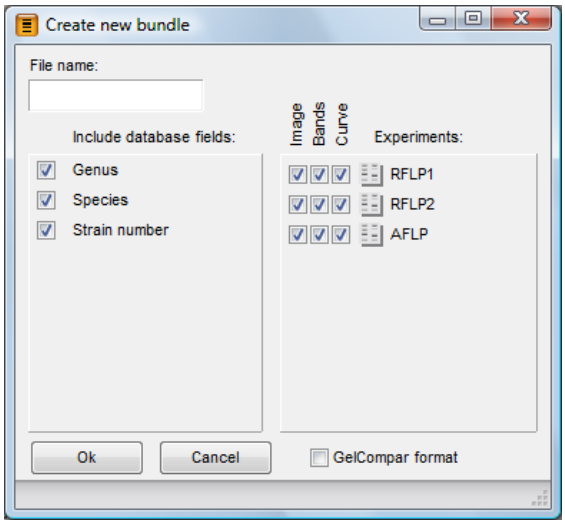


Figure 2-37. The *Create new bundle* dialog box.

2.5.2.3 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. FPQuest also recognizes and reads GelCompar and Molecular Analyst Fingerprinting bundles.*

2.5.2.4 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 2-2 for the directory structure).

Besides the numerical information of the experiments, a bundle contains all the information of the experiment type, so that FPQuest 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 automatically be 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.

The bundle holds the complete information about the reference system used and the molecular weight regression, so that FPQuest can automatically remap the bundle fingerprints to be compatible with the database fingerprints.

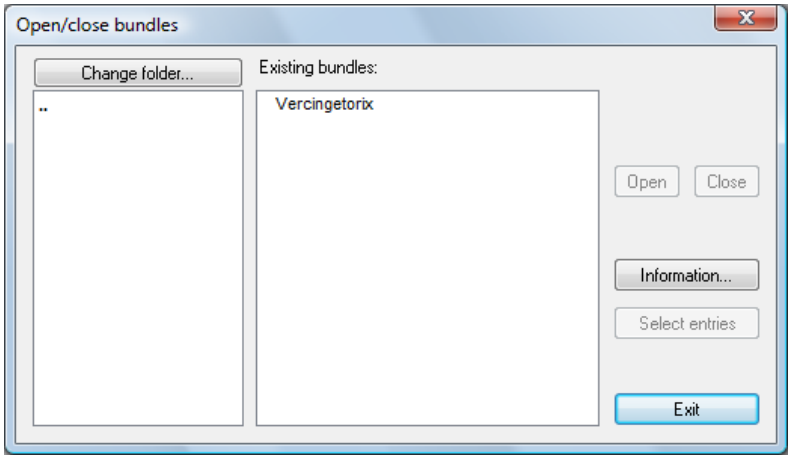


Figure 2-38. The *Open/close bundles* dialog box.

2.5.2.5 As an example for database exchange, copy **Vercingetorix.BDL** to the **Bundles** directory of database **Example**.

2.5.2.6 Close FPQuest and restart the main program under database **Example**.

2.5.2.7 In database **Example**, select ***File > Open bundle*** or press the  button.

In the *Open/close bundles* dialog box (Figure 2-38), you can browse to the local or network path where the bundle files can be found with the ***<Change folder>*** button. The default path is the **Bundles** subdirectory of the current database. In the right panel, you can select a bundle in the list of available bundles in the specified path.

2.5.2.8 Select **Vercingetorix** and press the ***<Information>*** button.

This opens the *Bundle information* dialog box for the selected bundle (Figure 2-39). 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.

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, FPQuest would add a new information field to the database. To avoid this, you can rename the information fields in the bundle.

2.5.2.9 Select **Strain number** and press the ***<Rename>*** button under the information fields panel.

2.5.2.10 Enter **Strain no** and press ***<OK>***.

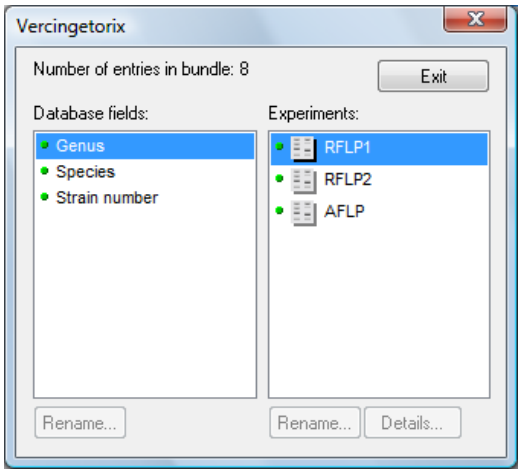


Figure 2-39. The *Bundle information* dialog box.

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 FPQuest 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 experiments panel.

2.5.2.11 ***<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 "+" in the *Open/close bundle* dialog box.

In the database, entries from a bundle are recognized by the name "Bundle" in the *Location* information field. If the *Location* information field is not displayed in the database, it can be shown by clicking on the column properties button  in the database information fields

header and selecting it from the pull-down menu. For all functions, they behave like normal database entries. If you exit FPQuest, 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.

2.5.2.12 You can select all entries from an opened bundle by pressing the **<*Select entries*>** button in the *Open/close bundle* dialog box.

2.5.2.13 To close a loaded bundle, select it in the list and press the **<*Close*>** button.

2.5.2.14 Press **<*Exit*>** to close the *Open/close bundle* dialog box.

NOTE: If you want a bundle to be always opened with the database when FPQuest is started up, you should rename it to contain the prefix @_ before its name and the .bdl bundle file should be placed in the Bundles folder of the corresponding database.

2.5.3 Export and import using XML files

The tools to export and import database entries as XML files are available as a plugin. To activate the XML Tools plugin, select ***File > Install / remove plugins*** in the *FPQuest main* window (see also 1.5.3 on how to install plugins). A detailed description on how to use the XML Tools can be found in the XML Tools plugin manual.

2.6 Taking backups from a FPQuest database

In many cases, FPQuest 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. and take backups on regular intervals.

The location where the data is stored - and therefore the directories to backup - is different whether a local or a connected database is used. For more details on how information is stored in local and connected databases, see Section 2.1.

2.6.1 Backing up a local database

In a local database setup, all data files that belong to a particular FPQuest database are stored on the hard disk in subdirectories of a single top directory that has the database name (see also Figure 2-2). If FPQuest is opened with this database, this directory is indicated in the status bar on the bottom of the *FPQuest main* window. Alternatively, the corresponding directories of all databases can also be displayed at once in the FPQuest Startup screen, by clicking on the Column properties button () in the information fields header and selecting ***Path*** from the drop-down list. [HOMEDIR] in the path refers to the home directory as specified in the settings. To find out what the current home directory is, or to modify the home directory, press the Settings button () and select ***Change home directory***. A dialog box appears which shows the currently selected home directory. For more information about the FPQuest home directory, see 1.1.4.

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: Backups restored from CD or DVD may be read-only. In this case you will have to specify the files to be write-accessible before you run FPQuest with the restored database.*

It is possible to create a duplicate of a local database in a similar way. To this end, copy the entire contents of the database's top directory to a new directory. In the FPQuest Startup 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 **<*Browse*>** 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?*" In the *Setup new database* dialog box, select ***Local database (single user only)*** to finish the creation of the database.

2.6.2 Backing up a connected database

In a connected database setup, the actual data may be stored outside the FPQuest data folder (see Figure 2-2). The location of the connected database and associated source files can be found as follows:

2.6.2.1 Open the database in FPQuest and select ***Database > Connected database*** in the *FPQuest main* window.

2.6.2.2 In the *Connected databases* dialog box, select the currently defined connected database and click **<*Edit*>**.

The ODBC connection string in the *Connected database configuration* dialog box (top left panel, see Figure 2-34) contains the database name and location. The location of the source files is shown in the bottom right of the same dialog box.

To ensure completeness, both the connected database and the source files should be backed up.

In case of an **automatically created connected database** (default setting when creating a new database), the connected database (.mdb) and the source files folder are located in the top directory that has the database name. This is indicated in the *Connected database configuration* dialog box as [DBPATH]. [DBPATH] refers to the database folder in the FPQuest home directory as specified under Settings () in the Startup screen. Therefore, backing up this complete directory (including subdirectories) is sufficient 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.

In case of **custom created databases or when FPQuest was connected to an already existing database**, the user needs to check the connected database and source file location in the *Connected database configuration* dialog box and back them up separately.

Professional DBMS such as SQL Server, Oracle, MySQL, etc. can be configured to take automatic backups on

regular time intervals. We refer to the DBMS documentation for the setup of such automatic backups.

3. EXPERIMENTS

3.1 Setting up fingerprint type experiments

3.1.1 Introduction

The type of experimental data that can be analysed in FPQuest is referred to as *fingerprint types*. 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. Fingerprint types can be derived from TIFF or bitmap files as well, which are two-dimensional bitmaps. The condition is that one must be able to translate the patterns into densitometric curves.

The user can create more than one experiment of the same type. For example, you can create two different fingerprint types, to analyze Pulsed Field Gel Electrophoresis (PFGE) gels obtained with two different restriction enzymes. Each fingerprint type can have its own specific settings such as reference marker, MW regression, stain, band matching tolerance, similarity coefficient, clustering method, etc.

3.1.2 Defining a new fingerprint type

The steps involved in data processing of fingerprint types will be illustrated with an example TIFF file from the **Sample and Tutorial data\Sample gel image file** folder on the CD-ROM. This directory contains a gel **Gel_01.tif**. The same gel file is also available from the download page of the website (www.bio-rad.com/softwaredownloads).

3.1.2.1 Create a new database (see 1.5.2).

3.1.2.2 In the *FPQuest main* window, select ***Experiments > Create new fingerprint type*** from the main menu, or press the  button from the *Experiments* panel toolbar and select ***New fingerprint type***.

3.1.2.3 The *New fingerprint type* wizard prompts you to enter a name for the new type. Enter a name, for example **RFLP**.

3.1.2.4 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).

3.1.2.5 After pressing ***<Next>*** again, the wizard asks whether the fingerprints have inverted densitometric values. This is the case when you are using ethidium bromide stained gels, photographed under UV light (such as the example provided). The bands then appear as fluorescent lighting on a black background. Since FPQuest 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 can leave the colors unaltered.

3.1.2.6 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.

3.1.2.7 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 *Experiments* panel (Figure 1-15) now lists **RFLP** as a fingerprint type.

3.1.2.8 Click the  button in the *Experiment files* panel or select ***File > Add new experiment file*** in the *FPQuest main* window.

3.1.2.9 Select the file **Gel_01.tif** from the **Sample and Tutorial data\Sample gel image file** folder on the CD-ROM or from the downloaded and unzipped folder.

3.1.2.10 The software now asks "Do you want to edit the image before adding it to the database". Answer ***<Yes>*** to open the *Image import* editor.

The selected file is opened in the *Fingerprint image import* editor, an editor which allows the user to perform a number of preprocessing functions on the image (Figure 3-1). These functions include flipping, rotating and mirroring the image, inverting the image color, converting color images to grayscale, and cropping the image to defined areas.

> *NOTES: (1) It is possible to skip the* Fingerprint image import *editor and copy the file directly to the database, by answering* **<No>** *to the question in*

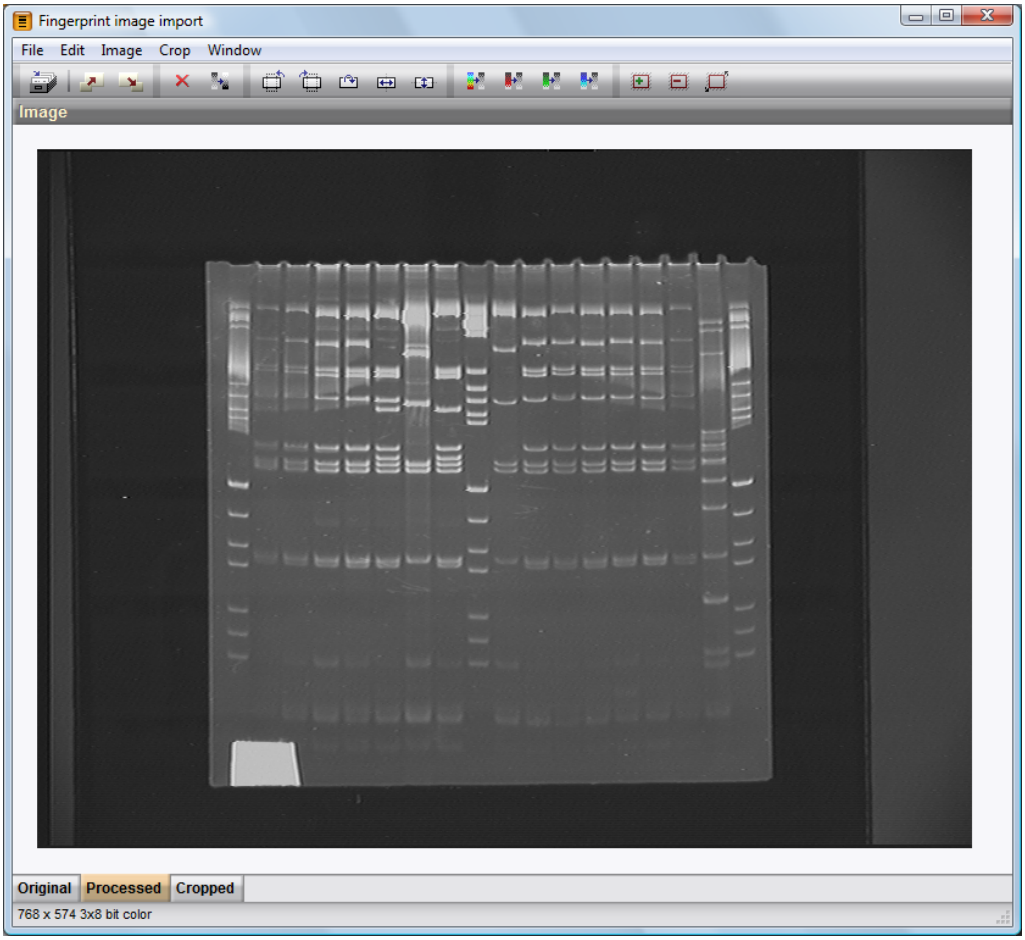


Figure 3-1. The *Fingerprint image import* window.

3.1.2.10. In case you skip this step, make sure the file is an uncompressed grayscale TIFF file, which is the only format recognized by the FPQuest database. Continue in this case with paragraph 3.1.3.

(2) The Fingerprint import image *editor supports most known file types such as JPEG, GIF, PNG and compressed TIFF files in grayscale or RGB color. For the conversion to an uncompressed grayscale TIFF file see 3.1.2.12 (***Image > Convert to gray scale***).*

The *Fingerprint image import* window consists of three tabs: **Original**, **Processed**, and **Cropped**.

3.1.2.11 In the **Original** tab, the unprocessed image is shown. In the **Original** view, you can zoom in ( ) or zoom out ( ), and save the image to the database (or ***File > Add image to database***). The image can only be saved when it is in gray scale mode (see below).

3.1.2.12 In the **Processed** tab, the same options are available as in the **Original** tab, plus a number of image editing tools. These include:

- Inverting the color (or ***Image > Invert***) to invert images that have a black background, for example gels that were stained with ethidium bromide.

- Rotating the image 90° left (or ***Image > Rotate > 90° left***), 90° right (or ***Image > Rotate > 90° right***), or 180° (or ***Image > Rotate > 180°***).

- Mirroring the image horizontally (or ***Image > Mirror > Horizontal***) or vertically (, ***Image > Mirror > Vertical***).

- Average RGB colors to gray scale (or ***Image > Convert to gray scale > Averaged***), or convert a single channel to gray scale, either red (or ***Image > Convert to gray scale > Red channel***), green (or ***Image > Convert to gray scale > Green channel***) or blue (or ***Image > Convert to gray scale > Blue channel***).

3.1.2.13 The editor also allows you to crop the image to a selected area, to which the following functions are available:

- ***Crop > Add new crop*** or  , to add a new crop mask to the image. The crop mask can be moved by clicking anywhere inside the rectangle and dragging it to another position, or resized by clicking and dragging the bottom right corner of the rectangle.

- ***Crop > Rotate selected crop*** or  , to rotate the crop mask over a defined angle. Rotating the crop mask over an angle different from 90° or 180° will cause the program to recalculate densitometric values based upon interpolation, which means that the quality of the image may slightly decrease. This action is therefore not recommended unless it is inevitable.

- ***Crop > Delete selected crop*** or  is to delete the crop mask that is currently selected. Note in this respect that the program allows multiple crop masks to be defined for a single image. The final image that will be saved to the database, will be composed of all cropped areas aligned horizontally next to each other.

3.1.2.14 With ***Image > Expand intensity range***, it is possible to recalculate the pixel values of the image so that they cover the entire range within the OD depth of the file, e.g. 8-bit = 256 gray levels, 16-bit = 65,536 gray levels.

3.1.2.15 The image can be reset to its original state with ***Image > Load from original*** or by pressing  .

3.1.2.16 To edit this gel, convert it to gray scale by averaging the 3 channels () and define a crop mask within the gel borders, excluding the black area at the left bottom, but including the full patterns.

3.1.2.17 The third tab, *Cropped*, displays the result of the image as defined by the crop mask(s). When you are satisfied with the result of the preprocessing, you can save the image to the database using the  button.

3.1.2.18 Give the gel a name and exit the *Fingerprint image import* window with ***File > Exit***.

One gel becomes available in the *Experiment files* panel, Gel_01. The file is marked with **N**, which means that it has not been edited yet (see Figure 3-2).

Figure 3-2. The *Experiment files* panel after import of a gel.

Any other gel TIFF file you want to process can be imported in the same way in the current database. The program will list these TIFF files in the *Experiment files* panel.

NOTE: Experiment files added to the Files *panel can be deleted by selecting the file and choosing* **File > Delete experiment file** *from the main menu or clicking on the*  *icon in the* Files *panel toolbar. 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** *or clicking on the*  *icon again.*

3.1.3 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.

3.1.3.1 Click on Gel_01 in the *Files* panel (see Figure 3-2), and then select ***File > Open experiment file (data)*** in the main menu.

Since the gel is new (unprocessed), FPQuest does not 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**.

3.1.3.2 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 3-3), showing the image of the gel.

NOTE: In a local database, the gel can be mirrored with **File > Tools > Vertical mirror of TIFF image** *or*

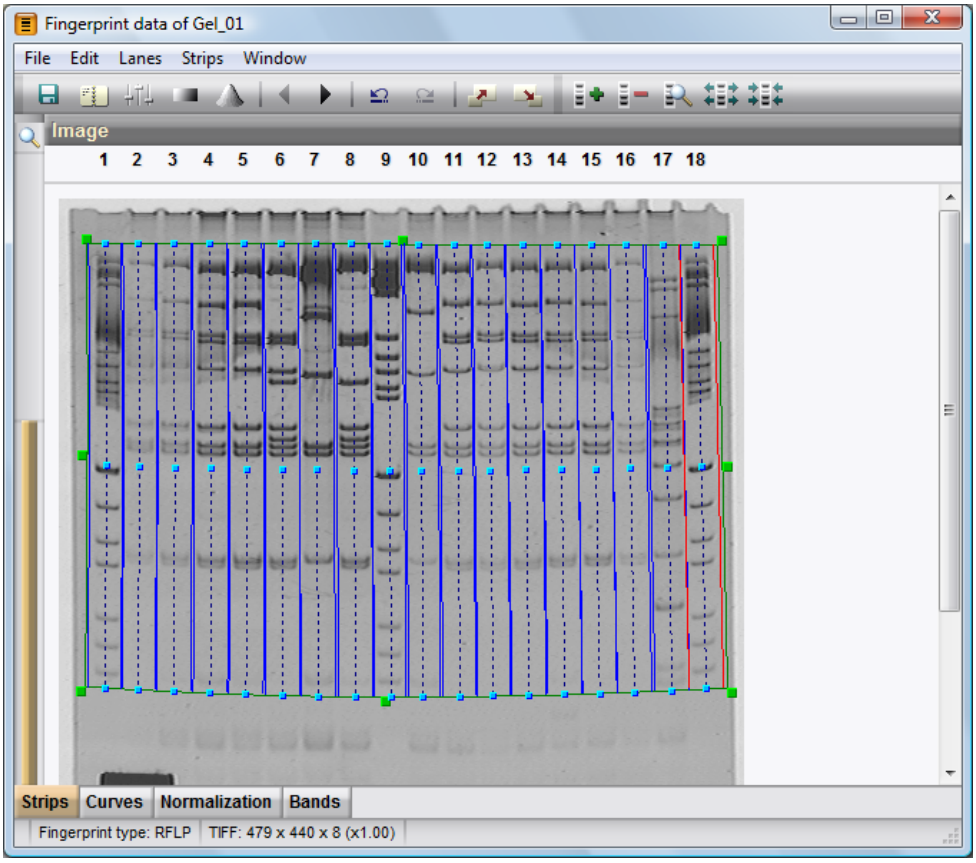


Figure 3-3. The *Fingerprint data editor* window. Step 1: defining pattern strips.

File > Tools > Horizontal mirror of TIFF image. *These commands are equivalent to the commands available in the* Image import *editor (see 3.1.2.12).*

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 tabs in the bottom left corner of the window: **1. Strips** (defining lanes), **2. Curves** (defining densitometric curves), **3. Normalization**, and **4. Bands** (defining bands and quantification). The tabs themselves can be used for navigation between the different steps and allow you to 'skip' steps, e.g. to return in one click from **Normalization** to **Strips** when it turns out a lane was not properly defined. When processing a new gel image, however, it is not recommended to skip any steps in the process.

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.

3.1.4 Defining pattern strips on the gel

3.1.4.1 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, respectively. Shortcuts are CTRL+PageUp and CTRL+PageDown on the keyboard. The zoom slider (left of the *Image* panel in default configuration) offers a convenient alternative for zooming in and out on the gel image. See 1.6.7 for a detailed description of the zoom slider functions.

3.1.4.2 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, press the space bar or right-click and select ***Navigator*** from the floating menu.

3.1.4.3 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 3-4).

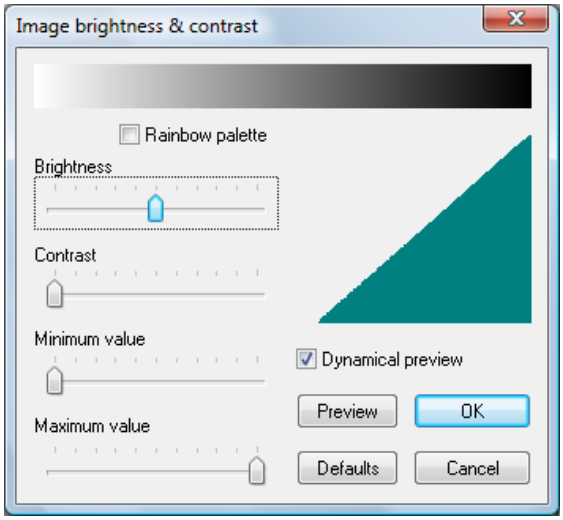


Figure 3-4. ***Image brightness & contrast*** **dialog box.**

3.1.4.4 In the *Image brightness & contrast* dialog box, click ***Dynamical preview*** to have the image directly updated with changes you make.

3.1.4.5 Use the ***Minimum value*** slide bar to reduce background if the background of the whole image is too high.

3.1.4.6 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.

3.1.4.7 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* Gel tone curve *editor, as explained further, is a more powerful image enhancement tool for which the settings are saved for each particular gel.*

3.1.4.8 With ***File > Show 3D view*** or  , a three dimensional view of the gel image can be obtained in a separate *3D view* window (Figure 3-5).

3.1.4.9 In the *3D view* window, you can use the **Left**, **Right**, **Up** and **Down** arrows keys on the keyboard, to turn the position of the image in all directions. The image can also be rotated horizontally and vertically by dragging the image left/right or up/down using the mouse.

3.1.4.10 You can change the zoom factor using ***View > Zoom in*** (PgDn) or ***View > Zoom out*** (PgUp).

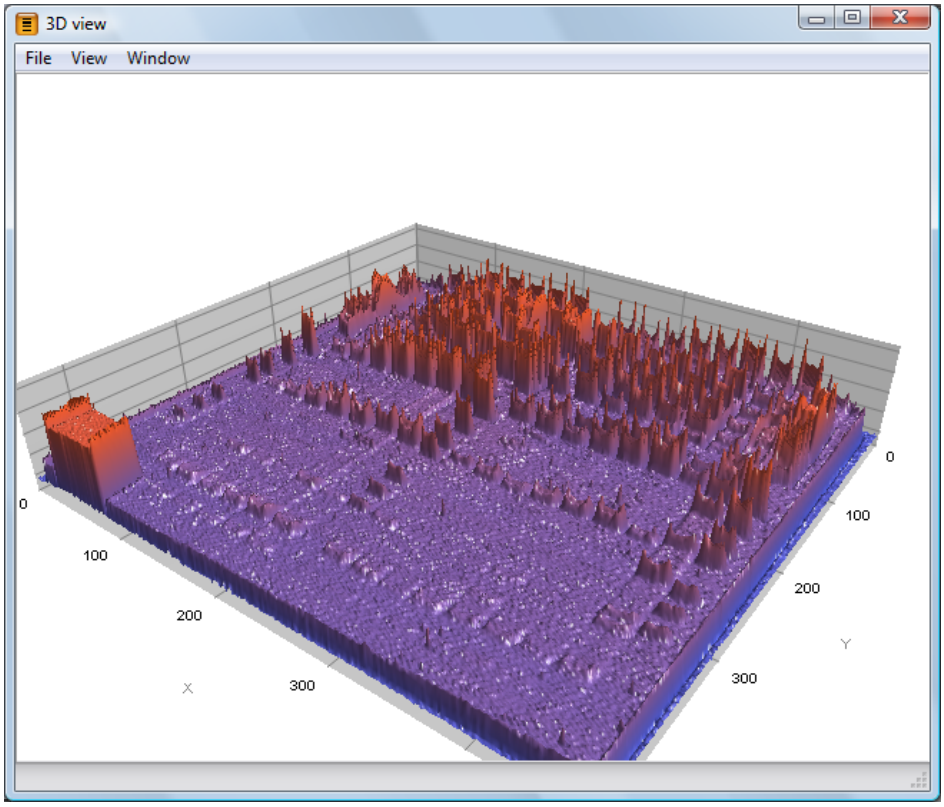


Figure 3-5. The ***3D view*** **window.**

3.1.4.11 You can also change the vertical zoom (Z-axis showing the peak height) with ***View > Higher peaks*** (INS) or ***View > Lower peaks*** (DEL).

NOTE: In the three further steps of the Fingerprint data editor *window (2. Curves, 3. Normalization, and 4. Bands), the* 3D view *window can also be popped up, showing only the selected lane image rather than the entire gel image.*

3.1.4.12 Close the *3D view* window with ***File > Exit***.

3.1.4.13 To save the work done at any stage of the process, you can select ***File > Save***, press CTRL + S, 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 gel resolution is explained further (see 3.1.10.2). If the question appears (not the case for the example gel), answer ***<Yes>***.

The green rectangle is the *bounding box*, which delimits the region of interest of the gel: tracks and gelstrips will be extracted within the bounding box.

3.1.4.14 To move the bounding box as a whole, hold down the CTRL key while dragging it in any of the green squares (*distortion nodes*).

3.1.4.15 Adjust the box by dragging the distortion nodes as necessary: corner nodes can be used to resize the box in two directions, whereas inside nodes can only be used to resize one side of the box.

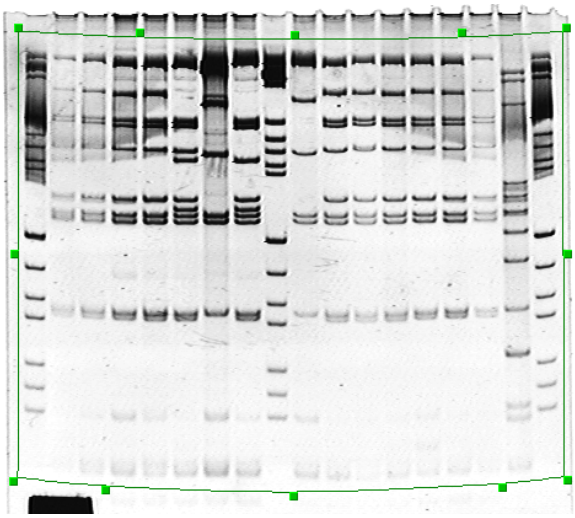


Figure 3-6. Defining the bounding box to follow contours of distorted gel.

3.1.4.16 By using the SHIFT key, one can even distort the sides of the rectangle. Holding the SHIFT key while dragging the corner nodes will change the rectangle into a non-rectangular quadrangle (parallelepiped).

3.1.4.17 A curvature can be assigned to the sides of the bounding box by holding the SHIFT key while dragging one of the inside nodes in any direction (see Figure 3-6, top and bottom sides).

3.1.4.18 On the top and bottom sides of the bounding box, more nodes can be added using ***Lanes > Add bounding box node***. While holding down the SHIFT key, a node can be dragged to the left or to the right using the mouse.

3.1.4.19 A node can be deleted from the bounding box using ***Lanes > Delete bounding box node***.

NOTES:

(1) Following the curvature of a distorted gel is not crucial, as this is normally corrected in the normalization step (see further, 3.1.6) in case there are sufficient reference lanes on the gel. However, as it will provide a first rough normalization, it can aid the automatic or manual assignment of bands as explained in 3.1.6. Also, the software allows the bounding box curvature to be used for rectifying sloping or "smiling" lanes (e.g. Figure 3-6, outer lanes), if this option is enabled (see the Fingerprint conversion settings *dialog box, Figure 3-7, and explanation below).*

(2) If you are running an upgrade from an older FPQuest version (prior to 4.0) and using a connected database, the column BOUNDINGBOX in the connected database may not be long enough to hold an increased number of nodes. To resolve this, perform **<Auto construct tables>** *in the* Connected database setup *window (see Figure 2-22).*

3.1.4.20 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.

3.1.4.21 Enter 18 as the number of tracks in Gel_01 press ***<OK>***.

Each lane found 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 3-8). By default, the strip thickness is 31 points, which is too wide in this example.

3.1.4.22 Call the *Fingerprint conversion settings* dialog box with ***Edit > 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 3-7).

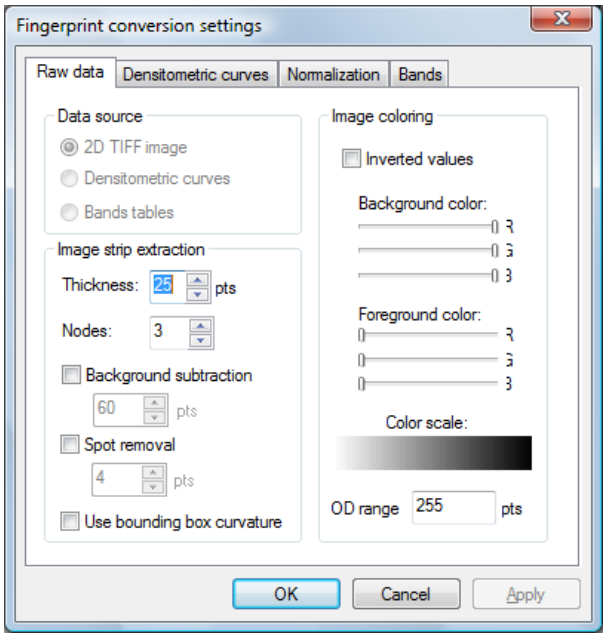


Figure 3-7. The *Fingerprint conversion settings* dialog box, *Raw data* tab.

3.1.4.23 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 3-8 for an optimally adjusted example.

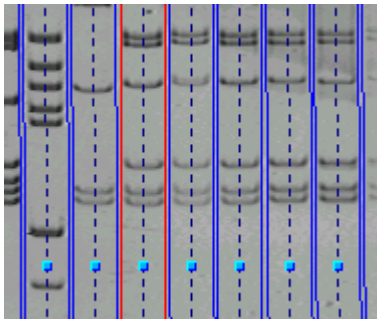


Figure 3-8. Optimal strip thickness settings, detail.

3.1.4.24 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 scans 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 spot removal, the less the distortion.

(2) If background subtraction on the gelstrips is applied, it is not necessary anymore to perform background subtraction 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.

Using the option ***Use bounding box curvature***, it is possible to have the program correct smiling or sloping bands due to distortion in the gel. The bands will be rectified according to the bounding box curvatures defined (3.1.4.17). An example is given in Figure 3-6, where the bounding box has been assigned a curvature to follow the distortions in the outer lanes. The result of enabling the correction for bounding box curvature is shown in Figure 3-10, where it can be clearly seen that the bands of the outer lanes have been straightened.

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

3.1.4.26 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.

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

3.1.4.28 Remove a selected lane with ***Lanes > Delete selected lane*** or DEL or  if necessary.

3.1.4.29 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***.

3.1.4.30 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 powerful tool to edit the appearance of the image is the *Gel 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 *Gel 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 (65,536) 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.

3.1.4.31 Select the *Image brightness and contrast* box with ***Edit > Change brightness & contrast*** or with  , and press ***<Defaults>*** to restore the defaults.

3.1.4.32 In the *Fingerprint data editor* window menu, select ***Edit > Edit tone curve***. The *Gel tone curve* editor appears as in Figure 3-9.

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.

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

3.1.4.34 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.

3.1.4.35 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

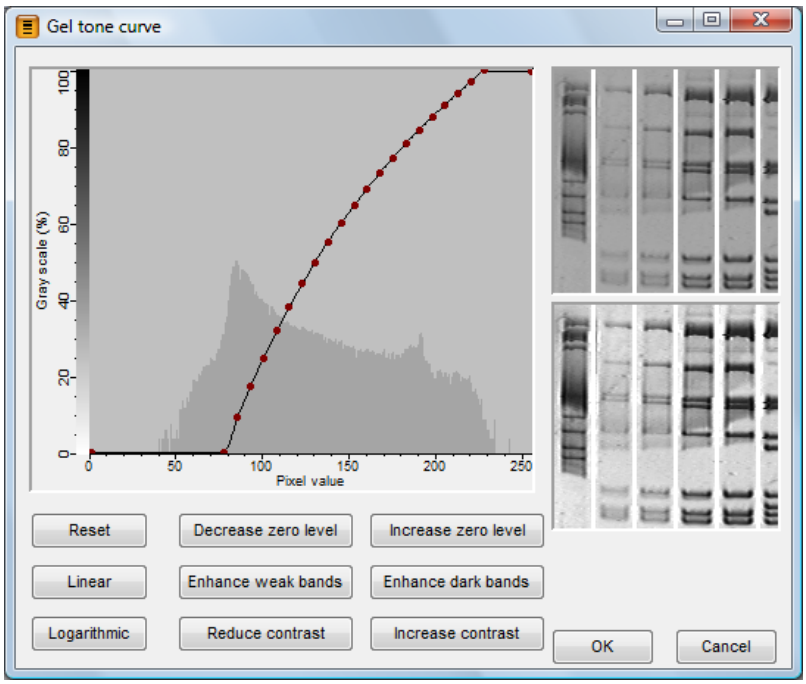


Figure 3-9. The *Gel tone curve* editor.

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.

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

3.1.4.37 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 in the Tone curve *window, 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.*

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

3.1.5 Defining densitometric curves

In this step, the window is divided in two panels (Figure 3-10): 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.

3.1.5.1 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.

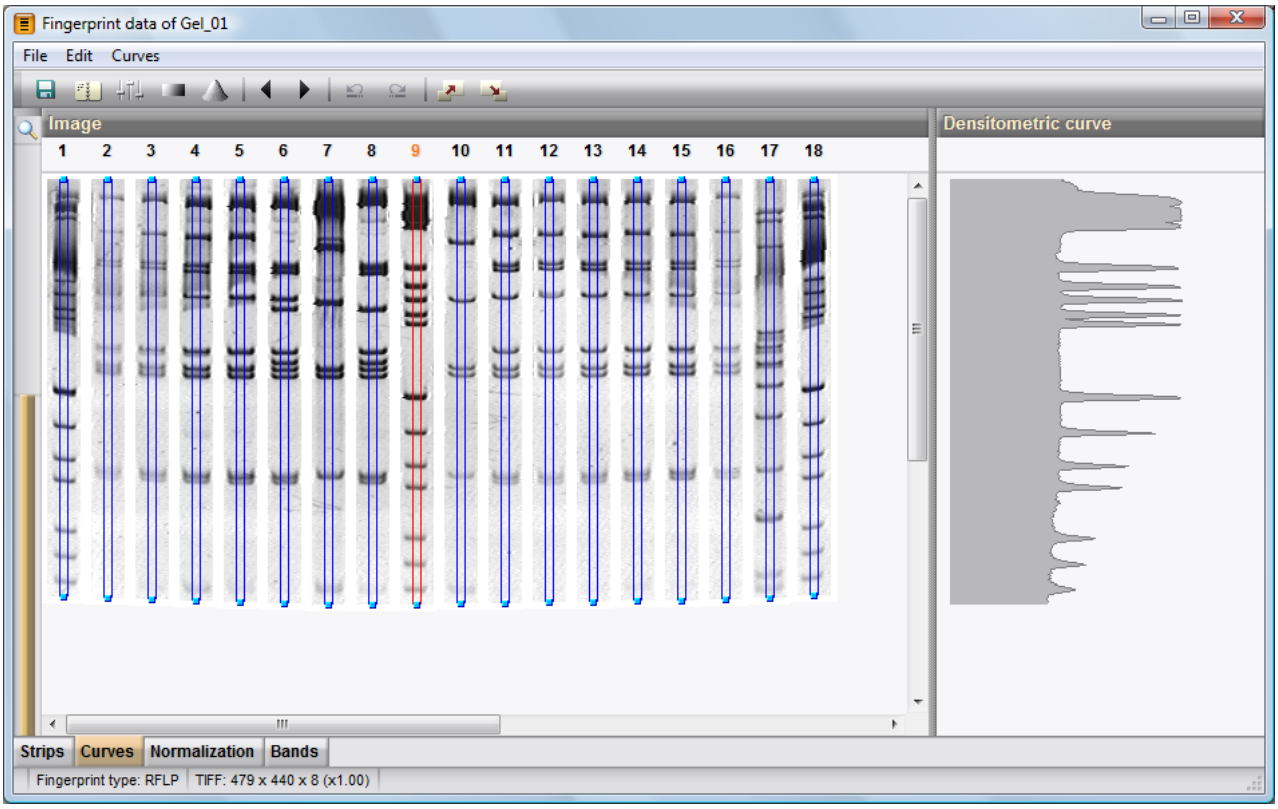


Figure 3-10. The *Fingerprint data editor* window. Step 2: defining densitometric curves.

3.1.5.2 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.

3.1.5.3 Call the *Fingerprint conversion settings* dialog box with ***Edit > Edit settings*** or  . This time, the *Densitometric curves* tab is displayed (Figure 3-11).

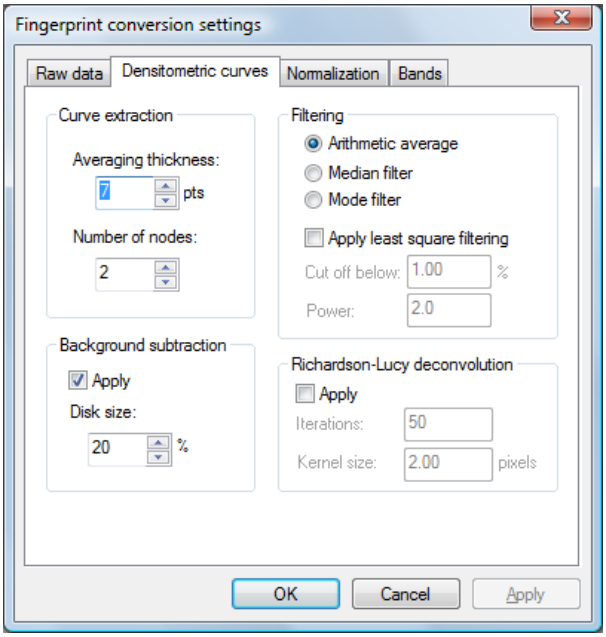


Figure 3-11. The *Fingerprint conversion settings* dialog box, *Densitometric curves* tab.

3.1.5.4 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 3-12).

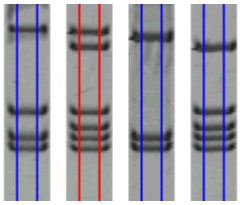


Figure 3-12. Optimal settings for curve averaging thickness.

3.1.5.5 Select ***Edit > 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 3.1.2.6), because we will see how we can have the program propose the optimal settings.

Filtering is a method to make an average of the values within a 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 3-13 illustrates the effect of the Median filter on a small spot. The latter two 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.

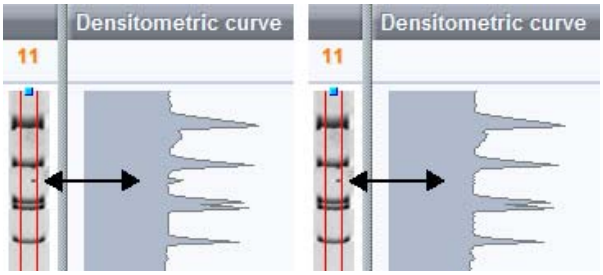


Figure 3-13. Result of Arithmetic average filtering (left) and Median filtering (right).

3.1.5.6 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 necessary. For now, we leave this parameter disabled.

Richardson-Lucy deconvolution is a method to *deblur* (sharpen) one-dimensional and two-dimensional arrays. This function sharpens and enhances the contrast of peaks in the densitometric curves. While peaks will become sharper, noise also will increase. Deconvolution actually does the opposite of least-square filtering. Since the method is iterative, the number of ***Iterations*** can be set (default 50). The more iterations, the stronger deconvolution will be obtained. The ***Kernel size*** (default 2.00) determines the resolution of the deconvolution: the smaller this value is set, the more shoulders will be split into separate peaks.

3.1.5.7 Press **<*OK*>** to save the settings.

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

3.1.5.8 Select ***Curves > Spectral analysis***. This shows the *Spectral analysis* window (Figure 3-14).

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 3-14 shows an optimal setting of 0.89%.

The Background scale is an estimation of the disk size for background subtraction. The figure shows a setting of 11%.

3.1.5.9 Call ***Edit > Edit settings*** again and specify the background subtraction and the least square filtering.

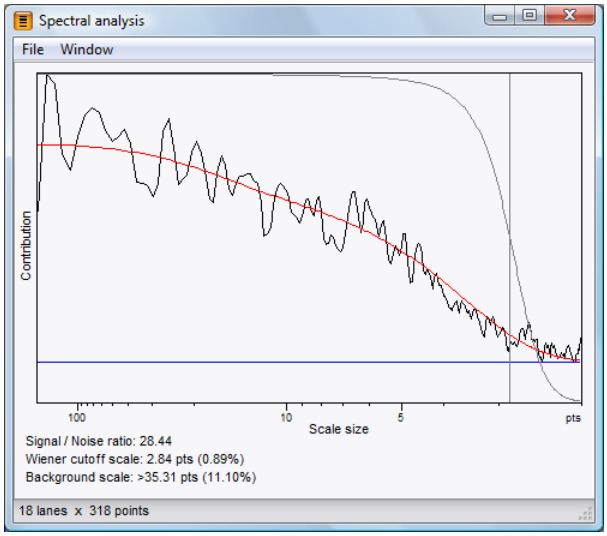


Figure 3-14. Spectral analysis of the patterns of a gel.

3.1.5.10 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.

3.1.5.11 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.

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

3.1.6 Normalizing a gel

In the Normalization step, the *Fingerprint data editor* window consists of three panels (Figure 3-15): 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).

We proceed as follows:

3.1.6.1 Select the first reference pattern (lane 1 on the example) and ***Reference > Use as reference lane*** or  (keyboard shortcut CTRL+R). Repeat this action for all other reference lanes (lanes 9 and 18 on the example).

3.1.6.2 Select ***Normalization > Show normalized view*** or press  .

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

3.1.6.4 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.

3.1.6.5 Use the provided scheme (see Figure 3-16) to enter all reference positions on the example gel.

Figure 3-15. The *Fingerprint data editor* window. Step 3: normalization.

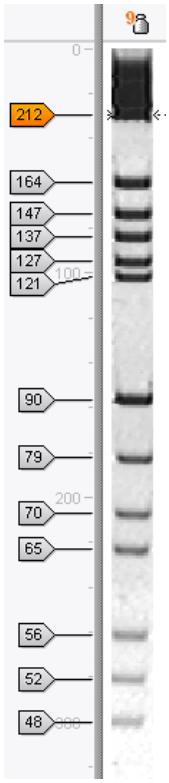


Figure 3-16. Band sizes of the reference positions on the example gel.

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 (3.1.16).

> *NOTE: Our current gel shows "**No active reference system defined**" in the left panel (see Figure 3-15). 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:

3.1.6.6 Click on a label of a reference position, or wherever on the gel at the height of the reference position.

3.1.6.7 Then, hold the CTRL key and click on the reference band you want to assign to that reference position.

3.1.6.8 Repeat this action for all other reference bands you want to assign to the same reference position.

3.1.6.9 Repeat actions 3.1.6.6 to 3.1.6.8 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.

3.1.6.10 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:

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

3.1.6.12 Remove all the manual assignments by ***Normalization > Delete all assignments***.

To let the program assign the bands and reference positions automatically, select ***Normalization > Auto assign*** or  . This will open the *Auto assign reference bands* dialog box (Figure 3-17). Under ***Search method***, two options are available: ***Using bands*** and ***Using densitometric curve***.

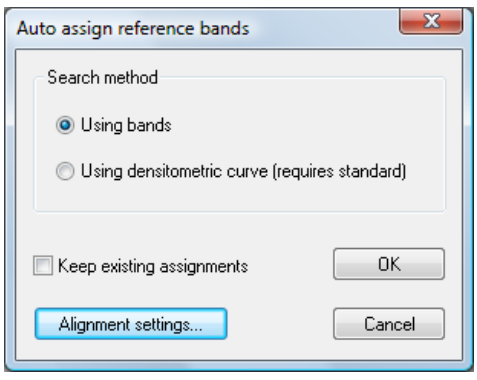


Figure 3-17. The *Auto assign reference bands* dialog box.

In the ***Using bands*** option, 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.

In the ***Using densitometric curve*** option, a different algorithm is used, which matches the densitometric curve of standard pattern with the curves of the reference patterns. Obviously, the option requires a standard to be defined (see 3.1.10.3 to 3.1.10.7 on how to define a standard). This method employs a pattern matching algorithm that works best for complex banding patterns, but is less suitable for simple patterns such as molecular weight ladders.

An option independent of the search method is ***Keep existing assignments***. When this option is chosen, any assignments made previously are preserved. This option allows the user to assign a few bands manually and let the program automatically assign the remaining bands on the reference patterns. This way of working is useful to provide some initial help to the algorithm in case of very distorted or difficult gels.

Pressing ***<Alignment settings>*** opens the *Alignment settings* dialog box (Figure 3-18). Parameters can be adjusted for the peak detection, global alignment and local alignment algorithms:

The **Peak detection parameters** determine what is recognized by the program as a peak.

- ***Threshold*** is the minimal height, expressed as a percentage of the highest peak in the profile, for which an elevation in the profile is still considered to be a peak. The default value is 2%.

- The ***Valley depth*** is important for peak separation: it is the minimal depth of the depression between two subsequent maxima, for which the program divides a single peak into two separate peaks. In case one maximum is higher than the other, the height between the lowest maximum and the minimum is used. Similar to the threshold, the valley depth is expressed as a percentage of the highest peak in the profile. The default value is 2%.

In a **Global alignment**, the profile as a whole is expanded (stretched) or compressed (shrunk) and displaced (shifted) to give the best possible fit with the reference positions. Depending on the status of the corresponding checkbox, a global alignment is performed or not. However, not performing a global alignment is only useful in very specific cases. Regardless of the status of the checkbox, when ***Keep existing alignments*** is checked in the *Auto assign reference bands* dialog box (Figure 3-17), a global alignment is not performed. Instead, the program uses the distances as obtained after the first band assignment.

- The slider for ***allowed expansion/compression*** lets the user determine the maximally allowed expansion or

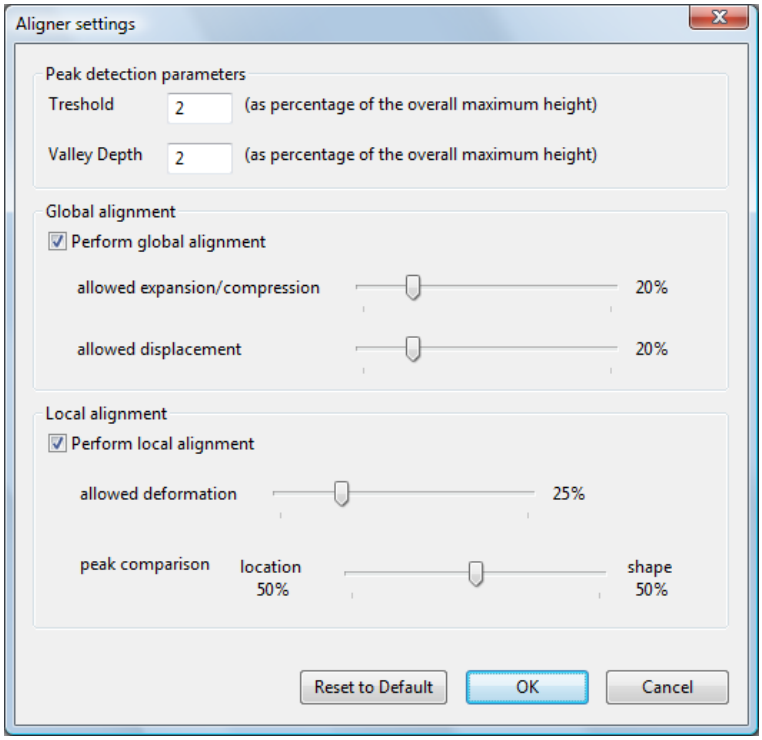


Figure 3-18. ***Alignment settings*** **dialog box.**

compression of the profile, expressed as a percentage of the total profile length. The default value is 20%.

- The slider for ***allowed displacement*** lets the user determine the maximally allowed displacement (shift) of the profile, expressed as a percentage of the total profile length. The default value is 20%.

In a **Local alignment**, the profile is locally expanded (stretched) or compressed (shrunk) to match optimally with the reference positions. Using the corresponding checkbox, you can either perform or not perform a local alignment.

- The ***allowed deformation*** is the maximally allowed deformation, expressed as a percentage of the profile length. The default value is 25%.

- The ***peak comparison*** parameter allows the user to assign more weight on the peak **location** (position) or on the peak **shape**. The shape parameter is calculated based on a curve regression and a peak size parameter. By default, both location and shape are accounted for evenly (50%). The peak comparison parameter is only considered when ***Using densitometric curves*** was checked in the *Auto assign reference bands* dialog box (Figure 3-17).

Generally, the default settings perform well with most fingerprint types. Default settings can be restored by pressing ***<Reset to default>***. Close the *Alignment settings* dialog box with ***<OK>***.

3.1.6.13 Select ***Using bands*** in the *Auto assign reference bands* dialog box and press ***<OK>***. Carefully inspect the assignments made, and if some are incorrect, correct them manually, as explained in 3.1.6.6 to 3.1.6.8.

3.1.6.14 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 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.

3.1.6.15 Show the distortion bars with ***Normalization > Show distortion bars***.

Slight transitions from bright yellow to bright blue are normal, as long as the color does not 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 (3.1.6.11), assign the correct band manually, and show the normalized view again (3.1.6.14). The ***Show distortion bars*** setting (on or off) is stored along with the *fingerprint type*.

> *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 with the option* **Keep existing assignments** *checked.*

3.1.6.16 Save the normalized gel with ***File > Save*** (F2) or  .

3.1.6.17 It is possible to generate a text file or a printout of the complete alignment of the gel, by selecting the command ***File > Export report*** or ***File > Print 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 (3.1.7), corresponding to the fourth phase in the processing of a gel (see 3.1.3). If you are going to use a curve-based coefficient, you can skip paragraph 3.1.7 and continue with 3.1.11.

3.1.7 Defining bands and quantification

In step 3 **(Normalization)**, press  , which brings you to the fourth step: **Defining bands and quantification**. This is the last step in processing a gel, which involves defining bands and quantifying band areas and/or volumes (see Figure 3-22).

3.1.7.1 Call the *Fingerprint conversion settings* dialog box with ***Edit > Edit settings*** or  . The fourth tab, ***Bands*** is shown, which allows you to enter the ***Band search filters*** and the ***Quantification units*** (Figure 3-19).

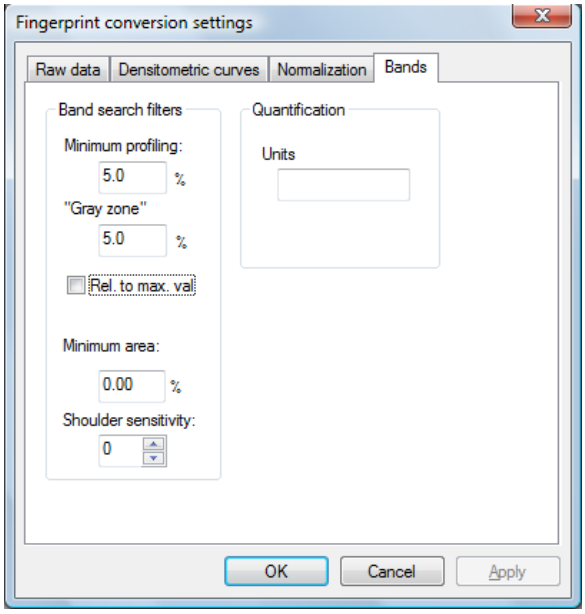


Figure 3-19. The *Fingerprint conversion settings* dialog box, *Bands* tab.

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). If, for example, you increase the OD range, peaks will look smaller on the densitometric profiles, and a smaller minimum profiling will need to be set in order to find the same number of bands. 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 with different intensities occur on the same gel, 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 3-19, 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 3-20).

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.

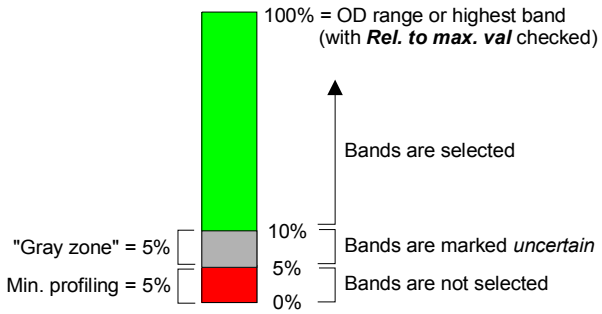


Figure 3-20. Understanding the meaning of the "gray zone" of uncertain bands in relation to the minimum profiling.

3.1.7.2 Change ***Minimal profiling*** to adjust the minimal peak height (in percent of the highest peak of the pattern), and/or ***Minimal area*** to adjust the minimal area, in percent 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).

3.1.7.3 Press **<*OK*>** to accept the settings.

3.1.7.4 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 3-21). 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.

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

You can search for bands on an individual lane by pressing **<*Search on this lane*>**, or on all lanes of the gel at once by pressing **<*Search on all lanes*>**.

3.1.7.5 Press **<*Search on all lanes*>** to start the search on the full gel.

*NOTE: If bands were already defined on the gel, the program will now ask "**There are already some bands defined on the gel. Do you want to keep existing bands?**". If you answer* **<No>**, *the existing*

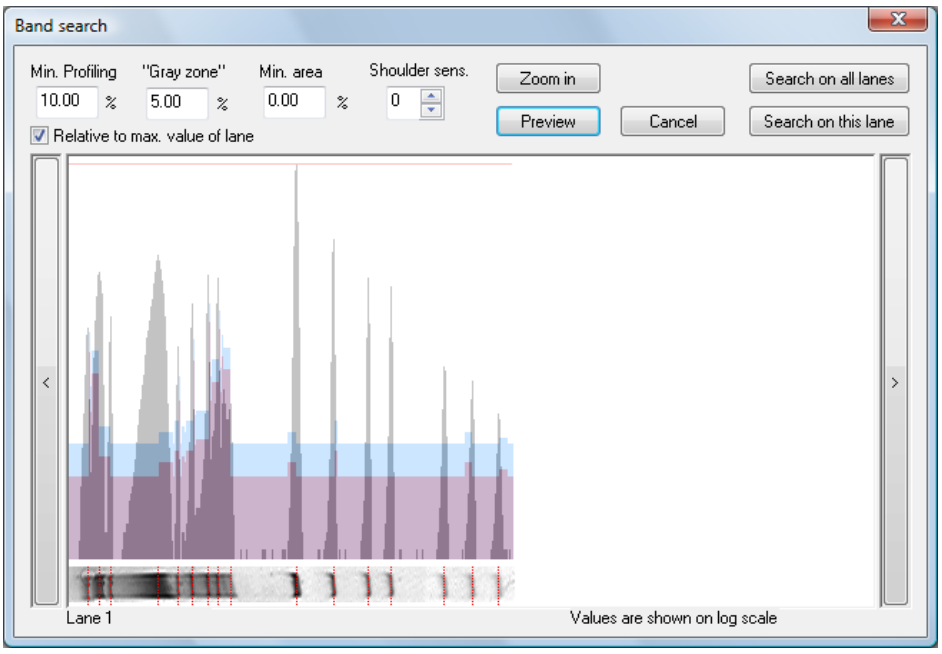


Figure 3-21. *Band search preview* window.

bands will be deleted before the program starts a new search. By answering **<Yes>**, *you can change the search settings and start a new search while any work done previously is preserved.*

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 3-22).

3.1.7.6 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 (3.1.7.6). 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 (4.1.9) 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 at the same position, will be treated as having two matching bands.*

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

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

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

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

3.1.8 Advanced band search using size-dependent threshold

In many electrophoresis systems, staining intensity of the bands is dependent on the size of the molecules. In DNA patterns stained with ethidium bromide for example (e.g. Pulsed-Field Gel Electrophoresis, PFGE), larger DNA molecules can capture many more ethidium bromide molecules than small DNA molecules, resulting in large size bands to appear much stronger than small size bands.

In other electrophoresis systems, the definition of the bands (sharpness) might depend on the size, which can also result in apparent different height depending on the position on the pattern.

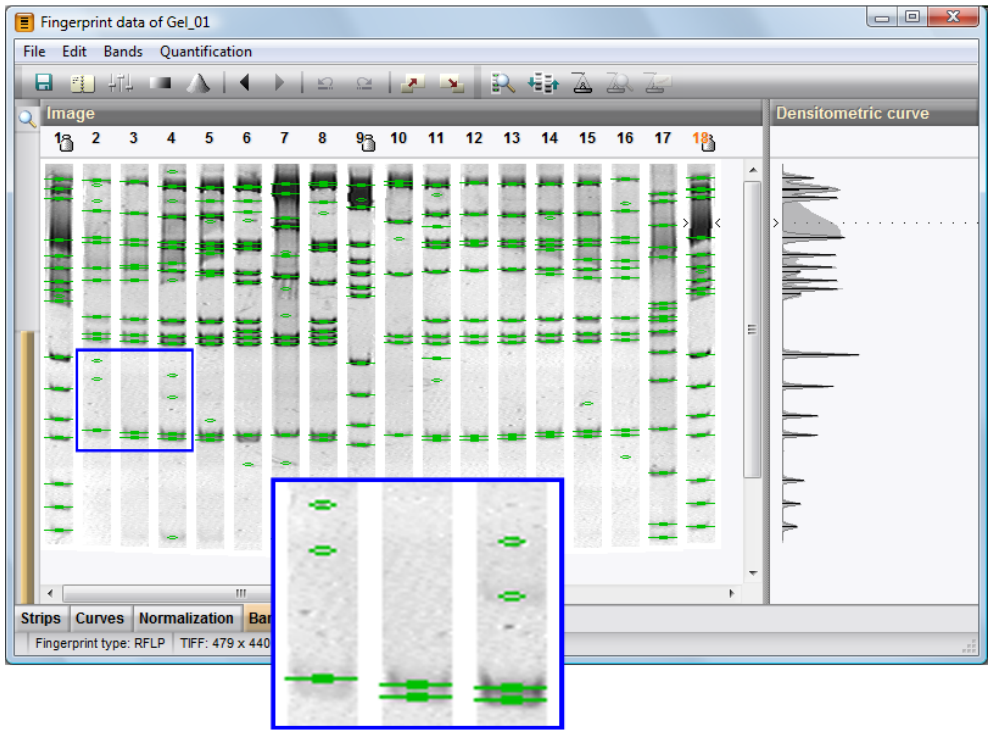


Figure 3-22. The *Fingerprint data editor* window. Step 4. Bands.

In such systems, a method that uses a single threshold parameter for finding bands on the patterns (i.e. the minimum profiling) might not work well: in case of PFGE for example, in the high molecular weight zone it might detect spots and irrelevant fragments whereas in the low molecular weight zone real bands might remain undetected.

In order to provide a more accurate band search for patterns with systematic dependence of intensity according to the position, FPQuest provides a way to calculate a regression that reflects the average peak intensity for every position on the patterns in a given fingerprint type. The only requirement for this method is that a sufficient number of gels already needs to be processed, with the bands defined appropriately, before the regression can be calculated. The user can make a selection of entries from the database, and based upon that selection and the bands they contain in the fingerprint type, the regression is established.

To obtain the regression, we proceed as follows.

3.1.8.1 Open **DemoBase** in the *FPQuest main* window.

3.1.8.2 Select ***Edit > Search entries*** or press F3 or  . This pops up the *Entry search* dialog box (see 2.2.9 for detailed explanation on search and select functions).

3.1.8.3 In the *Entry search* dialog box, check **RFLP1** and press ***<Search>***. All entries having a pattern of **RFLP1** associated are now selected in the database, which is visible as a colored arrow left from the entry fields (see 2.2.9).

3.1.8.4 In the *Experiments* panel, double-click on **RFLP1** to open the *Fingerprint type* window.

3.1.8.5 In the *Fingerprint type* window, select ***Settings > Create peak intensity profile***. This pops up the *Peak intensity profile* window, a plot of all intensities of the selected patterns in function of the position on the pattern (Figure 3-23).

3.1.8.6 Initially, the threshold factor is a flat line at 1.0. By pressing ***<Calculate from peaks>***, a non-linear regression is automatically calculated from the scatter plot (Figure 3-23).

3.1.8.7 The regression line contains 5 nodes, of which the position can be changed independently by the user. To change a node's position, click and hold the left mouse button and move the node to the desired position.

3.1.8.8 The regression can be reset to a flat line using the ***<Reset>*** button. To confirm and save the regression, press ***<OK>***.

The regression can be edited anytime later by opening the *Peak intensity profile* window again (3.1.8.5). As a result of creating a peak intensity regression curve, the minimum profiling threshold (3.1.7.2) will be dependent on the curve. The value entered for the minimum profiling will correspond to the highest value on the intensity profile regression curve (the outermost left point in Figure 3-23). Therefore, after creating an intensity profile regression, you may have to increase the minimum profiling setting to find the bands optimally: noise and irrelevant peaks will be filtered out in the high intensity areas whereas faint bands will still be detected in the low intensity areas.

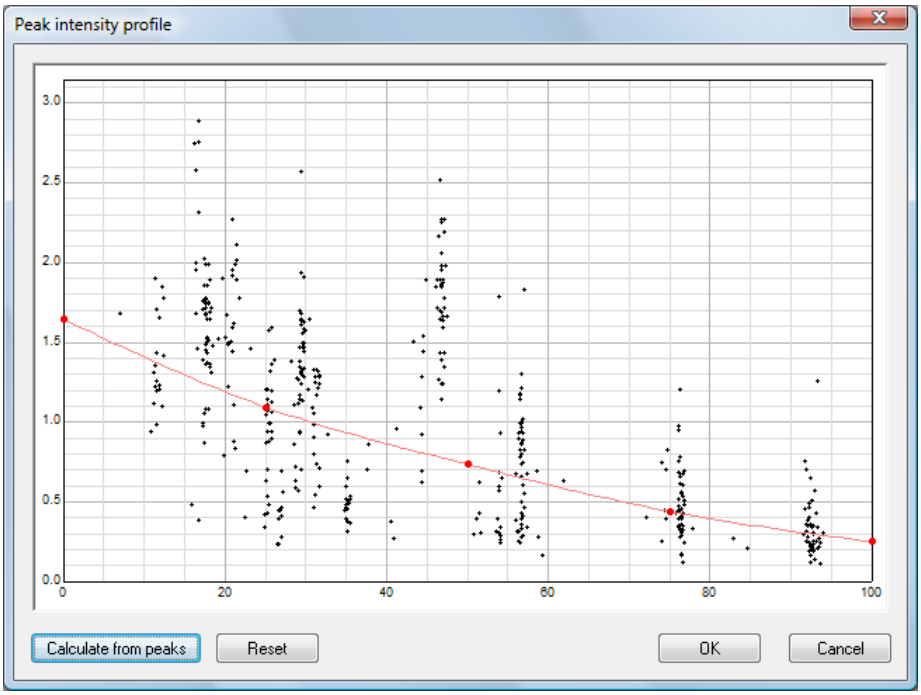


Figure 3-23. The *Peak intensity profile* window with peak intensity regression curve.

3.1.9 Quantification of bands

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

3.1.9.1 Select a band on a pattern.

3.1.9.2 Show rescaled curves with ***Edit > Rescale curves***.

3.1.9.3 Zoom in on the band by pressing repeatedly or use the zoom slider (see 1.6.7 for instructions on the use of zoom sliders). Figure 3-24 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.

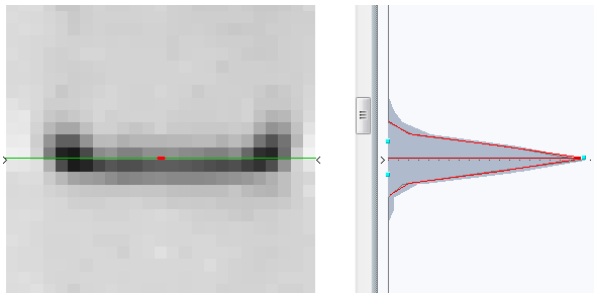


Figure 3-24. Zoomed band with its densitometric curve and best-fitting Gaussian approach.

3.1.9.4 Save the gel with ***File > Save*** (F2) or .

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

3.1.9.6 Bring the window in *Quantification mode* with or ***Quantification > Band quantification***. The quantification button now shows as and two additional band quantification buttons are shown.

3.1.9.7 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.

3.1.9.8 To change the contour of a band manually, first select the band and zoom in heavily (3.1.9.1 and 3.1.9.3).

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

3.1.9.10 For known reference bands, you can enter a concentration value by selecting the band and ***Quantification > Assign value*** (or from the floating menu that appears by right-clicking, or just double-click on the band). Known reference bands are marked with .

3.1.9.11 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 3-25) shows the real concentration in function of the band volumes, using cubic spline regression functions.

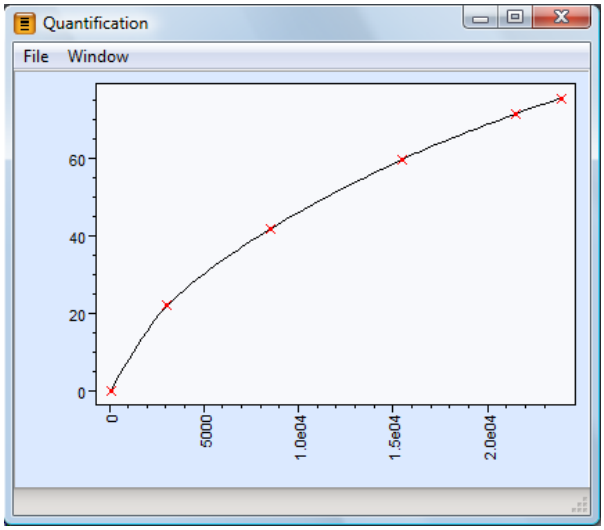


Figure 3-25. *Band quantification* window: concentration in function of known band volumes.

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

3.1.9.13 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 > Export report*** or ***File > Print 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.

3.1.9.14 Exit the *Fingerprint data editor* 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.

3.1.9.15 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 editor *window. Conversely, the current default settings can be copied to the current gel with* **Edit > Load default settings**.

3.1.10 Editing the fingerprint type settings

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

3.1.10.1 In the *FPQuest main* window, select **RFLP** in the *Experiments* panel (see Figure 1-15). Double-click on **RFLP**, or select ***Experiments > Edit experiment type*** in the main menu. This opens the *Fingerprint type* window (Figure 3-26).

3.1.10.2 The *Fingerprint 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, the densitometric curves also 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, FPQuest will ask you to copy the resolution of the current gel to the default for the fingerprint type (see 3.1.4.13). 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 FPQuest 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  . Further settings include the comparison settings, and the position tolerance settings, which will be discussed later.

The *Fingerprint type* window shows the defined reference positions in relation to the distance on the pattern (in percentage), and calls this reference system R01.

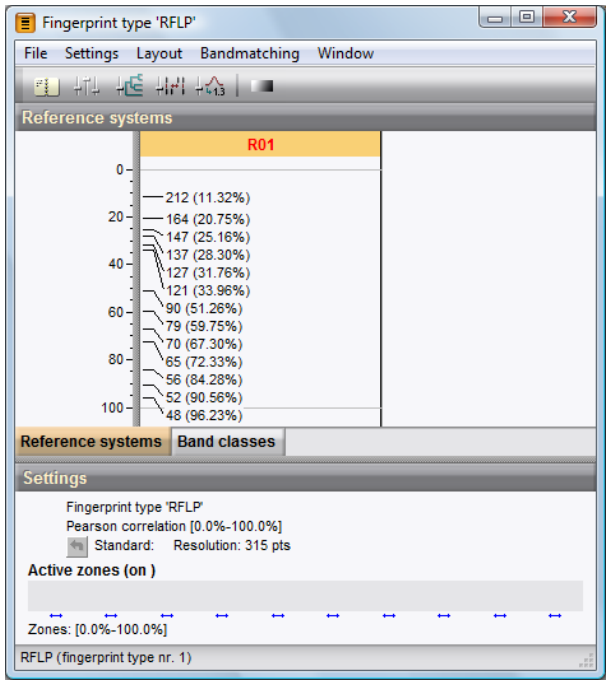


Figure 3-26. The *Fingerprint type* window; Standard is not yet defined.

Other reference systems (if created automatically) 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:

3.1.10.3 Close the *Fingerprint type* window for now (***File > Exit***).

3.1.10.4 Select the gel file in the *Files* panel of the *FPQuest main* window and choose ***File > Open experiment file (entries)*** from the main menu or press  in the toolbar of the panel.

This opens the *Fingerprint entry file* window, listing the lanes defined for the example gel (Figure 3-27).

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.

3.1.10.5 In the *FPQuest main* window, add a new database entry with ***Database > Add new entries*** (see 2.2.1.3 to 2.2.1.4).

3.1.10.6 Edit the new entry's information fields (see 2.2.3.1 to 2.2.3.2) and enter STANDARD as genus name.

3.1.10.7 Drag the link arrow of **lane 9** to the new database entry 'STANDARD': pattern 9 is now linked to this database entry.

3.1.10.8 In the *Fingerprint entry file* window, select the lane marked as STANDARD and choose ***Database > Set lane as standard***. The program will ask a confirmation.

Alternatively, the standard can also be assigned using a drag-and-drop operation from the *Fingerprint type* window, as follows:

3.1.10.9 Close the *Fingerprint entry file* window with ***File > Exit***.

3.1.10.10 In the *FPQuest main* window, open the *Fingerprint type* window again for **RFLP** (3.1.10.1).

3.1.10.11 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 3-28). From this point on, all further gels

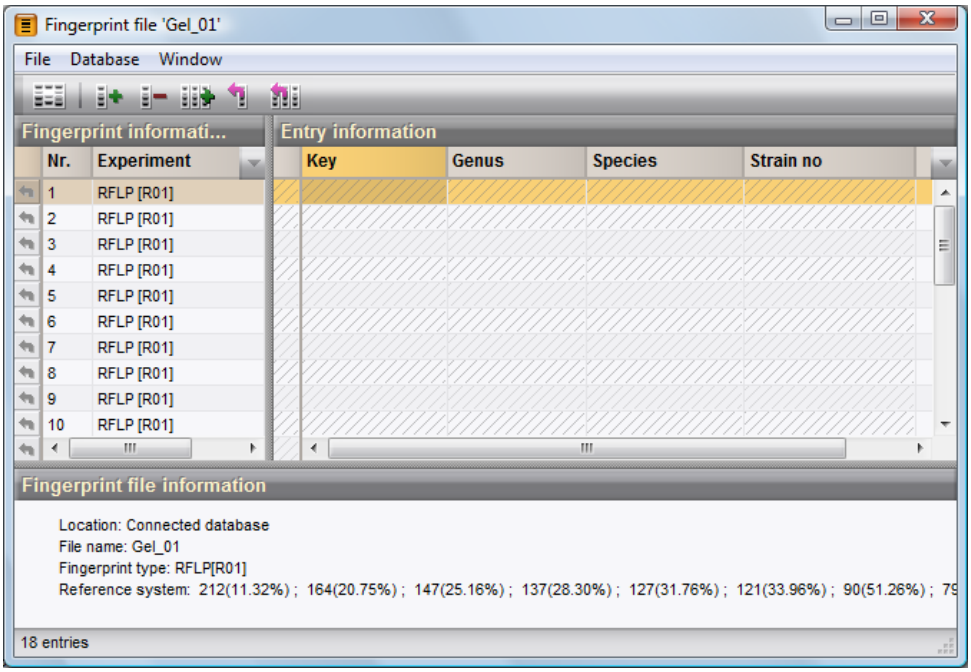


Figure 3-27. The *Fingerprint entry file* window.

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.

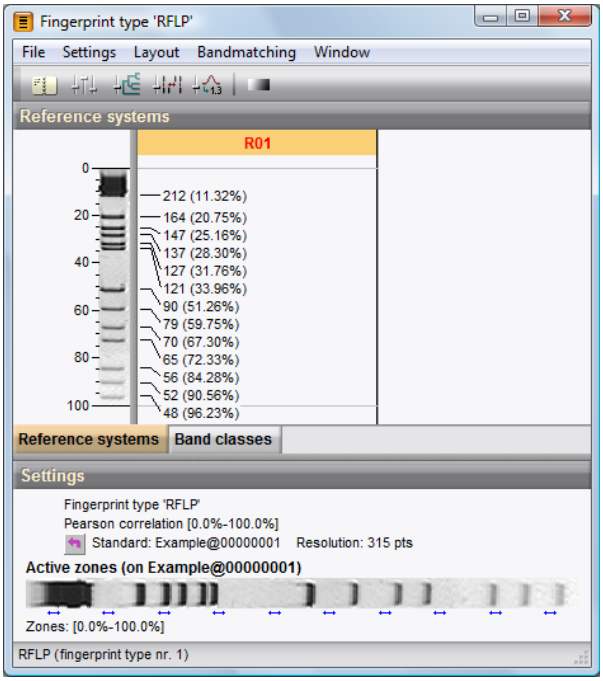


Figure 3-28. The *Fingerprint type* window; Standard is defined.

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.

3.1.10.12 In the *Fingerprint 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 3-29).

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

3.1.10.13 You can add the markers manually (***Metrics > Add marker***), but if you have entered the molecular weights as names for the reference positions (see 3.1.6.4 and 3.1.6.5), the obvious solution is to copy these molecular weights: ***Metrics > Copy markers from reference system.***

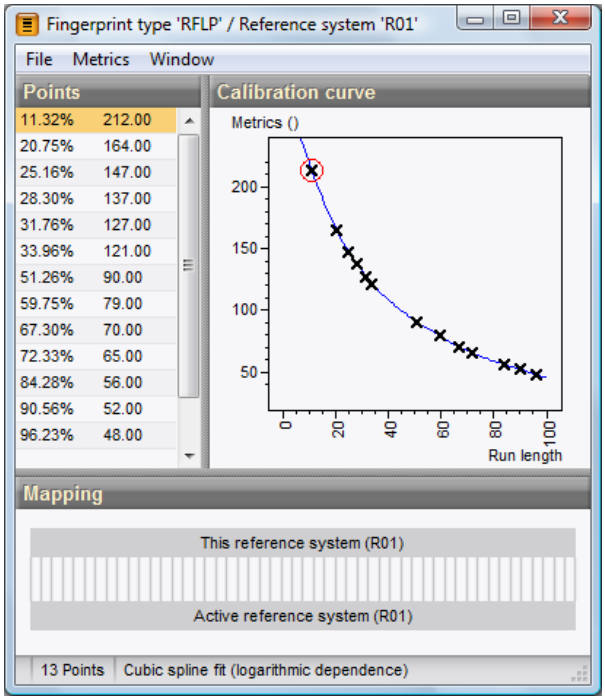


Figure 3-29. The *Reference system* window, showing molecular weight regression and remapping function to the active reference system (if different).

The result is a regression curve, shown in Figure 3-29. 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.

3.1.10.14 For this example, choose ***Metrics > Cubic spline fit*** with ***Logarithmic Dependence***.

3.1.10.15 Choose a unit with ***Metric > Assign unit***, and enter **bp** (base pairs).

3.1.10.16 Close the *Reference system* window, and close the *Fingerprint type* window.

NOTE: The Band classes *panel in the* Fingerprint type *window (displayed as a tab in Figure 3-28) will be discussed in 4.3.5.*

3.1.11 Adding gel lanes to the database

In paragraph 2.2.1, 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:

3.1.11.1 Select ***Database > Add new entries*** or  in the toolbar.

3.1.11.2 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.

3.1.11.3 Select the gel file in the *Experiment files* panel (Figure 1-15) and choose ***File > Open experiment file (entries)*** from the main menu or press  in the toolbar of the *Experiment files* panel.

This opens the *Fingerprint entry file* window, listing the lanes defined for the example gel (Figure 3-27).

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:  .

3.1.11.4 Drag the link arrow of **lane 2** (lane 1 is a reference) to the new database entry: as soon as you pass over a database entry, the cursor shape changes into  .

3.1.11.5 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 file* 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, FPQuest 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), FPQuest 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 1-15) shows a colored dot for the experiment of this entry.

3.1.11.6 You can click on such a colored dot, which pops up the *Gelstrip* for that experiment (see 3.3.1).

3.1.11.7 You can edit the information fields for this entry in several places: in the *Database entries* panel of the *FPQuest main* window (see 2.2.3.1 to 2.2.3.2), in the *Fingerprint entry file* window or in the *Information fields* panel of the *Comparison* window (see 4.1.3). In both cases, double-clicking on the entry calls the corresponding *Entry edit* window and clicking twice on the same information field enables direct editing.

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:

3.1.11.8 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 restriction enzyme 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***.

3.1.12 Adding information to fingerprint files and fingerprint lanes

In FPQuest, it is possible to assign information fields to fingerprint files in an easy way. This is useful to store information such as gel processing parameters, person who ran the gel, etc.

3.1.12.1 Right-click in the information fields header of the *Fingerprint files* panel.

3.1.12.2 From the floating menu, select ***Add new information field*** (see Figure 3-30).

3.1.12.3 Enter a name for the new information field (e.g. "Done by") and press **<*OK*>**.

The newly created information field is added in the information fields header. Clicking twice in an information field enables editing.

In addition to fingerprint file (i.e. gel-) specific information, it is possible to store information specific to individual fingerprint lanes. Recording lane-specific information could be useful e.g. to comment on PCR (RAPD, AFLP) or restriction digest (RFLP) efficiency of individual reactions. This feature is only accessible when working with a connected database (see Section 2.3).

3.1.12.4 Double-click on the fingerprint file to open the *Fingerprint entry file* window.

3.1.12.5 Right-click in the information fields header of the *Fingerprint information* panel and select ***Add fingerprint information field*** from the floating menu (or select ***File > Add fingerprint information field*** from the menu).

3.1.12.6 Enter a name for the new fingerprint lane information field (e.g. 'Comment').

The information field is added in the *Fingerprint information* panel. Clicking twice in the information field enables editing.

3.1.12.7 The lane information can also be visualized and edited from the *Experiment card* (Figure 3-40), by clicking the right mouse button inside the gelstrip window and selecting ***Fingerprint information fields*** in the floating menu that pops up.

3.1.12.8 Fingerprint lane fields can also be used in the *Advanced query tool* (see 2.2.10), using the search option **<*Fingerprint field*>**. This search option is only available if fingerprint lane information fields are defined. The first time after defining fingerprint fields, you will have to restart the program in order make the button **<*Fingerprint field*>** available in the advanced query tool.

3.1.13 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 multichannel automated 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, **CrvConv** is required to convert the multichannel sample chromatogram files into the FPQuest curve format. It can read chromatogram files from ABI, Beckman, and Amersham MegaBace. **CrvConv** splits the multichannel sample files into separate gel files for each available channel (color). 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 sequencer patterns. FPQuest 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. The fingerprint type **AFLP** in the example database **DemoBase** was processed as described above.

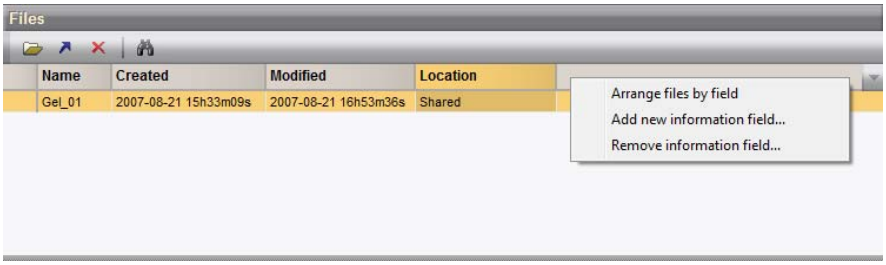


Figure 3-30. Adding information fields in the *Fingerprint files* panel.

In case (2), the conversion from the sequencer 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 usually not scanned in exactly the same position. The further steps are the same as for the automated sequencer 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 sequencer gels

3.1.13.1 Create a new database (see 1.5.2).

A set of example files together composing one gel can be found on the CD-ROM in the **Sample and Tutorial data\AB sequencer trace files** directory. The same files are also available from the download page of the website (www.bio-rad.com). We are going to import these Applied BioSystems files in our database.

In order to import chromatogram files from Applied BioSystems, Beckman, and Amersham MegaBace via the CrvConv program, the **Import** plugin needs to be installed.

3.1.13.2 Install the **Import** plugin (see paragraph 1.5.3 for more information).

3.1.13.3 Choose ***File > Import > Import Fingerprint files from Automated Sequencers***.

A dialog box pops up, listing two different import options:

- If you are importing Applied BioSystems sequencer files into a connected database (both conditions should be met), you can select ***Use automated import for AB files only***. With this option checked, FPQuest imports Applied BioSystems chromatogram files without opening the CrvConv program.

- If you want to import other types of sequencer files and/or work in a local database, you should check ***Open Curve Convertor (all formats)***. This option opens the **CrvConv** program to import your chromatogram files in the FPQuest software. Applied BioSystems, Beckman, and Amersham MegaBace chromatogram files are supported.

More information about the automated import of AB files can be found in the separate Import plugin manual. A pdf version of this manual becomes available when you click on **<*Manual*>** in the *Plugin installation* toolbox (Figure 1-13).

3.1.13.4 Check ***Open Curve Convertor (all formats)*** and press **<*OK*>**. The **CrvConv** window opens.

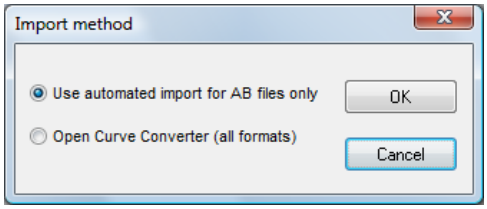


Figure 3-31. Two import methods for the import of multichannel chromatogram files.

3.1.13.5 Open the sample chromatogram files in the **CrvConv** window with ***File > Import curves from file***. A set of example files can be found on the CD-ROM in the **Sample and Tutorial data\AB sequencer trace files** directory. Alternatively, download the data from the website and browse to the unzipped folder. Select all files.

3.1.13.6 The program may produce a warning that the curve order is not specified, and that the default setting CTAG will be used. If you want to change the colors for the curves, you can use ***View > Customize colors***.

3.1.13.7 If necessary, you can change the order of the lanes with ***Edit > Move curve up*** and ***Edit > Move curve down***, or using CTRL+Up or CTRL+Down on the keyboard.

3.1.13.8 You can remove lanes if necessary with ***Edit > Remove curve*** (shortcut is DEL on the keyboard).

3.1.13.9 Select ***File > Export curves*** to save the curve gel files. Navigate to a directory on your hard drive, enter a name (e.g. **ABIGel01**) and press **<*Save*>**.

3.1.13.10 The program now asks "Do you want to reverse the curves?". If you know the top of the lanes is at the end of the curves, press **<*Reverse*>**, otherwise, press **<*Don't reverse*>**.

The program adds a number 01, 02, 03, 04 (the program supports up to 8 channels per lane) to each gel, depending on the color, and adds the extension .CRV to each gel.

3.1.13.11 In the *FPQuest main* window, create a new fingerprint type (3.1.2), e.g. **ABI**, specifying ***Densitometric curves*** when the wizard asks "***What kind of fingerprint type do you have?***".

3.1.13.12 Specify ***12-bit*** OD range (4,096 gray levels) and leave the other settings unaltered.

When finished, the new fingerprint type **ABI** is listed in the *Experiments* panel. In a next step we are going to import the curve files.

3.1.13.13 Click the  button in the *Experiment files* panel or select ***File > Add new experiment file*** in the *FPQuest main* window.

3.1.13.14 In the *Import Fingerprint entry file* window, select *Curve files* as type of files and navigate to the path where the curve files are stored (see 3.1.13.9). Select a curve file (e.g. ABIGel01_xx) and press ***<Open>***.

3.1.13.15 Repeat the previous step for the second curve file.

Two files are now listed in the *Files* panel. First we are going to process the reference file and then the data file(s).

3.1.13.16 Open the reference gel with ***File > Open experiment file (data)***, and assign it to fingerprint type ABI.

The gel with the reference patterns is the gel containing 13 bands per lane. If you have opened the data gel and not the reference gel, close the window and repeat the previous step for the other gel.

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.

3.1.13.17 Select ***Lanes > Auto search lanes*** or  to let the program automatically find the lanes.

You will notice that some setting options, applicable for TIFF files, are not available here: e.g. ***Gelstrip thickness*** and ***Number of nodes***.

3.1.13.18 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 3.1.3.

3.1.13.19 Move on to the **Normalization** step with  .

3.1.13.20 Locate a suitable standard, place the gel in ***Normalized view***  , and define the reference bands (see 3.1.6.4). The example uses the reference mix from ABI, containing 13 bands with known molecular weight.

3.1.13.21 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*** or press  .

3.1.13.22 Update the normalization with  and save the normalized reference gel.

3.1.13.23 Select the second gel, the gel containing the data, with ***File > Open experiment file (data)***, and assign it to fingerprint type ABI.

3.1.13.24 Link this gel to the reference gel with the command ***File > Link to reference gel*** and enter the name of the reference gel in the dialog box that pops up.

> *NOTE: If the reference gel is selected in the* Experiment files *panel, the name of the reference gel is automatically shown in the dialog box.*

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.

3.1.13.25 Whenever needed, you can pop up a reference gel to which a data gel is linked with ***File > Open reference gel***.

3.1.13.26 If you have made changes to the reference gel without saving them, you can update the changes to the data gel with ***File > Update linked information***. Once you save the changes to the reference gel, the data gel(s) are updated automatically.

More information on the processing of the gels can be found in paragraph 3.1.7 and sections 4.1 - 4.2.

B. RFLP gel scans containing internal markers

The way of processing the gels is similar as described for the 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.

3.1.13.27 Open the TIFF image of the reference gel and assign it to a fingerprint type.

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

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

3.1.13.29 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.

3.1.13.30 Proceed with the full normalization of the reference gel as described in 3.1.3. Save the file.

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

3.1.13.32 Link this gel to the reference gel with the command ***File > Link to reference gel*** and enter the name of the reference gel in the dialog box that pops up.

NOTE: If the reference gel is selected in the Experiment files *panel, the name of the reference gel is automatically shown in the dialog box.*

The tracking info, curve settings and alignment of the reference gel are now automatically superimposed on 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.

3.1.13.33 Whenever needed, you can pop up the reference gel to which a data gel is linked with ***File > Open reference gel***.

3.1.13.34 If you have made changes to the reference gel without saving them, you can update the changes to the data gel with ***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 generally recommended.*

3.1.14 Import of molecular size tables as fingerprint type

FPQuest 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.

3.1.14.1 In the **Example** database, create a new fingerprint type **AB-Genescan.** Leave every setting as default except in the second step, where you should specify ***Densitometric curves*** and ***12-bit (4096 values)***.

We will now create a new reference system to allow the import of an AB Genescan table, part of which is shown in Figure 3-32. The whole file (5 patterns) can be found in the **Sample and Tutorial data\Sample text files for import** directory on the installation CD-ROM as **Genescan.txt**. This text file is also available from the download page of the website (www.bio-rad.com/softwaredownloads). 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 3-32), next is a column with the running time, next is the size in base pairs, then the height, the volume, and the running time again.

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 3-32. Lane in an AB Genescan table and conversion of running distances to FPQuest positions.

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

The running distance needed by FPQuest 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:

$$\%RD = RT_{min}/RTx100$$

In the example, RT_{min} = 30. For the reference lane 17B, this yields the extra column under the running time column (Figure 3-32).

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.

3.1.14.2 In the *Fingerprint type* window, select ***Settings > New reference system (positions)***.

The input box shown in Figure 3-33 allows all known reference bands to be entered.

3.1.14.3 Press the **<*Add*>** button and enter all running distances and sizes of lane 17B, as shown in Figure 3-33.

3.1.14.4 Enter a name for the reference system, e.g. **AB**.

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

> *NOTE: 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.*

3.1.14.6 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).

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

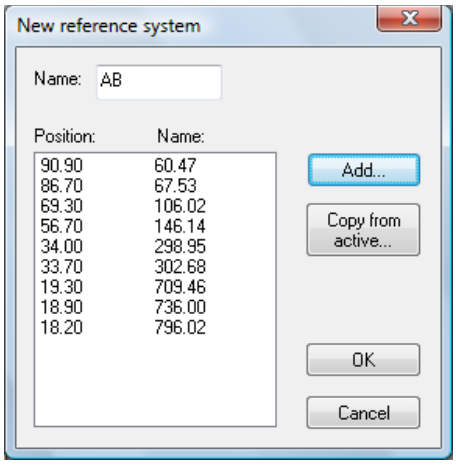


Figure 3-33. Defining a new reference system based upon known band positions and sizes.

3.1.14.8 In the *Reference system* window, copy the entered molecular weights with ***Metrics > Copy markers from reference system***.

FPQuest is now configured to import the Genescan tables.

3.1.14.9 Exit the *Reference system* window and the *Fingerprint type* window.

To import Applied BioSystems Genescan files, there are scripts available on the website of Bio-Rad. These scripts can be launched from the *FPQuest main* window, using the menu ***Scripts > Browse Internet***, or  . The script to import Genescan data can be found under **Import tools** and is called **Import ABI Genescan tables**. A description of how to use this script is available on the website.

3.1.14.10 When running the script, you can use the example **Genescan.txt** file in the **Sample and Tutorial data\Sample text files for import** directory on the CD-ROM. This text file is also available from the download page of the website (www.bio-rad.com/softwaredownloads).

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.

3.1.14.11 In the *FPQuest main* window, open the *Fingerprint type* window for **AB-Genescan**.

3.1.14.12 In the AB-Genescan *Fingerprint type* window, select ***Settings > New reference system (curve)***.

The *New reference system* window (Figure 3-34) allows the size range to be specified as well as the type and strength of the regression.

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

3.1.14.14 Press the **<*Add*>** button to add the sizes for all reference bands available in the fingerprint type (see lane 17B, Figure 3-32).

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

3.1.14.15 Optimize the ***Calibration curve*** and the strength (in percent) to obtain the best spread of the reference bands.

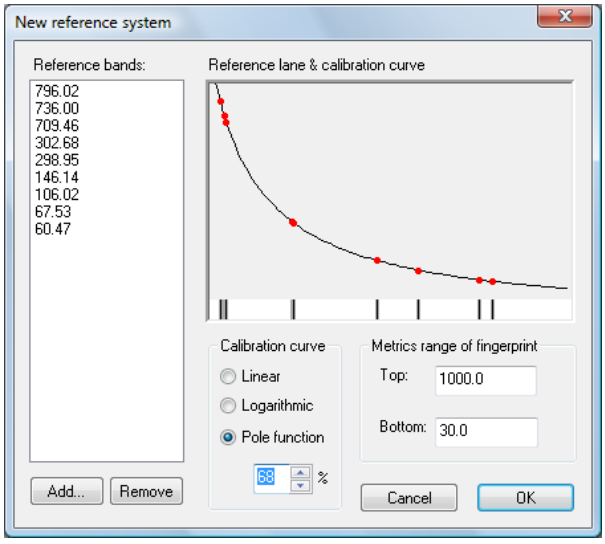


Figure 3-34. Defining a new reference system using a synthetic regression curve between user-defined size limits.

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

3.1.14.17 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 3.1.14.9 and further. 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 FPQuest. Both gels have their bands on different positions because of the different logarithmic function that was used to reconstruct the gels.

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

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

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

3.1.15 Conversion of gel patterns from GelCompar versions 4.1 and 4.2

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 FPQuest.

3.1.16 Dealing with multiple reference systems within the same fingerprint type

Under normal circumstances, a reference system is created once initially, and is never changed afterwards. In some cases however, it can be required that a second reference system is created. Some examples are:

(1) The gel used originally for defining the reference positions appears to be an aberrant one, so that repositioning the reference positions is required to allow most other gels to be normalized easily.

(2) One or more bands defined as reference positions are found to be unreliable or inappropriate and should be deleted or replaced with another band.

(3) The user switches to a new reference pattern for the fingerprint type.

(4) Gels of the same fingerprint type are imported from another database and need to be analyzed together with gels from the local database.

Case (1), shown in Figure 3-35, results in two reference systems with the same reference position names, but having different % distances on the gel. Gels processed under both reference systems are perfectly compatible and there is no loss of accuracy compared to gels analyzed under the same reference system.

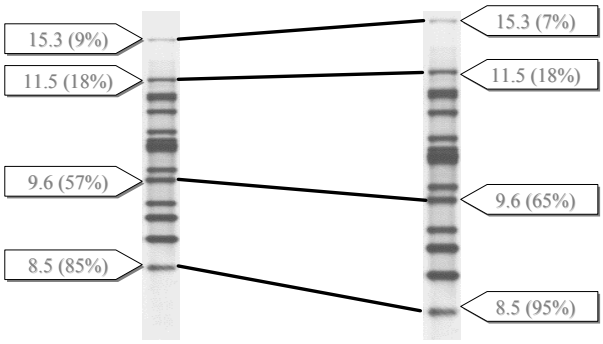


Figure 3-35. Example of different reference systems in the same fingerprint type for which remapping causes no loss of accuracy. See text for explanation.

The same situation can raise if gels are imported from another database, which have been processed under a different reference system [case (4)], but where the same marker pattern is used and the reference positions have been given the same name (even though the % distances are different).

Case (2) may result in a new reference system with more or less bands, or with bands having a different name (Figure 3-36). In either case, the new reference system will not be automatically compatible with the original, and compatibility can only be obtained by creating a molecular weight regression curve for both reference systems (see 3.1.10.12 to 3.1.10.16 on how to create a regression curve). Both reference systems can then be remapped onto each other, which inevitably causes some loss in accuracy. The degree of compatibility depends on the number of reference positions in both systems, the amount of overlap between regression curves, the predictability of the regression curve using one of the available methods, the spread of calibration points (reference positions), the definition of the reference bands, etc.

Case (3) obviously causes a situation where reference positions have different names, since one can assume that a new marker has different bands, and results in a situation where remapping is required.

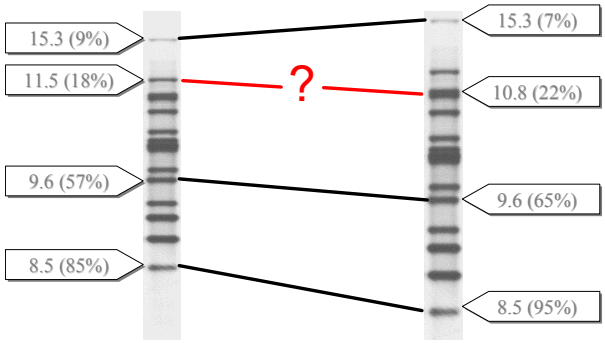


Figure 3-36. Example of different reference systems in the same fingerprint type for which remapping relies on molecular weight regression curves for both reference systems and as such, causes some loss of accuracy. See text for explanation.

When more than one reference system is present in a fingerprint type, one of the reference systems is specified as the "active" reference system. The active reference system is the one to which all new gels will be normalized. By default, the first created reference system is the active one. The name of the active reference system is shown in red in the *Fingerprint type* window.

3.1.16.1 To change the active reference system, open the *Fingerprint type* window, and select the reference system to become the active one. Choose ***Settings > Set as active reference system***.

3.1.16.2 To remove a reference system that is not used anymore, select the reference system in the *Fingerprint type* window, and choose ***Settings > Remove reference system***.

The program asks "**Do you want to check if this reference system is in use?**". For large connected databases, this may take a long time. If you answer ***<No>*** to this question, the selected reference system is removed, regardless of whether it is used in gels or not. By opening and saving a gel that was processed under the removed reference system however, it will be restored. By answering ***<Yes>***, the program checks the database for gels normalized with the reference system, and if any such gels are found, the reference system is not removed.

NOTE: To avoid any possible conflict situations, it is recommended to allow the program to scan the database for the presence of gels normalized with the reference system, and not to remove any reference systems that are in use.

3.2 Setting up composite data sets

3.2.1 Introduction

Composite data sets do not necessarily correspond to an actual experiment, but are character tables derived from one or more physical experiments. They provide a convenient way to analyze the combined results of several fingerprint types and offer a number of additional tools, e.g. for the analysis of band matching tables (see Section 4.3). These tools include a function to discriminate groups based upon differential characters, a function to perform transversal clustering (see 4.4.3), and bootstrap analysis (a cluster significance tool, see 4.1.13).

3.2.2 Defining a new composite data set

We will now describe the setup of composite data sets in function of cluster analysis based upon multiple experiments. As an example, we will create a character table for the two RFLP experiments defined in the **DemoBase** database.

3.2.2.1 In the *FPQuest main* window, with the database **DemoBase** loaded, select ***Experiments > Create new composite data set***, or press the  button in the *Experiments* panel toolbar and select ***New composite data set***.

3.2.2.2 Enter a name, for example **RFLP-combined** and press ***<OK>***.

The *Composite data set* window is shown for **RFLP-combined** (see Figure 3-38). 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.

3.2.2.3 Select **RFLP1** from the experiment list and ***Experiment > Use in composite data set***. Repeat this action for **RFLP2**.

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 4.4.2). 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.

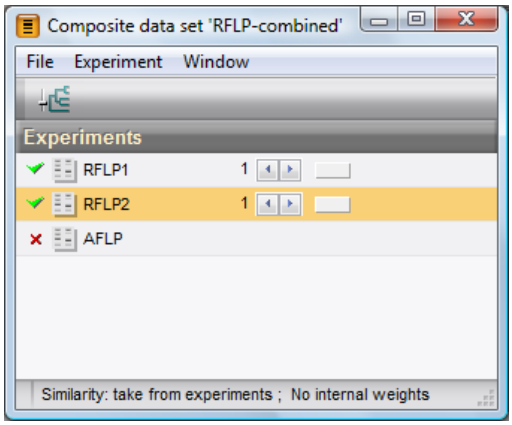


Figure 3-38. ***Composite data set*** **window.**

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:

3.2.2.4 Select ***Experiment > Correct for internal weights***. The caption now shows **Similarity: take from experiments; Correct for internal weights**.

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 4.4.2 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, or Euclidian distance (for more information on these coefficients, see Section 4.3) 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.

The comparison settings for the composite data set can be accessed with ***Experiment > Comparison settings*** or the  button, but also in the *Comparison* window. See Section 4.3 for a detailed explanation.

3.2.2.5 Close the *Composite data set* window with ***File > Exit***. The new composite data set is shown in the *Experiments* panel of the *FPQuest main* window.

3.3 Experiment display and edit functions

In 2.2.3, we have explained how you can edit the information fields for each database entry by double-clicking on the entry (2.2.3.1), 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:

3.3.0.1 Close the *FPQuest main* window.

3.3.0.2 Back in the Startup screen, select **DemoBase** and click on  or just double-click **DemoBase** to start FPQuest with this database loaded.

3.3.1 The gelstrip

3.3.1.1 If we open the *Entry edit* window for any database entry (except a standard), the window lists all available experiment types for this entry, each of which contains two buttons (Figure 3-39).

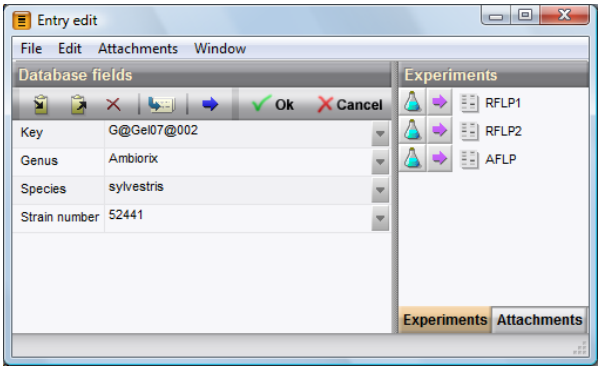


Figure 3-39. *Entry edit* window.

3.3.1.2 With the  button, you can display the *Gelstrip* of an experiment (Figure 3-40).

A gelstrip can also be opened from the *FPQuest main* window, by clicking on the colored dot in the *Experiment presence* panel (see Figure 1-15).

3.3.1.3 You can move the gelstrip by clicking and holding the left mouse button on the card, and then dragging it to its new position.

3.3.1.4 When you hover over the gelstrip with the mouse, a small tag displays the key of the entry, fingerprint type, gel name and lane number.

Gelstrips can be displayed in two modes, a raw mode, i.e. not normalized, and a normalized mode (see Figure 3-40). In the normalized mode, the band information is also shown. Band sizes are shown as molecular sizes (metrics) if the metrics regression curve is available for the reference system, or as relative distances from the top if no metrics regression curve is available.

3.3.1.5 To switch between the raw and normalized view, open the *Fingerprint type* window (Figure 3-26) and select ***Layout > Show normalized gelcards***. If the feature is enabled, the menu item is flagged.

3.3.1.6 Close a gelstrip by clicking in the small triangle-shaped button in the left upper corner.

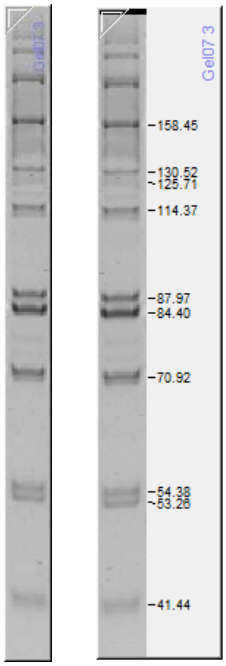


Figure 3-40. Gelstrip of a database entry in raw mode (left) and normalized mode (right).

You can open a gelstrip for an entry, close its *Entry edit* window, and then show the corresponding gelstrip 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 gelstrips that can be shown together.

3.3.1.7 In case multiple gelstrips are shown on the screen, it is possible to line them up by right-clicking on a gelstrip and choosing ***Line up***. All gelstrips can be closed at once using ***Close all*** in the floating menu.

3.3.1.8 The size of the gelstrip can be increased or decreased using the keyboard, by pressing the **numerical + key** (increase) or the **numerical - key** (decrease).

3.3.1.9 Right-clicking on the gelstrip pops up a floating menu, from which you can choose ***Export normalized curve***, ***Export normalized band positions***, and ***Export normalized band metrics***. Selecting any of the above commands exports the corresponding information to the clipboard, from where it can be pasted as text, e.g. in Notepad.

3.3.1.10 In a connected database it is possible to show or edit the fingerprint lane information fields with ***Fingerprint information fields*** (see also 3.1.12 on fingerprint lane information fields and Section 2.3 on connected databases).

4. COMPARISONS

4.1 General comparison functions CL

4.1.1 Definition

A *Comparison* in FPQuest includes every function which allows to compare database entries. This involves the display of experiment images of selected entries, the calculation and display of cluster analyses, and the calculation of principal component analysis (PCA) and multi-dimensional scaling (MDS) projects.

Two different windows are available in FPQuest for comparison of entries: The *Pairwise comparison* window offers a detailed comparison overview for all experiments available for two selected entries. Whenever more than two entries need to be compared, the *Comparison* window should be called.

The *Pairwise comparison* window and the *Comparison* window in FPQuest present 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 is always created from a selection of database entries. These can be selected manually (see 2.2.8) or via the automatic search and selection functions (see 2.2.9 and 2.2.10).

4.1.2 The *Pairwise comparison* window

From within any window where you can select entries, you can display a detailed comparison between two entries. This pairwise comparison shows all images of the fingerprint types as well as the similarities obtained using the specified coefficients.

4.1.2.1 In the Startup screen, double-click on **DemoBase** to open the database for analysis. Alternatively, you can select **DemoBase** from the list and then press the  button. In case **DemoBase** was already open, clear any previous selection with F4.

4.1.2.2 Select any two entries you want to compare.

4.1.2.3 In the *FPQuest main* window, select ***Comparison > Compare two entries*** or use the **CTRL+2** (numerical 2) or **CTRL+ALT+C** shortcuts. These shortcuts work from within any window. The *Pairwise comparison* window appears (Figure 4-1).

The *Pairwise comparison* window consists of two dockable panels: the *Experiments* and the *Comparison* panel. For detailed information about the display of dockable panels, see 1.6.4.

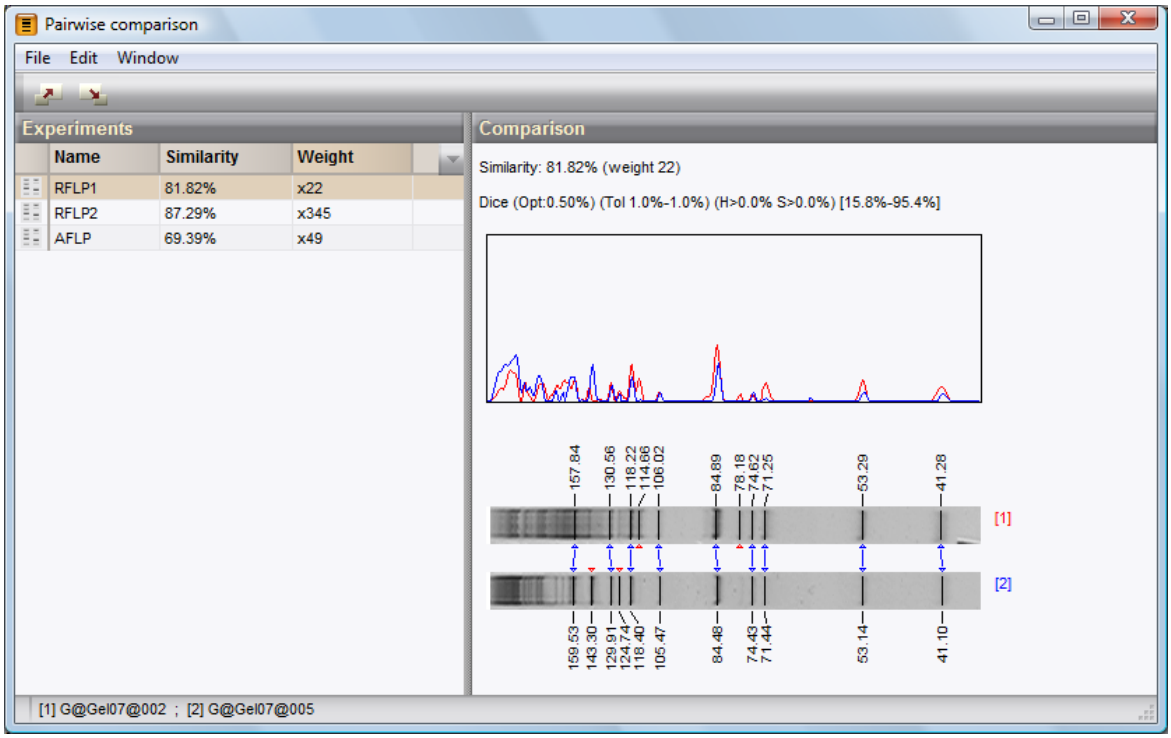


Figure 4-1. ***Pairwise comparison*** **window.**

The *Experiments* panel (left panel in default configuration) displays the names of all fingerprint types present in the database ('Name') and the type ('Type'). When a fingerprint type is present for both entries, the similarity value for this fingerprint type is shown in the information field 'Similarity'. 'Weight' displays the assigned weight to each experiment type (see 3.2.2). These information fields can be displayed or hidden by pressing the column properties button ( ) and selecting the fields from the pull-down menu (see 1.6.6 for more information on grid panels).

4.1.2.4 Click on a fingerprint type in the *Experiments* panel to display the corresponding images in the *Comparison* panel (right panel in default configuration).

The *Comparison* panel also lists the comparison settings used to calculate the similarity value. If a band matching coefficient was chosen in the experiment settings (e.g. Dice coefficient in Figure 4-1), the detailed comparison of the band matching is shown.

> *NOTE: The comparison settings are defined in the* Comparison settings *dialog box. This dialog box can be accessed from each* Experiment type *window (via* ***Settings > Comparison settings*** *or*) *and from the* Comparison *window (via* ***Clustering > Calculate > Cluster analysis (similarity matrix)****).*

4.1.3 The *Comparison* window

When more than two entries should be compared, this is achieved through the *Comparison* window. We will use the **DemoBase** database to explain this window.

4.1.3.1 In the Startup screen, double-click on **DemoBase** to open the database for analysis. Alternatively, you can select **DemoBase** from the list and then press the button. In case **DemoBase** was already open, clear any previous selection with F4.

With all entries except the standards selected, we will create a new comparison. This selection can be done manually as described below or via the search and selection functions (see 2.2.9 and 2.2.10 for a description).

4.1.3.2 In the *Database entries* panel of the *FPQuest main* window, click on the first database entry and, while holding the SHIFT key, click on the last entry to select all database entries. Alternatively, press CTRL+A on the keyboard to select all database entries at once.

4.1.3.3 Unselect the first entry marked as STANDARD by clicking it and selecting ***Edit > Select/Unselect entry*** or press the space bar on the keyboard. Repeat the same actions for the second and third STANDARD.

4.1.3.4 Select ***Comparison > Create new comparison*** (ALT+C) or press the button from the *Comparisons* panel toolbar. A *Comparison* window is created, with the selected database entries (Figure 4-2).

The *Comparison* window is divided in six main panels: the *Dendrogram* panel, which shows the dendrogram if calculated, the *Experiment data* panel, showing the images of the experiments, the *Information fields* panel, which shows the database fields in the same layout as in the database (see 2.2.6), the *Similarities* panel, which shows the similarity values, the *Experiments* panel, which shows the available experiment types and the *Groups* panel, which shows the groups if defined. Initially, the *Dendrogram* panel, the *Experiment data* panel, the *Similarities* panel and the *Groups* panel are empty.

4.1.3.5 You can drag the horizontal and vertical separator lines between the panels, in order to divide the space among the panels optimally.

4.1.3.6 All panels in the *Comparison* window are dockable and their position can therefore be changed according to your own preferences. For more information on the display of dockable panels, see 1.6.4.

The *Information fields* panel in the *Comparison* window is similar to the *Database entries* panel in the *FPQuest main* window and contains the database information in tabular format (grid panel). For detailed information on the display options of grid panels, see 1.6.6.

> *NOTE: The* Dendrogram, Experiment data, Information fields *and* Similarities *panel behave as a group, i.e. these panels cannot be docked outside this group and they cannot be displayed in a window of their own (undocked).*

4.1.3.7 In the *Information fields* panel, 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.

4.1.3.8 Clicking the column properties button ( ) located on the right hand side in the information fields header in the *Information fields* panel gives access to functions allowing information fields to be displayed or hidden, frozen, or moved to the left or to the right (see 1.6.6 for details).

4.1.3.9 As explained in 2.2.6.6, it is possible to freeze one or more information fields in the *FPQuest main* window using ***Edit > Freeze left panel***, so that they always remain visible left from the scrollable area. The same fields will be frozen in the *Comparison* window. This feature can be combined with the possibility to change the order of information fields, which makes it possible to freeze any subset of fields.

From the *Experiments* panel, you can select one of the available experiment types, to show an image, calculate

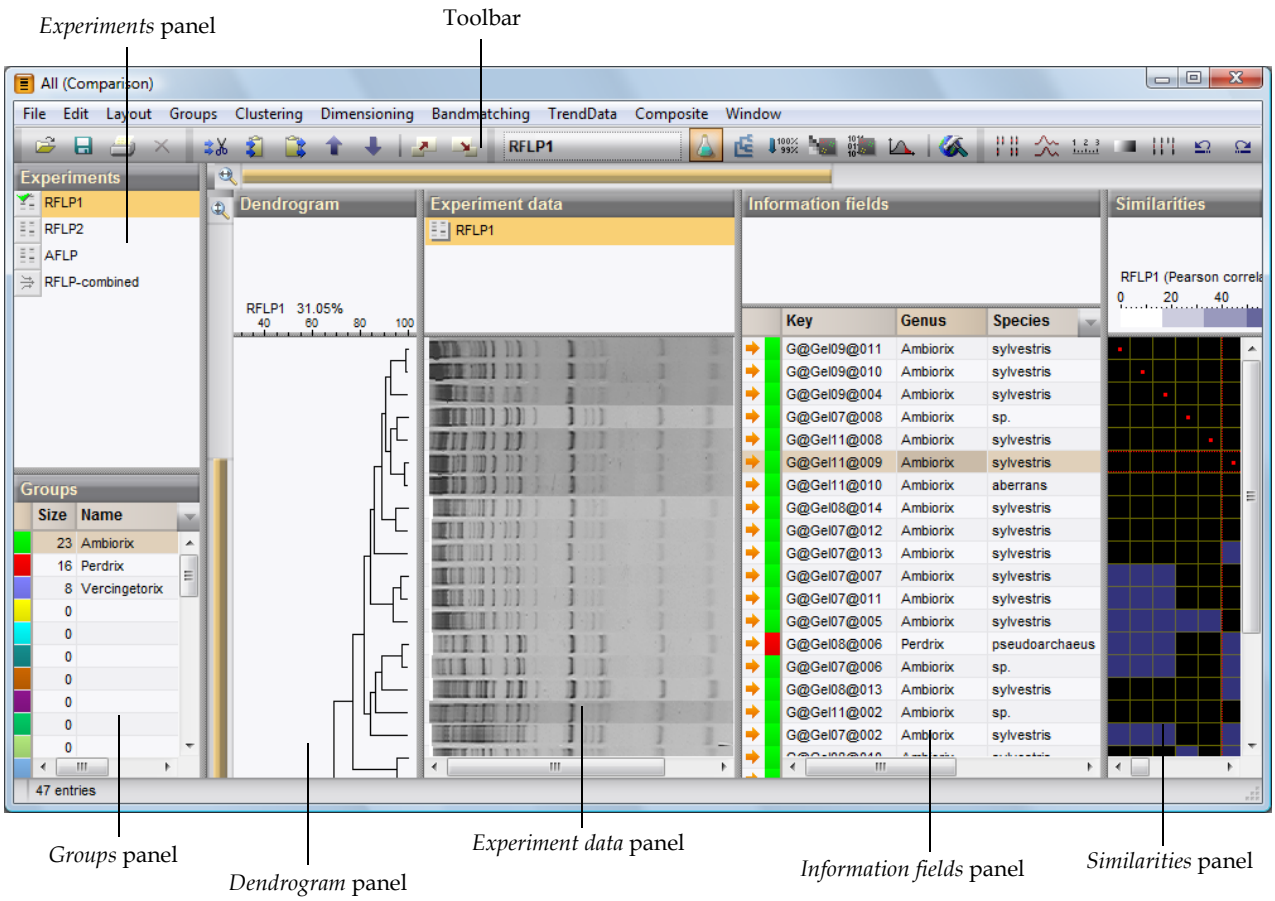


Figure 4-2. The *Comparison* window with dendrogram, gel image, groups and entry names displayed.

a dendrogram, or show a matrix. Each experiment type in the *Experiments* panel contains two objects: a button and the experiment type name, on the right hand side of the button. In case of a fingerprint type, the button is shown as ; for composite data sets as . Clicking one of these buttons shows the image of the corresponding experiment type in the *Experiment data* panel.

NOTE: The experiments in the Experiments *panel of the* Comparison *window are listed in the same order as they are listed in the* Experiments *panel of the* FPQuest main *window. This feature also allows one to control the order in which fingerprint data are displayed in a composite data set (see 4.4.2).*

4.1.3.10 Press the button of **RFLP1**; the pattern images are shown for **RFLP1**. When the image of an experiment type is displayed, the button shows like .

4.1.3.11 Press the button of **RFLP2**; the pattern images of **RFLP2** are shown right from those of **RFLP1**. The button of **RFLP2** now shows like .

NOTE: To display more than one image at a time, we recommend to maximize the Comparison *window, and to use maximal space for the* Experiment data *panel by minimizing the* Dendrogram *and* Similarities *panels (see 4.1.3.5).*

4.1.3.12 If insufficient space is available to show both images at the same time, you can scroll through the *Experiment data* panel, or use the zoom functions and (***Layout > Zoom in*** and ***Layout > Zoom out***). Shortcut keys for these actions are CTRL+PgUp and CTRL+PgDn, respectively. The zoom sliders indicated with and can be used to zoom selectively in the horizontal or vertical direction, respectively. See 1.6.7 for a detailed description of zoom slider functions.

4.1.3.13 In the caption of the *Experiment data* panel, you can drag the separator line between the images to the left or to the right, in order to reserve more or less horizontal space for a particular experiment image. The original aspect ratio (proportion height to width) of the image will not be maintained by this action.

4.1.3.14 To select an experiment type in the *Comparison* window, you can either click on the experiment type name in the *Experiments* panel, on the image itself or select the experiment type from the drop-down menu when pressing the *Active experiment* button RFLP1 in the toolbar.

When an experiment type is selected, both the image caption in the *Experiment data* panel (if the image is shown) and its name in the *Experiments* panel are highlighted. All functions listed under ***Clustering***, ***Dimensioning***, ***Bandmatching***, and ***Composite*** as well as some ***Layout*** functions, apply to the selected experiment type.

4.1.4 Adding and removing entries

Selections of entries made in the *Database entries* panel of the *FPQuest main* window are also shown in the *Information fields* panel of the *Comparison* window and vice versa. The entries in a newly created comparison are all marked with a colored selection arrow, since they were all selected in the database. You can manually select and unselect entries in the *Information fields* panel (see Figure 4-2), using the CTRL and SHIFT keys as described in 2.2.8. Selections can be added or removed from an existing comparison.

4.1.4.1 First unselect all entries by pressing the F4 key.

4.1.4.2 Select some entries from the comparison (see 2.2.8).

4.1.4.3 With ***Edit > Delete selection*** (shortcut DEL on the keyboard), the selected entries are removed from the comparison. The program will ask for confirmation to remove the selection from the comparison. You will not be able to undo this operation.

4.1.4.4 With ***Edit > Cut selection*** or  (shortcut CTRL+X on the keyboard), the selected entries are removed from the comparison and are copied to the clipboard.

4.1.4.5 With ***Edit > Paste selection*** or  (shortcut CTRL+V on the keyboard), the same entries are placed back into 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 (see also 4.1.5).

Entries can be added to an existing comparison at any time. The entries first need to be copied to the clipboard from the *FPQuest main* window or from another comparison.

4.1.4.6 To copy entries to the clipboard, select the entries (e.g. in the *FPQuest main* window) first and use the ***Edit > Copy selection*** command or  (shortcut CTRL+C on the keyboard).

To cut entries from one comparison into another, use ***Edit > Cut selection*** or  (shortcut CTRL+X on the keyboard) in the one comparison and ***Edit > Paste selection*** or  (shortcut CTRL+V on the keyboard) in the other comparison.

New database entries can be added to an existing *dendrogram* (see 4.1.9 on how to calculate a dendrogram) in this way: 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.

> *NOTE: Entries can also be selected from the* Dendrogram *panel: hold the CTRL key and left-click on a branch node to select/unselect a cluster on the dendrogram at once (see also 4.1.11).*

4.1.5 Rearranging entries in a comparison

The cut and paste functions can be used to rearrange entries in the *Comparison* window (4.1.4.4 to 4.1.4.5). Some other convenient functions are available for rearranging entries in a comparison, as explained below.

4.1.5.1 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*.

4.1.5.2 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 numerically and alphabetically, 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**].

4.1.5.3 A group of selected entries (colored arrows) can be placed at the position of the cursor (the entry you last clicked on) with ***Edit > Bring selected entries to cursor***.

4.1.5.4 A group of selected entries (colored arrows) can be moved to the top of the comparison with ***Edit > Bring selected entries to top*** (shortcut CTRL+T on the keyboard).

4.1.5.5 An individual entry can be moved up and down by left-clicking on it, and selecting ***Edit > Move entry up***

 or ***Edit > Move entry down***  . Pressing the Arrow Up and Arrow Down keys on the keyboard while holding down the SHIFT key does the same.

4.1.5.6 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.

4.1.6 Saving and loading comparisons

A comparison can be saved and all calculations done on the data it contains, will be stored along. This includes similarity matrices in all experiment types where they have been calculated, any dendrogram that has been calculated (see 4.1.9), band matchings and polymorphism analyses (see Section 4.3). In a connected database (see Section 2.3) it is even possible to share comparisons with other users who share the same database.

4.1.6.1 Select ***File > Save as*** or press  to save the comparison (shortcut CTRL+S on the keyboard). In a connected database, you can save the comparison either locally or in the connected database (see Figure 4-3).

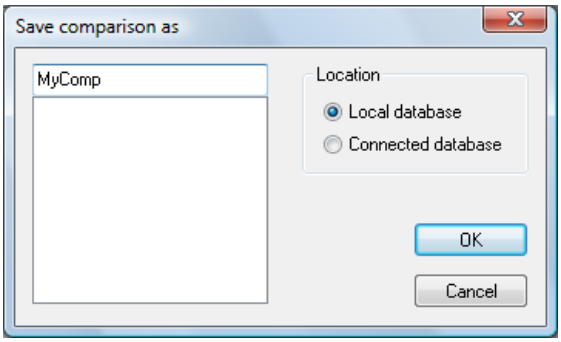


Figure 4-3. *Save comparison* dialog box in a connected database.

4.1.6.2 Enter a name, e.g. **MyComp**.

4.1.6.3 Close the comparison with ***File > Exit***. Comparison **MyComp** is now listed in the *Comparisons* panel of the *FPQuest main* window.

The default information field 'Location' in the *Comparisons* panel of the *FPQuest main* window lists the location (i.e., Shared or Local) of the saved comparisons.

If a comparison is saved in the connected database (see Figure 4-3), the comparison will be visible by other users who are connected to the same database. Only the first user, however, will be able to save changes in the comparison; for subsequent users it will be read-only.

4.1.6.4 To open an existing comparison, select it from the list in the *Comparisons* panel of the *FPQuest main* window and press the  button. Alternatively, just double-click on the comparison name.

4.1.7 Interaction between subsets and comparisons

Comparisons can be created from subsets and *vice versa* (for more information about subsets, see 2.2.11). To create a subset from a comparison there is a direct function available in the *Comparison* window.

4.1.7.1 First, make sure a subset is open. If you want to create a subset that contains only the members of the comparison, create a new subset (see 2.2.11 on how to create subsets).

4.1.7.2 In the *Comparison* window, select ***File > Add entries to current subset***.

The current subset now contains the entries of the comparison, in addition to entries that may have been present in the subset before.

Conversely, a comparison can be created from a subset as follows:

4.1.7.3 Open a subset in the database (see 2.2.11).

4.1.7.4 Click on the first entry and subsequently, while holding the SHIFT key, on the last entry in the *Database entries* panel to select all entries from the subset. Alternatively, press CTRL+A on the keyboard.

4.1.7.5 Select ***Comparison > Create new comparison*** or press the  button from the *Comparisons* panel toolbar (shortcut ALT+C).

A new comparison is created, containing all entries from the subset.

4.1.8 Cluster analysis: introduction

In many cases, the user would want to perform a *cluster analysis* based on a certain experiment. The term cluster analysis is a collective noun for a variety of techniques that have the common feature to produce a hierarchical tree-like structure (*dendrogram*) from the set of sample data provided. The tree usually allows the samples to be classified based upon the *clusters* produced by the method. Apart from this common goal, the principles and algorithms used, as well as the purposes, may be very different (for more information about cluster analysis, see 4.6.1). Cluster analysis *sensu latu* has therefore been subdivided in different sections in this manual:

- Cluster analysis *sensu stricto* is based upon a matrix of similarities between database entries and a subsequent algorithm for calculating bifurcating hierarchical dendrograms representing the clusters of entries (Section 4.1 to Section 4.3).

- Phylogenetic cluster methods are methods which attempt to create trees that optimize a specific phylogenetic criterion. These methods start from the data set directly rather than from a similarity matrix (Section 4.5).

- Minimum spanning trees are trees calculated from a distance matrix, that possess the property of having a total branch length that is as small as possible (Section 4.7).

4.1.9 Calculating a dendrogram

4.1.9.1 Open the database **DemoBase**.

4.1.9.2 Select all entries except STANDARD and create a new comparison (see 4.1.3.2 to 4.1.3.4).

We will save this comparison (see 4.1.6) since it will be used throughout this manual.

4.1.9.3 Select ***Edit*** **>** ***Save*** or press the  button. The dialog box that appears prompts for a name for the comparison. Enter **All** for example.

4.1.9.4 Select an experiment type in the *Experiments* panel (e.g. **RFLP1**) and show the image by pressing the image button ( for **RFLP1**).

4.1.9.5 Select ***Clustering*** **>** ***Calculate*** **>** ***Cluster analysis (similarity matrix)***. You can also press the  button, in which case the following menu pops up (Figure 4-4).

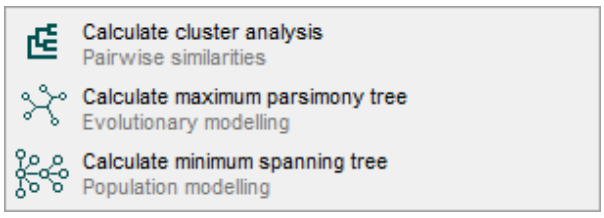


Figure 4-4. Cluster analysis menu popped up from the dendrogram button.

The first choice is the matrix-based cluster analysis discussed in this chapter, whereas the second and third choices are discussed in Section 4.3 and Section 4.7, respectively.

A *Comparison settings* dialog box pops up. For each experiment type, different settings are listed in this dialog box. More information about these settings can be found in the cluster analysis sections of fingerprint types (see 4.2) and composite data sets (see 4.3).

4.1.9.6 For this example, you can leave the default settings. Press ***<OK>*** 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 in the bottom of the window that proceeds from left to right.

When finished, the dendrogram and the similarity matrix are shown (Figure 4-2). For more information about the different panels in the *Comparison* window, see 4.1.3.

The experiment type 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.

4.1.9.7 To save the comparison with the dendrogram, select ***File*** **>** ***Save*** or press the  button. Comparison **All** now contains a dendrogram for fingerprint type **RFLP1**.

4.1.10 Calculation priority settings

FPQuest performs almost all its calculations in multi threaded mode. This means that you can further use FPQuest or any other program while time-consuming calculations are going on. In order to speed up the calculations, or make multi tasking smoother, you may want to modify the priority settings for the calculations. The calculation priority settings are grouped with other preference settings in the *FPQuest main* window.

4.1.10.1 In the *FPQuest main* window, select ***File*** **>** ***Preferences*** and click on ***Calculation priority settings*** in the list on the left side of the *Preferences* dialog box (see Figure 4-5).

The dialog box offers the choice between five priority levels. 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 FPQuest calculations while it has nothing else to do.

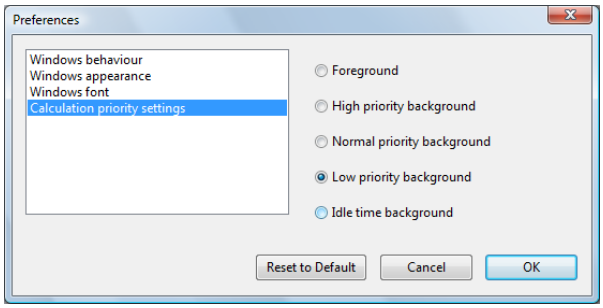


Figure 4-5. Calculation priority settings in the *Preferences* dialog box.

4.1.10.2 Select ***Low priority background*** and press ***<OK>***.

While the program is calculating, you can abort the calculations at any time using the  button.

4.1.11 Dendrogram display functions

4.1.11.1 In the **DemoBase** database, open comparison **All** or any other comparison for which a dendrogram is calculated (see 4.1.9 on how to calculate a dendrogram).

4.1.11.2 Press F4 to unselect any previous selection of database entries.

Entries can be selected from within the *Dendrogram* panel of the *Comparison* window:

4.1.11.3 To select an individual entry, hold the CTRL key and click on a dendrogram tip (where a branch ends in an individual entry). Alternatively, right-click on the dendrogram tip and choose ***Select branch into list*** from the floating menu. Repeat this action to unselect the entry.

4.1.11.4 To select a cluster on the dendrogram at once, hold the CTRL key and left-click on a branch node. Alternatively, right-click on a branch and choose ***Select branch into list*** from the floating menu. Repeat this action to unselect a branch.

When a dendrogram node or tip is clicked on, a diamond-shaped cursor appears on that position. The average similarity at the cursor's place is shown in the upper left corner of the *Dendrogram* panel. You can also move the cursor with the arrow keys.

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.

4.1.11.5 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:

4.1.11.6 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.

4.1.11.7 Select a cluster of closely related entries and ***Clustering > Collapse/expand branch***.

4.1.11.8 With ***Clustering > Show similarity values***, the average similarity of every branch is indicated on the dendrogram.

Another function, ***Clustering > Reroot tree***, only applies to so-called *unrooted trees*, i.e. neighbor joining and maximum parsimony trees. 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 outgroup 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**.

4.1.11.9 In the *Experiments* panel, select **RFLP2**.

4.1.11.10 Select ***Clustering > Calculate > Cluster analysis (similarity matrix)*** 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*".

4.1.11.11 Click somewhere in the middle of the branch. A secondary, X-shaped cursor appears.

4.1.11.12 Select ***Clustering > Reroot tree***, and the new root connects the outgroup with the rest of the entries.

4.1.11.13 The software automatically limits the displayed similarity range to the depth of the dendrogram. If you want to change this range, select ***Clustering > Set minimum similarity value***.

4.1.11.14 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***.

If a dendrogram is calculated for more than one experiment type in a comparison, you can toggle between the available dendrograms as follows:

4.1.11.15 In the *Experiments* panel, click on the experiment type for which you want to display the dendrogram (e.g **RFLP1**) and select ***Layout > Show dendrogram***. Alternatively, right-click on the experiment type and select ***Show dendrogram*** from the floating menu that appears.

4.1.12 Working with Groups

An important display function in the *Comparison* window is the creation of Groups. Groups basically are subsets of a comparison, that can be defined from clusters, from database fields, or just from any subdivision the user desires. Groups are normally displayed using rectangles of different colors next to the entries, each group having its own color. They can also be displayed using different symbols, or using alphanumerical codes. In the first place, Groups facilitate the comparison between a dendrogram or a dimensioning and a certain characteristic (database information). Groups also make the homology display between dendrograms obtained from different experiments easier. In addition, Groups are necessary in a number of derived statistical analysis functions, such as Group separation and Discriminant Analysis (4.1.15). Finally, Groups form an easy link between dimensional representations such as PCA, SOM or graphs and scatter plots on the one hand, and database field information on the other hand. To make the distinction between groups as clusters on the one hand and groups as defined by the Groups tool on the other hand, the latter Groups are always written with a capital.

First we will see how to define Groups based on the clusters in a dendrogram.

4.1.12.1 If a UPGMA dendrogram of **RFLP1** is already calculated for comparison **All**, show it as follows: right-click on **RFLP1** in the *Experiments* panel, and select ***Show dendrogram***. Otherwise, calculate a dendrogram as described in 4.1.9.4 to 4.1.9.6.

This dendrogram reveals three major clusters: *Vercingetorix, Ambiorix,* and *Perdrix* (with some exceptions).

4.1.12.2 Make sure that no entries are selected by pressing F4.

4.1.12.3 Hold the CTRL key and click on the node that connects all entries belonging to the *Vercingetorix* cluster. The entries of this cluster are now selected (as indicated by the colored arrows in the *Information fields* panel).

4.1.12.4 In the menu, select ***Groups > Assign selection to.*** The menu lists 30 different colors and accompanying symbols, from which you can choose one (e.g. the first one, green).

The selected color is shown next to all selected entries in the *Information fields* panel.

4.1.12.5 The *Groups* panel displays the number of entries next to the group color (see Figure 4-6).

4.1.12.6 Click twice in the information field **Name**, to add descriptive information to the defined group (see Figure 4-6).

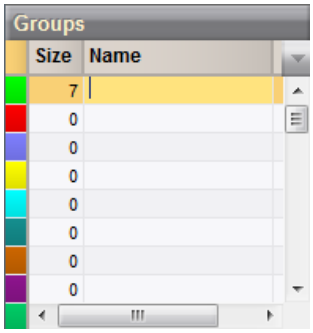


Figure 4-6. The *Groups* panel.

4.1.12.7 Press F4 to clear the selection and click on the node connecting all *Ambiorix* entries while holding the CTRL key.

4.1.12.8 Select ***Group > Assign selection to*** and choose the second color (red).

4.1.12.9 Repeat actions 4.1.12.2, 4.1.12.3, and 4.1.12.4 for the third cluster mainly composed of *Perdrix*. Use for example the third color (purple).

4.1.12.10 You can repeat these actions for two outliers of *Perdrix*, using another color. The *Groups* panel is updated whenever a new group is created.

Whatever dendrogram you now display, you will be able to recover the groups of the **RFLP1** dendrogram at a glance.

4.1.12.11 Right-click on **RFLP2** in the *Experiments* panel, and select ***Show dendrogram*** or calculate a dendrogram based on **RFLP2** if not yet present. The *Perdrix* and *Ambiorix* strains are not well separated by this technique: the second and the third Group 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:

4.1.12.12 Select a database field by clicking on the database field name, for example 'Genus'.

4.1.12.13 In the ***Groups*** menu, select ***Create groups from database field*** or right-click in the database field name

'Genus' and select ***Create groups from database field***. The *Create groups from field* dialog box appears (see Figure 4-7).

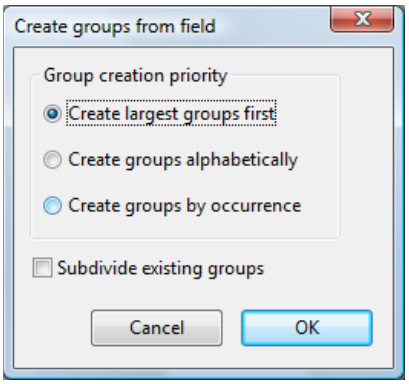


Figure 4-7. The *Create groups from field* dialog box.

The ***Group creation priority*** setting determines the order in which Groups are assigned. With ***Create largest group first***, the group containing the largest number of entries will be Group 1, the second largest group will be Group 2, etc. With ***Create groups alphabetically***, Groups will be created according to the alphabetical order of the information field. ***Create groups by occurrence*** assigns Groups in the order in which they are listed in the comparison; the first occurrence of a Group member determines its Group number. When ***Subdivide existing groups*** is disabled (not checked), any previously defined Groups will first be removed and the program will assign the new Groups based upon the selected database field only. If you check ***Subdivide existing groups***, 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.

4.1.12.14 Leave ***Subdivide existing groups*** disabled and press **<*OK*>**. The program creates three Groups according to the genus names.

> *NOTE: The maximum number of Groups that can be defined is 30. In case a database field contains more than 30 different names (text strings), the program will only assign Groups to the 30 most prevalent (***Create largest groups first*** *checked), to the first 30 in alphabetical order (***Create groups alphabetically*** *checked) or to the first 30 in occurring order (***Create groups by occurrence*** *checked).*

The *Groups* panel displays the number of entries for each group and the genus name as Group name (Figure 4-8).

4.1.12.15 Select the **Species** database field and ***Group > Create from database field*** again.

4.1.12.16 Select ***Subdivide existing groups*** and press **<*OK*>**.

Every unique species name now is assigned to a different Group. In addition, if two different genus names would have the same species name, they would

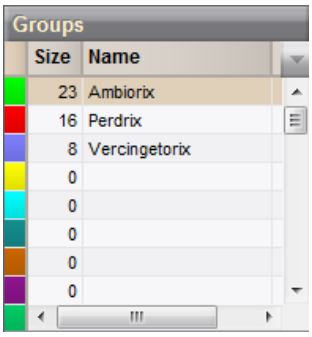


Figure 4-8. The *Groups* panel with three defined groups.

belong to a different Group too, since we kept the existing Groups based upon the genus database field (see Figure 4-9).

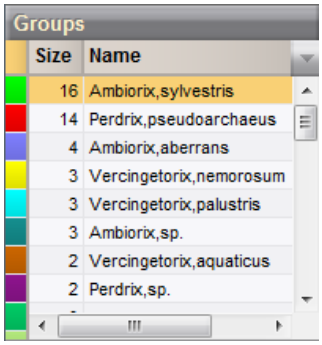


Figure 4-9. The *Groups* panel after executing the *Subdivide existing groups* command.

Since different colors are not equally distinguishable by different persons it may be useful to customize the Group colors in a user-defined scheme.

4.1.12.17 To define an own Group color scheme, select ***Groups > Edit group colors***. This brings up the *Group colors* dialog box (Figure 4-10). For each color, three slide bars (red, green and blue, respectively) can be adjusted to produce any desired color.

4.1.12.18 A thus obtained color scheme can be saved by pressing the **<*Save as*>** button, and entering a name.

A user defined color scheme can be selected from the drop-down list of ***Saved color schemes***.

4.1.12.19 To delete a saved color scheme, first select it, and then press the **<*Delete selected*>** button.

4.1.12.20 To bring up the default color scheme, press **<*Default*>**. Another predefined scheme, using pastel colors, can be loaded by pressing **<*Pastels*>**.

4.1.12.21 It is also possible to generate a scheme of transition colors by pressing **<*Range*>**. The program will ask to enter the number of colors to include in the range. Enter a number between 2 and 30.

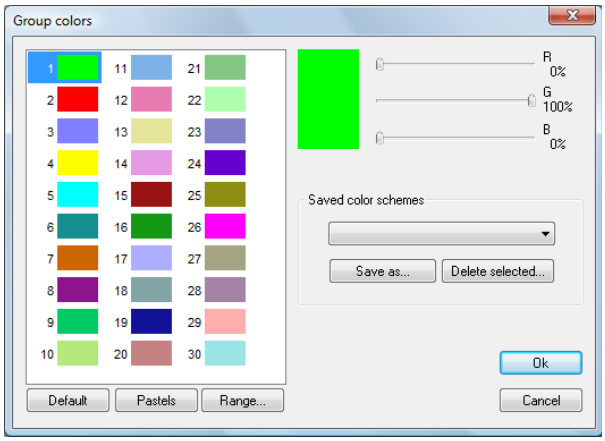


Figure 4-10. The *Group colors* dialog box.

As an alternative to using group colors, it is possible to assign a symbol to each group.

4.1.12.22 Uncheck ***Groups > Show groups using colors*** in the menu of the *Comparison* window**.**

The colors in the *Information fields* panel are replaced by symbols. To view the list of symbols in the *Groups* panel, select the button in the *Groups* panel and select ***Sign*** from the pull-down menu.

In many cases, the entry keys or particular information fields may be too long to be displayed in particular comparison types, e.g. phylogenetic trees, PCA plots and rendered trees. In such cases, 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 can be shown as follows:

4.1.12.23 In the *Comparison* window, select ***Layout > Use group numbers as key***.

The keys in the *Information fields* panel are replaced by a letter followed by a number. The letter corresponds to the Group letter. To view the list of letters in the *Groups* panel, select the button in the *Groups* panel and select *Letter* from the pull-down menu

A legend to the Group numbers can be obtained with ***File > Export database fields*** in the *Comparison* window.

Alternatively, a selected information field can be displayed instead of the key:

4.1.12.24 In the *Comparison* window, click on the information field which you would like to display as key (e.g. 'Strain number').

4.1.12.25 Select ***Layout > Use field as key*** from the menu in the *Comparison* window. The strain number is now displayed in the 'Key' field and in e.g. a PCA plot, rendered tree, etc.

4.1.13 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 or the trueness of these groups. Therefore, the software offers a range of methods that express the stability or the error at each branching level. The simplest indication of the significance of branches is showing the average similarities of the dendrogram branches (see 4.1.11.8).

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.

4.1.13.1 Right-click on **RFLP1** in the *Experiments* panel, and select ***Show dendrogram***.

4.1.13.2 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 4-11). 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.

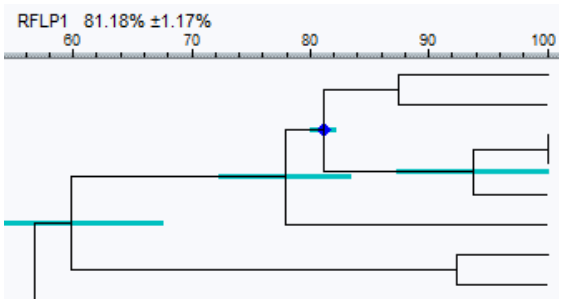


Figure 4-11. Dendrogram with error flags, detail. The average similarity and standard deviation is shown at the cursor's position (top).

4.1.13.3 Select ***Clustering > Calculate error flags*** again to remove the error flags.

The *Cophenetic Correlation* is also a parameter to express the consistency 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 FPQuest, 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.

4.1.13.4 Select ***Clustering > Calculate cophenetic correlations***.

The cophenetic correlation is shown at each branch (Figure 4-12), together with a colored dot, of which the color ranges between green-yellow-orange-red according to decreasing cophenetic correlation. This makes it is easy to detect reliable and unreliable clusters at a glance.

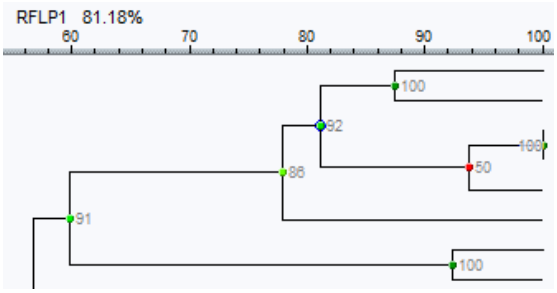


Figure 4-12. Dendrogram showing cophenetic correlation values, detail.

Bootstrap analysis[1] measures cluster significance at a different level. Instead of comparing the dendrogram to its 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 are replaced with others[2]. 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 cannot be performed on fingerprint type data directly; a band matching needs to be performed first (Section 4.3).

> *NOTES: To be able to perform bootstrap analysis on your data, do NOT choose* ***Average from experiments*** *in the* Composite data set comparison settings *dialog box (see Section 4.3). The bootstrap analysis option is only accessible when using a similarity coefficient.*

We will illustrate bootstrap analysis with the composite data set **RFLP-combined** in the **DemoBase** database. If **RFLP-combined** is not yet present, see Section 3.2 on how to set up a composite data set. First, a band matching needs to be calculated:

4.1.13.5 Select the fingerprint type **RFLP1** in the *Experiments* panel and select ***Bandmatching > Perform band matching*** or press the  button.

4.1.13.6 In the *Perform band matching* dialog box, select ***Find classes on all entries*** and press **<*OK*>** (see 4.3.2 for more information on creating a band matching).

4.1.13.7 Repeat steps 4.1.13.5 to 4.1.13.6 for fingerprint type **RFLP2**.

The composite data set **RFLP-combined** now contains the band matching information of both RFLP techniques.

4.1.13.8 Select **RFLP-combined** in the *Experiments* panel and select ***Clustering > Calculate > Cluster analysis (similarity matrix)***.

4.1.13.9 In the *Composite data set comparison* dialog box, select ***Pearson correlation*** with ***Standardized characters*** and ***UPGMA*** as clustering method. Press **<*OK*>** to calculate the dendrogram.

4.1.13.10 When the dendrogram appears, select ***Clustering > Bootstrap analysis*** and enter the number of simulations (samplings) to perform. A reasonable number of samplings is 100.

4.1.13.11 Press **<*OK*>** and wait until 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 4-12).

4.1.14 Matrix display functions

The similarity matrix is displayed in the *Similarities* panel, located at the right hand side of the *Comparison* window (see Figure 4-2).

4.1.14.1 If the similarity matrix is not shown for the selected experiment, you can display it with ***Layout > Show matrix***. This option is only available when a dendrogram was calculated for the selected experiment. A dendrogram can be calculated as described in 4.1.9.4 to 4.1.9.6.

> *NOTE: It is also possible to show the average similarities for the branches directly on the dendrogram; see 4.1.11.8.*

4.1.14.2 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

1. Efron, B. 1979. Bootstrap methods: another look at the jackknife. Ann. Statist. **7**:1-26
2. Felsenstein, J. 1985. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution* **39:** 783-791

settings for the shadings is graphically represented in the caption of the *Similarities* panel (Figure 4-13).

Figure 4-13. Adjustable similarity shading scale.

There are two ways to change the intervals for shading:

4.1.14.3 Drag the interval bars on the scale; the matrix is updated instantly.

4.1.14.4 Select ***Layout > Similarity shades*** in the menu. The maximum/minimum values for each interval can be entered as numbers.

4.1.14.5 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.

4.1.14.6 With the option ***Layout > Show matrix rulers*** (default enabled), a set of horizontal and vertical rulers appear on the similarity cell where clicked. These rulers connect the two entries from which the similarity value is derived.

If you want to find the similarity value on the matrix between two entries in the comparison, click first on the point on the diagonal of the matrix corresponding to the first entry, and then on the second entry inside the *Information fields* panel (Figure 4-14). The similarity value is the intersection between the horizontal and the vertical rulers.

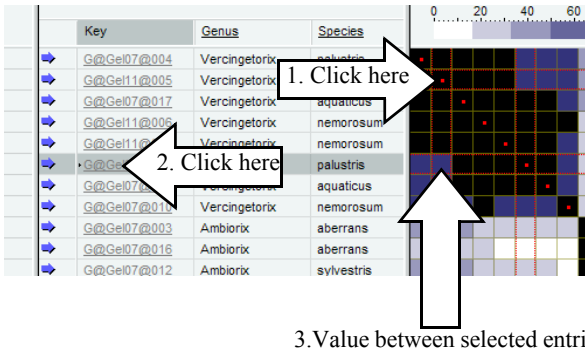


Figure 4-14. Workflow for finding a similarity value between two entries.

4.1.14.7 By double-clicking on a similarity block or value, you can pop up the detailed comparison between the two entries (4.1.2).

4.1.14.8 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:

4.1.14.9 Export the information fields with ***File > Export database fields***.

4.1.15 Group statistics

As mentioned in 4.1.12, a number of statistical functions will need the presence of Groups. These Groups statistics functions are based upon the Groups the user has defined. We have explained earlier how to define Groups (see 4.1.12.2 to 4.1.12.10), and if you have gone through the dendrogram display functions (4.1.11), the Groups are already present on the dendrogram.

The group separation statistics determine the stability of the defined groups, whether they are defined manually, derived from clusters, or created from an information field. They involve the *Jackknife* method and the "*Group violations*" measurement.

4.1.15.1 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 4-15).

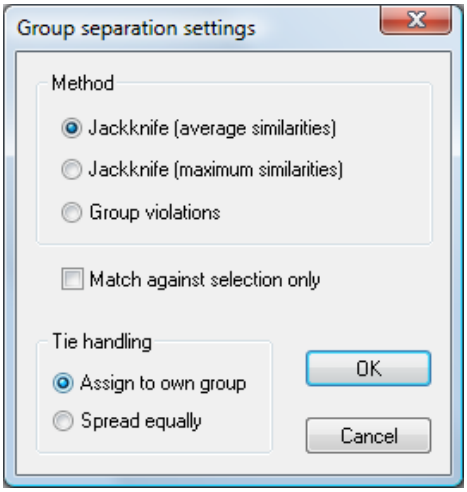


Figure 4-15. *Group separation settings* dialog box.

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 (unless ***Match against selection only*** is checked). The percentage of cases that entries are identified to the group they were originally assigned to, is a measure of the internal stability (significance) of that group. The percentage of cases that entries are identified to another group than originally assigned to, is indicative of lack of internal stability.

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 the own 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***.

4.1.15.2 Click ***<OK>*** with the default settings to display the *Group separation statistics* window (Figure 4-16).

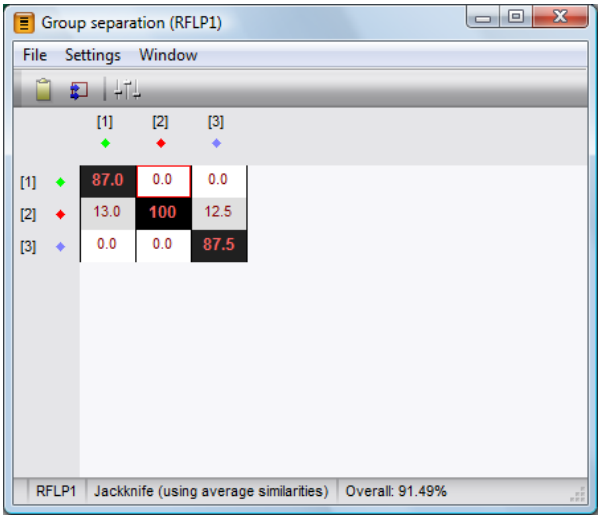


Figure 4-16. *Group separation statistics* window.

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 1 are given in column 1 (Figure 4-16), for members of group 2 in column 2, etc. In Figure 4-16 for example, 13% of group 1 members are identified as group 2, but 0% of group 2 members are identified as group 1.

The overall quality of the Group separation is indicated in the status bar of the window; it is the average of the diagonal, i.e. the total percentage of correct identifications.

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.*

4.1.15.3 Click  or select ***Settings > Statistics*** to call the *Settings* dialog box again.

4.1.15.4 Under ***Method***, select ***Group violations***. Figure 4-16 is based on group violations between three groups partitioned as above (**RFLP1**).

The group violations method compares all the similarity values within a group with those between a group and the other groups. All the values occurring in the overlap zones (see Figure 4-17) are considered "violations" of the integrity of the group.

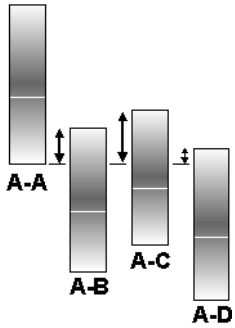


Figure 4-17. Schematic representation of internal similarity range of group A (A-A), and similarity ranges with other groups (A-B, A-C, and A-D). The overlapping values are group violations.

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.

4.1.15.5 The *Group statistics* can be copied to the clipboard using ***File > Copy to clipboard*** or by pressing the  button.

4.1.16 Printing a cluster analysis

When printing from the *Comparison* window, FPQuest first shows a print preview. This print preview shows the same information as is displayed 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.

4.1.16.1 In the *Comparison* window, select ***File > Print preview*** or press the button, which opens the *Comparison print preview* window (Figure 4-18).

The *Comparison print preview* window is divided in two panels, which are both dockable (see 1.6.4 for display options of dockable panels). The *Overview* panel shows an overview of the pages that will be printed, with the actual page in yellow. In the *Print preview* panel, the actual page is shown.

4.1.16.2 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.

4.1.16.3 It is possible to zoom in and out on a page using ***Edit > Zoom in*** and ***Edit > Zoom out*** (shortcuts CTRL+PgUp and CTRL+PgDn on the keyboard) or by using the zoom slider (see 1.6.7) in the *Preview* panel.

4.1.16.4 When zoomed, the horizontal and vertical scroll bars allow you to scroll through the page.

4.1.16.5 The whole image can be enlarged or reduced with ***Layout > Enlarge image size*** or ***Layout > Reduce image size***.

4.1.16.6 If a similarity matrix is available, it can be shown and printed with ***Layout > Show similarity matrix*** or .

4.1.16.7 With ***Layout > Show comparison information***, the name of the comparison (if already saved) and the number of entries are indicated on top of the first page.

4.1.16.8 It is possible to display a header line with the database field names when ***Layout > Show field names*** is selected.

On top of the preview page, there are a number of small yellow slide bars (Figure 4-18). 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 hidden);

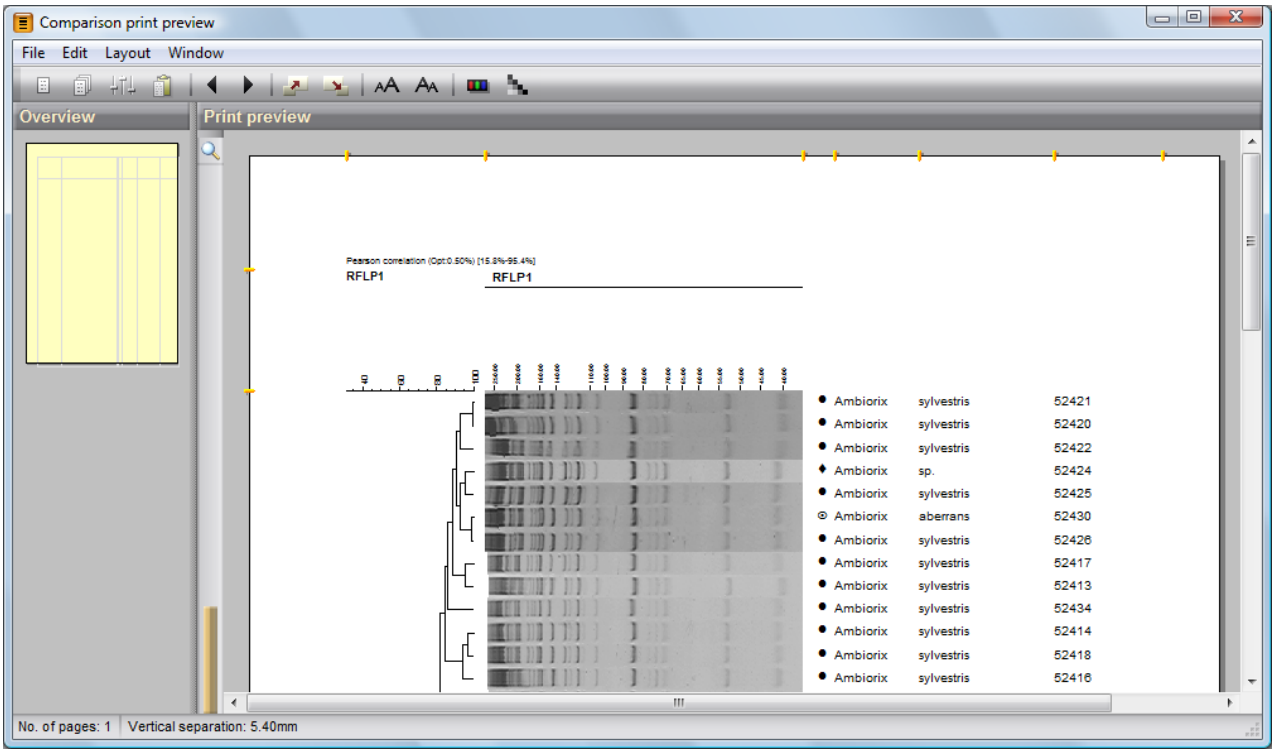


Figure 4-18. ***The Comparison print preview*** **window.**

- Right margins of different information fields (except the hidden fields);
- If the similarity matrix is 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 the 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.

4.1.16.9 You can preview and print the image in full color with ***Layout > Use colors*** or  .

4.1.16.10 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.

4.1.16.11 If the preview is covering more than one page, you can click on a specific page in the *Overview* panel to select a page from the range.

4.1.16.12 With ***File > Print this page*** or  , the current page is printed.

4.1.16.13 Use ***File > Print all pages*** or  to print all pages at once.

4.1.16.14 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 FPQuest 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 (or a more recent version of the same software) to import the graphical data.

With the ***Copy page to clipboard*** 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 (4.1.16.5).

> *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* FPQuest main *window (under* **Fingerprint types***) and select* **Layout > Show space between gelstrips**.

4.1.16.15 Select ***File > Exit*** to close the *Comparison print preview* window.

4.1.17 Exporting rendered trees

In publications and presentations, particularly in a phylogenetic context, a dendrogram is sometimes represented as a real tree with a stem and branches. Such representations can be achieved in FPQuest using the *rendered tree* option in the *Comparison* window (Figure 4-19). This option should be used with care, as it will only produce acceptable pictures from a very limited number of entries and with fairly equidistant members.

Rendered trees can be created from a standard rooted tree in the *Comparison* window as well as from unrooted phylogenetic trees (maximum parsimony).

4.1.17.1 In the *Comparison* window, create a dendrogram containing a small and not too heterogeneous group of entries (e.g. 10 entries).

4.1.17.2 Select ***Clustering > Rendered tree export***. A *Rendered tree settings* dialog box (Figure 4-20) prompts for a number of settings:

- ***Hide branches if shorter or equal to*** allows all entries that are very similar to be grouped together at one branch tip. This allows simpler trees to be produced and may avoid starlike branch tips to occur.
- ***Hide distance labels if shorter or equal to*** sets a minimum for the distance values to be shown on the branches. If many short distances occur, the labels may overlap, which can be avoided by only allowing larger distance to be shown. If the value is set to 100 (or more), no distance labels will be shown.
- ***Tree type*** can be ***Rooted*** or ***Unrooted***. If the dendrogram is rooted by nature (e.g. UPGMA) it makes no sense to export an unrooted tree from it.
- ***Display type*** can be ***Rendered***, i.e. with a thicker stem and smooth, gradually narrowing branches, or ***Outlines***, where stem and branches are represented by straight lines.
- ***Display field*** is a pull-down listbox where one of the available database fields can be chosen.

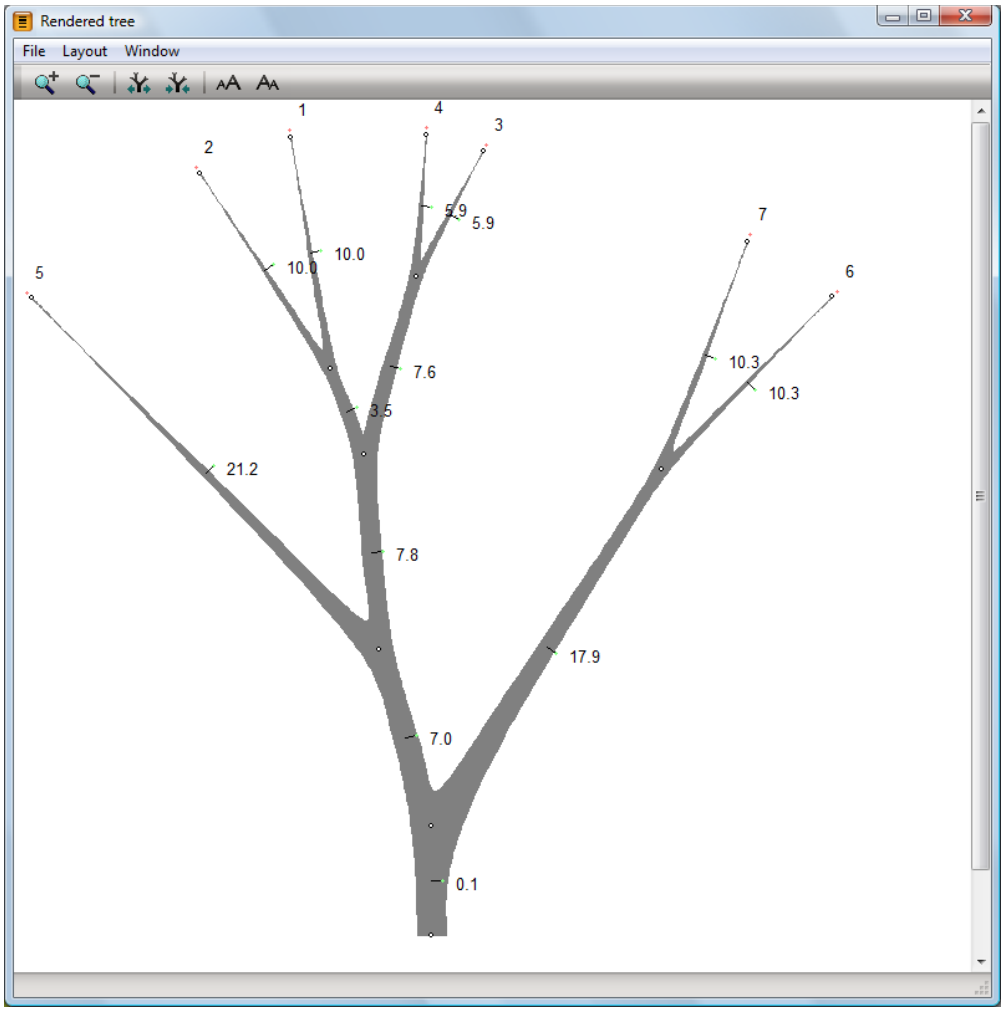


Figure 4-19. The *Rendered tree* window.

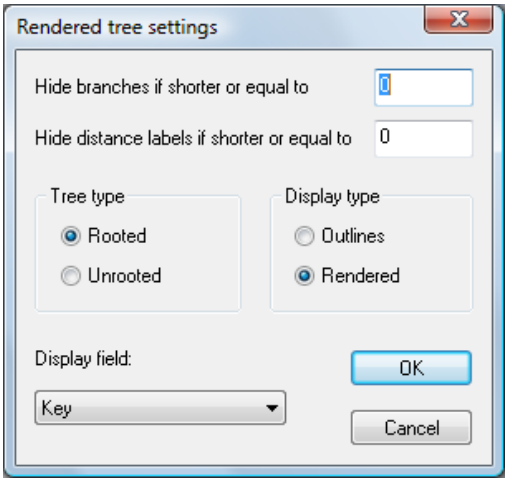


Figure 4-20. *Rendered tree export settings* dialog box.

NOTE: In case the information fields are too long, it is possible to replace them by a group code, if groups are defined (4.1.12). The use of group codes is explained in 4.1.12.23.

4.1.17.3 Rendered trees can also be exported from maximum parsimony trees (see Section 4.3). If a *rooted* rendered tree is exported, the highlighted branch of the unrooted tree will be used to create the root.

4.2 Cluster analysis of fingerprints CL FP

4.2.1 Fingerprint comparison settings

The comparison settings of fingerprint types will be illustrated using the **DemoBase** database.

4.2.1.1 Open the database **DemoBase**.

4.2.1.2 Open the comparison **All** if already existing. Alternatively, select all entries except STANDARD (see 4.1.3.2 to 4.1.3.4) and create a new comparison (4.1.3.4).

4.2.1.3 Select **RFLP1** in the *Experiments* panel and show the normalized gel image by pressing the corresponding button.

4.2.1.4 Select ***Clustering*** **>** ***Calculate*** **>** ***Cluster analysis (similarity matrix)***. You can also press the button, in which case the following menu pops up (Figure 4-21).

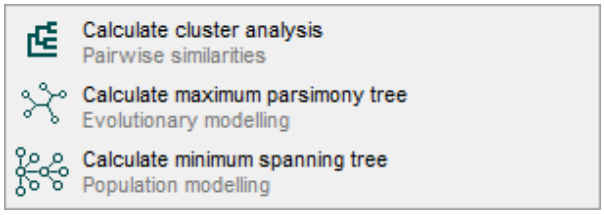


Figure 4-21. Cluster analysis menu popped up from the dendrogram button.

The *Comparison settings* dialog box allows you to specify the similarity coefficient to calculate the similarity matrix, and the clustering method to be applied (see Figure 4-22).

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:

1. The ***Jaccard*** coefficient

$$s_J = \frac{N_{AB}}{N_A + N_B - N_{AB}}$$

2. The ***Dice*** coefficient

$$s_D = \frac{2N_{AB}}{N_A + N_B}$$

3. The ***Jeffrey's X*** coefficient

$$S_X = \frac{1}{2}\left(\frac{N_{AB}}{N_A} + \frac{N_{AB}}{N_B}\right)$$

4. The ***Ochiai*** coefficient

$$S_O = \frac{N_{AB}}{\sqrt{N_A N_B}}$$

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. The ***Relaxed doublet matching*** option allows a single band to match with two bands of a doublet, on condition that both bands of the doublet fall within the tolerance window from the single band.

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***. The option ***Advanced*** is explained in Section 4.6.

4.2.1.5 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 which these bands can be considered as matching.

4.2.1.6 Press the **<*Position tolerances*>** button.

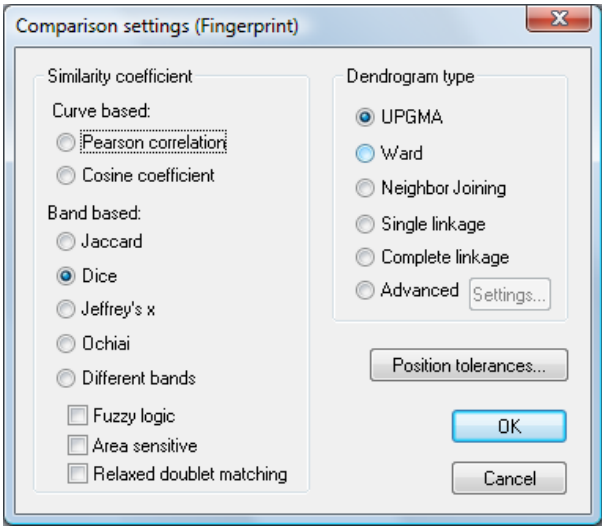


Figure 4-22. The *Comparison settings* dialog box.

The *Position tolerance settings* dialog box for the fingerprint type is popped up (Figure 4-23).

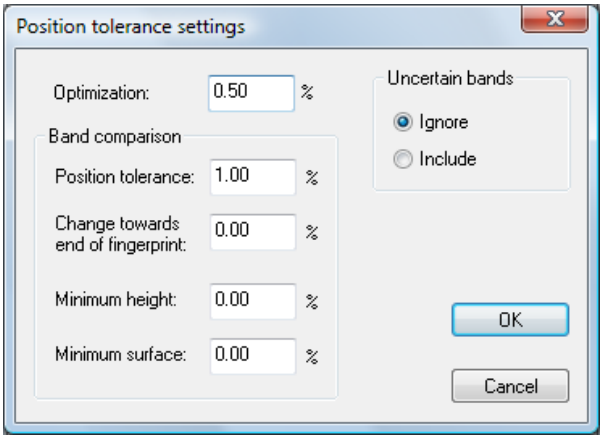


Figure 4-23. *Position tolerance settings* dialog box of a fingerprint type.

The ***Position tolerance*** is the maximal shift allowed (in percentage of the pattern length) between two bands to consider them 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 4.2.4, 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 4-24.

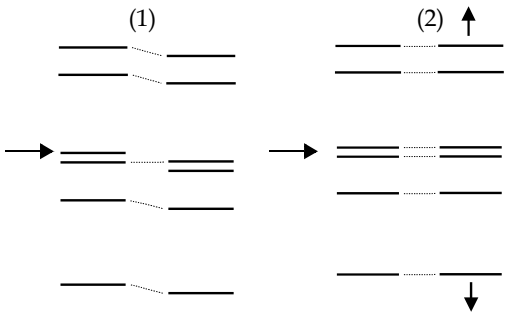


Figure 4-24. Effect of position tolerance (1) and optimization (2) on the matching between patterns.

In 4.2.4, we also discuss how the program can 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 (3.1.10.1).*

The ***Uncertain bands*** option allows you to either include uncertain bands or ignore them (see 3.1.7.1). When ***Ignore*** is chosen, uncertain bands are not taken into account. 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 4-22).*

4.2.1.7 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*>**.

4.2.1.8 Press **<*OK*>** again in the *Comparison settings* dialog box to start the cluster analysis.

When finished, the dendrogram and the similarity matrix are shown. For more information about the panels in the *Comparison* window, see 4.1.3.

4.2.2 Fingerprint display functions

Additional information for fingerprint types can be shown in the *Experiment data* panel and data can be

exported as text file. These features will be illustrated with the fingerprint type **RFLP1**.

4.2.2.1 Make sure the image of **RFLP1** is shown in the *Experiment data* panel (see 4.1.3.10).

4.2.2.2 Press the  button or ***Layout > Show metrics scale*** to display the molecular weight scale of the selected fingerprint type.

4.2.2.3 Select ***Layout > Show bands*** or press  to show or hide the band positions in the *Experiment data* panel.

4.2.2.4 One can also show only the band positions in the *Experiment data* panel without showing the actual image. Select ***Layout > Show image*** or press  to show or hide the image.

4.2.2.5 When bands are shown on the image, they can be exported as a tab-delineated 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 (in case a regression curve is calculated for the reference system used; see 3.1.10).

4.2.2.6 Select ***Layout > Show densitometric curves*** or press  to show or hide small densitometric curves in the *Experiment data* panel. One can also show only the curves without showing the actual image (see 4.2.2.4).

4.2.2.7 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 (e.g. in case of profiles from automated sequencers), it can be useful to display the curves as pseudo gelstrips (reconstructed images). This option is selected in the *Fingerprint type* window as follows:

4.2.2.8 In the *FPQuest main* window, double-click on a fingerprint type (e.g. **RFLP1**) in the *Experiments* panel.

4.2.2.9 In the *Fingerprint type* window, select ***Layout > Show curves as images***. If densitometric curves are now shown in a comparison (4.2.2.6), 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:

4.2.2.10 In the *Fingerprint type* window, select ***Layout > Rescale curves***. If densitometric curves are now shown in a comparison (4.2.2.6), they will all be displayed with equal intensity.

4.2.2.11 The image of patterns can be shown with a space between the gelstrips. To do so, open the *Fingerprint type* window in the program's *FPQuest main* window and select ***Layout > Show space between gelstrips***.

4.2.3 Defining 'active zones' on fingerprints

When clustering fingerprints, one is not necessarily interested in comparing 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.

For each fingerprint type, it is possible to define *excluded regions* which are applied for all comparisons using this fingerprint type.

4.2.3.1 Select any entry in the **DemoBase** database that contains a fingerprint of **RFLP1**.

4.2.3.2 Open the *Fingerprint type* window for **RFLP1** in the *Experiments* panel.

At the bottom of the window, the fingerprint of the selected database entry is shown (Figure 4-25).

4.2.3.3 To exclude a region for comparison, hold the left mouse button and the SHIFT key simultaneously while dragging the mouse pointer over the fingerprint.

The excluded region becomes cross-hatched in red. In the bottom part of the window, the parts of the fingerprints that are included for comparison are shown as percentages (see Figure 4-25).

4.2.3.4 To include a region, hold the left mouse button (without holding the SHIFT key), while dragging the mouse pointer over the fingerprint.

4.2.3.5 You can for example exclude the top 15% and the end 15% of the fingerprints.

> *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.*

4.2.3.6 You can specify the exact start and end of the active zone(s) using a script available on Bio-Rad' website. The scripts can be launched from the *FPQuest main* window, using the menu ***Scripts > Browse***

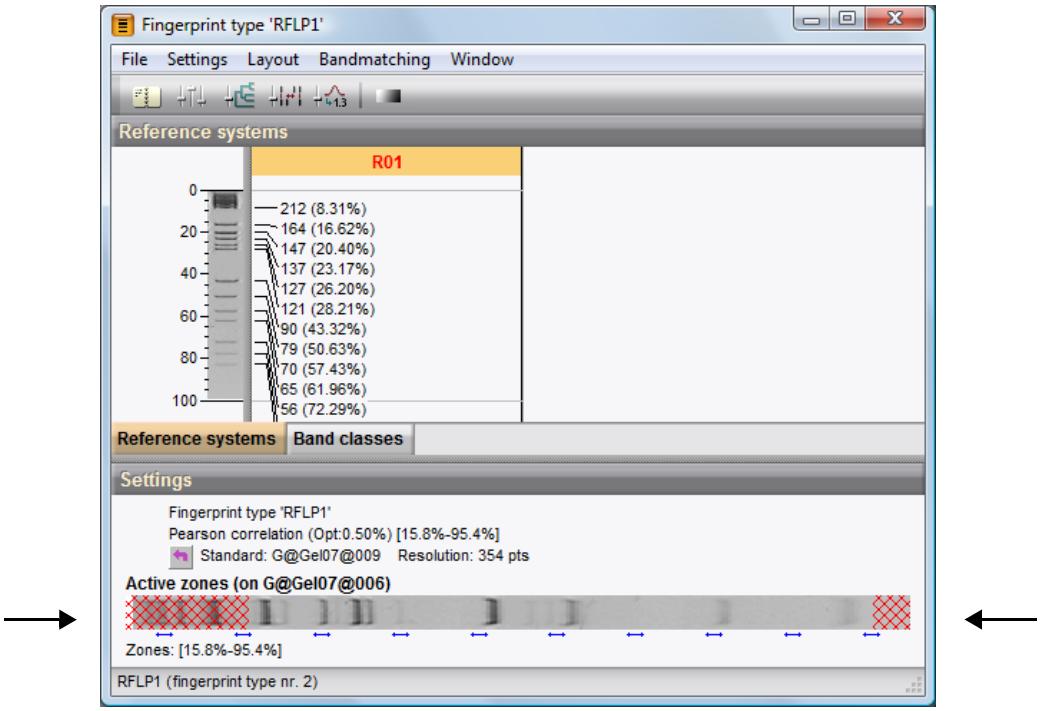


Figure 4-25. ***Fingerprint type*** **window with excluded regions defined (see arrows).**

Internet, or  , and then selecting ***Fingerprint related tools > Set active zones***.

4.2.3.7 Back in the *Comparison* window, select **RFLP1** and ***Clustering > Calculate > Cluster analysis (similarity matrix)***, to recalculate the dendrogram using the excluded regions.

4.2.4 Calculation of optimal position tolerance optimization and settings

FPQuest possesses a very interesting option to calculate automatically the optimal settings for position tolerance and optimization 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. The program will calculate similarity matrices with varying position tolerances. Within a limited range, 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 launched 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 4.1.11) already defined, e.g. based upon cluster analysis or partitioning (see 4.1.15). 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* x *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 small numbers of entries. Alternatively, both parameters can be calculated separately.

Given the time needed to calculate *n x 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:

4.2.4.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison containing all entries except the STANDARDs (see 4.1.9).

4.2.4.2 In case no groups are defined, select the **Genus** database field and ***Groups > Create from database field***.

4.2.4.3 Select **RFLP1** in the *Experiments* panel, and ***Clustering > Tolerance & optimization analysis***.

The *Comparison settings* dialog box appears (see Figure 4-22) where you can select the coefficient and clustering method. Only the coefficient will influence the calculation of the optimization.

4.2.4.4 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 4-26) showing the group separation in function of the allowed optimization in the right diagram.

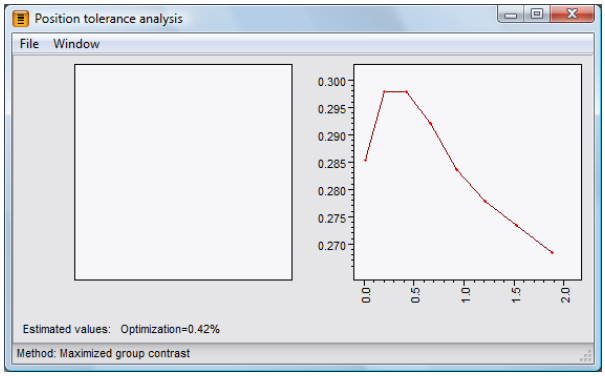


Figure 4-26. Position tolerance analysis. Optimization analysis shown for a curve-based coefficient.

The ideal optimization value is shown (bottom) and is automatically saved in the settings for the experiment type.

4.2.4.5 Close the window with ***File > Exit*** and select ***Clustering > Tolerance & optimization analysis*** again.

4.2.4.6 This time, select the ***Dice*** coefficient and press **<*OK*>**.

4.2.4.7 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 4-27) shows the group separation in function of the allowed band matching tolerance in the left diagram.

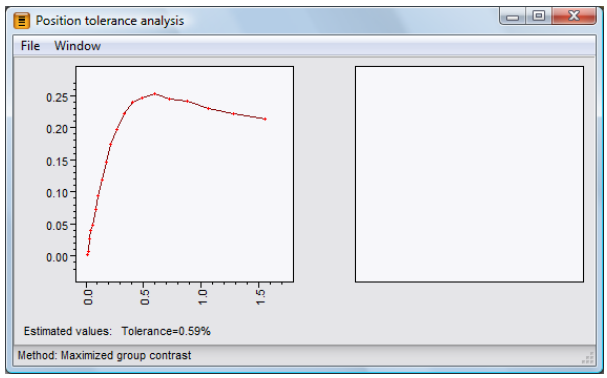


Figure 4-27. Position tolerance analysis. Position tolerance analysis shown for a band matching coefficient.

The position tolerance value is shown (bottom) and is automatically saved in the settings for the experiment type.

4.2.4.8 Close the window with ***File > Exit***.

4.3 Band matching and polymorphism analysis CQ FP

4.3.1 Introduction

Band matching is a comparison function for fingerprint types, which can be executed on any selection of entries from the database. In a first step, FPQuest divides all the bands found among the selected patterns into *classes of common bands* (1 to 8 in Figure 4-28). 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 4-28.

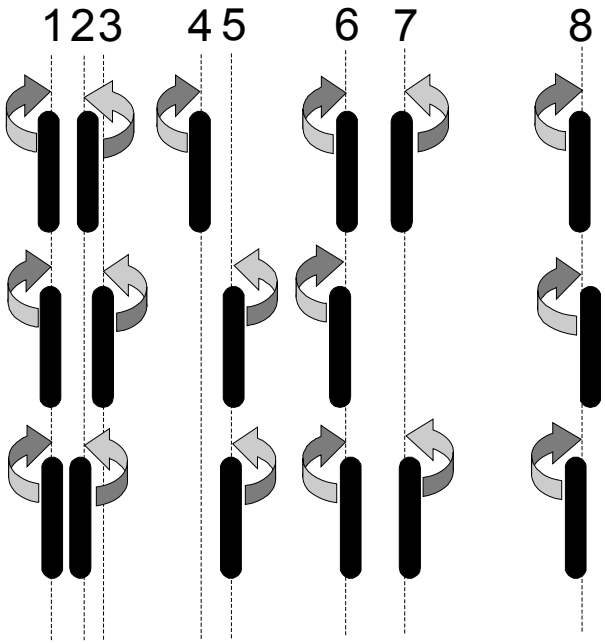


Figure 4-28. Comparative quantification: bands are assigned to classes.

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 4-28, 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 4-28) is shown in Figure 4-29.

	1	2	3	4	5	6	7	8
Pattern 1	+	+	-	+	-	+	+	+
Pattern 2	+	-	+	-	+	+	-	+
Pattern 3	+	+	-	-	+	+	+	+

Figure 4-29. Binary presence/absence table of banding patterns.

Instead of using binary (+/−) data, the same tables can be generated using band intensities obtained from the curves (band heights or surfaces) or from the two-dimensional pattern contours (volumes or concentrations).

The use of band matching tables is obvious: it provides a binary or numerical character table for fingerprint type patterns, which allows a number of statistical techniques to be applied, including Minimum Spanning Trees (Section 4.7), Maximum Parsimony trees (Section 4.3), dimensioning techniques such as Principal Components Analysis and related techniques (Section 4.8), and bootstrap analysis on dendrograms (4.1.13).

To visualize a band matching table as a character matrix (binary or quantitative), it is necessary that a composite data set is associated with the fingerprint type. Therefore, the use of composite data sets is described here in association with band matching tables (see 4.3.9). However, it is possible to apply the techniques mentioned in the previous paragraph directly on the fingerprint type without having a composite data set associated to it.

4.3.2 Creating a band matching

4.3.2.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison containing all entries except the STANDARDs (see 4.1.9).

4.3.2.2 Select **RFLP1** in the *Experiments* panel and press the button or ***Layout > Show image***.

4.3.2.3 Choose ***Bandmatching > Perform band matching*** or press .

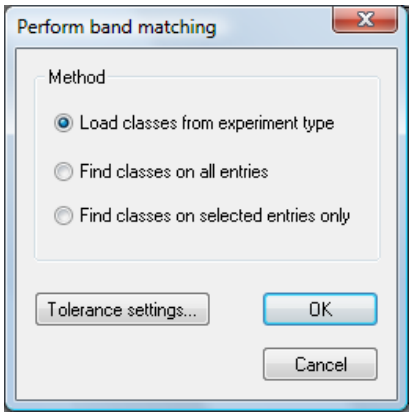


Figure 4-30. ***Perform band matching*** **dialog box of a fingerprint type.**

The *Perform band matching* dialog box pops up (see Figure 4-30), listing three different band matching options.

- ***Load classes from experiment type***: the band classes stored with the experiment type are loaded (see 4.3.5). This way you can have perfect control on what bands to use in the analysis.

- ***Find classes on all entries:*** a band matching is performed on all entries within the comparison.

- ***Find classes on selected entries only***: a band matching is performed on the currently selected entries only.

4.3.2.4 Press **<*Tolerance settings*>** to open the *Position tolerance settings* dialog box (see Figure 4-31).

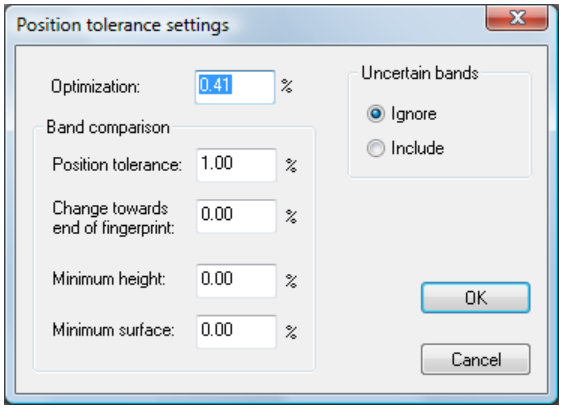


Figure 4-31. ***Position tolerance settings*** **dialog box of a fingerprint type.**

The ***Position tolerance*** is the maximal shift allowed (in percentage of the pattern length) between two bands allowed to consider them 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 position tolerance, see the example in Figure 4-24.

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 3.1.7.1). When ***Ignore*** is chosen, uncertain bands are ignored. This means that in composing a band matching table, the software will omit the uncertain bands, considering them as characters that are unknown. 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.

4.3.2.5 For this example, 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*>**.

4.3.2.6 Because no band classes are yet defined for **RFLP1** (see 4.3.5), select ***Find classes on all entries*** in the *Perform band matching* dialog box and press **<*OK*>**.

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 4-32) and the bands are linked to a class in red.

> *NOTE: Band classes are only defined within active zones of the fingerprint type. Active zones can be set in the* Fingerprint type *window of the corresponding fingerprint type (see 4.2.3).*

4.3.2.7 Zoom in on the image as necessary using the zoom functions  and  (***Layout > Zoom in*** and ***Layout > Zoom out***) or by using the zoom sliders (see 1.6.7 for instructions on how to use the zoom sliders). The latter option allows you to zoom separately in the horizontal () and vertical () direction. Horizontal zooming can also be achieved via ***Layout > Stretch (X dir)*** (keyboard shortcut CTRL+SHIFT+PgUp) and ***Layout > Compress (X dir)*** (keyboard shortcut CTRL+SHIFT+PgDn).

Zooming in horizontally can be an interesting option for long patterns with numerous small bands, such as **AFLP** in the **DemoBase**. This causes the image to be enlarged in the horizontal direction only, so that sharp bands become better visible, without loosing the overview of a large number of patterns.

4.3.2.8 Press the  button or ***Layout > Show metrics scale*** to display the molecular weight scale of the fingerprint type.

After having performed a band matching, all band classes are labelled with a band class label. The band

class labels are listed on top of the image. If a band class is selected, its label is highlighted (see Figure 4-33).

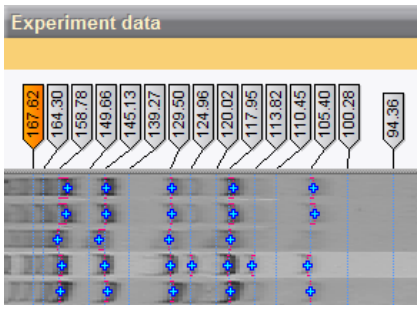


Figure 4-33. Band class labels.

If a regression curve is calculated for the reference system(s) of the selected fingerprint type (see 3.1.10.12 - 3.1.10.16), the metric positions of the band classes are displayed in the labels (e.g. 167.62; 164.30; ...).

4.3.2.9 Double-click on a band class label, or select ***Band-matching > Band class information***, or press CTRL+I to open the *Band class information* dialog box.

The *Band class information* dialog box contains detailed information on the band class (see Figure 4-34):

- **Name:** If a regression curve is calculated for the reference system(s), the default name of the band class label is the metric position of the band class. This metric position is the average position of all bands belonging to that band class. Each band class name is editable and can be changed to any name of your choice. Band class names can be changed in the *Band class information* dialog box, but also from within the *Fingerprint type* window (see 4.3.5). When changing the band class names, the metric positions of the band classes remain present in the database (see *Position (metrics)* column in Figure 4-37).

- **Position:** The position reflects the relative position of the band class, derived from the regression curve.

- **Occurrence:** The occurrence corresponds to the relative occurrence of bands in the band class, expressed as a percentage of the run length.

- **Position spread box:** The position spread box lists the standard deviation of the bands to the band class and the scores of the 50th, 90th and 98th percentile. The scores are the relative positions below which respectively 50, 90 and 98% of the bands are found.

4.3.2.10 Close the *Band class information* dialog box by pressing **<*OK*>**.

4.3.3 Manual editing of a band matching

Due to shape or distribution, the program does not always assign the bands to the correct class. Therefore, you can manually correct the assignments.

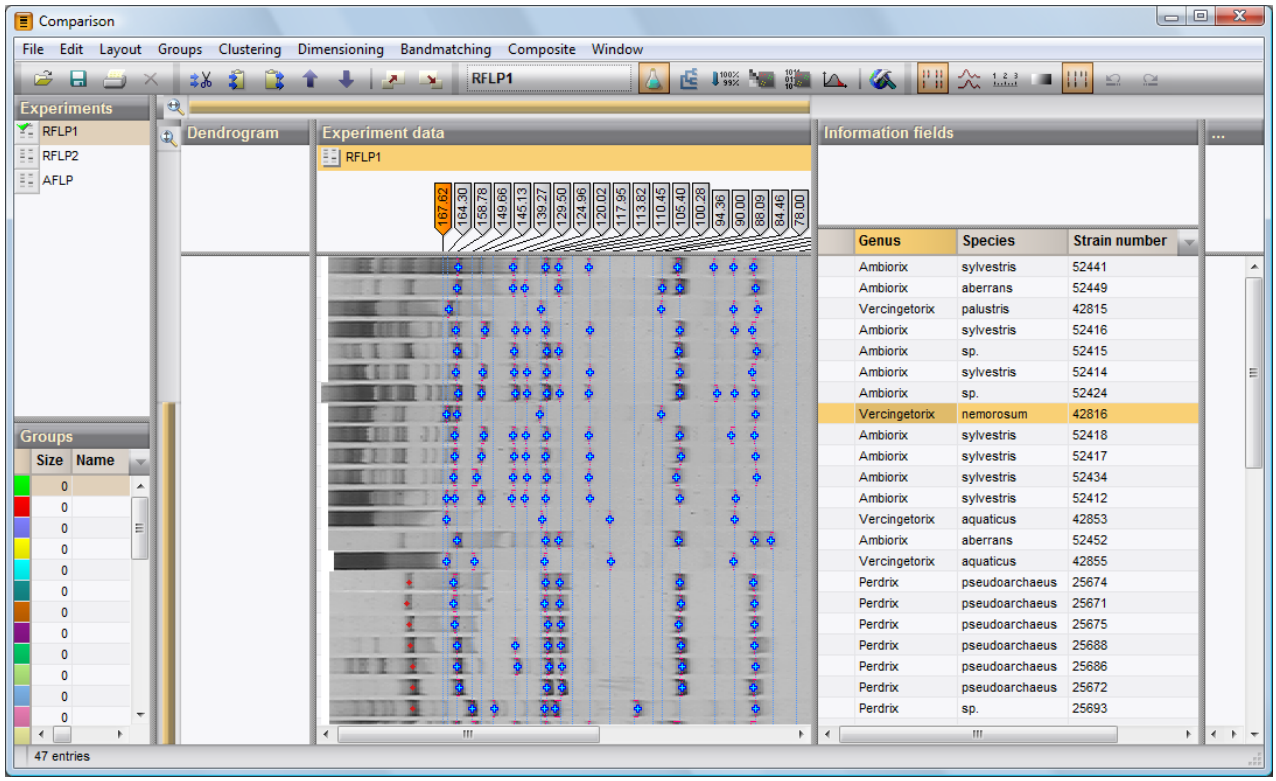


Figure 4-32. Band matching analysis in the *Comparison* window.

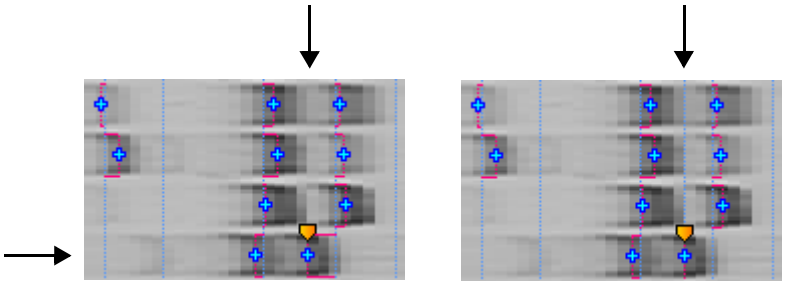


Figure 4-36. Splitting up a band class into two band classes.

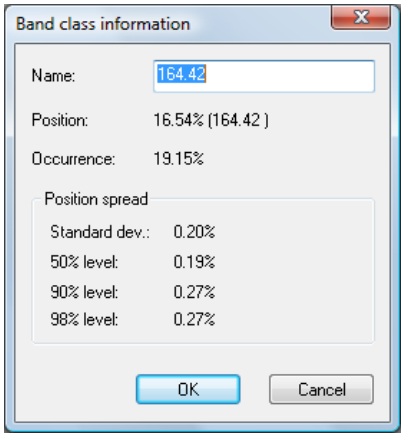


Figure 4-34. The *Band class information* dialog box.

For the manual band matching editing tools, a multi-level undo and redo function is available. The undo function can be accessed with ***Bandmatching* > *Undo*** or CTRL+Z or the  button. The redo function is accessible through ***Bandmatching* > *Redo*** or CTRL+Y or the  button.

In Figure 4-35, the band marked with the arrow is assigned to the left of two close classes, whereas it should be assigned to the right class.

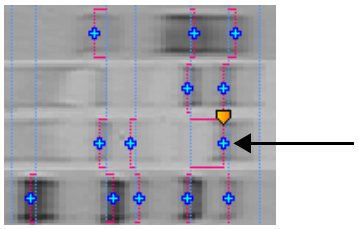


Figure 4-35. Detail of band class assignments.

NOTE: You can easily see which bands 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.

Reassigning a band to another class can be done with a simple drag-and-drop procedure:

4.3.3.1 Select the band that was wrongly assigned. While pressing the mouse button, drag it to the band class where it should be assigned to and release the mouse button.

If you do not wish to use a single band in a band matching analysis, you can undo its assignment as follows:

4.3.3.2 Click on the band that you want to unassign and drag it outside of the gelstrip.

A whole band class is deleted as follows:

4.3.3.3 Click on a band belonging to the band class.

4.3.3.4 Right-click on the band, and select ***Band classes* > *Remove band class*** from the floating menu. You can also press SHIFT+DEL on the keyboard to remove the selected band class.

If different bands are incorrectly assigned to the same class, you can create a second class as follows (Figure 4-36):

4.3.3.5 Select a band which should belong to a new class (see Figure 4-36).

4.3.3.6 Right-click on the band, and select ***Band classes* > *Add new band class*** from the floating menu or press SHIFT+ENTER on the keyboard.

The program asks "*Do you want to auto assign bands to the new class?*". If you press **<*No*>**, the new band class will contain only the selected band. If you select **<*Yes*>**, 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 drag-and-drop procedure explained in 4.3.3.1.

If bands are incorrectly assigned to different classes, you can merge the classes as follows (Figure 4-36):

4.3.3.7 Choose a band which occurs quite in the middle of the two classes.

4.3.3.8 Right-click on the band, and select ***Band classes* > *Add new band class*** from the floating menu or press SHIFT+ENTER on the keyboard. Press **<*No*>** when the program asks to auto assign bands to the new class.

4.3.3.9 Choose a band which belongs to the left class.

4.3.3.10 Right-click on the band, and select ***Band classes > Remove band class*** from the floating menu, or press SHIFT+DEL.

4.3.3.11 Choose a band which belongs to the right class (left-click).

4.3.3.12 Right-click on the band, and select ***Band classes > Remove band class*** from the floating menu, or press SHIFT+DEL.

4.3.3.13 Select the new band class to which all the bands should belong (left-click). The selector becomes blue.

4.3.3.14 Right-click, and select ***Band classes > Auto assign all 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:

4.3.3.15 Right-click on the band and select ***Band classes > Remove band from class*** or DEL.

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:

4.3.3.16 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.

4.3.3.17 If all assignments are corrected, you can save the band matching with ***File > Save*** or  .

NOTE: A band matching is saved along with the comparison. When a comparison is opened and a band matching is available for the experiment type selected, the  *button shows up*  . *The graphical representation of the band matching can be displayed again by* **Layout > Show bands** *or by pressing the*  *button.*

4.3.4 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.

4.3.4.1 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 performed 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:

4.3.4.2 Select a few entries in the database. If you have executed the previous step (4.3.4.1) there are still some entries selected and placed on the clipboard.

4.3.4.3 [In case you would have copied something else in the meantime, select ***Edit > Copy selection*** or  in the *FPQuest main* window.]

4.3.4.4 With ***Edit > Paste selection*** or  in the other comparison, the selected entries are placed back in the band matching.

4.3.4.5 Select ***Bandmatching > Search band classes***. The *Perform band matching* dialog box as in Figure 4-30 is shown. Select ***Find classes on all entries*** and press ***<OK>***.

The program now asks "*Remove existing band classes?*".

4.3.4.6 In order to preserve the existing band matching, it is important to answer ***<No>*** to this question.

4.3.5 Saving band classes to a fingerprint type

After having defined band classes in the *Comparison* window, you can save the band classes to the corresponding fingerprint type.

4.3.5.1 Select ***Bandmatching > Save band classes to experiment type***.

4.3.5.2 The program asks for confirmation, select ***<Yes>***.

4.3.5.3 Open the *Fingerprint type* window for **RFLP1** and select the *Band classes* panel (see Figure 4-37).

All band classes defined for **RFLP1**, together with their relative and metric positions, are listed in the *Band classes* panel. The names of the band classes can be edited by clicking twice in the information fields (see 1.6.6 on direct editing of information fields).

If band classes are saved for a fingerprint type, the band classes can be loaded when checking ***Load classes from experiment type*** in the *Perform band matching* dialog box (see Figure 4-30).

The ability to edit, save and load band classes has several interesting implementations:

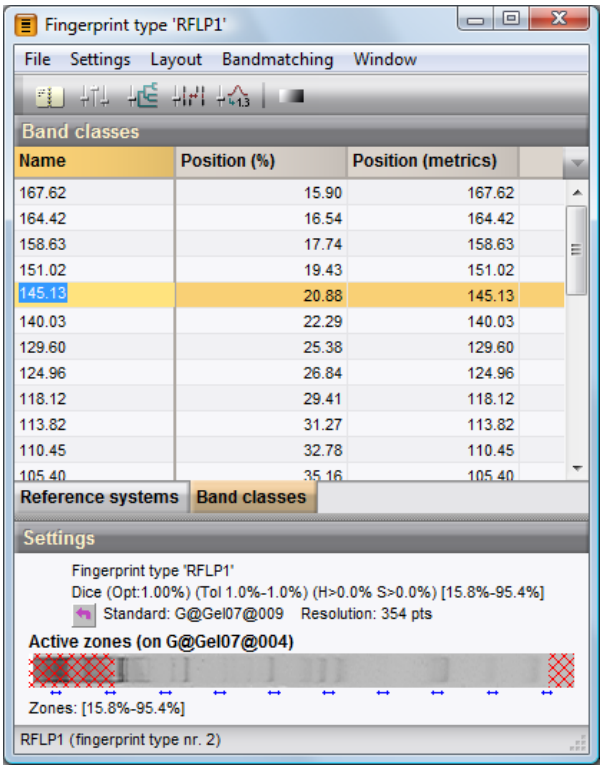


Figure 4-37. The *Band classes* panel.

- Perfect control on what bands to use in the analysis.
- Every new set of fingerprint profiles can easily be compared based upon the predefined set of band classes.
- Band matching tables can be reduced to the relevant information, e.g. bands that carry genetic marker information.

4.3.5.4 To add a band class to the list, select ***Bandmatching > Add new band class*** in the *Fingerprint type* window.

4.3.5.5 Select ***Bandmatching > Remove band class*** to remove a band class from the list.

4.3.6 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 FPQuest: 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.

4.3.6.1 In the band matching analysis created in 4.3.2, select ***Bandmatching > Band class filter***.

This pops up the *Band filtering settings* dialog box (Figure 4-38). It exists of two parts: the upper part "***Remove all bands below…***" is to filter individual bands within a given 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.

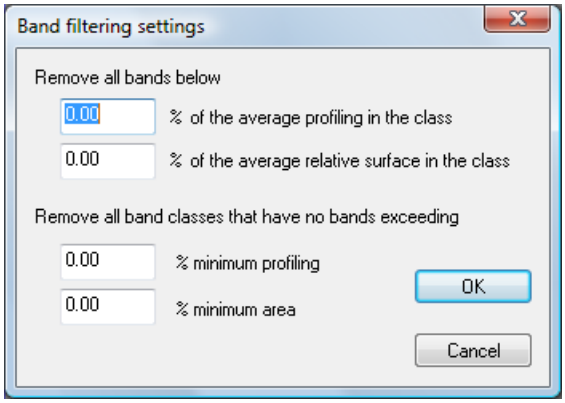


Figure 4-38. *Band filtering settings* dialog box for band matching.

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).

4.3.6.2 Within a band class, you can ***Remove all bands below*** a certain ***% of the average profiling in the class***.

If you enter 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 surface as filtering factor.

4.3.6.3 Within a band class, you also can furthermore ***Remove all bands below*** a certain ***% of the relative surface in the class***.

In this case, if you enter 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 genetic 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, FPQuest offers the possibility to have all band classes excluded from the band matching table that do not contain at least one clear relevant band.

4.3.6.4 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 enter 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.*

4.3.6.5 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.

4.3.7 Exporting band matching information

Band matching information can be exported as a binary (presence/absence) table or as a quantitative character table.

4.3.7.1 In the band matching analysis created in 4.3.2, select ***Bandmatching > Export band matching***. The program will ask *"Export quantitative band information?"*.

4.3.7.2 Press **<*No*>** to export the band matching information as a binary (presence/absence) table in tab-delimited format.

The exported band intensity values are based on the *Comparative quantification settings* for the used fingerprint type. This option can be defined in the *Fingerprint type* window (see 3.1.10.2), but it can also be changed in the *Comparison* window, as follows:

4.3.7.3 Select ***Bandmatching > Comparative Quantification settings***. This opens the *Comparative Quantification settings* dialog box (Figure 4-39).

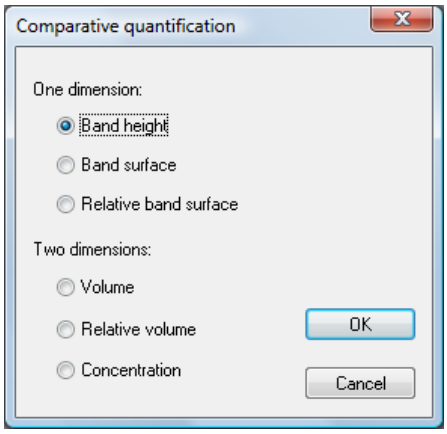


Figure 4-39. The *Comparative quantification settings* dialog box.

One dimension quantification is based on the densitometric curves extracted from the patterns (see paragraph 3.1.7): ***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, but expressed 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 3.1.7): ***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.

4.3.7.4 Close the *Comparative quantification settings* dialog box with **<*OK*>** or **<*Cancel*>**.

4.3.8 Tools to display selective band classes

4.3.8.1 If present, remove the existing band matching analysis by selecting **RFLP1** in the *Experiments* panel and ***Bandmatching > Perform band matching*** or  . The program will ask for confirmation. Press **<*OK*>**.

4.3.8.2 Clear any selection made by pressing F4, and manually select some entries in the comparison (see 2.2.8 for manual selection functions).

4.3.8.3 Select ***Bandmatching > Perform band matching*** or press  to create a new band matching.

4.3.8.4 In the *Perform band matching* dialog box (see Figure 4-30), check the option ***Find classes on selected entries only***.

With this option, the program will only create band classes for bands found on the entries in the selection.

4.3.8.5 With ***Bandmatching > Auto assign all bands to all classes***, you can let the program assign the bands of the non-selected entries to the corresponding band classes.

FPQuest 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 (see 4.3.3 for manual editing of band classes).

4.3.8.6 Clear any list of selected patterns with F4.

4.3.8.7 First, remove the existing band matching analysis by selecting **RFLP1** in the *Experiments* panel and ***Bandmatching > Perform band matching*** or  .

4.3.8.8 Select ***Bandmatching > Perform band matching*** or press  to create a new band matching, including all band classes.

4.3.8.9 Select two entries having a few different bands.

4.3.8.10 Select ***Bandmatching > Polymorphic bands only (for selection list).*** Only the band classes that are polymorphic between the selected two patterns are now displayed.

4.3.9 Creating a band matching table for polymorphism analysis

Before a presence/absence table as shown in Figure 4-29 can be displayed in FPQuest, 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 (see Section 3.2). Such a character table is necessary to convert the band classes and represent them as presence/absence tables.

4.3.9.1 If not already available, define a composite data set for the two RFLP techniques (**RFLP-combined**) as described in 3.2.2.

When a comparison is opened after the composite data set **RFLP-combined** had been defined, **RFLP-combined** is listed in the *Experiments* panel of the *Comparison* window. Since we defined **RFLP1** and **RFLP2** as being the experiment types used in this composite data set, the band matching values for **RFLP1** as calculated in the previous paragraphs (4.3.2 to 4.3.7) are automatically filled in as character values.

4.3.9.2 In the *Experiments* panel, with the band matching for **RFLP1** shown, press the  button of **RFLP1-table**. The binary band matching table appears as in Figure 4-40.

4.3.9.3 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 4-40) downwards.

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 (*).

4.3.9.4 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 tab-delineated table looks as shown in Figure 4-41 and is in fact very similar to the one obtained via the command ***Bandmatching > Export band matching***.

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.

4.3.9.5 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) (see Figure 4-42). 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 3.1.10.2) or in the *Comparison* window (see 4.3.7.3).

```
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
```

HEADER:
Band classes

```
0.00  0.00 25.27  3.48  2.99  8.95  0.00  0.00  8.31  0.00 13.31  0.00
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
```

TABLE:
Rows=entries

Figure 4-43. Numerical band matching character table exported from FPQuest (space-delineated).

	RFLP1:167.62	RFLP1:164.42	RFLP1:158.63	RFLP1:151.02
G@Gel07@004	0	1	0	0
G@Gel11@005	0	1	1	0
G@Gel07@017	0	1	0	0
G@Gel11@006	0	1	1	0
G@Gel11@011	1	1	0	0
G@Gel08@016	0	1	1	0
G@Gel07@015	0	1	0	0
G@Gel07@010	0	1	1	0
G@Gel08@003	0	0	1	0
G@Gel08@006	0	0	1	0
G@Gel08@015	0	0	1	0

Figure 4-41. Binary band matching character table exported from FPQuest (tab-delineated).

4.3.9.6 Make sure that **RFLP-combined** is selected in the *Comparison* window. Select ***Composite > Show quantification (values)*** to display the numerical intensities of the bands.

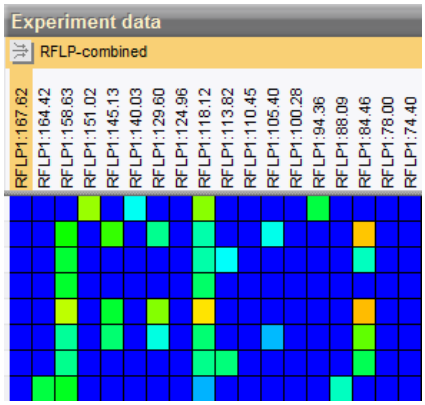


Figure 4-42. Intensity of bands shown in color.

4.3.9.7 With ***Composite > Export character table***, a numerical band matching table is created in text format, separated by tabs or spaces (Figure 4-43).

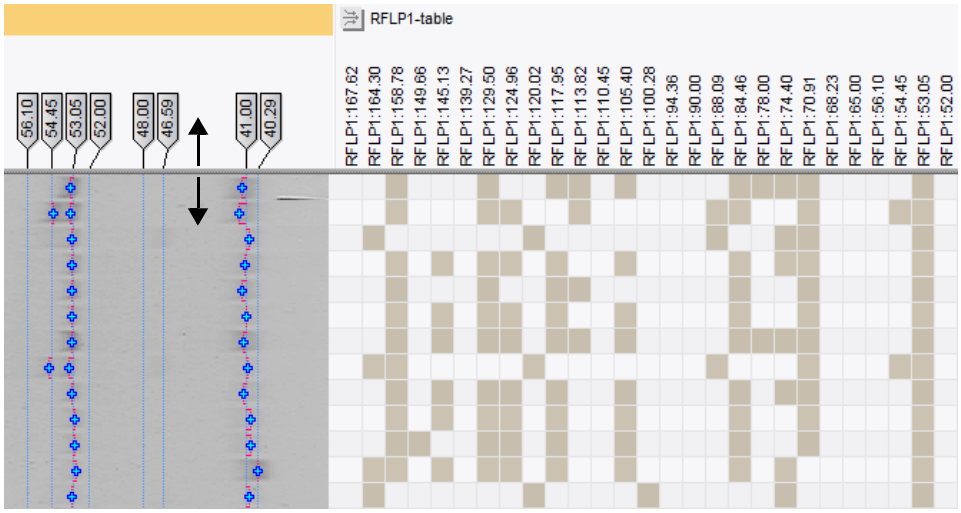


Figure 4-40. Binary band matching table; detail.

4.3.10 Finding discriminative bands between entries

The use of a composite data set allows discriminative bands to be searched for in a band matching table.

4.3.10.1 In database **DemoBase**, have a *Comparison* window open with all non-"STANDARD" entries selected (e.g. comparison **All**, see 4.1.9) and the composite data set **RFLP-combined** shown (see 4.3.9).

4.3.10.2 Make sure that the image of the composite data set is shown, by pressing the button of **RFLP-combined** in the *Experiments* panel.

4.3.10.3 Minimize or reduce the *Comparison* window so that the *FPQuest main* window (at least the menu and toolbar) becomes visible.

4.3.10.4 Press F4 to make sure that no entries are selected.

4.3.10.5 In the *FPQuest main* window, select ***Edit > Search entries*** (F3), enter Vercingetorix in the **Genus** field and press ***<Search>***.

All *Vercingetorix* entries are selected in the *Database entries* panel of the *FPQuest main* window and in the *Information fields* panel of the *Comparison* window (see 2.2.9 and 2.2.10 for more information about the automatic search and selection functions).

4.3.10.6 To group the selected entries, choose ***Edit > Bring selected entries to top*** in the *Comparison* window or press CTRL+T on the keyboard.

4.3.10.7 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 4-44).

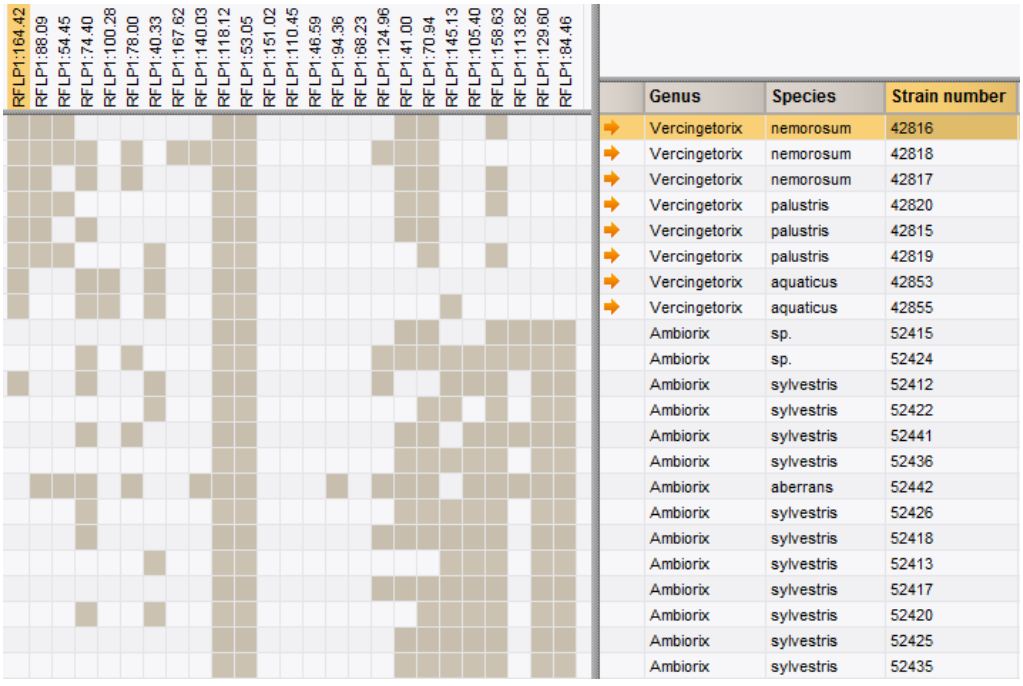
In a composite data set it is possible to list the entries according to the intensity of a selected band. This feature allows for a particular band the entries to be found for which the band is present or not.

4.3.10.8 Show the band table as intensity table with ***Composite > Show quantification (colors)***.

4.3.10.9 Click on a band class in the band classes header (Figure 4-44) and select ***Composite > Sort by character***.

The entries are now sorted by increasing intensity of the selected band class.

Furthermore, it is possible to perform a transversal (or two-way) clustering of a band matching table. See 4.4.3 for a detailed description of the transversal clustering of composite data sets.



Genus	Species	Strain number
Vercingetorix	nemorosum	42816
Vercingetorix	nemorosum	42818
Vercingetorix	nemorosum	42817
Vercingetorix	palustris	42820
Vercingetorix	palustris	42815
Vercingetorix	palustris	42819
Vercingetorix	aquaticus	42853
Vercingetorix	aquaticus	42855
Ambiorix	sp.	52415
Ambiorix	sp.	52424
Ambiorix	sylvestris	52412
Ambiorix	sylvestris	52422
Ambiorix	sylvestris	52441
Ambiorix	sylvestris	52436
Ambiorix	aberrans	52442
Ambiorix	sylvestris	52426
Ambiorix	sylvestris	52418
Ambiorix	sylvestris	52413
Ambiorix	sylvestris	52417
Ambiorix	sylvestris	52420
Ambiorix	sylvestris	52425
Ambiorix	sylvestris	52435

Figure 4-44. Discriminative bands for selected entries, positive discrimination left, negative discrimination right.

4.4 Cluster analysis of composite data sets

4.4.1 Principles

A clustering based upon a similarity matrix can be performed on an individual experiment type or on a combination of experiment types. The methods that FPQuest uses to arrive at dendrograms representing combined techniques are represented schematically in Figure 4-45.

- 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.

Both steps 3 and 4 require a *composite data set* to be generated.

4.4.2 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

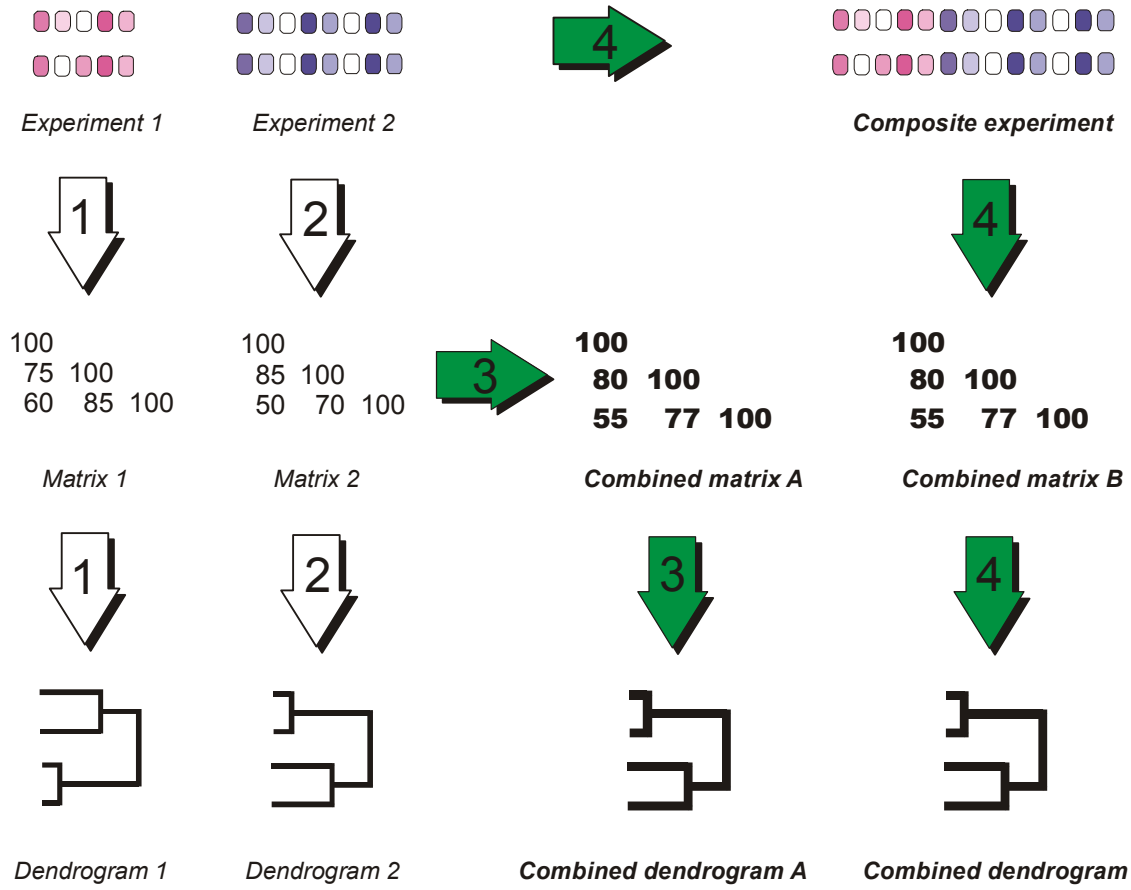


Figure 4-45. Scheme of possibilities in FPQuest to obtain combined dendrograms from multiple experiments.

of a composite data set and its functions is described in Section 3.2, and if you have gone through that paragraph, a composite data set **RFLP-combined** should be available in the **DemoBase** database, including both RFLP techniques.

4.4.2.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison containing all entries except the STANDARDs (see 4.1.9).

4.4.2.2 Select **RFLP-combined** in the *Experiments* panel and show the character image by pressing the button of **RFLP-combined**.

> *NOTE: The order in which experiment data are displayed in a composite data set is the same as the experiment order in the* Experiments *panel of the* FPQuest main *window (see 1.6.2): re-sorting the* Experiments *panel will result in an updated display order of the experiment data in all composite data sets when the comparison is opened again.*

4.4.2.3 Right-click on the image and select ***Show quantification (colors)*** from the floating menu.

4.4.2.4 Select ***Clustering > Calculate > Cluster analysis (similarity matrix)***.

The *Composite data set comparison* dialog box (Figure 4-46) allows you to choose between step 3 (averaging the matrices of the experiments) and step 4 (merging the experiments to a composite experiment) of Figure 4-45. With the ***Similarity*** option ***Average from experiments***, the matrices from the individual experiments are averaged according to the defined weights (step 3 in Figure 4-45). With one of the coefficients under ***Binary coefficient***, ***Numerical coefficient*** (non-binary coefficients), or ***Multi-state coefficient***, step 4 in Figure 4-45 will be followed using a composite band matching table. For a description of the coefficients, see 4.2.1.

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. PCR-DGGE, the fingerprint pattern may consist of a few very bright bands and a number of minor bands. It is likely that the bright bands will account for most of the discrimination between the samples studied, whereas the minor bands, which may be equally valuable for typing purposes, are masked. (2) When creating band matching tables from different fingerprint types, the optical densities of the fingerprints 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. For example, in the **DemoBase** database, this situation would occur when creating a composite data set for **AFLP** (12-bit optical density) and one of the RFLP experiments (8-bit optical density). The feature ***Standardized characters*** standardizes each character by subtracting its mean value and dividing by its standard deviation. The

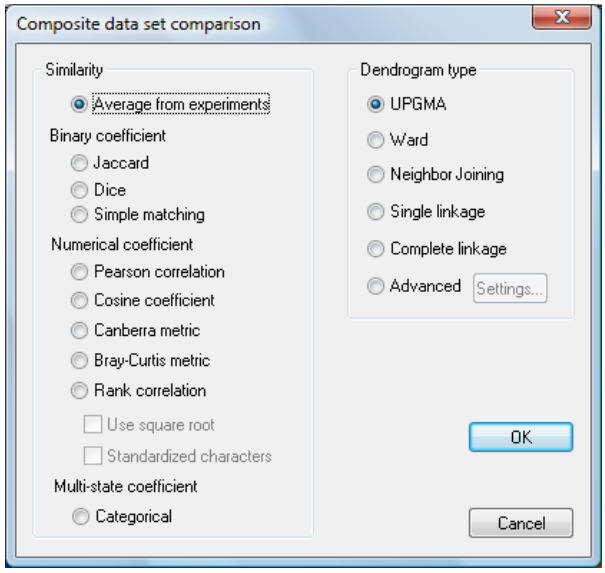


Figure 4-46. ***Composite data set comparison*** **dialog box.**

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 ***Pearson correlation*** and ***Cosine coefficient*** (or ***Euclidean distance*** in case of non-composite data sets).

The ***Rank correlation*** coefficient first transforms an array of characters into an array of ranks according to the magnitude of the character values. The rank arrays are then compared using the Pearson product-moment correlation coefficient. The ***Rank correlation*** is known to be a very robust coefficient, but with low sensitivity.

4.4.2.5 Select ***Pearson correlation*** with ***Standardized characters*** and ***UPGMA*** as clustering method.

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***.

> *NOTE: It can be proven that in case of binary data sets the option* **Average from experiments** *offers exactly the same results when* **Correct for internal weights** *is enabled in the* Composite data set settings *(see 3.2.2.4).*

In the case however, that the *ranges* of the combined experiments are different, e.g. a range between 0 and 255 (8-bit) for one experiment and between 0 and 65,535 (16-bit) 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 (***Average 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 in Section 3.2).

The ***Categorical*** coefficient can be chosen in case all the characters of the individual experiment types are *multi-state* 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.

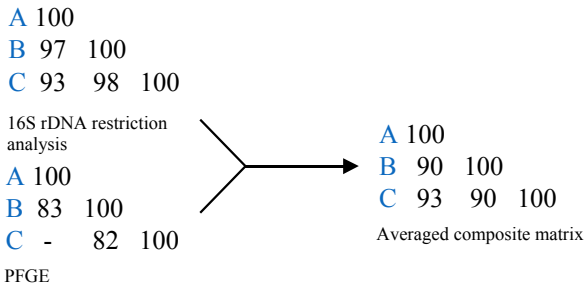
4.4.2.6 Select ***Average from experiments*** and press **<*OK*>** to calculate the cluster analysis.

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 3.2.2.4). This option proportionally assigns more weight to the RFLP technique with the highest number of bands. However, since **RFLP1** contains a comparable number of bands as **RFLP2**, the effect on the similarity values is only marginally.

> *NOTE: If you want to see the difference when* **Correct for internal weights** *is* not *enabled, save and close the* Comparison *window, open* ***RFLP-combined*** *in the* Experiments *panel, and uncheck* **Experiment > Correct for internal weights**. *Open the comparison again and select* **Calculate cluster analysis** *again for* ***RFLP-combined***.

It is obvious that the possibility of approach 4 described in 4.4.2, i.e. merging two character sets into 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 4.4.1). 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. PFGE restriction fragment similarity on the one hand and 16S rDNA restriction fragment similarity on the other hand. In many cases, PFGE similarity values will range from 100% to 40% or less, whereas 16S rDNA restriction fragment similarity will range between 100% and 90% or even higher. It is clear that the small but very significant similarity differences in 16S rDNA restriction fragment similarity will be masked by the much larger differences (including experimental error) of PFGE similarity, 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 FPQuest, A, B, and C. Consider the following matrices for these three entries, generated from 16S rDNA restriction analysis and PFGE. One of the matrices, for example the PFGE matrix, is incomplete.



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 restriction analysis, 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 FPQuest, 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 FPQuest as follows:

4.4.2.7 If not existing yet, create a new composite data set **All-Exp**, in which you add all experiments available in **DemoBase**.

4.4.2.8 Open a the comparison **All** or create a comparison containing all but the STANDARD lanes, and calculate a matrix (***Calculate cluster analysis***) for each experiment.

4.4.2.9 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.

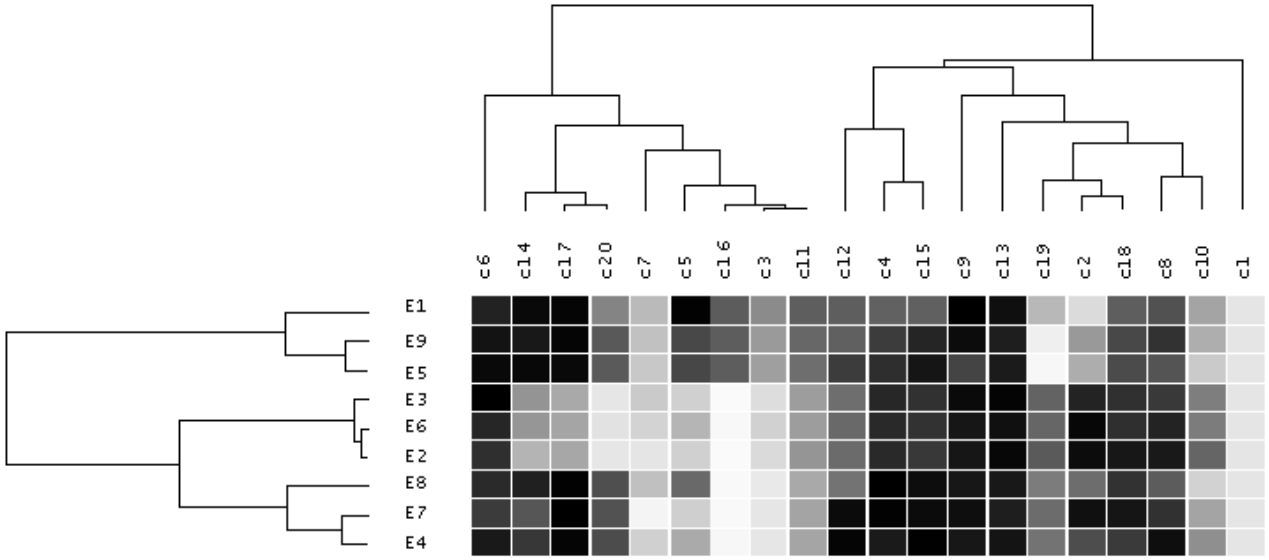


Figure 4-48. Transversal clustering of entries (horizontal) and characters (vertical).

NOTE: The feature to correct for internal weights (3.2.2.4) does not apply to a consensus matrix.

4.4.3 Transversal clustering

The input for a cluster analysis in a composite data set is a *data matrix*. A data matrix of *n* entries having *p* characters looks like in Figure 4-47: the entries are presented as *rows* and the characters as *columns*. In FPQuest, the data matrix should not necessarily be complete: some missing character values are allowed, for example if test results are ambiguous or not available.

	Char 1	**Char 2**	**…**	**Char *p***
Entry 1	Val 11	Val 12	…	Val 1*p*
Entry 2	Val 21	Val 22	…	Val 2*p*
…	…	…	…	…
Entry *n*	Val *n*1	Val *n*2	…	Val *np*

Figure 4-47. Data matrix of *n* entries and *p* characters.

A simple and efficient way to visualize associated groups of characters (columns) with groups of entries (rows) in a data matrix is to construct a two-way clustering of the data matrix, i.e. in which the entries are clustered by means of their character values (the conventional clustering as described in 4.1.8; also called *Q*-clustering), and the characters are clustered by means of their values per entry (*R*-clustering).

The result is a data matrix in which both the entries and the characters are ordered according to their relatedness (Figure 4-48), which we will call *transversal clustering*. This representation makes it easy to visually associate clusters of characters with clusters of entries. For example, the first group of entries (E1, E9, and E5) is separated from the others by a cluster of characters (C5, C16, C3, and C11) which are all more positive in the first cluster than in the other clusters. Another group of three characters (C14, C17, and C20) separates the second group of entries (E3, E6, and E2) from the other clusters because they are less positive.

In FPQuest, it is possible to calculate a transversal clustering from a composite data set. As an example, we use the composite data set **RFLP-combined** in **DemoBase** including **RFLP1** and **RFLP2** as described in 4.4.2.

4.4.3.1 Create a *Comparison* window with a selection of entries and select the composite data set in the *Experiments* panel. You can show the character image by pressing the button of **RFLP-combined**.

4.4.3.2 Calculate a cluster analysis of the entries as described in 4.1.9.

4.4.3.3 Choose ***Composite > Calculate clustering of characters*** or click the button. A dialog box offers a choice between the ***Pearson correlation*** for numerical characters, the ***Jaccard***, ***Dice***, and ***Simple Matching*** coefficients for binary data, and the ***Categorical*** coefficient for multi-state or categorical characters. For a description of the coefficients, see 4.2.1.

4.4.3.4 Select ***Pearson correlation*** and press **<*OK*>** to calculate a character dendrogram, which appears horizontally in the caption of the data matrix display of the composite data set.

4.4.3.5 It may be useful to drag the separator bar between the image panel and its caption down to obtain more space for the character dendrogram and the character names.

4.5 Phylogenetic clustering methods CL

4.5.1 Introduction

In addition to the *Neighbor Joining* method, which we described previously (see 4.1.11), FPQuest offers an alternative phylogenetic clustering method, based on the concept of maximizing *parsimony*. Usually, this clustering method will be applied to sequence data. However, maximum parsimony can be applied to any data set that can be presented as a *binary* or *categorical data matrix*. As such it can be applied to fingerprint type data on condition that a band matching is performed (see Section 4.3).

4.5.2 Maximum parsimony clustering

Since maximum parsimony requires a binary or categorical data matrix as input, it can only be applied to fingerprint type data for which a band matching is performed (see Section 4.3). The program will use the *binary* band presence table associated with the band matching as input for maximum parsimony. We will describe maximum parsimony clustering with the fingerprint type **AFLP** of the **DemoBase** database.

4.5.2.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.5.2.2 Click on the fingerprint type **AFLP** in the *Experiments* panel. If not already present, calculate a band matching for **AFLP** as explained in 4.3.2.

4.5.2.3 To calculate a maximum parsimony dendrogram, select ***Clustering > Calculate > Maximum parsimony tree (evolutionary modelling)***. Alternatively, you can press the  button, in which case the floating menu as shown in Figure 4-4 pops up. Select ***Calculate maximum parsimony tree*** from the floating menu.

The *Maximum parsimony cluster analysis* dialog box for character data appears (Figure 4-49).

4.5.2.4 Under ***Data set***, you can specify how to treat the data, i.e. ***Convert to binary*** or ***Treat as categorical***. In case of fingerprint type data, these options are redundant.

4.5.2.5 FPQuest uses methods that are described in the literature to optimize the topology of parsimonious trees. An alternative method, which sometimes finds even more parsimonious trees, but which is considerably slower, is the mathematical principle of ***Simulated annealing***.

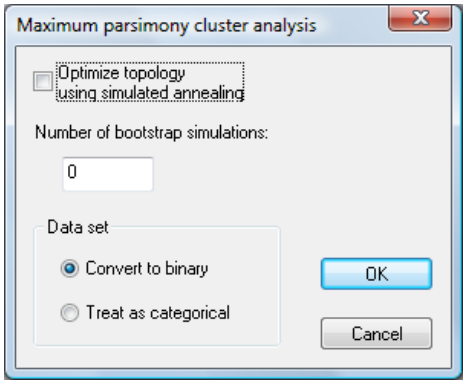


Figure 4-49. The *Maximum parsimony cluster analysis* dialog box for character type data.

4.5.2.6 In addition, FPQuest can do a *Bootstrap* analysis on the parsimony clustering, for which you can enter the ***Number of bootstrap simulations***. If zero is entered, no bootstrap values are calculated.

Caution: enabling simulated annealing and at the same time entering a number of bootstrap simulations will increase the computing time dramatically. We do not recommend to combine these options.

4.5.2.7 Press **<*OK*>** to start the calculations.

The result is an *Unrooted dendrogram* window, of which the parsimony is given in the status bar (Figure 4-50). If groups were defined previously (see 4.1.12), the entries are represented in the group colors.

4.5.2.8 To zoom in or out on the tree, use the  and  buttons or ***Layout > Zoom in*** and ***Layout > Zoom out***.

4.5.2.9 You can toggle between the colors and the black-and-white representation mode with ***Layout > Show group colors*** or  . When the group colors are shown, this button is displayed as  .

In black-and-white mode, the groups are represented (and printed) as symbols.

4.5.2.10 The drop-down list 

allows you to select a coloring based on groups or any of

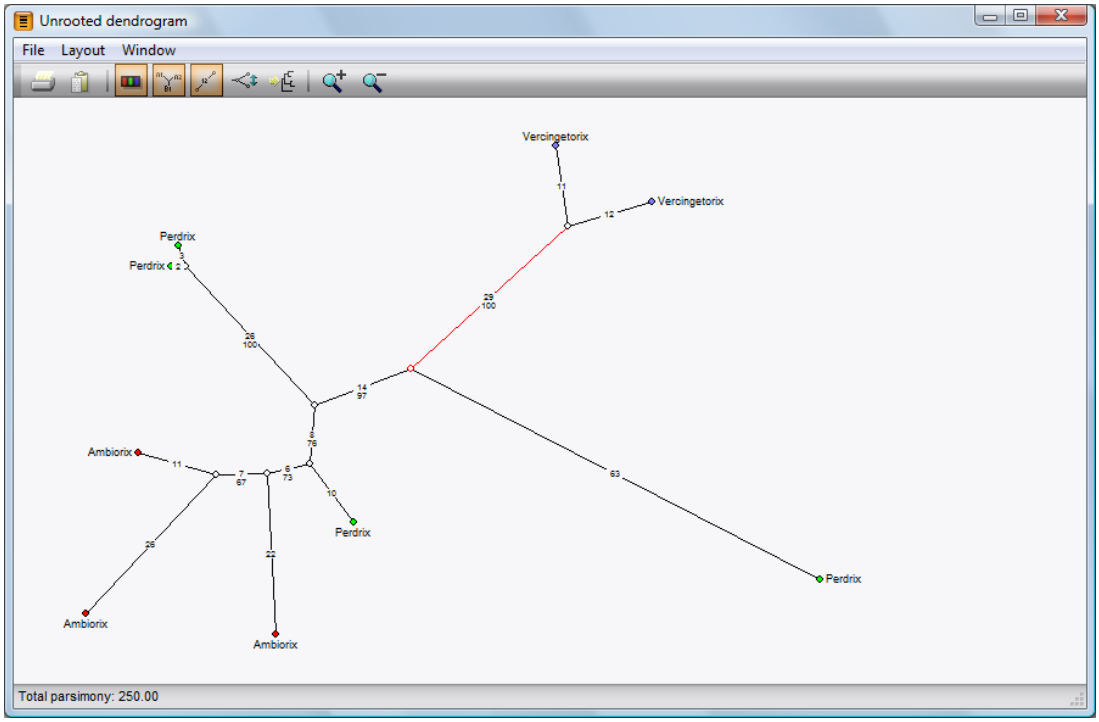


Figure 4-50. Unrooted maximum parsimony tree. Number of mutations are indicated on the branches (top) as well as the bootstrap values (bottom).

the available field states (see 2.2.5 on how to define field states). 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 they 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:

4.5.2.11 In the parent *Comparison* window, select ***Layout > Use group numbers as key***.

4.5.2.12 A legend to the group numbers can be obtained with ***File > Export database fields*** in the parent *Comparison* window.

Alternatively, a selected information field can be displayed instead of the key:

4.5.2.13 In the parent *Comparison* window, click on the information field which you would like to display as key (e.g. 'Genus').

4.5.2.14 Select ***Layout > Use field as key*** from the menu in the *Comparison* window. The 'Genus' field is now displayed in the maximum parsimony tree (see Figure 4-50).

4.5.2.15 With ***Layout > Show branch lengths*** or  , the lengths of the branches, as numbers of base conversions, are shown. In case bootstrap values were calculated, this option displays the bootstrap values as well.

In more complex trees, the spread of the branches may not be optimal. The program can iteratively optimize the spread of the branches:

4.5.2.16 In the *Unrooted dendrogram* window, select ***Layout > Optimize branch spread*** or  .

The user can rotate and swap the branches manually if the tree layout is not satisfactory.

4.5.2.17 Left-click in the proximity of a node or a branch tip.

4.5.2.18 While holding down the mouse button, rotate the branch to the desired position.

4.5.2.19 If you select entries in the parent *Comparison* window or in the *BioNumerics main* window, these entries are shown within a square in the *Unrooted dendrogram* window.

4.5.2.20 You can also select entries directly in the *Unrooted dendrogram* window, by holding the CTRL key while clicking in the proximity of a node. All entries branching off from this node will be selected.

4.5.2.21 Repeat this action to unselect entries.

4.5.2.22 To copy the unrooted tree to the clipboard, select ***File > Copy image to clipboard*** or  .

4.5.2.23 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:

4.5.2.24 Select a branch by clicking in the proximity of one of the two nodes it connects. The selected branch is red.

4.5.2.25 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 tree.

NOTE: All dendrogram display functions (see 4.1.11) also apply to unrooted trees, except the incremental clustering: one cannot delete or add entries while the tree is automatically updated.

In publications and presentations, particularly in a phylogenetic context, a dendrogram is sometimes represented as a real tree with a stem and branches. Such representations can be achieved from a maximum parsimony tree using the *rendered tree* option. This option should be used with care, as it will only produce acceptable pictures from a limited number of entries and with fairly equidistant members.

4.5.2.26 If you want to create a *rooted* rendered tree from the parsimony tree, you first have to select the branch on the tree from which the root will be constructed. Usually, the longest branch on the tree is taken as root.

4.5.2.27 Create a rendered tree from the *Unrooted parsimony tree* window using ***File > Export rendered tree***.

The functions of the *Rendered tree* window are described in 4.1.17.

4.6 Advanced clustering and consensus trees CL

4.6.1 Introduction

Cluster analysis is one of the most popular ways of revealing and visualizing hierarchical structure in complex data sets. As explained before (4.1.8), cluster analysis is a collective noun for a variety of algorithms that have the common feature of visualizing the hierarchical relatedness between samples by grouping them in a dendrogram or tree. The most universally applied methods are pairwise clustering algorithms that use a distance or similarity matrix as input (Figure 4-51). UPGMA (Unweighted Pair Group Method using Arithmetic Averages), Complete Linkage, Single Linkage, and Ward's method are examples of such methods. The advantage of these methods is that they can be applied to any type of data, as long as there exists a suitable similarity or distance coefficient that can generate a similarity (distance) matrix from the data. As such, similarity-based clustering can be applied to incomplete data sets or data that is not presented in the form of a data matrix (e.g., electrophoresis band sizes).

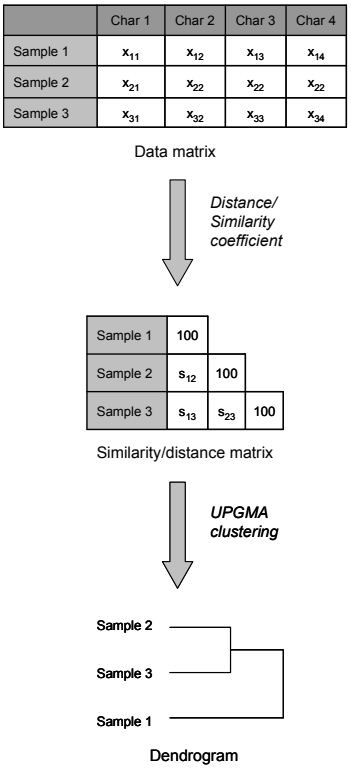


Figure 4-51. Steps in similarity based cluster analysis.

In the analysis steps outlined in Figure 4-51, one should consider the matrix of pairwise similarities (or distances) as the complete comparative information between all the samples analyzed. Obviously, for larger numbers of samples, interpreting a similarity matrix becomes hardly simpler than looking at the original data. This is why a similarity matrix is not usually calculated as a final result, but as an intermediate step for grouping algorithms such as cluster analysis or multi-dimensional scaling.

The real simplification of the data is obtained by cluster analysis. Both the power and the weakness of a dendrogram lie in its ability to present an easy to interpret, well-structured, hierarchical grouping of the samples. Indeed, simplification means loss of information, and there is no way to present the data in a simple and easily interpretable way, yet holding all the information. As a consequence, every dendrogram resulting from a non-artificial data set will contain errors, the amount of error being proportional to the complexity of the similarity matrix. A second source of error results from the fact that hierarchical clustering always imposes hierarchical structure, even if the data does not support it. The fact that even a perfectly random data set results in a dendrogram with branches, is a clear example of the danger that hierarchical clustering holds. Various statistical methods allow the error associated with dendrogram branches or their uncertainty to be estimated, e.g., standard deviation values and the cophenetic correlation (see 4.1.13). Other methods, such as bootstrap, allow the probability of dendrogram branches, as a result of the data set, to be indicated.

4.6.2 Degeneracy of dendrograms

Another problem with pairwise hierarchical clustering methods such as UPGMA is the degeneracy of the solution. Whereas UPGMA results in just one tree, in many cases there exist a number of equally good alternative solutions. Such degeneracies are very likely to occur in cases where the similarity matrix contains multiple identical values. In practice, binary and categorical data sets and banding patterns treated as absent/present states result in frequent occurrence of identical similarity values, whereas quantitative measurements registered as decimal numbers almost never yield identical similarity values. To understand how the occurrence of identical similarity values can result in multiple possible trees, we consider the example of three banding patterns (Figure 4-52). As can be seen from this simple example, s[A,B] and s[B,C] are both 0.75, whereas s[A,C] is 0.50. The way how UPGMA constructs a dendrogram is by first searching for the highest similarity value in the

matrix, and linking the two samples from which it results. In the present example, [A,B] and [B,C] are equivalent solutions, two partial dendrograms can be constructed: one with [A,B] linked at 75% (solution 1) and the other with [B,C] linked at 75% (solution 2). In the next step of UPGMA, the remaining sample is linked at the average of its similarity with the samples already grouped. In solution 1, this leads to C being linked at 62.5% to [A,B], whereas in solution 2, A is being linked at 62.5% to [B,C]. Both dendrograms suggest a quite different hierarchical relatedness but actually none of them truly reflects the relationships suggested by the data set and the similarity matrix.

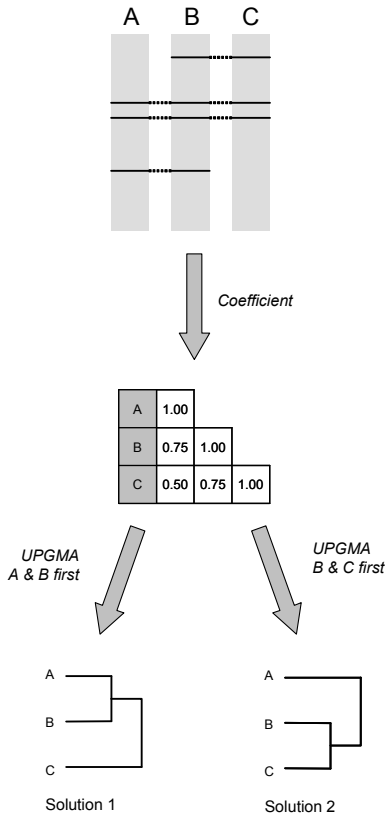


Figure 4-52. A scenario of three banding patterns resulting in two possible UPGMA solutions.

Another inconsistency in pairwise clustering results from the inability to deal with infringements upon the transitivity rule of identity. When sample A is identical to sample B, and sample B is identical to sample C, the transitivity rule predicts that A will be identical to C as well. Infringements upon this rule are particularly found in the comparison of banding patterns, where the identity of bands is judged based upon their distance, using a position tolerance value that specifies a maximum distance between bands to be considered identical. The example below (Figure 4-53) illustrates the result of a UPGMA clustering of three banding patterns for which one band is slightly shifted. With a position tolerance as indicated on the figure, the pairs of patterns [A,B] and [A,C] will have a 100% score, whereas [A,C] will have only 75% similarity as the distance between their lower bands is greater than the position tolerance specified. Similarly as explained above, the UPGMA algorithm has two choices to perform the first linkage, and the results are displayed as solution 1 and solution 2. Neither of the two dendrograms reflects the discrepancy indicated by the similarity values, but instead, each dendrogram falsely suggests a hierarchical structure that is not supported by the data.

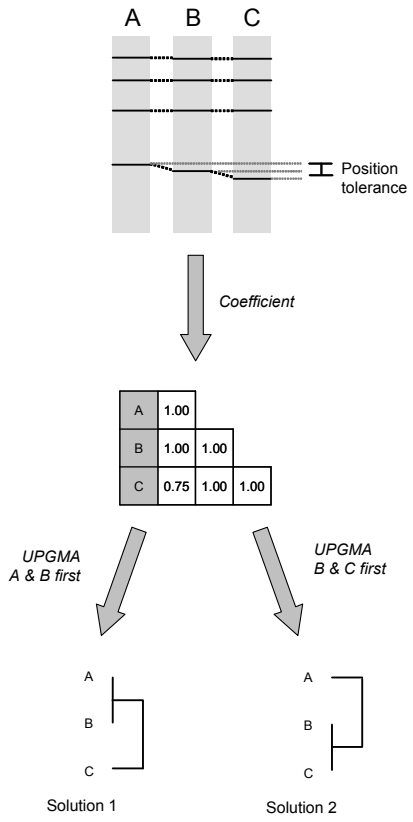


Figure 4-53. Infringement upon the transitivity rule for sample identity and resulting dendrograms.

4.6.3 Consensus trees

A more truthful representation of the relationships given in Figure 4-52 and Figure 4-53 can only be obtained by respecting the indeterminacy resulting from the identical similarity values. Using the conventional pairwise linkage dendrogram representation, this cannot be achieved, and therefore, a new dendrogram type has been introduced in FPQuest, allowing more than two entries or branches to be linked together. The resulting tree can be called a *consensus tree* because it allows all entries that are part of a degeneracy to be linked at one similarity level in a single consensus branch (Figure 4-54). To obtain such a consensus representation of the different trees possible, FPQuest will first calculate all possible solutions and draw a consensus tree that uses pairwise linkage as the primary criterion, but applies multi linkage in those cases where branches or entries are degenerated.

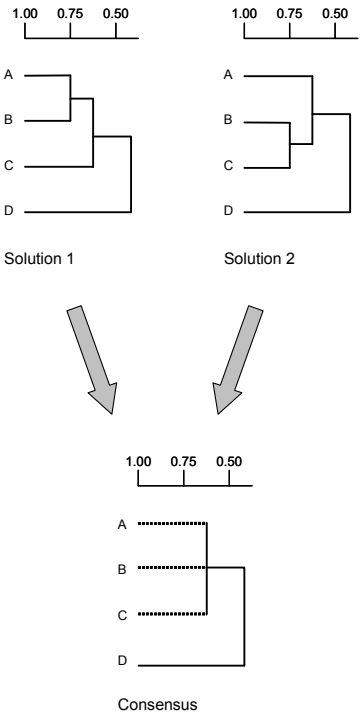


Figure 4-54. Displaying different UPGMA solutions as a consensus branch.

Another advantage of the presentation method that supports multi linkage of entries or branches is that it can be used to calculate consensus trees from trees generated from different data sets as well. The same algorithms can be applied to compare the different and common branches on the trees, and the example shown in Figure 4-54 could as well be a case where Solution 1 and Solution 2 result from different data sets.

4.6.4 Advanced clustering tools

The advanced clustering tools in FPQuest offer some additional functionality compared to the standard clustering tools in the *Comparison* window. This functionality is related to the possibility of linking more than two entries or branches together, as shown in Figure 4-54. As such it becomes possible to display multiple solutions of a cluster analysis in a consensus representation, as well as representing two trees from different data sets in one consensus tree. In addition, each tree obtained using the advanced clustering tools is automatically saved, which makes it possible to have more than one stored tree per experiment type. This feature is useful if one wants to compare trees generated using different similarity coefficients or using varying parameters such as position tolerance for banding patterns.

4.6.5 Displaying the degeneracy of a tree

In FPQuest, select a data type that can potentially result in multiple tree solutions, for example, the fingerprint type **RFLP1** in **DemoBase**.

4.6.5.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.6.5.2 Select the fingerprint type **RFLP1** from the *Experiments* panel in the *Comparison* window and choose ***Clustering > Calculate > Cluster analysis (similarity matrix)***. This pops up the *Comparison settings* dialog box (Figure 4-22), which shows five clustering options (UPGMA, Ward, Neighbor Joining, Single Linkage and Complete Linkage) and an option ***Advanced***.

4.6.5.3 If the option ***Advanced*** is checked, a button ***<Settings>*** becomes available, which will open the *Advanced cluster analysis* dialog box (Figure 4-55).

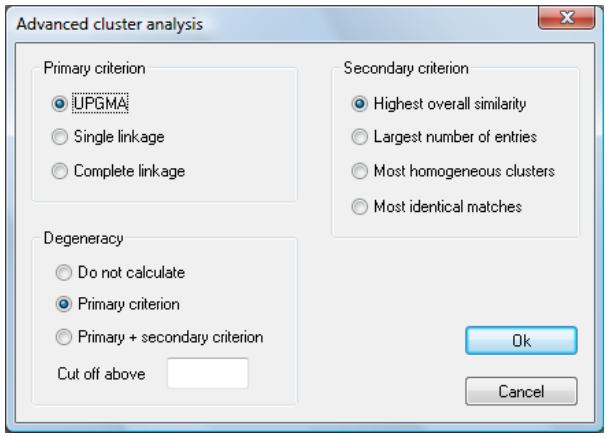


Figure 4-55. The *Advanced cluster analysis* dialog box.

Under *Primary criterion*, the criterion for clustering can be chosen, which can be ***UPGMA***, ***Single Linkage*** or ***Complete Linkage***. All three methods are pairwise clustering algorithms, i.e. which will construct dendrograms by grouping branches and/or entries pair by pair, using the highest similarity as criterion. In ***UPGMA*** the similarity between clusters is calculated as the average of all individual similarities between the clusters, whereas in ***Single Linkage*** it is the highest similarity found between the clusters. In ***Complete Linkage***, it is the lowest similarity found between the clusters.

The *Secondary criterion* applies to those cases where two clusters have the same (highest) similarity with a third, in which case two different tree solutions exist. The program will then apply one of the following criteria to solve the indeterminacy left by the standard clustering algorithm (i.e., the primary criterion):

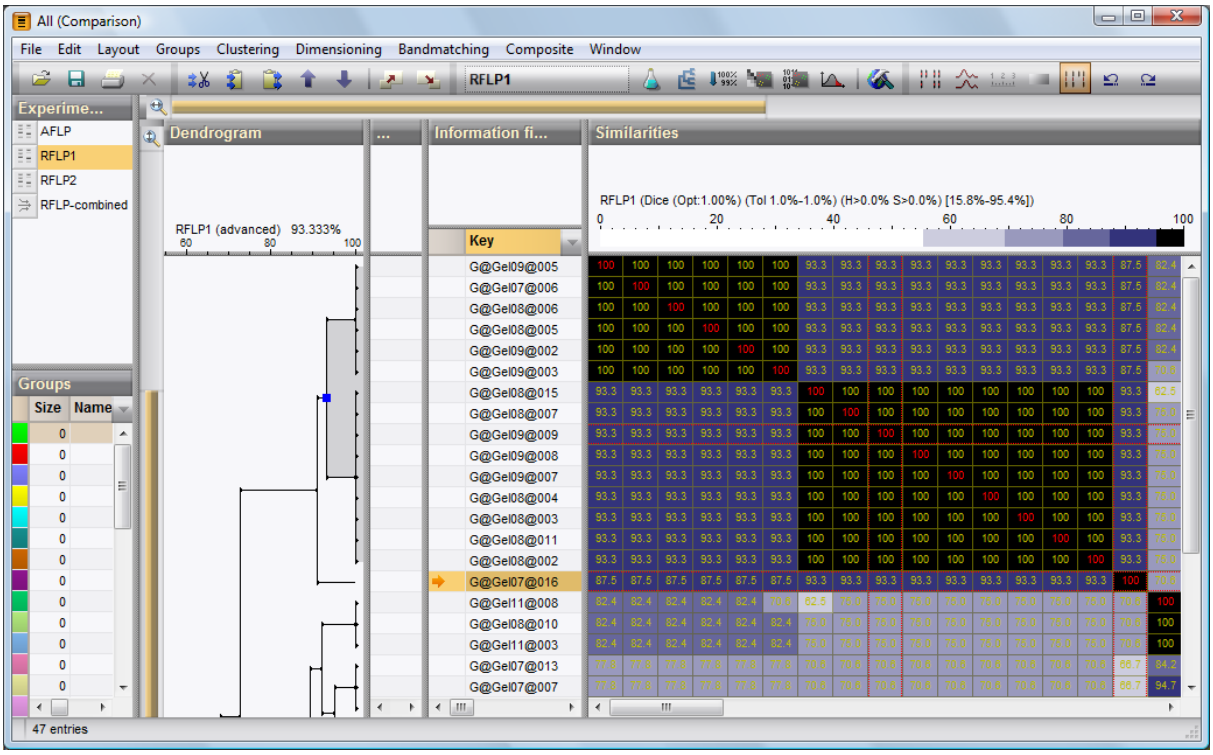


Figure 4-56. Advanced tree representation with a highlighted cluster, indication of the number of degenerated entries relative to the cluster, and the degenerated entry selected.

(1) ***Highest overall similarity***: the two clusters will be joined that result in the cluster with the highest overall similarity with all other members of the comparison.

(2) ***Largest number of entries***: the two clusters will be joined that result in the cluster with the largest number of entries.

(3) ***Most homogeneous clusters***: the two clusters will be joined that result in a cluster that has the highest internal homogeneity.

Note that criteria (1) and (3) are complementary to each other as (1) will only consider the external similarity values of the resulting clusters whereas (3) will only consider their internal similarity values.

Under *Degeneracy*, three options allow one to deal with degenerated trees:

(1) ***Do not calculate*** will not look for degeneracies and will display just one solution. The differences with a conventional cluster analysis are that (i) the solution presented is the best according to the secondary criterion specified, and (ii) the resulting tree is saved automatically as an *advanced tree* and can be used together with other advanced trees to calculate a *Consensus Tree*.

(2) The option ***Primary criterion*** will calculate all degeneracies resulting from the primary criterion only and will not consider any secondary criterion specified.

(3) ***Primary + secondary criterion*** will use the specified secondary criterion to solve the degeneracies resulting from the primary criterion and will only display the degeneracies that remain after the secondary criterion. It is very unlikely that there will remain any degeneracies with this option checked.

The ***Cut off above*** parameter specifies the maximum allowed number of degenerate entries relative to a cluster. A *degenerate entry* is an entry that does not belong to a given cluster in the present tree, but that does belong to the cluster in at least one alternative solution. If zero is entered as cutoff value, no degenerate entries are allowed and as a consequence, a consensus tree is generated that includes all possible solutions. If the field is left blank, the degeneracy of the tree will not be reduced at all. If a number is entered, for example 2, all clusters for which there are more than 2 degenerate entries will be displayed as consensus clusters with the degenerate entries included.

Each cluster that has degenerate entries relative to it, will have an indication of the number of degenerate entries (see Figure 4-56, which shows one degenerated entry for the selected cluster).

4.6.5.4 When a cluster is selected by clicking on its branching node, the cluster is filled in gray (Figure 4-56), which makes it easier to see which entries belong to it.

4.6.5.5 If there are degenerated entries relative to the highlighted cluster, you can find them by choosing ***Clustering > Advanced trees > Select degenerate entries***. All degenerate entries relative to the cluster are now added to the selection.

The interpretation of degeneracies and tracking back their reason is sometimes difficult. The larger the tree and the deeper the branch, the more complex the degeneracies will be. The example screen in Figure 4-56 is a capture taken from experiment **RFLP1** in the **DemoBase**. The highlighted cluster has one degenerated entry, which is selected. The cluster consists of two subclusters which have an overall average similarity of 93.3%. The single degenerate entry, however, also has an average similarity of 93.3% with the second subcluster. The present solution has first linked subcluster 1 to subcluster 2 and then linked the single entry to the merged cluster. According to the criterion of UPGMA, however, an equivalent solution would be to first link the single entry to subcluster 2 and then link subcluster 1 to this new cluster. When the same clustering is done with zero as cutoff value, the cluster looks like in Figure 4-57. Note that the three subclusters are now linked together at the same level. The clusters that connect always at the displayed similarity level in the solution obtained using the secondary criterion are represented by solid lines (in the present case, the single entry), whereas subclusters that cluster at higher levels using the secondary criterion are connected by an interrupted line.

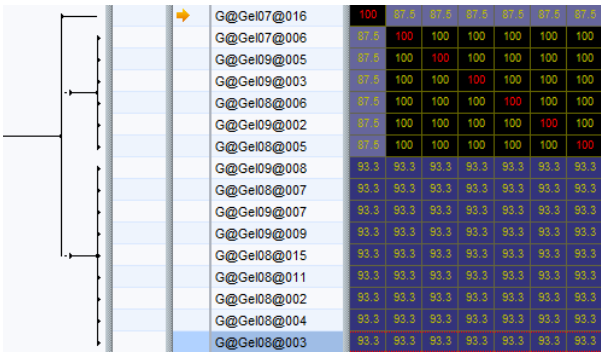


Figure 4-57. Detail of cluster highlighted in Figure 4-56, calculated with a cut off value of zero.

4.6.6 Creating consensus trees

The advanced clustering tool allows a *consensus tree* to be calculated from two or more individual dendrograms. These trees can be conventional clusterings or advanced trees, and can be generated from the same experiment type or from different experiment types. In case you want to calculate different dendrograms from the same experiment type, you should use the Advanced Clustering tools. To create a consensus tree, the program will look for all branches that hold exactly the same entries in both trees and represent them as branches in the consensus tree.

4.6.6.1 As an example, we can calculate two dendrograms in **DemoBase**: one from experiment **RFLP1** using Dice and the other from experiment **RFLP2** using the Pearson correlation. You can calculate the trees using the conventional clustering tools or using the Advanced Clustering tools.

4.6.6.2 Select ***Clustering > Advanced trees > Create consensus tree***, which pops up a dialog box listing the *Stored trees* (Figure 4-58).

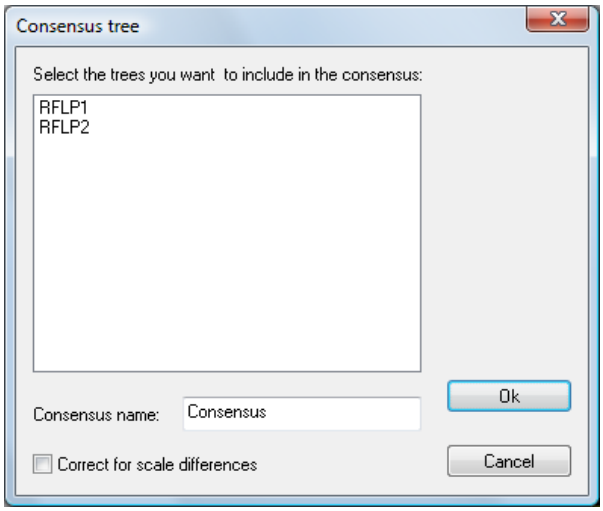


Figure 4-58. *Stored trees* dialog box to calculate a consensus tree.

4.6.6.3 Select the two calculated trees, which have the name of the experiment types they were derived from, and enter a name for the consensus tree to be generated (the default name is "Consensus"). With the option ***Correct for scale differences***, the dendrograms will first be rescaled so that they have the same similarity ranges. The result is that dendrograms covering a narrow similarity range will have more impact on the consensus tree when this option is checked.

After clicking ***<OK>***, the consensus tree is calculated, and only the clusters that contain exactly the same entries in both dendrograms are displayed.

4.6.7 Managing advanced trees

Advanced trees exist as long as a *Comparison* window is opened. Unlike conventional trees however, they are not stored along with a comparison and will disappear after the *Comparison* window is closed.

4.6.7.1 An advanced tree can be displayed by selecting it from the list that appears in ***Clustering > Advanced trees***. The currently displayed tree is flagged in the menu. The currently displayed tree can be deleted with ***Clustering > Advanced trees > Delete current***.

A number of dendrogram editing functions under the ***Clustering*** menu are not applicable to advanced trees.

4.7 Minimum spanning trees for population modelling CL

4.7.1 Introduction

Minimum spanning trees (MSTs) are known for a long time in the context of mathematical topology. When a set of distances is given between *n* samples, a minimum spanning tree is the tree that connects all samples in such a way that the summed distance of all branches of the tree is minimized.

In a biological context, the MST principle and the maximum parsimony (MP) principle share the idea that evolution should be explained with as little events as possible. There are, however, major differences between MP and MST. The MP method allows the introduction of hypothetical samples, i.e. samples that are not part of the data set. Such hypothetical samples are created to construct the internal branches of the tree, whereas the real samples from the data set occupy the branch tips. The phylogenetic interpretation of the internal branches is that they are supposed to be common ancestors of current samples, which do not exist anymore but which are likely to have existed in the past, under the criterion of parsimony.

The MST principle, in contrast, requires that all samples are present in the data set to construct the tree. Internal branches are also based upon existing samples. This means that, when a MST is calculated for evolutionary studies, there are two important conditions that have to be met: (1) the study must focus on a very short time-frame, assuming that all forms or states are still present, and (2) the sampled data set must be complete enough to enable the method to construct a valid tree, i.e. representing the full biodiversity of forms or states as closely as possible. Through these restricting conditions, the method of MST is only applicable for specific purposes, of which population modelling (micro-evolution) and epidemiology are good examples.

The trees resulting from MP on the one hand, and MST on the other hand, also have a topological difference. The MP method assumes that two (related) samples are evolved from one common ancestor through one or more mutations at either side. This normally results in a bifurcating (dichotomic) tree: the ancestor at the connecting node, and the samples at the tip. A MST chooses the sample with the highest number of related samples as the root node, and derives the other samples from this node. This may result in trees with star-like branches, and allows for a correct classification of population systems that have a strong mutational or recombinational rate, where a large number of single locus variants (SLV) may evolve from one common type[1].

An important restriction is that true MST's, e.g. according to the Prim-Jarnik algorithm can only be calculated from a true distance matrix. A criterion for a true distance matrix is that, given three samples A, B, and C, the distance from A to C should never be longer than the summed distance from A to B and B to C. This restriction implies that MSTs are not compatible with all data types. For example, a distance matrix based upon pairwise compared DNA fragment patterns does not fulfill this criterion, and hence, will not result in a true minimum spanning tree. On the other hand, a distance matrix based upon a global band matching table can be used. In theory, all experiments that produce *categorical* data arrays (i.e. *multistate* character arrays) or *binary* data arrays are suitable for analysis with the MST method. The most typical applications for use with MSTs, however, are categorical Multilocus Sequence Typing (MLST) data used in population genetics and epidemiological studies. MST's can also be very useful for analyzing VNTRs (Variable Number Tandem Repeats) in MLVA studies (Multi-Locus VNTR Analysis).

Notwithstanding the restrictions with respect to the distance matrix, FPQuest allows MSTs to be calculated from any similarity matrix. The result from similarity matrices that are known to produce untrue distance matrices (e.g. binary comparison of banding patterns) should not be regarded as true MSTs but provide interesting trees anyway.

4.7.2 Minimum spanning trees in FPQuest

The MST method usually provides many equivalent solutions for the same problem, i.e. one data set can be clustered in to many MSTs with a different topology but with the same total distance. Therefore, a number of priority rules, with respect to the linkage of types in a tree, have been adopted from the BURST program (see the MLST website http://www.mlst.net or Feil et al., 2003[2]) to reduce the number of possible trees to those that have the most probable evolutionary interpretation. These rules assign priority, in decreasing order, to (1) types that have the highest number of single locus vari-

1. Maynard Smith, J., N.H. Smith, M. O'Rourke, and B.G. Spratt BG. 1993. PNAS 90: 4384-4388.
2. Feil, E.J. J.E. Cooper, H. Grundmann, D.A. Robinson, M.C. Enright, T. Berendt, S.J. Peacock, J. Maynard Smith, M. Murphy, B.G. Spratt, C.E. Moore, and N.P.J. Day. 2003. J. Bacteriol. 185:3307-3316.

ants (SLVs) associated, (2) the highest number of DLVs (double locus variants) associated (in case of equivalent solutions), and (3) the highest number of samples belonging to the type. In FPQuest, the *most frequent states* can also be used as a priority rule, and each of these rules can be assigned the first priority.

As discussed in the introduction, a pure minimum spanning tree assumes that all types needed to construct a correct tree, are present in the sampled data. Conversely, algorithms like MP will introduce hypothetical nodes for every internal branch, while the samples from the data set define the branch tips.

The major problem with the MST algorithm in this view is that it requires a very complete data set to obtain a probably correct tree topology. In reality, a number of existing types may not have been included in the sampled data set. If such missing samples represent central nodes in the "true" MST, their absence may cause the resulting tree to look very different, with a much larger total spanning.

The MST algorithm in FPQuest offers an elegant solution to this problem, by allowing hypothetical types to be introduced that cause the total spanning of the tree to decrease significantly. In the context of MLST, these are usually missing types for which a number of SLV (single locus variants) are present in the data set. From an evolutionary point of view, it is very likely that such types indeed exist, explaining the existence of SLVs.

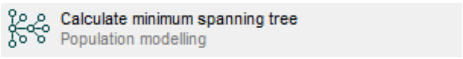
4.7.3 Calculating a minimum spanning tree from character tables

The **DemoBase** does not contain a categorical data set such as MLST type data. However, the MST method can also be applied to binary data. You can create a binary data set, e.g. using the **RFLP1** fingerprint data, by calculating a global band matching table.

4.7.3.1 Open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.7.3.2 Perform a global band matching from the fingerprint type **RFLP1** for all entries in comparison **All**, as described in 4.3.2.

4.7.3.3 To calculate a minimum spanning tree, select ***Clustering > Calculate > Minimum spanning tree (population modeling)*** or press the  button and from the floating menu that appears, select



.

The *Minimum spanning tree* dialog box appears as depicted in Figure 4-59. This dialog box consists of four panels, about (1) the treatment of ***Hypothetical types***, (2) the ***Coefficient*** to calculate the distance matrix, (3) the ***Priority rule*** for linking types in the tree, and (4) the settings for the ***Creation of complexes***

- **Hypothetical types:**

With the checkbox ***Allow creation of hypothetical types (missing links)***, you can allow the algorithm to introduce hypothetical types as branches of the MST, as described in 4.7.2. When enabled, the following criteria can be specified:

- ***Create only if total distance is decreased with at least*** (default 1) ***changes***: Only in the case the introduction of a hypothetical type decreases the total spanning of

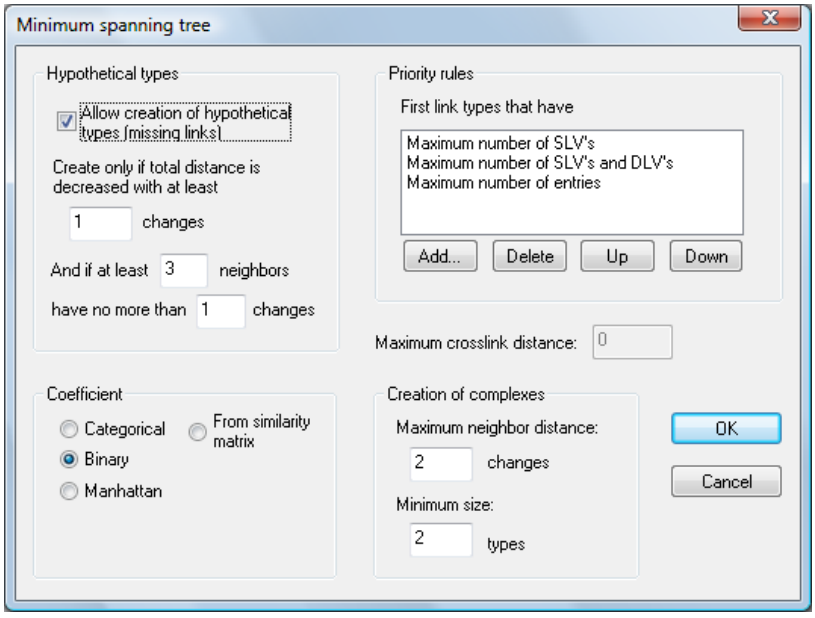


Figure 4-59. The *Minimum spanning tree* dialog box, with *Allow creation of hypothetical types* checked.

the tree with one change, the hypothetical type will be accepted.

- ***And if at least*** (default 3) ***neighbors have no more than*** (default 1) ***changes***: The algorithm will only accept hypothetical types that have at least 3 neighbors (closest related types) that have no more than 1 changes (see also 4.7.2 for the interpretation of this rule).

- **Coefficient:**

The choice is offered between ***Categorical***, for categorical data and ***Binary***, for binary data. When ***Manhattan*** is checked, the sum of the absolute differences between the values of any two corresponding states is calculated, and the thus obtained distances are used to calculate the MST. This option can be used to cluster non-binary, non-categorical data with integer values. If non-integer (decimal) values are used, the program will round them to the closest integers.

In the ***Manhattan*** option, an ***Offset*** and a ***Saturation*** value can be specified. For each character compared between two types, the *offset* value determines a fixed distance that is added to the distance of these characters. If the distance is zero, however, the transformed distance remains zero. In addition, for each character compared between two types, the *saturation* determines the maximum value the distance can take. In other words, above the saturation distance, different characters are all seen equally different. The relation between offset, saturation, and distance of characters is illustrated in Figure 4-60. The offset and distance can be used to tune the summed distance result between fully categorical (offset = 1 and saturation = 1) and fully numerical (offset = 0 and saturation infinite).

- **Priority rules**

In case of equivalent solutions in terms of calculated distance, the priority rules allow you to specify a priority based upon other criteria than distance. One or more rules can be added, with a maximum of 3. The order of appearance of the rules determines their rank.

4.7.3.4 A rule can be added by pressing the ***<Add>*** button. One of the following rules can be selected (Figure 4-61):

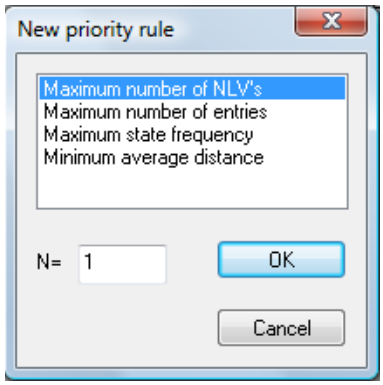


Figure 4-61. Priority rule selection box.

- ***Maximum number of NLV's*** (Ntuple Locus Variants). N has to be chosen by the user. For example, if N is 1, the rule becomes "*Maximum number of SLV's*". This means that, in case two types having an equal distance to a linkage position in the tree, the type that has the highest number of *single locus variants* (i.e. other types that differ only in one state or character) will be linked first. If N=2, the rule becomes "*Maximum number of SLV's and DLV's*".

- ***Maximum number of entries***: The program counts how many entries each unique type contains, and the type that has the highest number of entries will be assigned priority, in case of equivalent linkage possibilities.

- ***Maximum state frequency***: The program calculates a frequency table for each state of each character. Types are thus ranked based upon the product of

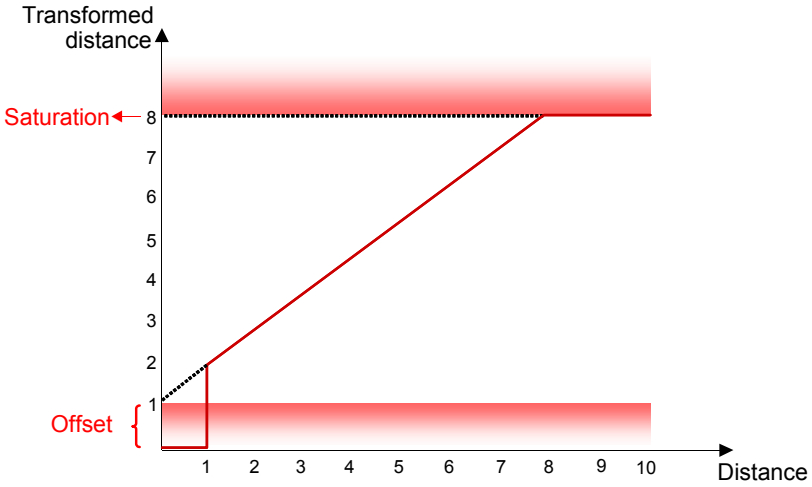


Figure 4-60. Graphical representation of the meaning of *offset* and *saturation* values.

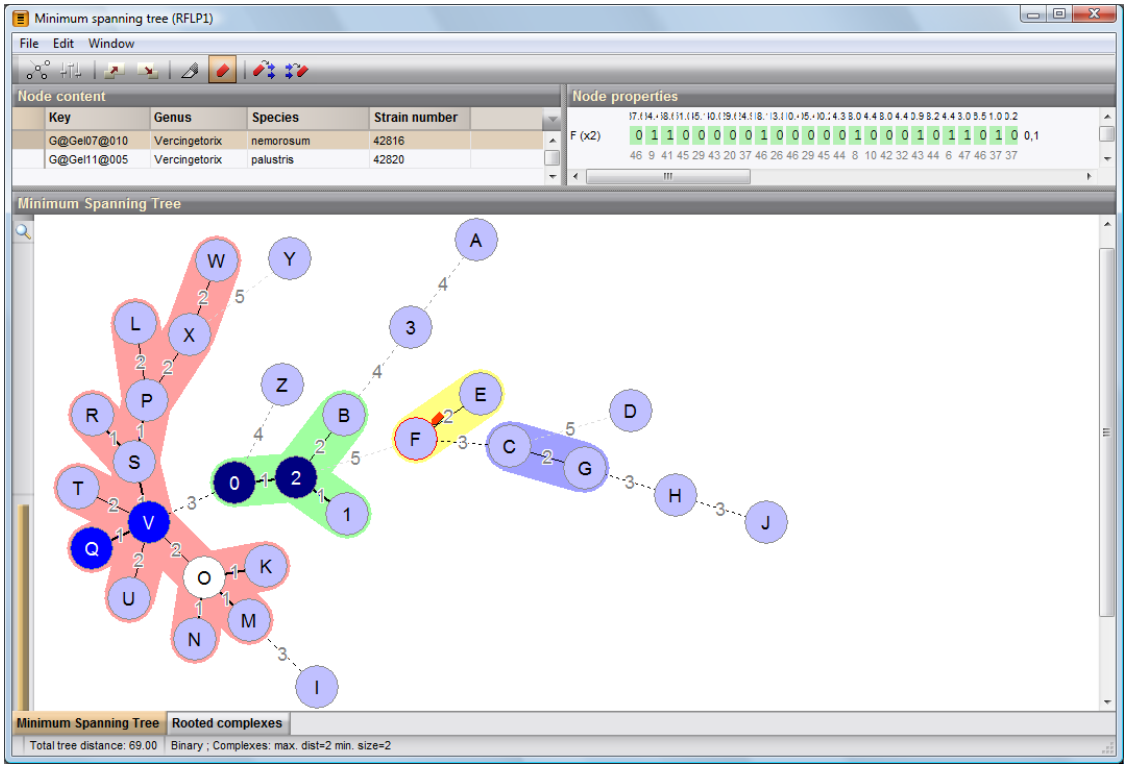


Figure 4-62. The *Minimum spanning tree* window, *Minimum Spanning Tree* panel displayed.

frequencies of their characters. In case of equivalent possibilities, types that have the highest state frequency rank are linked first.

- ***Minimum average distance***: the type that has the lowest average distance with the other types will be linked first, in case of equivalent solutions.

4.7.3.5 Since the order of appearance in the list of defined priority rules is determinative for the order of execution, it is possible to move rules up or down using the **<*Up*>** and **<*Down*>** buttons.

- **Maximum crosslink distance**

With this option, you can allow the program do display alternative equivalent solutions under the clustering criterion used, which are then displayed as crosslinks. Suppose that the program has linked group B to group A because they differ in one state. If group B has also one state difference with another group, C, it will be shown as a crosslink between B and C. Crosslinks are indicated as dark red lines.

4.7.3.6 You can specify the maximum number of states difference before a crosslink will be displayed. For example, if 2 is entered, only crosslinks between groups that have 1 or two states difference will be indicated. If 0 is entered, crosslinks will not be shown.

- **Creation of complexes**

In epidemiological population genetics based upon MLST, a *clonal complex* can be defined as a single group of isolates sharing identical alleles at all investigated loci, plus single-locus variants that differ from this group at only one locus[1]. In another, more relaxed definition[2,3], a clonal complex includes all types that differ in *x* loci or less from at least one other type of the complex (*x* is usually taken as 1 or 2). Under this definition, not all types of a complex are necessarily SLVs or DLVs from one another. The latter definition is used in FPQuest.

The maximum number of changes allowed to form complexes can be specified; the default value is 2. In addition, one can also specify a minimum number of types that should be included before the groups is defined as a complex. The default value is 2.

4.7.4 Interpreting and editing a minimum spanning tree

After pressing **<*OK*>** in the *Minimum spanning tree* dialog box, the *Minimum spanning tree* window will pop up. In the example shown in Figure 4-62 and Figure 4-63, a band matching table of **RFLP1** in the **DemoBase** database was created and analyzed as ***Binary***, while the other parameters were left to the defaults.

1. Feil, E.J., J. Maynard Smith, M.C. Enright, and B.G. Spratt. 2000. Genetics 154: 1439-1450.
2. Feil, E.J. J.E. Cooper, H. Grundmann, D.A. Robinson, M.C. Enright, T. Berendt, S.J. Peacock, J. Maynard Smith, M. Murphy, B.G. Spratt, C.E. Moore, and N.P.J. Day. 2003. J. Bacteriol. 185:3307-3316.
3. BURST (Based Upon Related Sequence Types) program description, see the MLST website http://www.mlst.net.

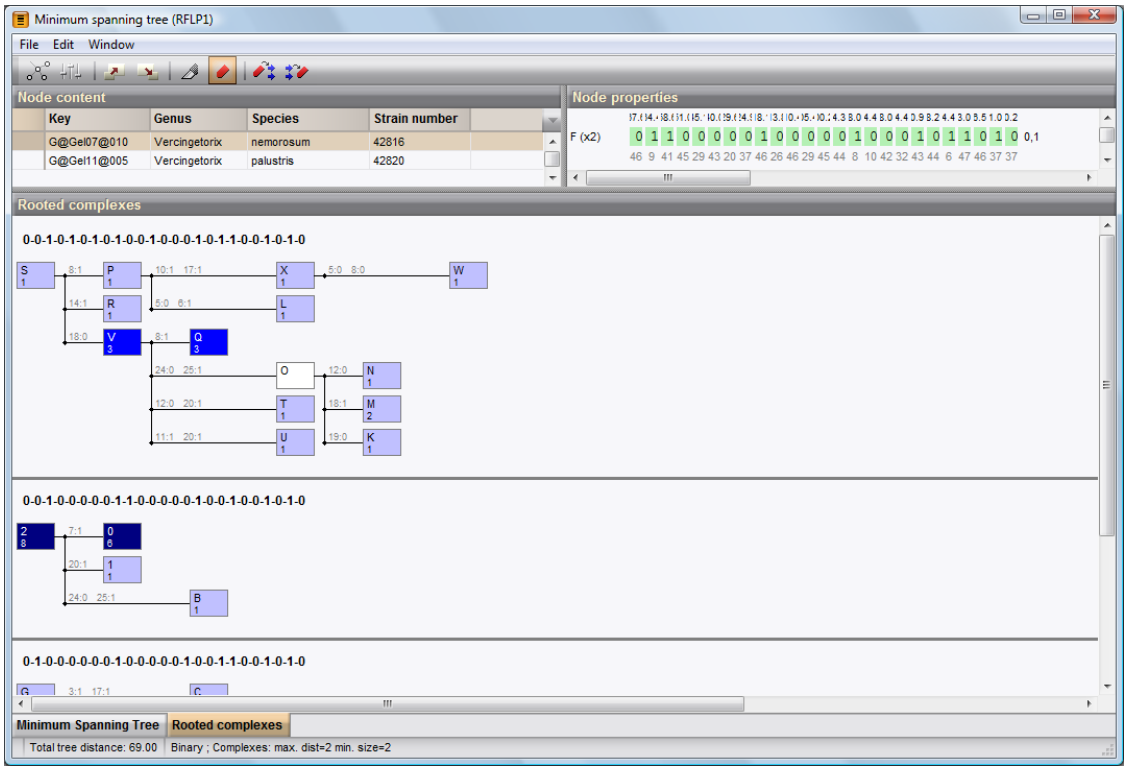


Figure 4-63. The *Minimum spanning tree* window, *Rooted complexes* panel displayed.

The window is divided in four panels, of which the *Minimum Spanning Tree* panel displays the actual MST, the *Node content* panel lists the entries belonging to the selected node or nodes, the *Node properties* panel shows the type for the selected node or nodes and the *Rooted Complexes* panel (in default configuration tabbed view with the *Minimum Spanning Tree* panel) displays the composition of the complexes. All four panels in the *Minimum spanning tree* window are dockable. See 1.6.4 for general display options of dockable panels.

•Display options

In the *Minimum Spanning Tree* panel, each type is represented by one node or branch tip, displayed as circles that are connected by branches. In the default settings, but with ***Letter code*** selected under ***Type labeling***, the following information can be derived from the tree view:

- When sufficiently zoomed (using the zoom buttons  and  , the zoom slider (see 1.6.7) or the keyboard shortcuts CTRL+PgUp and CTRL+PgDn), a letter code will appear within each circle, uniquely identifying each type. In case of more than 26 types in total, a two-letter code is used, of which the second can be a digit 1-9 as well. The codes are assigned alphabetically according to the *Priority rule* specified (see 4.7.3).

- The length of the branches is proportional to the distance between the types, and the thickness, dotting, and graying of the branch lines also indicate the distance between the nodes.

- The number of entries contained in a type (node) is indicated using a color ranging from white over three blue shades to brown and red.

In the *Rooted complexes* panel, the complexes are displayed as defined under the specified calculation settings (see 4.7.3).

- Each complex is shown as a rooted tree, with the type having the highest priority, as defined by the *Priority rule* (4.7.3) defining the root. On top of the *Rooted complexes* panel, the character values of the root type are indicated. The branch lengths of the derived types (i.e., the types branching from the root) are in proportion to the distances of these types.

- For each type branching off from the root type, the change(s) is (are) indicated as two numbers separated by a colon. The first number is the character number, and the second number is the value towards the character has changed. For example, 6:003 means that character 6 has changed into 003 for this type. If more than one change has led to a derived type, the changes are indicated next to each other.

- Similarly as on the tree, the types are indicated with a color reflecting the number of entries contained in the type. In addition, the number of entries is written just below the type code.

In the *Node content* panel, the entries contained in the selected node(s) are shown in a grid or tabular format (see 1.6.6 for display options of grid panels). If the entries are selected in the *Comparison* window, this is indicated here as well, with the same colored arrows. Selections can be made in this list using the CTRL and SHIFT keys, and the entry card can be popped up by double-clicking on an entry.

The *Node properties* panel displays the details of the highlighted type(s) in the *Minimum Spanning Tree* panel or the *Rooted complexes* panel. If a type is selected in the *Minimum Spanning Tree* panel, it becomes highlighted by a red circle, and marked with a red flag. The same type becomes highlighted in the *Rooted complexes* panel, by a red rectangle. For the highlighted type, detailed information is shown in the *Node properties* panel.

- On top of the panel, the character names and character values (on green background) are shown for the highlighted node. The frequencies of the character values are indicated in gray.

- Left from the character list is the name of the type with the number of entries it contains between brackets.

- Right from the character list is the number of SLVs (single locus variants; types differing only in one character) and DLVs (double locus variants; types differing in two characters).

- In case more than one node is highlighted in the *Minimum Spanning Tree* panel or the *Rooted complexes* panel, the highlighted types are displayed under each other in the *Node properties* panel. Characters that are the same for more than 50% of the types are shown on a green background. Characters for which there is less than 50% consensus are shown on a white background. A character that is different from the majority in a type is indicated in red.

- **Edit options**

4.7.4.1 With ***Edit > Display settings*** or  , the display options can be customized in the *Display settings* dialog box (Figure 4-64).

4.7.4.2 Under ***Cell color***, you can use a color to display the number of entries, any groups or field states defined, or the groups pie charts. The colors are displayed both in the *Minimum Spanning Tree* panel and the *Rooted complexes* panel.

4.7.4.3 With ***Number of entries*** selected, a differential color will be assigned to the nodes according to the number of entries they contain. The intervals can be specified under ***Number of entries coding***.

4.7.4.4 With ***Groups*** selected, the colors assigned to the groups (see 4.1.11) in the comparison or to field states (if defined, see 2.2.5), will be given to the nodes. When a type consists of more than one group, it will become black. ***Groups (pie chart)*** is similar, except that, in case a type (node) consists of more than one group, the different groups will be represented in a pie chart. This option also works in combination with the ***Compact complexes*** option (4.7.4.9). In the *Rooted complexes* panel, the different group colors are also displayed in the type boxes, in a proportional way.

4.7.4.5 ***Number of entries coding*** is only enabled when ***Number of entries*** is selected under ***Cell color***.

NOTE: By default, the first color (white) is set as <= 0. This means that only empty nodes are white. This is useful to visualize hypothetical nodes (see 4.7.3) when this option is enabled. When no hypothetical nodes are allowed, it is more useful to enter a positive value, for example <= 1.

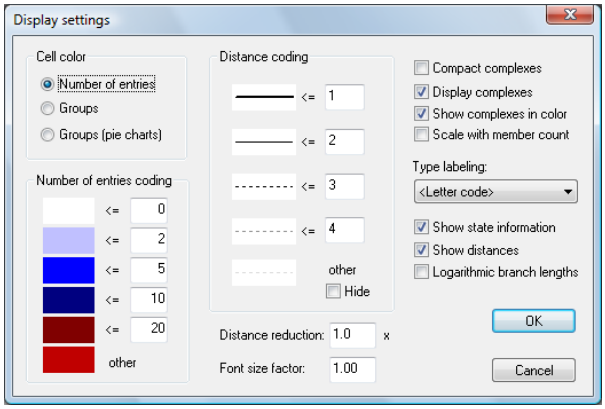


Figure 4-64. The *Display settings* dialog box in the *Minimum spanning tree* window.

4.7.4.6 Under ***Distance coding***, you can specify the distance that corresponds with the different line types offered by the program.

4.7.4.7 With ***Distance reduction***, you can change the length of the branches. In the *Minimum Spanning Tree* panel, this only changes the zoom, but in the *Rooted Complexes* panel, this value will determine the horizontal distance between the types displayed. With a distance reduction of e.g. 1.5x, the distance unit is decreased with a factor 1.5.

4.7.4.8 The option ***Display complexes*** allows you to choose whether the complexes are displayed or not. Note that this option only applies to the *Minimum Spanning Tree* panel; the complexes remain displayed in the *Rooted Complexes* panel.

4.7.4.9 Using ***Compact complexes***, you can choose to display a full complex as one node on the tree. The diameter of the circle is (slightly) proportional to the number of types the complex contains, and the compacted complexes are encircled.

4.7.4.10 With ***Use color***, you can display the image in color or grayscale mode.

4.7.4.11 ***Scale with member count*** is an option that lets the diameter of the circles depend on their size.

4.7.4.12 Under ***Type labeling***, it is possible to select the ***Letter code*** which is automatically assigned to the types by the program, or any of the information fields the database contains. In the latter case, types (nodes) that do not all have the same string will be marked with **???**. Note also that you may have to zoom in sufficiently to visualize longer labels than the letter codes. If the labels do not fit within the circle, they are represented by ... .

4.7.4.13 The option ***Show state information*** relates to the *Node properties* panel, where the states of any selected nodes can be displayed. When this option is unchecked, the states of the characters for the selected nodes are not displayed.

4.7.4.14 With ***Show distances***, you can have the distances indicated on the branches of the tree.

As indicated earlier, it is possible to highlight nodes on the tree or in the *Rooted Complexes* panel. You can use the SHIFT or CTRL keys to highlight multiple nodes, or drag a rectangle with the mouse in the tree or the complex panel. For a single highlighted node, you can select individual entries directly in the *Node content* panel.

4.7.4.15 For one or more highlighted nodes, it is also possible to select all the entries directly from the tree panel, by pressing the  button or choosing ***Edit > Select all entries in selected nodes*** from the menu.

4.7.4.16 Likewise, it is possible for any selected entry to highlight all the types where this entry occurs, using the  button or ***Edit > Select nodes that contain selected entries***.

4.7.4.17 With ***Edit > Select related nodes***, you can highlight all the types that have no more than a specified number of changes from the highlighted type(s). When choosing this menu command, the program asks to enter the maximum distance from the highlighted type(s).

4.7.4.18 On the tree, the highlighted nodes are marked by default with a red label and with a red circle as well. You can choose to hide or show this label using  button or with ***Edit > Label selected nodes***.

4.7.4.19 The *Cut branch tool* (***Edit > Cut branch tool*** or ) is a cursor tool that allows a branch of the tree to be "cut off" and displayed as one simple end node. A branch can be cut off by selecting the branch cut tool, moving the cursor towards one end of a branch and left-clicking. When cut off, the branch is displayed as a green node which always has the same size, regardless of the zoom. To disclose the branch again, simply double-click on the green node.

4.7.4.20 The drop-down list 

allows you to select a coloring based on groups or any of the available field states.

4.7.4.21 The complexes present on the minimum spanning tree can be converted into Groups using ***File > Convert complexes to groups***.

4.7.4.22 With ***Edit > Show crosslinks***, you can toggle between displaying and hiding the crosslinks. CTRL+C is a shortcut for this operation. This feature cannot be combined with the ***Compact complexes*** option (4.7.4.9).

4.7.5 Calculating a minimum spanning tree from a similarity matrix

As explained in the introduction (see 4.7.1), FPQuest allows MST's to be calculated from similarity matrices obtained from any type of data. The condition to use a true distance matrix as input is thereby not necessarily met. In particular when banding patterns are compared using a binary band matching coefficient, where infringements upon the transitivity rule happen frequently (see also Section 4.6), a MST cannot perfectly depict the "odd" relationships given in the similarity matrix. Since there is no tree algorithm that can deal with intransitivities, it is equally justified to apply the MST algorithm as, for example, UPGMA to such matrices.

To convert a similarity matrix into an integer distance matrix, the software uses bins of certain similarity intervals that will be converted into distance units. For example, with a similarity bin size of 1%, two entries that have a similarity of 99.6% will have a distance of zero. Two entries that have a similarity of 98.7% will have a distance of 1.

As an example, we will analyze the fingerprint type **RFLP1** in **DemoBase**.

4.7.5.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.7.5.2 Select **RFLP1** in the *Experiments* panel and choose ***Clustering > Calculate > Cluster analysis (similarity matrix)***.

Under **Similarity coefficient**, select ***Different bands***. This will generate a similarity matrix with discrete integer distance values, which will result into an easily interpretable MST.

4.7.5.3 It does not matter what tree algorithm is used; only the similarity matrix is needed. Press ***<OK>*** to calculate the matrix.

4.7.5.4 Select ***Clustering > Calculate > Minimum spanning tree (population modelling)***.

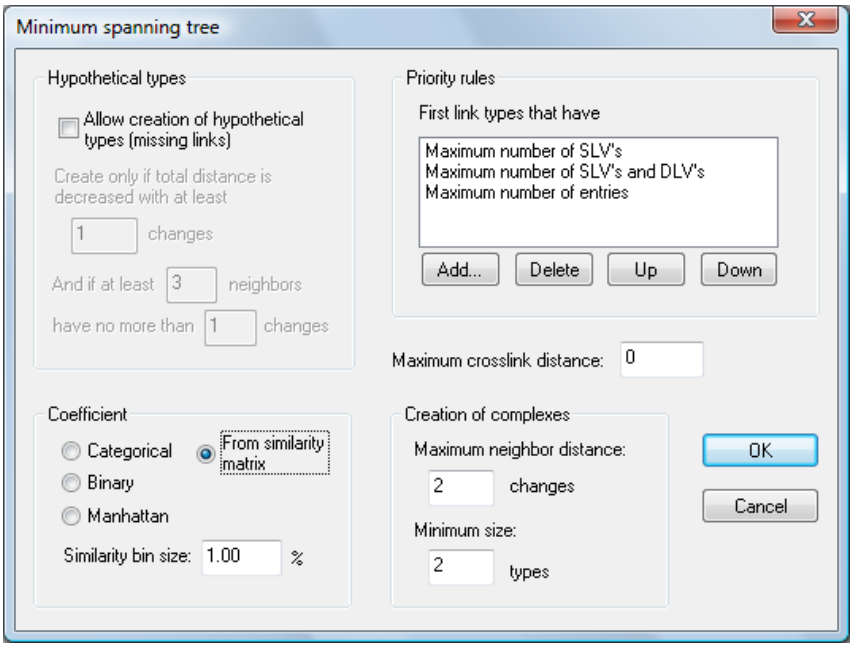


Figure 4-65. ***Minimum spanning tree*** **dialog box, starting from similarity matrix.**

In the *Minimum spanning tree* dialog box that pops up, the option ***From similarity matrix*** is selected and the other options relating to character data are disabled (Figure 4-65). The ***Similarity bin size*** allows the bin size for the conversion to a discrete distance matrix to be specified.

The other options in this dialog box are as explained in 4.7.3.

4.7.5.5 Leave the bin size to 1% and press **<*OK*>** to calculate the MST.

In the resulting MST, connecting lines between nodes can be directly translated into numbers of different bands: short thick line = 1 band different; thin full lines = 2 bands different; black dashed lines = 3 bands different etc. Of course, all entries that have no bands different fall in the same node.

4.8 Dimensioning techniques DI

4.8.1 Introduction

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 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 4.3.2).

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.

4.8.2 Calculating an MDS

4.8.2.1 Any experiment type for which a complete similarity matrix is available can be analyzed by MDS.

4.8.2.2 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.8.2.3 Select **AFLP** in the *Experiments* panel and check whether a similarity matrix is available for this experiment type by looking in the ***Layout*** menu if the menu command ***Show matrix*** is enabled (not grayed).

4.8.2.4 If ***Show matrix*** is grayed, first calculate a dendrogram with ***Clustering > Calculate > Cluster analysis (similarity matrix)***.

4.8.2.5 Select ***Dimensioning > Multi-dimensional scaling*** or  .

The program now asks "***Optimize positions***". FPQuest 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 slightly longer.

4.8.2.6 Press ***<Yes>*** to optimize the positions.

The MDS is calculated and the *Coordinate space* window is shown (see Figure 4-66).

4.8.3 Editing an MDS

The *Coordinate space* window (Figure 4-66) shows the entries as dots in a cubic coordinate system.

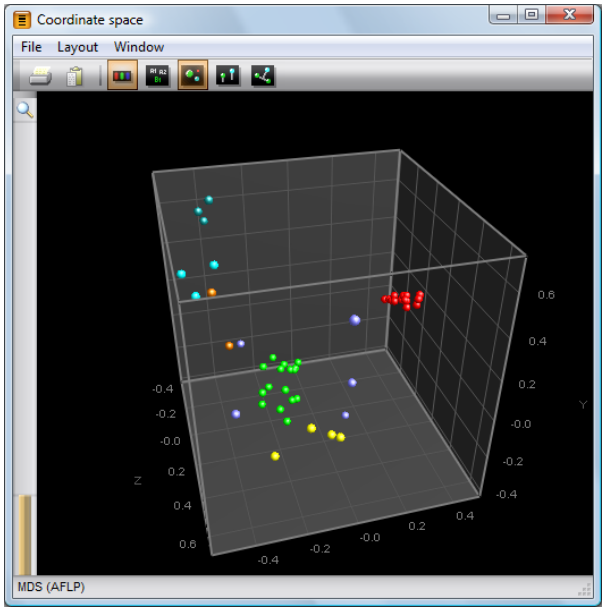


Figure 4-66. *Coordinate space* window, resulting from a PCA or MDS analysis.

4.8.3.1 To zoom in and zoom out on the image, press the PgDn and PgUp keys, respectively. Alternatively, the zoom slider can be used (see 1.6.7 for the zoom slider functions).

4.8.3.2 The image can be rotated in real time by clicking on the image and dragging in the desired direction with the mouse.

By default, the entries are represented as 3D spheres in a realistic perspective. They appear in the colors as defined for the groups on the dendrogram (4.1.11).

4.8.3.3 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:

4.8.3.4 In the parent *Comparison* window, select ***Layout > Use group numbers as key***.

4.8.3.5 A legend to the group numbers can be obtained with ***File > Export database fields*** in the parent *Comparison* window.

Alternatively, a selected information field can be displayed instead of the key:

4.8.3.6 In the parent *Comparison* window, click on the information field which you would like to see displayed (e.g. 'Strain number').

4.8.3.7 Select ***Layout > Use field as key*** from the menu in the *Comparison* window. The strain number is now displayed in the MDS plot.

4.8.3.8 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 generally easier to evaluate the groups using colors.

4.8.3.9 If field states with corresponding color coding were defined (see 2.2.5), the drop-down list 

allows you to select a coloring based on groups or any of the available field states.

4.8.3.10 Select an entry coordinate system using CTRL+left mouse click. Selected entries are contained in a blue cube.

4.8.3.11 To select several entries at a time, hold down the SHIFT key while dragging the mouse in the coordinate system. All entries included in the rectangle will become selected.

4.8.3.12 By double-clicking on an entry, its *Entry edit* window is popped up.

4.8.3.13 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.

4.8.3.14 ***With Layout > Show rendered image*** or  , you can toggle between the realistic three-dimensional perspective with entries represented by spheres, and a simple mode where entries are represented as dots.

4.8.3.15 With ***Layout > Preserve aspect ratio*** enabled, the relative contributions of the three components are respected, which means that the coordinate system is no longer shown as a cube.

4.8.3.16 Another very interesting display 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).

4.8.3.17 To copy the coordinate space image to the clipboard, select ***File > Copy image to clipboard*** or  .

4.8.3.18 The image can be printed with ***File > Print image*** or  . The image will print in color if the colors are shown on the screen.

4.8.4 Calculating a PCA

PCA is typically executed on complete character data. Fingerprints can only be analyzed by PCA if a band matching is first performed (see 4.3.2).

4.8.4.1 In the *FPQuest main* window with **DemoBase** loaded, open comparison **All**, or create a comparison with all entries except those defined as STANDARD (see 4.1.3.2 to 4.1.3.4).

4.8.4.2 Select **AFLP** in the *Experiments* panel and ***Dimensioning > Principal Components Analysis*** or  .

The *Principal Components Analysis* dialog box (Figure 4-67) allows a number of more advanced choices to be made.

The simplest choice is "***Use quantitative values***". By default, this choice is checked, and if the technique provides quantitative information, one will normally want to use this information for the PCA calculation. If this option is unchecked, only the presence or absence of a band is taken into account.

More sophisticated options are the possibilities to ***Subtract average*** character value over the ***Entries***, and

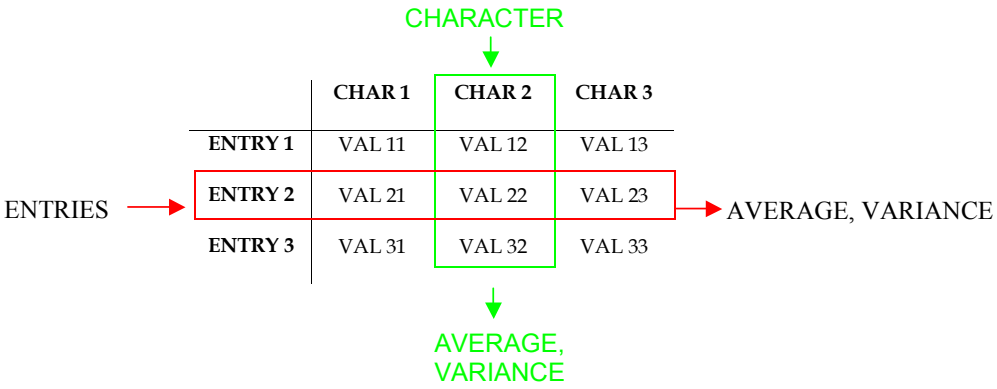


Figure 4-68. Character table showing the meaning of *Average* and *Variance* correction at the *Entries* and *Characters* level.

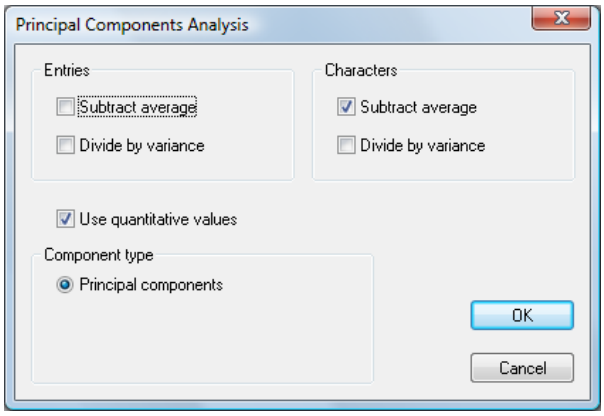


Figure 4-67. *Principal Components Analysis* dialog box.

to ***Subtract average*** character value over the ***Characters***. Figure 4-68 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. Dividing by the variance for each character 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 a number of gels, belonging to the same fingerprint type, for which the staining efficiency differs from gel to gel. If you decide to use quantitative values to calculate a PCA, gels that were efficiently stained will have higher band intensity values then gels stained less efficiently. Without correction, efficiently stained and less efficiently stained gels will fall apart in the study. Dividing by the variances normalizes the quantitative band matching information for such irrelevant differences, making band matching sets with different overall band intensities fall together as long as the relative intensities of the bands 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, relative band surfaces).*

The lower panel of the dialog box (Figure 4-67) displays the ***Component type***. In FPQuest, this option can only be ***Principal components*** (BioNumerics also allows Discriminant analysis to be performed).

4.8.4.3 In the *Entries* and *Characters* panels, check ***Subtract average*** under *Characters*, and leave the other options unchecked.

4.8.4.4 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 4-69.

The *Principal components analysis* window is divided in three dockable panels (for display options of dockable panels, see 1.6.4). In the *Components* panel (the left panel in default configuration), 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 *Entry coordinates* panel shows the *entries* plotted in an X-Y diagram corre-

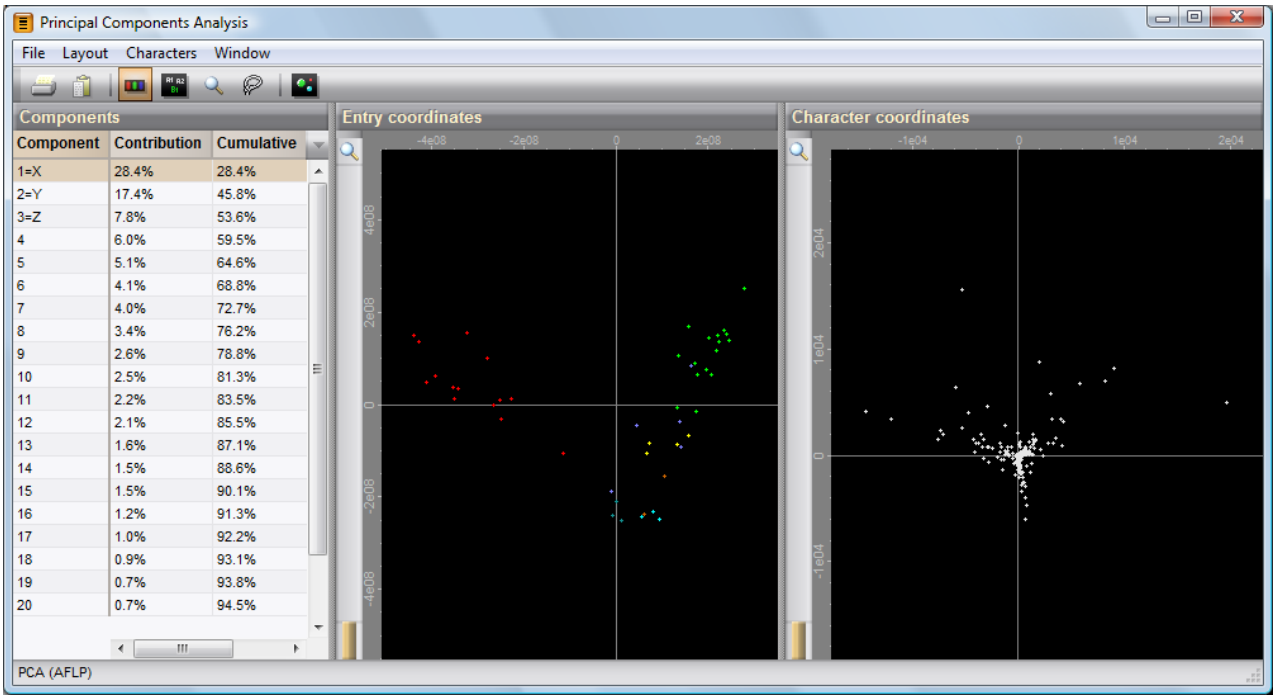


Figure 4-69. ***Principal components analysis*** **window.**

sponding to the first two components. The *Character coordinates* panel shows the characters plotted in the same X-Y diagram. From the *Character coordinates* 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).

4.8.4.5 If you want to assign another component as one of the axes, select the component in the *Components* panel, and ***Layout > Use component as X axis***, ***Layout > Use component as Y axis***, or ***Layout > Use component as Z axis*** (or right-click on the component).

• Layout tools:

4.8.4.6 Switching from color indication for the groups to symbol indication with ***Layout > Show group colors*** or


.

4.8.4.7 Selecting a coloring based on groups or any of the available field states from the drop-down list

. See 2.2.5 on how to define states for an information field.

4.8.4.8 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 underlaying* 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**. *Alternatively, click on an information field and select* **Layout > Use field as key** *in the underlying* Comparison *window.*

4.8.4.9 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.

4.8.4.10 With ***Layout > Zoom in / zoom out*** or  , you can zoom in on any part of the *Entry coordinates* or *Character coordinates* 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. In order to restore the original size of the image, simply left-click within the panel. Disable the zoom-mode afterwards. Alternatively, the zoom sliders of the *Entry coordinates* and *Character coordinates* panel can be used to zoom in or out on the plots (see 1.6.7 for a description of the zoom slider functions).

4.8.4.11 If you move the mouse pointer over the *Character coordinates* panel (characters), the name of the pointed character is shown.

•Editing tools:

4.8.4.12 Entries can be selected in a *Principal components analysis* 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. You can also hold down the CTRL key while clicking on an entry.

4.8.4.13 An even more flexible way of selecting entries is using the lasso selection tool. To activate the lasso selection tool, choose ***Layout > Lasso selection tool*** or press the  button. With the lasso selection tool enabled, selections of any shape can be drawn on the plot. The lasso selection tool menu item is flagged and the button shown as  when the tool is enabled. To stop using the lasso selection tool, you have to click the button a second time, or disable it from the menu.

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!**

4.8.4.14 If you want to add entries to an existing PCA, you can select new entries in the *FPQuest main* window and copy them to the clipboard using ***Edit > Copy selection*** or  .

4.8.4.15 In the *Comparison* window, select ***Edit > Paste selection*** or  . The new entries are placed in the *Comparison* window and in the *PCA* window.

4.8.4.16 To delete entries from a PCA, select the entries as in 4.8.4.12 and in the *Comparison* window, select ***Edit > Cut selection*** or  .

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 for composite data sets):

4.8.4.17 In the *Principal components analysis* window, first determine the component that best separates the groups.

4.8.4.18 Select that component in the *Components* panel and select ***Characters > Order characters by component***. The characters are now ordered by the selected component in the underlying *Comparison* window.

4.8.4.19 The entry plot can be printed with ***File > Print image (entries)*** or  and the character plot can be printed with ***File > Print image (characters)***.

4.8.4.20 Alternatively, the entry plot can be copied to the clipboard with ***File > Copy image to clipboard (entries)*** or  and the character plot can be copied to the clipboard with ***File > Copy image to clipboard (characters)***.

If you want to reconstruct or analyze the PCA system in another software package, it is possible to export the coordinates of the entries along a selected component (for example the X-axis):

4.8.4.21 Select a component and ***File > Export selected 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 selected entry coordinates***.

4.8.4.22 It is also possible to export all entry coordinates at once in a tab-delimited format using ***File > Export all entry coordinates***.

Similarly, one can export the coordinates for the characters for a certain component:

4.8.4.23 Select a component and ***File > Export selected character coordinates***.

4.8.4.24 To export all character coordinates at once, use ***File > Export all character coordinates***.

FPQuest allows you to display three components at the same time, by plotting the entries in a three-dimensional space.

4.8.4.25 To create a three-dimensional plot from the PCA, select ***Layout > Show 3D plot*** or  .

The *Coordinate space* window is shown. See 4.8.3 for instructions on editing a PCA in 3-D representation mode.

4.8.4.26 Close the *Coordinate space* window with ***File > Exit***.

4.8.4.27 Close the *PCA* window with ***File > Exit***.

4.9 Chart and statistics tools

4.9.1 Introduction

A number of simple chart tools are available in FPQuest to apply to the database information fields or to character data for the entries in a comparison. FPQuest also offers the possibility to perform some basic statistic analysis on the entries and variables used in a chart. Given the large variety of information and character types FPQuest can contain, there are many different types of charts that can be displayed, depending on the type of the variable(s) to present. For each chart one or more standard statistical tests are implemented. The next paragraphs are intended to provide some information on the terminology (4.9.2) and the mathematical background (4.9.3) of these tests.

The use of the chart and statistics tools is described in paragraphs 4.9.4 to 4.9.11.

4.9.2 Basic terminology

4.9.2.1 Literature

This manual is not aimed to be an introduction to basic statistics. For more detailed literature, we refer to the following handbooks:

- Press W., Teukolsky S.A., Vetterling W.T., Flannery B.P., 'Numerical recipes in C', Cambridge University Press, Cambridge.
- Sheskin D.J., 'Handbook of parametric and nonparametric statistical procedures', CRC Press, Boca Raton.
- Zwillinger D., Kokoska S., 'Standard probability and statistics tables and formulae', Chapmann & Hall/ CRC, Boca Raton.

4.9.2.2 Application of statistic tests

In general terms, the application of a statistic test can be outlined as follows:

- Make a proposition that will be referred to as the **null-hypothesis**. ***Statistical tests cannot be employed for proving that a certain hypothesis is true, but only for proving that all alternative hypotheses can be rejected.*** Therefore, the null-hypothesis is what one wants to reject.
- Determine what **statistic** will be used. A statistic is a value calculated from the data set by means of some formula and which is sensitive to the null-hypothesis that will be tested for.
- If the null-hypothesis is true, the probability function of the statistic is known.
- If the statistic is located on an unfavorable position in the probability function, i.e. if its probability is very small, the null-hypothesis can be rejected. The opposite is not true: the null-hypothesis cannot be accepted as fulfilled if the statistic has a favorable location in the probability distribution.

Note that not all tests are applicable in all situations. There may be restrictions to e.g. the amount of data in the sample, or to some basic properties of the data set. These restrictions are mentioned where the tests are described.

4.9.2.3 Parametric or non-parametric tests

Parametric tests basically suppose that the data are distributed normally; they generally make use of the values for the mean and the standard deviation.

Non-parametric tests are commonly based on a ranking of the data. These ranks are distributed uniformly, hence these tests are independent of any underlying distribution. The price to pay is that an estimate of the significance is more complicated and often relies on approximations. These methods also generally loose some strength because they loose some information about the data. In comparison with parametric tests they require more data to come to an equally significant result.

For these tests the values of the data points are usually replaced by their rank among the sample. The data points are ordered, the lowest in order is assigned rank one and the highest in order is assigned the rank that equals the total sample size.

If some of the data points originally have the same values, they can be assigned the mean of the ranks (called 'tie rank') they would have had if they were different. The sum of the assigned ranks is always equal to the total sample size.

4.9.2.4 Categorical or quantitative data

Within the chart tool, a distinction between three types of variables is made.

- **Categorical variable**: this type of variable divides a sample into separate categories or classes. Examples are database fields like e.g. genus, species, etc. Also intervals of quantitative variables can be treated as categorical data.

- **Quantitative variable**: this type of variable can take either continuous numerical values or binary values. Character data are a typical example for this type of variable. Continuous numerical values can be converted into interval data if necessary. If this option is chosen, an interval size can be specified.

- **Date variable**: a variable containing a date. This variable can be converted into interval data, which means that it can be interpreted as either a categorical variable or a quantitative variable. When converting into interval data, you can choose to group the dates by day, week, month, quarter or year.

With combinations of these variables several types of plots can be created, based upon:

One variable:

- *Bar graph*: for a single categorical variable

- *1-D numerical distribution*: for one quantitative variable

Two variables:

- *Contingency table*: for two categorical variables

- *2-D scatterplot*: for two quantitative variables

- *2-D ANOVA* plot: for one categorical and one quantitative variable.

Three variables:

- *3-D scatterplot*: for three quantitative variables

For an overview of graph types and associated tests for one and two variables, see Table 4-1.

Some types of plots can be extended in the sense that they can display information from an additional categorical variable by means of a color code. These plots are the 2-D scatterplot, the 3-D scatterplot, the 2-D ANOVA plot and the 1-D numerical distribution.

	Categorical	**Quantitative**
- - -	**Bar graph (4.9.3.1)** *Chi square test for equal category sizes*	**1-D numerical distribution (4.9.3.4)** *Kolmogorov-Smirnov test for normality*
Categorical	**Contingency table (4.9.3.3)** *Chi square test for contingency tables*	**2-D ANOVA plot (4.9.3.6)** *See Table 4-3*
Quantitative	**2-D ANOVA plot (4.9.3.6)** *See Table 4-3*	**2-D scatterplot (4.9.3.5)** *See Table 4-2*

Table 4-1. Schematic representation of variable types and corresponding graphs and tests for one and two variables.

	Parametric	**Non-parametric**
Means	*T test (4.9.3.5.1)*	*Wilcoxon signed-rank test (4.9.3.5.2)*
Correlations	*Pearson correlation test (4.9.3.5.3)*	*Spearman rank-order correlation test (4.9.3.5.4)*

Table 4-2. Overview of tests associated with 2-D scatterplots.

	Parametric	**Non-parametric**
2 categories	*T test (4.9.3.6.1)*	*Mann-Whitney test (4.9.3.6.2)*
>2 categories	*F test (4.9.3.6.4)*	*Kruskal-Wallis (4.9.3.6.4)*

Table 4-3. Overview of tests associated with 2-D ANOVA plots.

4.9.3 Charts and statistics

4.9.3.1 Bar graph: Chi square test for equal category sizes

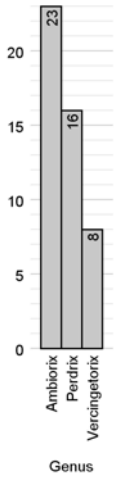


Figure 4-70. Example of a bar graph.

For a bar graph displaying the number of entries for a categorical variable, one typically likes to know if there are significant differences in the number of entries per category. Hence, the null-hypothesis is that all categories have an equal number of entries.

If this null-hypothesis holds, the *expected average count per category* (N_e) can be calculated as the total number of entries divided by the number of categories, $N_e = N/n$, with N the total number of entries and n the number of categories. The *chi square* statistic is calculated from the values for the expected average count (N_e) and the observed entries per category (N_{oi}), $\chi^2 = \sum_{i=1}^{n} \left(\left[N_{oi} - N_e \right]^2 / N_e \right)$, with n the number of categories.

If the null-hypothesis is true and under certain conditions (see the note below) this statistic approximately follows a chi square distribution with n-1 *degrees of freedom*. The *p-value* that is returned gives the probability that the statistic is at least as high as the observed one. If the p-value is low, the null-hypothesis can be rejected. The *significance s* of the test is calculated as the complement of the p-value,

$$s = 100 \times (1 - p) \cdot$$

The values for these parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.5.

```
Chi square: 7.191 (2 degrees of freedom)
P value= 0.027440
   Significance= 97.2560%

Expected average count per category: 15.67
```

Figure 4-71. Example of a test report for the chi square test for equal categorical sizes applied on a bar graph as shown in Figure 4-70.

NOTE: This test should not be used if the expected average count per category is less than 5. If this is the case, consider combining categories in order to increase the expected average count.

4.9.3.2 Bar graph: Simpson and Shannon Weiner indices of diversity

A commonly asked question about a number of entries occurring in different categories, is how they are distributed. Two widely used coefficients to measure the diversity are the *Shannon-Weiner index of diversity* and *Simpson's index of diversity*. Both coefficients take into account the *diversity*, i.e. the number of categories present in the sampled population, as well as the *equitability*, i.e. the evenness of the distribution of entries over the different categories.

Simpson's index of diversity is defined as the probability that two consecutive entries will belong to different categories. Given K categories present in a sampled population, the probability of sampling category i twice consecutively is as follows (n_i is the number of entries in category i):

$$P_i = \frac{n_i(n_i - 1)}{\sum_{j=1}^{K} n_j(n_j - 1)}$$

The probability of sampling any two samples of the same category is given by $P = \sum_{i=1}^{K} P_i$. Hence, the probability D of sampling two different categories is $D = 1 - P$, which is Simpson's index of diversity.

For a sampled population of N entries belonging to K categories, the Shannon-Weiner index of diversity is calculated as follows (n_i is the number of entries in category i):

$$H = -\sum_{i=1}^{K} \frac{n_i}{N} \ln\left(\frac{n_i}{N}\right)$$

4.9.3.3 Contingency table: chi square test for contingency tables

A contingency table contains information on the association between two categorical variables. Each cell contains the number of entries for a specific combination of row and column categories. For this kind of representation of the data, the obvious question is usually if the information contained in the rows and columns is correlated or not. The null-hypothesis is that there is no association between the rows and columns.

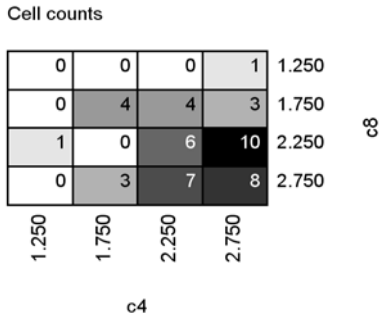


Figure 4-72. Example of a contingency table where intervals of a numerical variable are used to create categories.

If the null-hypothesis is true, the expected count per cell can be calculated. Therefore, we need to know the total number of cells n in the table, $n = n_i n_j$ with n_i the number of rows and n_j the number of columns. The summed numbers of counts in each row and column are called the *marginal row counts* (e.g. N_{rowi} stands for the marginal row count of row i) and *marginal column counts* (N_{colj}). If there is no association between rows and columns, the expected cell count n_{ij} for a cell on row i and column j can be calculated as $n_{ij} = N_{rowi} N_{colj} / N$, with N the total number of entries.

Using these expected cell counts (n_{ij}) and the observed counts per cell (N_{oij}), a *chi square* statistic is calculated, $\chi^2 = \sum_{i=1,j=1}^{n_i,n_j} \left(\left[N_{oij} - n_{ij} \right]^2 / n_{ij} \right)$, with n_i the number of rows and n_j the number of columns.

If the null-hypothesis is true and under certain conditions (see note below), this statistic approximately follows a chi square distribution with $N - n_i - n_j + 1$ *degrees of freedom*. The *p-value* that is returned gives the probability that the statistic is at least as high as the observed one. If the p-value is low, the null-hypothesis can be rejected. The *significance s* of the test can be calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

In case there is a significant association, its strength can be expressed using *Cramer's V*. The formula is $V = \sqrt{\chi^2 / \left[N \min(n_i - 1, n_j - 1) \right]}$, with χ^2 the value for the statistic, N the total number of entries, n_i the number of rows and n_j the number of columns. This gives a value between 0%, in case there is no association, and 100%, in case there is a perfect association. Cramer's V can be used to compare the strengths of different associations.

Values for the various parameters can be found in the test report. The marginal column and row counts are expressed in absolute counts and relative to the total number of counts in the table. How such a chart and report can be created is explained in section 4.9.6.

```
Chi square: 10.337 (9 degrees of freedom)
P value= 0.323868
   Significance= 67.6132%

Cramer's V: 27.08%

Total count: 47
Average cell count: 2.94

Marginal column counts:
  1.250    1   2.13%
  1.750    7  14.89%
  2.250   17  36.17%
  2.750   22  46.81%

Marginal row counts:
  1.250    1   2.13%
  1.750   11  23.40%
  2.250   17  36.17%
  2.750   18  38.30%
```

Figure 4-73. Example of a test report for the chi square test for contingency tables like shown in Figure 4-72.

The contingency table can be displayed showing the residuals for the cells. The residual is a measure for the deviation from the expected number of counts in that cell and is calculated as $\left[N_{oij} - n_{ij} \right] / \sqrt{n_{ij}}$, with N_{oij} the observed cell count and n_{ij} the expected cell count.

NOTE: This test should not be used if the expected average count per category is less than 5. If this is the case, consider combining categories in order to increase the expected average count. In practice, this also means that there should be no empty rows or columns in the contingency table.

4.9.3.4 1-D numerical distribution function: Kolmogorov-Smirnov test for normality

For a sample containing a single quantitative variable, an often recurring question is if it is normally distrib-

uted or not. In this case the null-hypothesis is that the sample is drawn from a normal distribution. The *mean value* $\langle x \rangle$ and *corrected standard deviation* $\sqrt{\sum_{i=1}^{n} (x_i - \langle x \rangle)^2 \Big/ (n-1)}$ (with x_i the observations and n the sample size) are calculated from the sample and are used to determine a normal distribution that can be used as a model (further referred to as model normal distribution) for the underlying distribution of the sample if the null-hypothesis holds.

The Kolmogorov-Smirnov test for normality is applied to test how different the cumulative distribution of the sample is from the cumulative distribution of the model normal distribution. For a sample where each observation is associated with a single number of events, the cumulative distribution $F(x_i)$ gives for each observation (x_i) the total number of events associated to all observations in the sample that are smaller or equal to the observation (x_i). Hence, the cumulative distribution gives at each observation the probability of obtaining that observation or a lower one.

The test statistic is the *maximum difference* in absolute value between the cumulative distribution of the sample and the cumulative distribution of the model normal distribution. In case the null-hypothesis is true and under certain conditions (see note below), the distribution function for this statistic can be calculated approximately. The *p-value* gives the probability that the statistic obtains a higher value than the observed one. If the p-value is low, the null hypothesis can be rejected. The *significance* of the test can be calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.10.

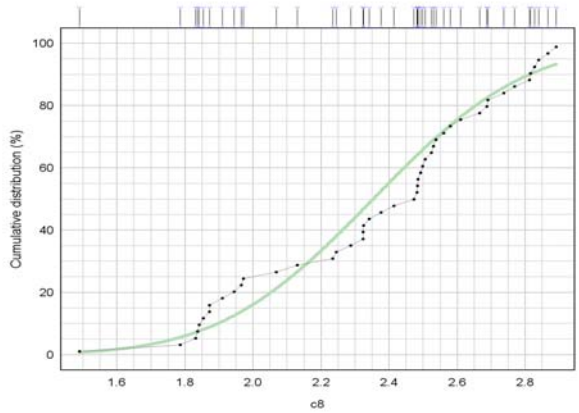


Figure 4-74. Example of a 1-D numerical distribution and model normal distribution.

```
Mean: 2.352766
Corrected standard deviation: 0.357346
Maximum difference: 0.1413
P value= 0.282993
  Significance= 71.7007%
```

Figure 4-75. Example of a test report for the Kolmogorov-Smirnov test for normality applied to a 1-D numerical distribution as shown in Figure 4-74.

NOTES:

(1) The Kolmogorov-Smirnov test for normality should not be used if the number of data points is smaller than 4. The test becomes more accurate if more data points are used.

(2) This test cannot be used to prove that a sample follows a normal distribution, since its aim is only to reject the null-hypothesis with a certain level of significance.

4.9.3.5 2-D scatterplot

Scatterplots contain information on two quantitative variables that are obtained for a set of entries. The position of each dot on the plot is determined by the observations. A scatterplot is dealing with **paired** data since a specific pair of observations characterizes each entry that is represented in the plot.

For this kind of plot one could ask (1) if the means are significantly different or (2) if there is any correlation between the two variables. For both questions, there is a parametric and a non-parametric test available.

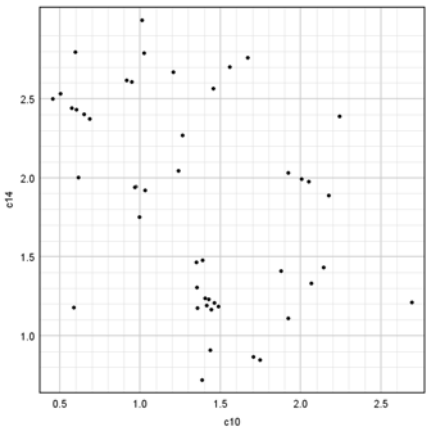


Figure 4-76. Example of a 2-D scatterplot.

4.9.3.5.1 Parametric test for means: T test

The null-hypothesis is that the two samples have the same mean values. Assume the sample observations are

x_i and y_i (i=1,…,n), with $\langle x \rangle$ and $\langle y \rangle$ the respective *mean values* and $s_x = \sum_{i=1}^{n} (x_i - \langle x \rangle)^2 \Big/ (n-1)$ and $s_y = \sum_{i=1}^{n} (y_i - \langle y \rangle)^2 \Big/ (n-1)$ the *corrected variances*.

For paired data, it is generally not guaranteed that all entries have a completely independent pair of observations. The test statistic should be corrected for the influences this may have on the variance of the observations. Therefore, the *corrected covariance Cov* of the sample, $Cov(x,y) = \left(\sum_{i=1}^{n} (x_i - \langle x \rangle)(y_i - \langle y \rangle) \right) \Big/ (n-1)$, is taken into account. The sample variance can be expressed by means of the *pooled corrected standard deviation* s_d. In this case, s_d can be calculated as $s_d = \sqrt{(s_x + s_y - 2Cov(x,y))/n}$.

A statistic is defined as $T = (\langle x \rangle - \langle y \rangle)/s_d$. If the null-hypothesis holds and under certain conditions (see note below) this statistic follows a t distribution with n-1 *degrees of freedom*. The *p-value* gives the probability that the statistic indeed has the observed value or higher. If the p-value is small, the null-hypothesis can be rejected. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.7.

```
Mean values:
  c10    1.3396
  c14    1.8512
Corrected variances:
  c10    0.2870
  c14    0.4246
Corrected covariance = -0.1476
Pooled corrected standard deviation =
0.1464

T = -3.495 (46 degrees of freedom)
P value= 0.001060
  Significance= 99.8940%
```

Figure 4-77. Example of a test report for the T test applied to a 2-D scatterplot like in Figure 4-76.

NOTES:

(1) This test should not be used if the data points are not normally distributed. In this case the Wilcoxon signed-rank test can be used.

(2) This test should not be used if the variances of the two samples are not the same.

4.9.3.5.2 Non-parametric test for means: Wilcoxon signed-rank test

The null-hypothesis is that the two samples have the same mean values. Assume the sample observations are x_i and y_i (i=1,…,n). The absolute values of the differences of these observations $|d_i| = |x_i - y_i|$ are ranked (zero values are eliminated from the analysis). As a first step, these ranks are assigned to rank variables R_i. Afterwards, these R_i get the sign of corresponding d_i. These two steps turn the R_i into ranks of positive or negative differences. The *sum of ranks of positive differences* (sum of all positive R_i) and the *sum of ranks of negative differences* (absolute value of the sum of all negative R_i) are determined and the smallest of these sums is called the Wilcoxon T test statistic.

If the null-hypothesis holds, the expected value for T is $n(n-1)/4$ (with n the number of pairs of observations), while the expected standard deviation on T is $\sqrt{n(n+1)(2n+1)/24}$. Hence, if the null-hypothesis holds and under certain conditions (see note below) the statistic defined as $(T - [n(n-1)/4]) / \sqrt{n(n+1)(2n+1)/24}$ approximately follows a normal distribution. The *p-value* gives the probability that the statistic is at least as high as the observed one. If the p-value is low, the null hypothesis can be rejected. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.7.

```
Sum of ranks of positive differences= 303.0
Sum of ranks of negative differences= 825.0

P value= 0.005746 (Normal approximation)
  Significance= 99.4254%
```

Figure 4-78. Example of a test report for the Wilcoxon signed-rank test applied to a 2-D scatterplot like in Figure 4-76.

NOTES:

(1) This test should not be used if the population distribution is not symmetric.

(2) The approximation by using a normal distribution is only valid if the sample contains more than 20 observations.

4.9.3.5.3 Parametric test for correlations: Pearson correlation test

The null hypothesis is that there is no linear relationship between the sample variables. Assume the observations in the sample are x_i and y_i (i=1,…,n), with $\langle x \rangle$ and $\langle y \rangle$ the *mean values*, $s_x = \sum_{i=1}^{n} (x_i - \langle x \rangle)^2 \Big/ n$ and $s_y = \sum_{i=1}^{n} (y_i - \langle y \rangle)^2 \Big/ n$ the *variances* and $Cov(x,y) = \left(\sum_{i=1}^{n} (x_i - \langle x \rangle)(y_i - \langle y \rangle) \right) \Big/ n$ the *covariance* of the sample. *Pearson's correlation r* is calculated as $r = Cov(x,y) \Big/ \sqrt{s_x s_y}$.

If the null-hypothesis holds and under certain conditions (see note below) the statistic defined as $|r| \sqrt{n-2} \Big/ \sqrt{1-r^2}$ approximately follows a t distribution with n-2 *degrees of freedom*. Since $|r|$ is used to calculate the statistic, the *p-value* can be calculated using a single tail of the t distribution. The p-value gives the probability that the statistic obtains a value at least as high as the observed one. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1-p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.7.

```
Mean values:
  c10     1.3396
  c14     1.8512
Variances:
  c10     0.2809
  c14     0.4156
Covariance= -0.1445

Pearson correlation= -42.288%
P value (single tail)= 0.001531 (T test
approximation)
   Significance= 99.8469%
```

Figure 4-79. Example of a test report for the Pearson correlation test applied to a 2-D scatterplot like in Figure 4-76.

In case there is a significant linear correlation, Pearson's *r* can be used to indicate its strength. A positive value for Pearson's *r* is associated with a positive correlation and would result in a regression line with positive slope. A negative value for Pearson's *r* is associated with a negative correlation and would result in a regression line with negative slope.

If the samples contain less than 30 observations, an alternative way for testing the null-hypothesis is offered by Monte-Carlo simulations. To do this, 10.000 samples with n pairs of randomly distributed observations are created. For each of these samples, a value for the statistic is obtained and is compared to the observed value. The *p-value* from the simulations is determined by the number of times the simulations give a larger value for the statistic than the value observed in the real sample. Also here, the *significance* is calculated as $s = 100 \times (1-p)$. The results for the simulated p-value and significance also appear in the test report.

NOTE: This test should not be used if the distributions of x_i or y_i have strong wings or if they are not normally distributed. However, this test is acceptable for sufficiently large samples.

4.9.3.5.4 Non-parametric test for correlations: Spearman rank-order correlation test

The null-hypothesis is that there is no linear correlation between the sample rank variables, or equivalently that there is no monotonic relation between the sample variables. The sample observations x_i and y_i (i=1,…,n) are replaced by their rank after ordering them from smallest to largest. This results in a sample of ranks R_i and S_i (i=1,…,n). The *Spearman rank-order correlation* coefficient is defined as $r_s = Cov(R,S) \Big/ \sqrt{s_R s_S}$, with $s_R = \sum_{i=1}^{n} (R_i - \langle R \rangle)^2 \Big/ n$ and $s_S = \sum_{i=1}^{n} (S_i - \langle S \rangle)^2 \Big/ n$ the *rank variances*, $Cov(R,S) = \sum_{i=i}^{n} (R_i - \langle R \rangle)(S_i - \langle S \rangle) \Big/ n$ the *rank covariance* and $\langle R \rangle$ and $\langle S \rangle$ the *rank mean values* of the rank variables R_i and S_i respectively.

The null-hypothesis can be tested using the statistic $|r_s|\sqrt{n-2}\Big/\sqrt{1-r_s{}^2}$. If the null-hypothesis holds, this statistic approximately follows a t distribution with n-2 *degrees of freedom*. Since $|r_s|$ is used to calculate the statistic, the *p-value* can be calculated using a single tail of the t distribution. The p-value gives the probability that the statistic obtains a value at least as high as the observed one. In this case, a single tail test is performed. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.7.

```
Rank mean values:
  c10   24.0000
  c14   24.0000
Rank variances:
  c10  184.0000
  c14  184.0000
Rank covariance= -77.1489

Spearman rank-order correlation= -41.929%
P value (single tail)= 0.001675 (T test
approximation)
   Significance= 99.8325%
```

Figure 4-80. Example of a test report for the Spearman rank-order correlation test applied to a 2-D scatterplot like in Figure 4-76.

If the samples contain less than 30 observations, an alternative way for testing the null-hypothesis is offered by Monte-Carlo simulations. To do this, 10.000 samples with *n* pairs randomly distributed observations are created. For each of these samples, a value for the statistic is obtained and is compared to the observed value. The *p-value* from the simulations is determined by the number of times the simulations give a larger value for the statistic than the value observed in the real sample. Also here, the *significance* is calculated as $s = 100 \times (1 - p)$. The results for the simulated p-value and significance also appear in the test report.

4.9.3.6 ANOVA plot

This kind of plot presents a categorical and quantitative variable. The categorical variable splits the sample in a number of groups while the quantitative variable describes a distribution within each group. This kind of data is called **unpaired**.

A typical question is whether the groups have the same average for the quantitative variable. In case there are only two groups for the categorical variable, the parametric T test or the non-parametric Mann-Whitney test can be applied. If there are three or more groups for the categorical variable, the parametric F test or the non-parametric Kruskal-Wallis test can be applied.

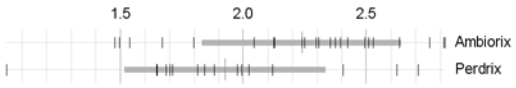


Figure 4-81. Example of an ANOVA plot with two categorical variables.

4.9.3.6.1 Parametric test for two groups: T test

The null-hypothesis is that the two groups have the same mean values. Assume the sample group observations are x_i (*i=1,…,n*) and y_j, (*j=1,…,m*), with $\langle x \rangle$ and $\langle y \rangle$ the respective *mean values* for the groups. The *pooled corrected standard deviation* is defined as

$$s_d = \sqrt{\left(\sum_{i=1}^{n}(x_i - \langle x \rangle)^2 + \sum_{j=1}^{m}(y_j - \langle y \rangle)^2\right)\left(\frac{1}{n} + \frac{1}{m}\right)\Big/(n+m-2)}$$

A statistic is defined as $T = (\langle x \rangle - \langle y \rangle)/s_d$.

If the null-hypothesis is true and under certain conditions (see note below) this statistic follows a t distribution with n-m-2 *degrees of freedom*. The *p-value* gives the probability that the statistic indeed has the observed value or higher. If the p-value is small, the null-hypothesis can be rejected. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.9.

```
Mean values:
  Ambiorix    2.552
  Perdrix     2.391
Pooled corrected standard deviation =
```

```
0.08509

T = 1.891 (37 degrees of freedom)
P value= 0.066419
   Significance= 93.3581%
```

Figure 4-82. Example of a test report for a T test applied on an ANOVA plot with two categorical variables like in Figure 4-81.

NOTES:

(1) This test should not be used if the data points are not normally distributed.

(2) This test should not be used if the variances of the two samples are not the same.

4.9.3.6.2 Non-parametric test for two groups: Mann-Whitney test

The null-hypothesis is that the two groups have the same median values. Assume the observations in the sample groups are x_i ($i=1,\ldots,n$) and y_j ($j=1,\ldots,m$). All observations are combined into one sample and are ranked. For each group, the *sum of ranks* is determined and the smallest of those sums is taken as the U statistic. If the null-hypothesis holds and under certain conditions (see note below) this statistic approximately follows a normal distribution with mean $nm/2$ and variance $n\,m(m+n+1)/12$. The *p-value* gives the probability that the statistic indeed has the observed value or higher. If the p-value is small, the null-hypothesis can be rejected. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1 - p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.9.

```
Sum of ranks:
  Ambiorix  509.5
  Perdrix  270.5

P value= 0.157560 (Normal approximation)
  Significance= 84.2440%
```

Figure 4-83. Example of a test report for a Mann-Whitney test applied on an ANOVA plot with two categorical variables like in Figure 4-81.

NOTE: This test should not be used if one of the groups contains less than 8 members.

If the sample contains less than 30 observations, an alternative way for testing the null-hypothesis is offered by Monte-Carlo simulations. To do this, 10.000 samples with two groups of *n* and *m* randomly distributed observations are created. For each of these samples, a value for the statistic is obtained and is compared to the observed value. The *p-value* from the simulations is determined by the number of times the simulations give a larger value for the statistic than the value observed in the real sample. Also here, the *significance* is calculated as $s = 100 \times (1 - p)$. The results for the simulated p-value and significance also appear in the test report.

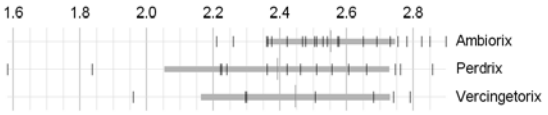


Figure 4-84. Example of an ANOVA plot with two categorical variables

4.9.3.6.3 Parametric test for more than two groups: F test

Assume that the sample contains *g* groups. The null-hypothesis is that all groups have the same mean. The group sizes are given by n_1, n_2, …, n_g, in total *n* observations for the complete sample. The *j*th observation in the *i*th group is denoted as x_{ij}. The sample *group means* are

$\langle x \rangle_{groupi} = \sum_{j=1}^{n_i} x_{ij} \Big/ n_i$, with x_{ij} all observations within group *i*. The mean of all observations is $\langle x \rangle = \sum_{i=1}^{g} \langle x \rangle_{groupi} \Big/ g$,

The *total sum of squares*, $SST = \sum_{i=1}^{g} \sum_{j=1}^{n_i} \left(x_{ij} - \langle x \rangle \right)^2$, is a measure for the variation in the sample around the mean of all observations. The *sum of squares among groups* $SSA = \sum_{i=1}^{g} n_i \left(\langle x \rangle_{groupi} - \langle x \rangle \right)^2$ measures the variation among the group means. The *total within-group sum of squares* $SSW = \sum_{i=1}^{g} \sum_{j=1}^{n_i} \left(x_{ij} - \langle x \rangle_{groupi} \right)^2$ gives the variation in the sample within the groups. From the definitions it is clear that *SST=SSA+SSW*.

If the null-hypothesis holds and under certain conditions (see note below) the statistic

$F = SSA(n-g)/SSW(g-1)$ approximately follows an F-distribution with g-1 and n-g *degrees of freedom*.

The *p-value* gives the probability that the statistic obtains a value at least as high as the observed one. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1-p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.9.

```
SST=    3.347
SSA=    0.255
SSW=    3.092

F= 1.814 (2;44 degrees of freedom)
P value= 0.174938 (F approximation)
  Significance= 82.5062%
P value= 0.175700 (Simulated)
  Significance= 82.4300%

Group means:
       Ambiorix  2.552
       Perdrix   2.391
       Vercingetorix   2.446
```

Figure 4-85. Example of a test report for an F test applied to an ANOVA plot with more than two categorical variables like in Figure 4-84.

In case the sample contains less than 30 observations, an alternative way for testing the null-hypothesis is offered by Monte-Carlo simulations. To do this, 10.000 samples with g groups and $n_1, n_2, \ldots, n_g$ randomly distributed observations in the groups are created. For each of these samples, a value for the F statistic is obtained and is compared to the observed value. The *p-value* from the simulations is determined by the number of times the simulations give a larger value for the F statistic than the value observed in the real sample. Also here, the *significance* is calculated as $s = 100 \times (1-p)$. The results for the simulated p-value and significance also appear in the test report.

4.9.3.6.4 Non-parametric test for more than two groups: Kruskal-Wallis test

Assume the sample contains g groups. The null-hypothesis is that all groups have the same median. The number of observations in the groups are given by $n_1, n_2, \ldots, n_g$, with n the total number of observations. All observations are ranked, the rank for the jth observation in the ith group is denoted by R_{ij} and R_i stands for the *group rank sum* of group i.

A statistic is defined as:

$$H = \left[\frac{12}{n(n+1)} \sum_{i=1}^{g} \frac{R_i}{n_i} \right] - 3(n+1).$$

If the null-hypothesis holds and under certain conditions (see below) the statistic approximately follows a chi-square distribution with g-1 *degrees of freedom*.

The *p-value* gives the probability that the statistic obtains a value at least as high as the observed one. The *significance* of the test is calculated as the complement of the p-value, $s = 100 \times (1-p)$.

The values for the parameters can be found in the test report. How such a chart and report can be created is explained in section 4.9.9.

```
H= 2.377 (2 degrees of freedom)
P value= 0.304666 (Chi square approxima-
tion)
   Significance= 69.5334%
P value= 0.312600 (simulated)
   Significance= 68.7400%

Group rank sums:
       Ambiorix 623.5
       Perdrix 328.5
       Vercingetorix 176.0
```

Figure 4-86. Example of a test report for the Kruskal-Wallis test applied to an ANOVA plot with more than two categorical variables like in Figure 4-84.

NOTES:

(1) In case there are only 3 groups, this test should not be used if one of the groups contains less than 6 observations.

(2) In case there are more than 3 groups, this test should not be used if one of the groups contains less than 5 observations.

If the sample contains less than 30 observations, an alternative way for testing the null-hypothesis is offered by Monte-Carlo simulations. To do this, 10.000 samples with g groups and $n_1, n_2, \ldots, n_g$ randomly distributed observations in the groups created. For each of these samples, a value for the H statistic is obtained and is compared to the observed value. The *p-value* from the simulations is determined by the number of times the simulations give a larger value for the H statistic than the one observed in the real sample. Also here, the *signif-*

icance is calculated as $s = 100 \times (1 - p)$. The results for the simulated p-value and significance also appear in the test report.

4.9.4 Using the plot tool

The plot and statistics tools are available directly from the *FPQuest main* window or from the *Comparison* window. In the *FPQuest main* window, it can be started using ***Comparison > Chart / Statistics***. When launched from the *FPQuest main* window, it works on the current selection made in the database. If launched in the *Comparison* window, it works on all entries contained in the comparison.

4.9.4.1 In the *Comparison* window, click the  button or select ***File > Chart / statistics***. This pops up a dialog box (see Figure 4-87) that is used to select the plot components. ***All components*** that can be included in a chart are listed on the left.

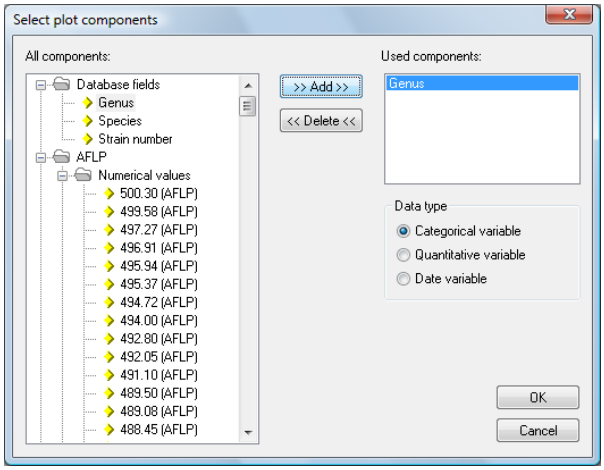


Figure 4-87. The *Select plot components* dialog box, that appears when the chart tool is started, is used to select the plot components for the chart.

4.9.4.2 To add a component to the chart, select a component from this list by clicking on it and add it to the list of ***Used components*** (displayed at the right) with the button ***<Add>***. Also in this list, components can be clicked for selecting them. The selected component can be removed from the ***Used components*** list with the button ***<Delete>***. For the selected component, the panel beneath the ***Used components*** list displays what data type it is.

4.9.4.3 Within this *Select plot components* dialog box you can convert a quantitative variable into an interval variable by checking the ***Convert to interval data*** checkbox. When this option is checked, the ***Interval size*** has to be specified. See lower right part of the panel displayed in Figure 4-87. The same procedure has to be followed if a ***Date variable*** has to be converted to an interval variable: four choices appear in a drop-down box: ***Group by day***, ***Group by month***, ***Group by quarter***, and ***Group by year***.

4.9.4.4 For this example, select one numerical variable. After clicking the ***<OK>*** button, the chart appears, as in Figure 4-88. In this section, the general features and appearances of the *Chart and Statistics* window are discussed. The content of the plot will be discussed in sections 4.9.5 - 4.9.11.

4.9.4.5 To copy the plot of this window select either ***File > Copy to clipboard (metafile)*** or ***File > Copy to clipboard (bitmap)***. A paper copy can be obtained by selecting ***File > Print***.

4.9.4.6 For some type of charts, you can export the data by selecting ***File > export data (formatted)*** or ***File > export data (TAB delimited)***. These menu items appear in grey instead of black if they cannot be applied for the current type of chart.

4.9.4.7 Selecting ***Plot > Edit components*** or clicking the  button pops up the *Select plot components* dialog box (see Figure 4-87). This can be used to change the ***Used components***. If the list of ***Used components*** is modified, it is possible that the plot changes into another type of chart because the chart functionality selects the optimal representation for a given set of variables. Of course it is possible to select another type of chart (see 4.9.4.8).

4.9.4.8 From the ***Plot*** menu item, another type of chart can be selected. The same options are also available from the toolbar, which is displayed vertically on the left side of the window in default configuration (see Figure 4-88). Its position can be modified as described in 1.6.5. The toolbar has following buttons: the *Display bar graph* button  , the *Display 2D contingency table* button  , the *Display 2D scatterplot* button  , the *Display 3D scatterplot* button  , the *Display ANOVA plot* button  , the *Display 1D distribution function* button  , the *Display 3D bar graph* button  , and the *Display colored bar graph*  . The button for the plot type that is presently shown is highlighted: e.g.  . If the chart type chosen is not compatible with the data type, the message "Invalid type of source data" appears.

4.9.4.9 Zooming in or zooming out on the plot can be done with ***View > Zoom in*** () or ***View > Zoom out*** (). Alternatively, the zoom slider can be used (see 1.6.7 for a description of zoom slider functions).

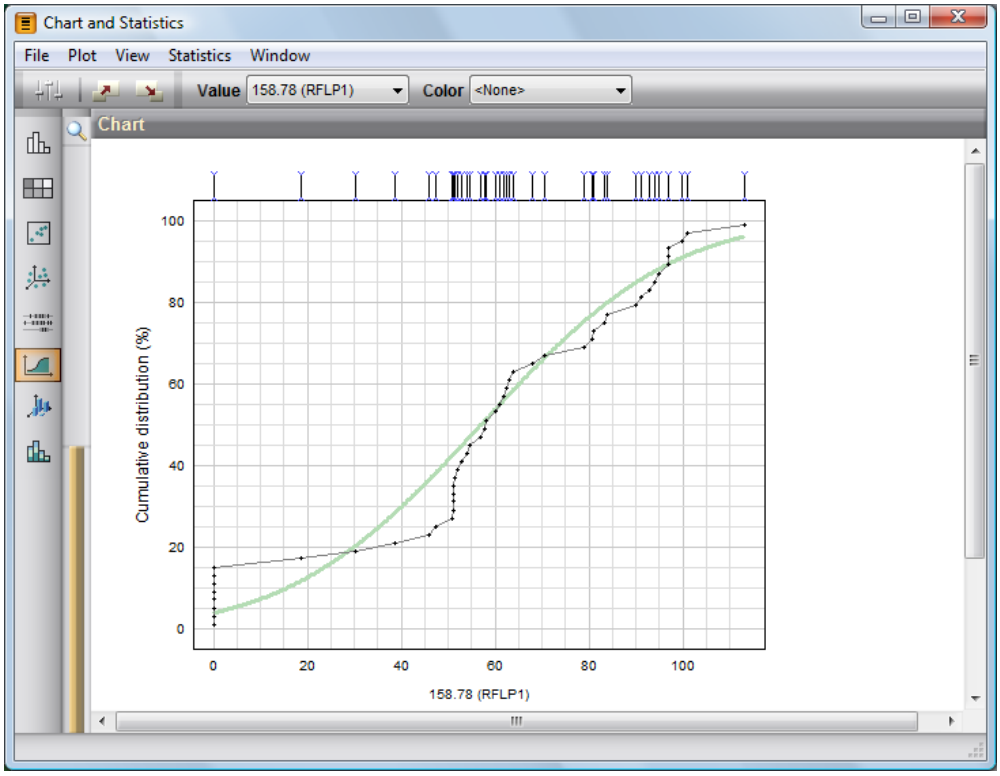


Figure 4-88. The *Chart and Statistics* window, displaying a 1-D numerical distribution function for a single quantitative variable.

The ***View*** menu item is divided into two parts, separated by a horizontal line. The part below the horizontal line contains menu items that change the view of the plot and that generally depend on the kind of chart that is displayed in the window. These commands will be discussed when the various charts are presented.

4.9.4.10 The last menu item is ***Statistics***. Under this item, a list of statistic tests that can be applied to the selected type of plot is given. These tests will be discussed when the various charts are presented.

4.9.4.11 Selections of entries can be made within the chart, except for 3-D bar graphs. These selections are also shown in the *Comparison* window and the *FPQuest main* window. If the selection is changed in the comparison, the chart is updated automatically. If another chart type is selected, the entries keep their selected/unselected state. Selections from the *Comparison* window and the *FPQuest main* window are also visualized in a *Chart and Statistics* window.

In the following sections the various types of charts and their statistics are described.

4.9.5 Bar graph

4.9.5.1 Open the chart tool by clicking  in the *Comparison* window.

4.9.5.2 Select a categorical variable, e.g. an information field and add it to the list of ***Used components***, then press **<*OK*>**.

4.9.5.3 This creates a *Chart and Statistics* window like shown in Figure 4-89. The component that is displayed is indicated beneath the toolbar. In case you selected more than one categorical variable in the ***Used components*** list (4.9.5.2) a drop-down list can be used to display another variable.

4.9.5.4 The entries corresponding to the bars in the chart can be selected (or unselected) by pressing the CTRL key while clicking or dragging the mouse.

4.9.5.5 Select ***Statistics > Chi square test for equal category size.*** This creates a *Statistics report*, as shown in Figure 4-90. A description of this test can be found in 4.9.3.1. The report can be exported by pressing **<*Copy to clipboard*>** and pasting it in another application.

4.9.5.6 With ***Statistics > Index of diversity*** a report window is generated which displays Simpson's index of diversity and the Shannon-Weiner index of diversity for the selected entries and categories. The report can also be copied to the clipboard.

4.9.6 Contingency table

4.9.6.1 Create a *Chart and Statistics* window with two categorical variables. This can be done from the *Compar-*

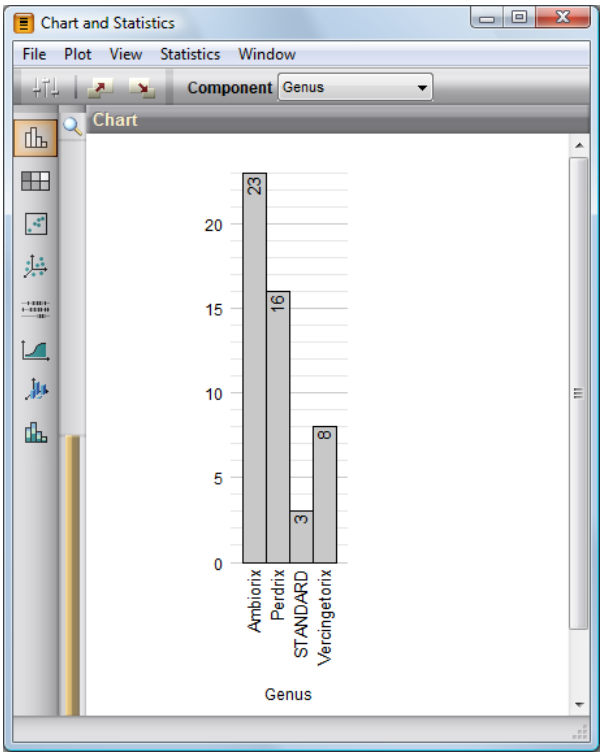


Figure 4-89. A bar graph for one categorical variable.

ison window by clicking  or within the *Chart and Statistics* window by editing the plot components after clicking the  button. Select two categorical variables into the ***Used components*** list. After clicking **<*OK*>**, a contingency table like in Figure 4-91 is created.

4.9.6.2 The contents of the ***X component*** and ***Y component*** are indicated in the window. A drop-down list makes it possible to assign another categorical variable

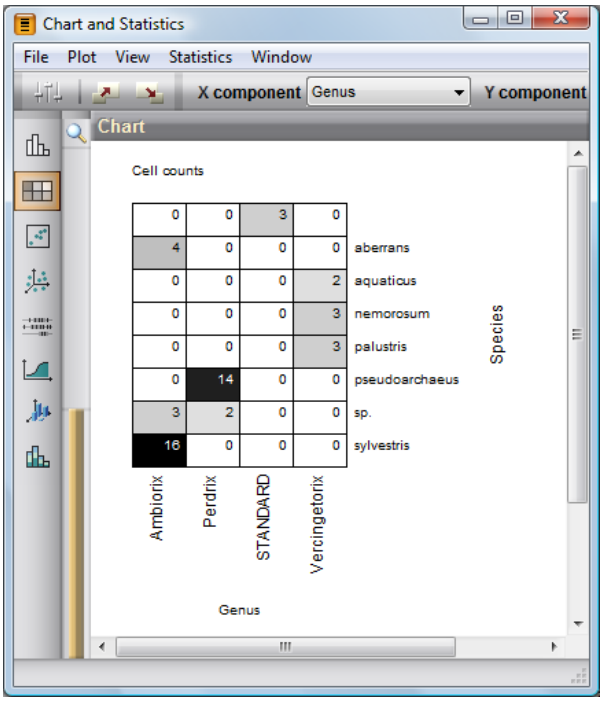


Figure 4-91. A contingency table for two categorical variables.

from the used components list to the ***X component*** and ***Y component***.

4.9.6.3 Cells can be selected (or unselected) in the table by pressing CTRL while left-clicking the mouse (CTRL+click).

The contingency table can be displayed showing row respectively column percentages by selecting ***View > Display row percentages*** respectively ***View > Display column percentages***.

4.9.6.4 The contingency table can be displayed in the *Chart and Statistics* window showing residuals in the

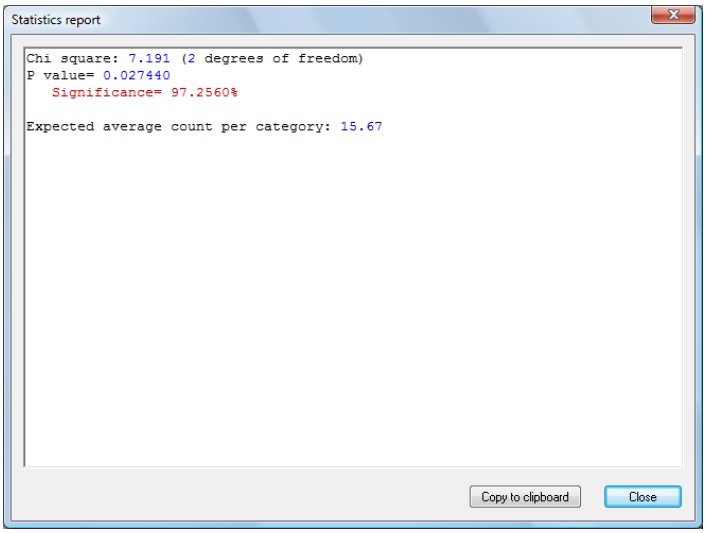


Figure 4-90. *Statistics report* **for Chi square test for equal category sizes.**

cells, with ***View > Display residuals***. The residual for a cell is a measure for the deviation from the expected number of counts in that cell and is calculated as $\left[N_{oij} - n_{ij}\right] / \sqrt{n_{ij}}$, with N_{oij} the observed cell count and n_{ij} the expected cell count. This view is closely related to the statistic test that can be applied to this chart (see 4.9.3.3).

4.9.6.5 Select ***Statistics > Chi square test for contingency tables*** to apply the statistical test that is available for this kind of plot. This creates a *Statistics report*, as shown in Figure 4-92. A description of this test can be found in 4.9.3.3.

4.9.6.6 In the *Chart and Statistics* window, you can create bar graphs for each of the two selected categorical variables by clicking the bar graph button  .

4.9.7 2-D scatterplot

4.9.7.1 Create a *Chart and Statistics* window with two quantitative variables. This can be done from the *Comparison* window by clicking  or within the *Chart and Statistics* window by editing the plot components after clicking the  button. Select two quantitative variables into the ***Used components*** list. After clicking **<*OK*>**, a 2-D scatter plot like in Figure 4-93 is created.

4.9.7.2 The contents of the ***X axis*** and ***Y axis*** are indicated in the toolbar. A drop-down list makes it possible to change the variables displayed on the axes.

4.9.7.3 Single dots can be selected in the chart using CTRL+click. Multiple dots are selected at once by holding the SHIFT key and drawing a rectangle around the dots with the mouse.

4.9.7.4 With the menu command ***View > Regression line***, a regression line can be added to the plot. A *Regression selection* dialog box pops up (Figure 4-94), offering a choice between several types of regression lines. After selecting a regression type and clicking **<*OK*>** in the dialog box, a small statistics report is generated. If a regression line is fitted, it is shown as a thick green line. The 1-sigma uncertainty levels are plotted as a thin green line.

4.9.7.5 Under the menu item ***Statistics***, a number of statistic test can be found: ***T test for mean value (paired samples)***, ***Wilcoxon signed ranks test (paired samples)***, ***Pearson correlation test*** and ***Spearman rank-order correlation test***. Each of these tests generates a statistics test report. A description of these tests can be found in 4.9.3.5.

4.9.7.6 If one or more categorical variables are present in the ***Used components*** list, additional information from one of these variables can be displayed in color code by selecting the variable form the color drop-down list. If

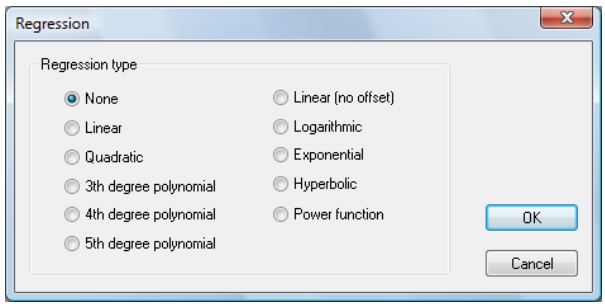


Figure 4-94. The *Regression selection* dialog box, where the type of regression line for the scatter plot can be selected.

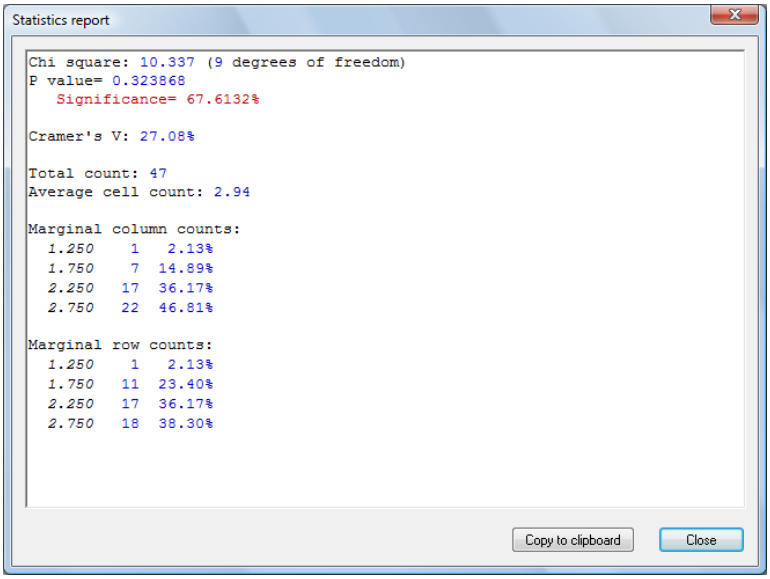


Figure 4-92. *Statistics report* **for the Chi square test for contingency tables.**

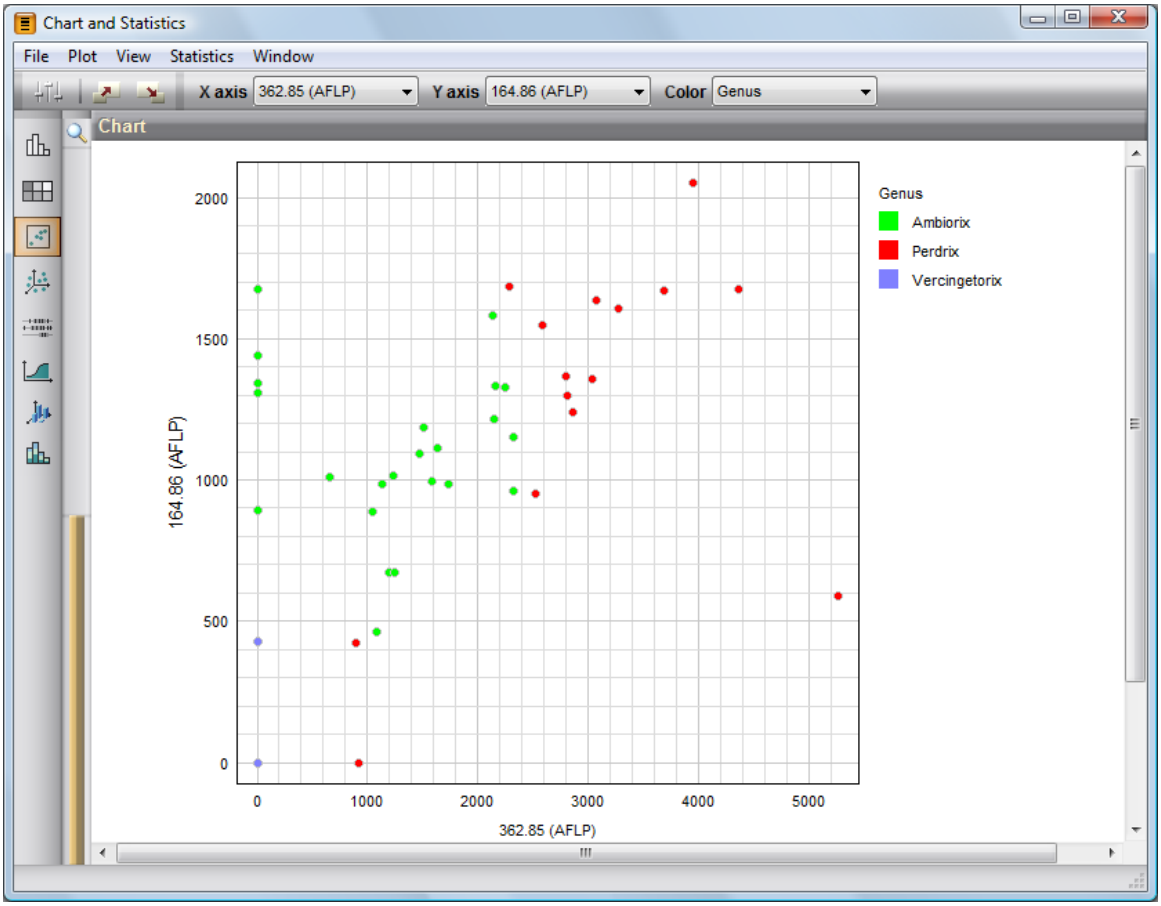


Figure 4-93. 2-D scatterplot for two quantitative variables.

this is the case, you can change the color labels with the command ***View > Label with continuous colors***.

4.9.7.7 For each of the quantitative variables used in this plot, a 1-D distribution function plot can be generated. This can be done by selecting ***Plot > 1D distribution function***, or by clicking the button. For more details on this kind of chart, see 4.9.10.

4.9.8 3-D scatterplot

4.9.8.1 Create a *Chart and Statistics* window with three quantitative variables. This can be done from the *Comparison* window by clicking or within the *Chart and Statistics* window by editing the plot components after clicking the button. Select three quantitative variables into the ***Used components*** list. This will create a 3-D scatterplot.

4.9.8.2 The variables that are displayed on the respective axes are indicated beneath the toolbar. The variables can be switched between the *X axis*, *Y axis* and *Z axis*. A drop-down list makes it possible to assign another quantitative variable from the *Used components* list to the respective axes.

4.9.8.3 Dots can be selected in the chart by CTRL + mouse click or by holding the SHIFT key and drawing a rectangle around the dots with the mouse. The corresponding entries are also selected in the *Comparison* window and the *FPQuest main* window. If they are removed from the comparison, the chart is updated automatically. Selections made in the *Chart and Statistics* window are automatically updated in the *Comparison* window and vice versa.

4.9.8.4 By clicking on the plot and holding the left mouse button, the plot can be rotated in different directions. The data points in the plot can be displayed as small dots or as larger spheres, which can be achieved by checking or unchecking the command ***View > Show rendered spheres***.

4.9.8.5 If one or more categorical variables are present in the ***Used components*** list, additional information from one of these variables can be displayed in color code by selecting the variable form the color drop-down list. If this is the case, you can change the color labels with the command ***View > Label with continuous colors***.

4.9.8.6 For each of the quantitative variables used in this plot, a *1-D distribution function* plot can be generated. This can be done by selecting ***Plot > 1D distribution function***, or by clicking the button. For more details on this kind of chart, see 4.9.10. For each couple

of categorical variables used in this plot, a 2-D scatterplot can be generated. This can be done by selecting ***Plot > 2D scatterplot***, or by clicking the button. For more details on this kind of chart, see 4.9.7.

4.9.9 ANOVA plot

4.9.9.1 Create a *Chart and Statistics* window with one categorical and one quantitative variable. This can be done from the *Comparison* window by clicking or within the *Chart and Statistics* window by editing the plot components after clicking the button. Select one categorical and one quantitative variable for the plot. This creates an ANOVA plot like in Figure 4-95. The data for each category is presented on a horizontal line. The scale for the line is indicated at the top. Each data point is indicated with a small vertical mark at the position according to its numerical value and the category it belongs to.

4.9.9.2 The categorical and quantitative variables that are displayed are indicated in the toolbar. A drop-down list makes it possible to assign other variables from the ***Used components*** list to the respective axes.

4.9.9.3 Vertical marks, indicating the database entries, can be selected in the chart by CTRL + mouse click or by holding the SHIFT key and drawing a rectangle around the marks with the mouse.

4.9.9.4 From the menu item ***Statistics***, the ANOVA test (***F test***), the ***Kruskal-Wallis test*** (in case more than two categorical variables are used), the ***T test*** or the ***Mann-Whitney test*** (in case only two categorical variables are used) can be launched. For these tests a statistics report is generated. A description of these tests can be found in 4.9.3.6.

4.9.9.5 If one or more categorical variables are present in the ***Used components*** list, additional information from one of these variables can be displayed in color code by selecting the variable form the color drop-down list. If this is the case, you can change the color labels with the command ***View > Label with continuous colors***.

4.9.9.6 For the quantitative variable used in this plot, a 1-D distribution function plot can be generated. This can be done by selecting ***Plot > 1D distribution function***, or by clicking the button. For more details on this kind of chart, see 4.9.10. For the categorical variables used in this plot, a bar graph can be generated. This can be done by selecting ***Plot > Bar graph***, or by clicking the button. For more details on this kind of chart, see 4.9.5.

4.9.10 1-D numerical distribution

4.9.10.1 Create a *Chart and Statistics* window with one quantitative variable. This can be done from the *Comparison* window by clicking or within the *Chart and Statistics* window by editing the plot components after clicking the button. Select only one quantitative variable for the plot. This will create a 1-D cumulative distribution function plot as previously shown in Figure 4-88. The dots present the data points, each dot has a corresponding vertical mark just above the chart. The

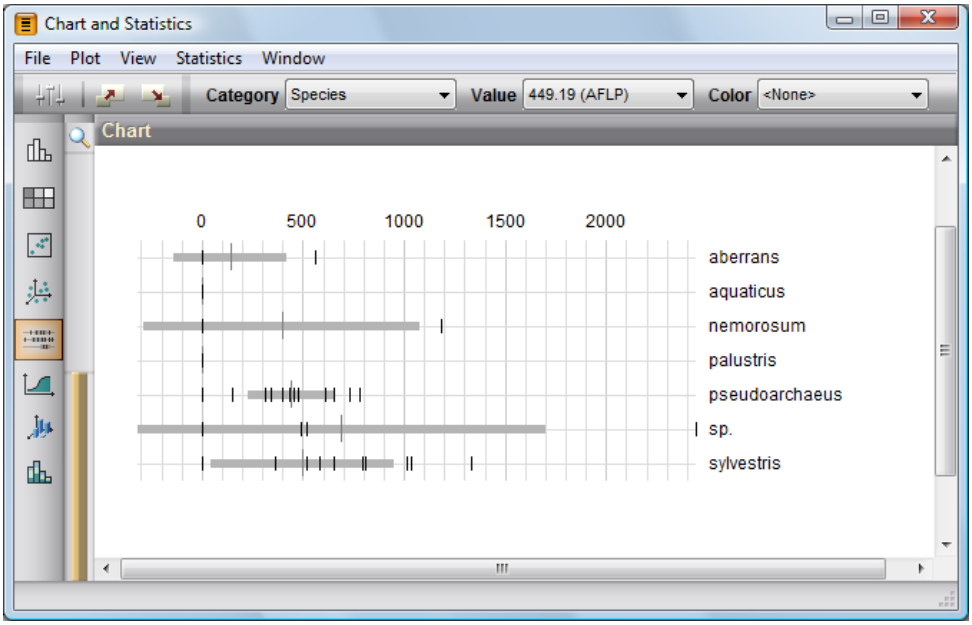


Figure 4-95. ANOVA plot for a categorical and a quantitative variable.

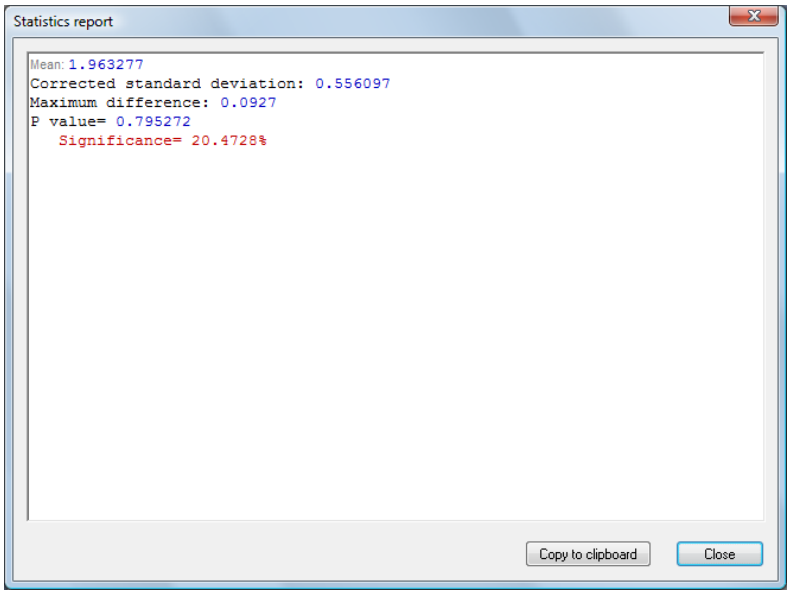


Figure 4-96. *Statistics report* **for a Kolmogorov-Smirnov test.**

smooth green line is the normal distribution that serves as a model for the data.

4.9.10.2 The variable that is displayed is indicated in the toolbar. A drop-down list is available to select another variable in case there is more than one numerical variable in the ***Used components*** list.

4.9.10.3 Data points can be selected in the chart using CTRL+click or by holding the SHIFT key and drawing a rectangle around the vertical marks with the mouse.

4.9.10.4 Select ***Statistics > Kolmogorov-Smirnov test for normality*** for applying the statistical test that is available for this kind of plot. This will create a *Statistics report*, as shown in Figure 4-96. A description of this test can be found in 4.9.3.4..

4.9.10.5 Instead of a cumulative distribution, the data can be presented as bar graph by unchecking the command ***View > Display cumulative distribution***.

4.9.10.6 Additional information from a categorical variable can be displayed in color code. In this case, with the menu item ***View***, you can change the color code into a continuous color code and back.

4.9.11 3-D Bar graph

4.9.11.1 For categorical variables, a 3-D bar graph can be plotted, see Figure 4-97 for an example. This can be done by selecting two categorical variables for the plot and by clicking the  button or by selecting ***Plot > 3D bar graph*** from the menu.

4.9.11.2 Under the menu item ***View***, there is the option to ***Label the X axis in color***, to ***Label the Y axis in color*** or to ***Label with continuous colors***.

4.9.11.3 By clicking on the plot and holding the left mouse button, the plot can be rotated in different directions.

4.9.12 Colored bar graph

4.9.12.1 For categorical variables, a colored bar graph can be generated, see Figure 4-98 for an example. This can be done by selecting two categorical variables for the plot and by clicking the  button or by selecting ***Plot > colored bar graph*** from the menu.

The components used as ***X component*** and ***Color*** are indicated in the toolbar. The drop-down list can be used to display another variable.

4.9.12.2 The entries corresponding to the colored bars in the chart can be selected (or unselected) by pressing the CTRL key while clicking or dragging the mouse.

4.9.12.3 Select ***View > Show percentages*** to scale the colored blocks in a relative fashion.

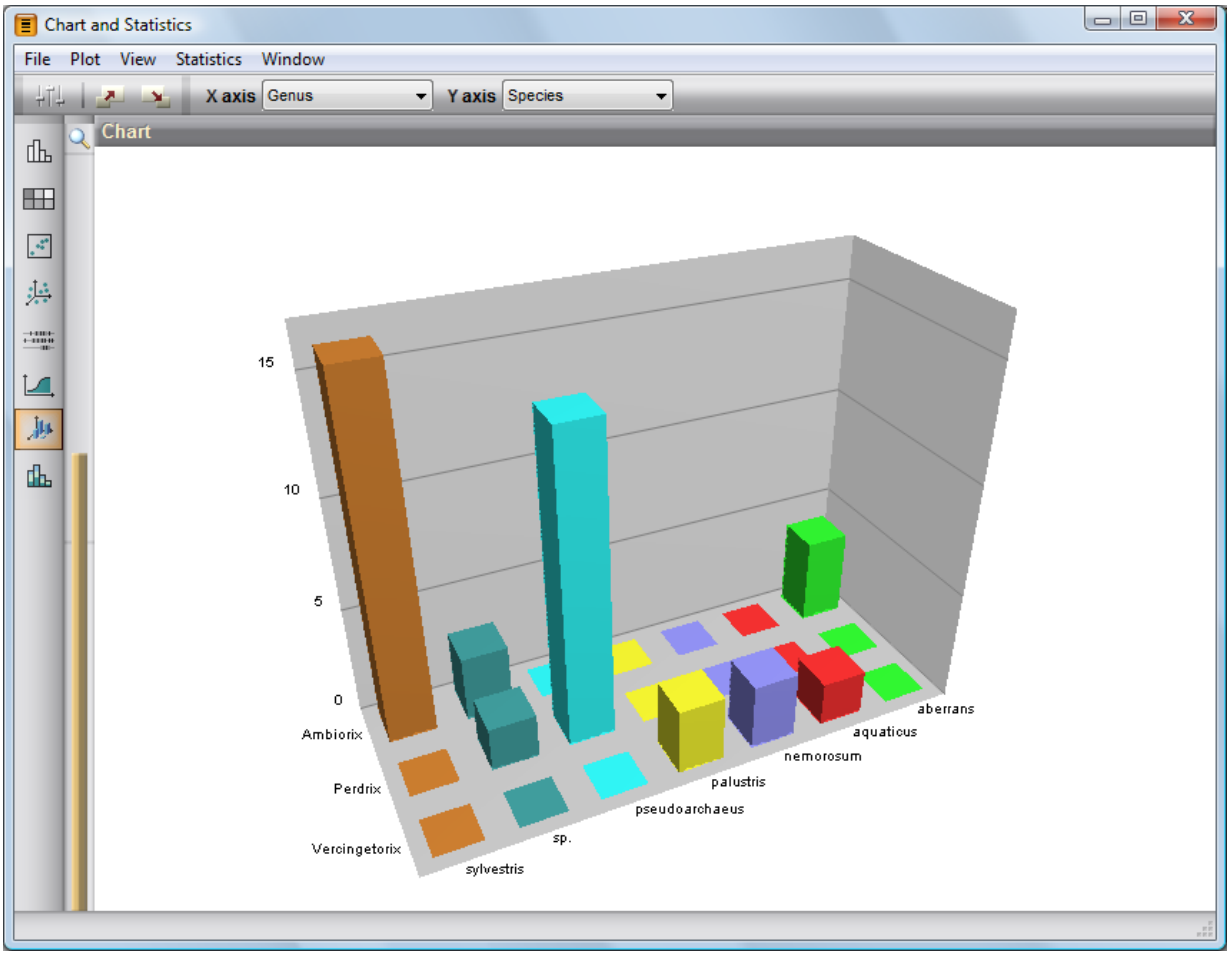


Figure 4-97. 3-D bar graph for two categorical variables.

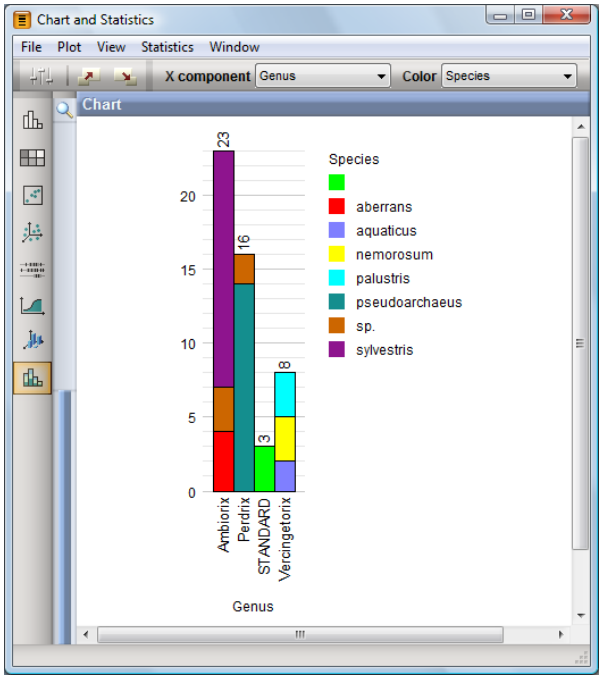


Figure 4-98. Colored bar graph for two categorical variables.

5. IDENTIFICATION

5.1 Identification with database entries

There are two methods for identification available in FPQuest. The most straightforward way (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 Section 5.2).

5.1.1 Creating lists for identification

In FPQuest, a comparison can be used as a "container" for a number of well-characterized database entries that are used to identify against.

5.1.1.1 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. For more information about the automatic search and select functions, see 2.2.9 and 2.2.10.

5.1.1.2 Select ***Comparison > Create new comparison*** (ALT+C) or press the  button in the *Comparisons* panel toolbar to create a comparison with the selected entries.

5.1.1.3 Select ***File > Save as*** or press  to save the comparison (shortcut CTRL+S on the keyboard). Enter **Ambiorix** as name for the comparison.

5.1.1.4 Exit the *Comparison* window.

5.1.2 Identifying unknown entries

First we select the 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**).

5.1.2.1 In the *FPQuest main* window, press F4 to clear the selection.

5.1.2.2 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). For more information about the automatic search and select functions, see 2.2.9 and 2.2.10.

5.1.2.3 Copy the selected entries to the clipboard using ***Edit > Copy selection*** or  .

5.1.2.4 Open the saved comparison **Ambiorix** by double-clicking on **Ambiorix** in the *Comparison* panel.

5.1.2.5 Paste the selected *Ambiorix* sp. entries into the comparison with ***Edit > Paste selection*** or  .

5.1.2.6 For identification purposes, we do not need the *Dendrogram* panel (left, see 4.1.3 and Figure 4-1), which you can minimize.

5.1.2.7 Create sufficient space for the *Similarities* panel (right, see 4.1.3 and Figure 4-2), where the similarity values will appear.

5.1.2.8 In the *Experiments* panel, select an experiment by means of which you want to identify the unknown entries. Select for example **RFLP1**.

5.1.2.9 Click on the first unknown *Ambiorix* sp. entry in the *Information fields* panel. This entry now becomes highlighted.

5.1.2.10 In the menu of the *Comparison* window, choose ***Edit > Arrange entries by similarity***.

The highlighted entry 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 *Similarities* panel.

5.1.2.11 You can click on the  button of **RFLP1** in the *Experiments* panel 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 fingerprint type or composite data set, in order to compare the results. The program uses the similarity coefficient which is specified in the *Experiment type* window (see Chapter 3).

5.1.2.12 A printout of the list of similarity values can be obtained with ***File > Print database fields***.

5.1.2.13 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 (see Figure 4-*

22). Before selecting **Edit > Arrange entries by similarity**, *you should enable* **Layout > Show distances**.

5.1.3 Fast band-based database screening of fingerprints

In case of large databases of fingerprint patterns, the most time-consuming part of a quick database screening of new or unknown patterns is reading or downloading all the fingerprint information. FPQuest offers a tool that overcomes this bottleneck by generating a cache containing band information of all available fingerprints belonging to a fingerprint type. When a database screening is performed, this cache is loaded rather than the full gel information. This cache-based fingerprint screening is extremely fast, even for the largest databases, but is limited to band-based comparisons of fingerprint patterns. In addition, the feature is only available in a connected database environment (see Section 2.3), where a special column holding the quick-access band information is generated (6.1.14). To try out this feature, you can e.g. install the **DemoBase_SQL** database, as described in 1.3.2.

5.1.3.1 The fast band-based identification can be enabled in the *Fingerprint type* window (*Experiments* panel), by selecting ***Settings > Enable fast band matching*** (this menu command appears only in a connected database). A question pops up "**Do you want to generate cached patterns for all current fingerprints?**". By answering ***<Yes>***, a cached pattern will be generated for all patterns present in the database that belong to the selected fingerprint type. If you answer ***<No>***, a cached pattern will be created only for new patterns that are added to the database.

5.1.3.2 The fast band matching identification tool is launched from the *FPQuest main* window, where a set of selected entries will be identified against all other database entries.

NOTE: For the fast band matching identification tool to work, metrics information (molecular weight regression) needs to be available for the active reference system of the selected fingerprint type (see 3.1.10).

5.1.3.3 A menu command ***Identification > Fast band matching*** (only in a connected database) pops up the *Fast band matching* dialog box (Figure 5-1). Under ***Experiment type***, select the fingerprint type you want to use for the band matching. With ***Used range***, you can specify a range of the pattern (in percentage distance from top) within which bands will be compared. The ***Tolerance*** is the same as the position tolerance explained in 4.3.2. With ***Maximum difference***, you can specify the maximum number of different bands between the unknown pattern and a database pattern to be included in the result set. Furthermore, the ***Result set*** can be limited to a certain number (default 20). In the input box ***SQL query***, it is possible to enter an SQL query, to limit the search to a subset of entries that match a specific string entered for an information field.

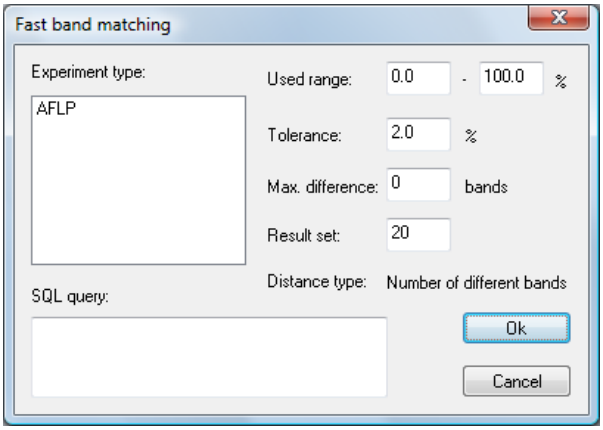


Figure 5-1. The *Fast band matching* dialog box.

The typical syntax of a restricting SQL query is:

```
"GENUS"='Ambiorix'
```

One can also combine statements, for example:

```
"GENUS"='Ambiorix' AND "SPECIES"='sylvestris'
"GENUS"='Ambiorix' OR 'Perdrix'
```

5.1.3.4 By pressing ***<OK>***, the fast band matching is executed and the identification result pops up in the *Fast matching* window (Figure 5-2). This window is subdivided in two dockable panels, of which the *Entries* panel lists the entries to be identified, and the *Matches* panel lists the result set for the selected entry in the *Entries* panel (for display options of dockable panels, see 1.6.4). The only matching criterion used is the number of different bands, which is listed in the 'Distance' column of the *Matches* panel.

NOTES:

(1) In cases where matching patterns are identical, there may be a small decimal distance. For each identical match, the software uses the band pair with the highest shift and adds this shift value to the match (i.e. to zero). This is an additional feature to sort identical patterns according to distance based upon shifts within the defined position tolerance.

(2) In the Entries *panel and* Matches *panel of the* Fast matching *window, the same information fields are displayed as in the* Database entries *panel of the* FPQuest main *window. To display or hide other information fields in a panel, click on the column properties button*  *in the information fields header.*

5.1.3.5 In both panels of the *Fast band matching* window, you can select or unselect entries using the mouse in combination with the SHIFT or CTRL keys. You can also pop up the *Entry edit* window by double-clicking on an entry or pressing ENTER.

5.1.3.6 A text report can be exported with ***File > Export*** or by pressing the  button. A tab-delimited text file is opened in Notepad, where the matched entries are listed together with the best matching database entries, sorted according to number of different bands.

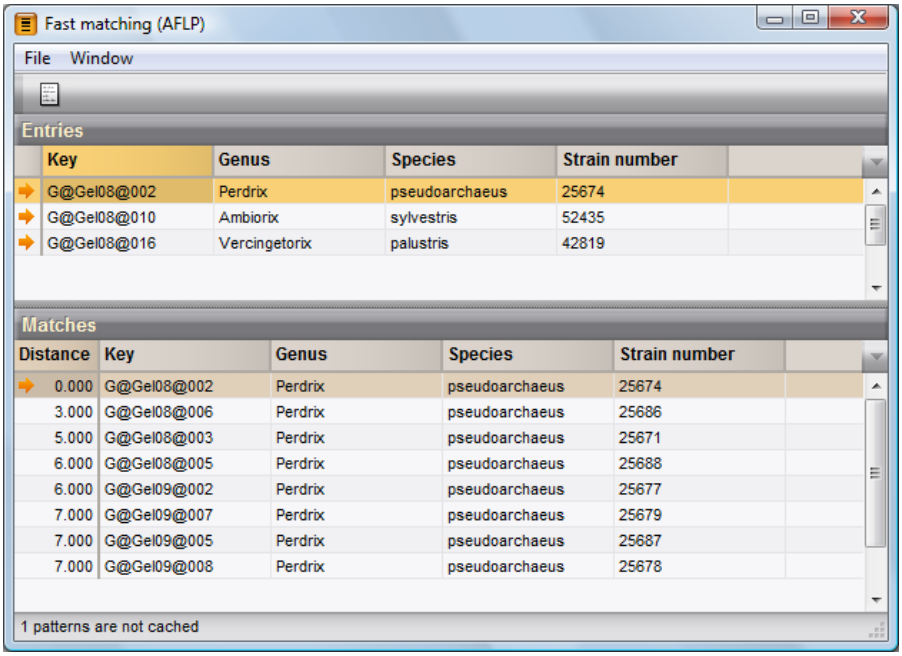


Figure 5-2. The *Fast matching report* window.

5.2 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 a system for identification, a new library is first created. Then, library units are defined within that library, to which the names of the taxa are given. Within each library unit, a selection of representative entries for that taxon is entered.

5.2.1 Creating a library

5.2.1.1 In the *FPQuest main* window with **DemoBase** loaded, select ***Identification > Create new library*** from the menu or press  in the toolbar of the *Libraries* panel.

In case of a connected database (see Section 2.3), a dialog box allows you to choose whether you want to store the library in the ***Local database*** (file-based), or in the ***Connected database*** (Figure 5-3). In the latter case, other users that are connected to the same database will be able to use the library too.

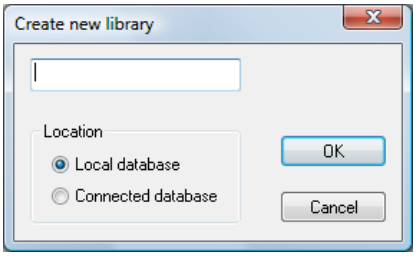


Figure 5-3. The *Create new library* dialog box.

5.2.1.2 Enter a name for the library, for example **DemoLib**.

The *Library* window of the new library appears (Figure 5-4). The *Experiments* panel (left in default configuration) shows the available experiments and the *Units* panel (right in default configuration) shows the library units defined within the library. Both panels are dockable (see 1.6.4 for display options). The layout of the *Experiments* panel can be modified by clicking on the column properties button  in the information fields header. The *Units* panel is initially empty.

Within the library, you can include or exclude experiments. Excluded experiments will not be used for identification.

5.2.1.3 Select an experiment which you do not want use for identification, for example a composite data set.

5.2.1.4 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.

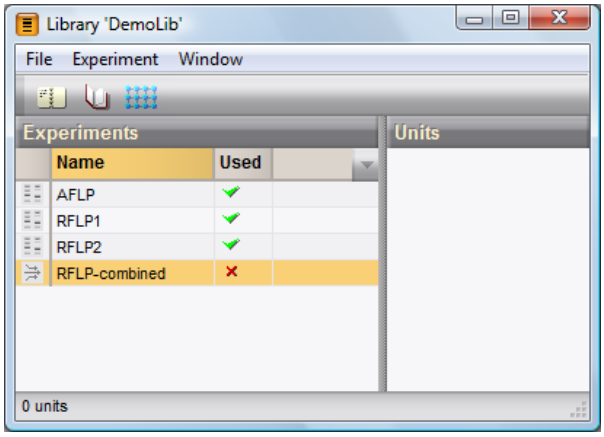


Figure 5-4. The *Library* window of a new library.

5.2.1.5 Select ***File > Add new library unit*** or  .

5.2.1.6 Enter a name of one of the species in the database, for example *Ambiorix sylvestris*.

The library unit now shows up in the *Units* panel.

5.2.1.7 Double-click on the unit, or click on it and select ***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 (see Section 4.1). This allows you to cluster the members of a library unit internally in order to check the homogeneity of a defined taxon.

5.2.1.8 In the database, select all *Ambiorix sylvestris* entries and copy them to the clipboard using ***Edit > Copy selection*** or  .

5.2.1.9 Paste the entries in the library unit with ***Edit > Paste selection*** or  .

5.2.1.10 Save the library unit with ***File > Save*** or  .

5.2.1.11 Repeat 5.2.1.5 to 5.2.1.10 to create library units for the other named species.

5.2.1.12 When finished, close the library with ***File > Exit***.

The library is now listed in the *Libraries* panel of the *FPQuest main* window. You can open the library and add or edit units whenever desired.

5.2.2 Identifying entries against a library

5.2.2.1 In the *FPQuest main* window, clear any selected entries in the database with F4.

5.2.2.2 Select a list of entries, for example all unnamed species (*Ambiorix* sp. and *Perdrix* sp.) and a few entries of the other species.

5.2.2.3 Click on **DemoLib** in the *Libraries* panel and select ***Identification > Identify selected entries***.

A dialog box appears, as shown in Figure 5-5. Under Method, you can choose between ***Mean similarity***, ***Maximum similarity***, ***K-Nearest Neighbor*** and ***Neural Network*** (if available; see 5.2.3).

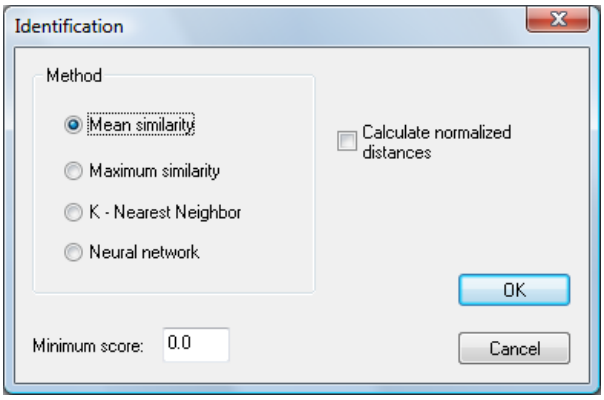


Figure 5-5. The *Identification* dialog box with the Mean similarity option selected.

5.2.2.4 With the option ***Mean similarity***, the program calculates a similarity between the unknown entry and each entry in the library unit, and then calculates the average similarity for the entire library unit. These average similarities are then used in the identification report.

5.2.2.5 With the option ***Maximum similarity***, the program will also calculate all similarities between the unknown and the library unit entries, but only the highest similarity value found is used in the identification report.

5.2.2.6 If ***Mean similarity*** or ***Maximum similarity*** is selected, an option ***Calculate normalized distances*** becomes available.

The *Normalized distance* is an indication for 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.

5.2.2.7 With the option ***K - Nearest Neighbor***, the user has to specify a value K, which is a number of entries from the whole library having the highest similarity with the unknown. Suppose that 10 is entered for K, the 10 best matching entries from the whole library will be retained. The library unit having the largest number of entries belonging to these K nearest neighbors is considered the best matching, and gets the highest score. The score is simply the number of entries of the library unit that belong to the K nearest neighbors.

5.2.2.8 If ***K - Nearest Neighbor*** is selected, an input field ***K value*** becomes available, where you can enter the number of nearest neighbors to look for.

> *NOTE: The value for K is supposed to be smaller than the number of entries contained in each of the library units. If this is not the case, the program will warn you for this conflict when the identification is executed.*

5.2.2.9 The ***Neural network*** option is explained in detail in 5.2.3. If this option is checked, a drop-down list becomes available, showing the existing neural networks, from which you can choose one.

5.2.2.10 Optionally, a ***Minimum score*** can be specified. If a library unit has a score that is lower than the minimum score specified, the library unit will be grayed in the *Identification* window and will not be listed in the identification report. Obviously, the score depends on the method selected. If a similarity method is selected, the score should be a floating value between 0 and 100; if ***K - Nearest Neighbor*** is selected, the value should be an integer value between 0 and K.

5.2.2.11 Click ***Mean similarity***, check ***Calculate normalized distances***, and press ***<OK>***.

The *Identification* window appears, showing the progress of the calculations in the progress bar in the bottom of the window. Once the calculations are done, the window is divided in three panels (Figure 5-6). The *Unknowns* panel (left) lists the unknown entries that you have selected for identification. The *Matches* panel (right) lists for each experiment type (organized in columns) the library unit that matches best with the unknowns. The dockable *Details* panel (bottom panel in default configuration) shows the identification details for the highlighted unknown/experiment type combi-

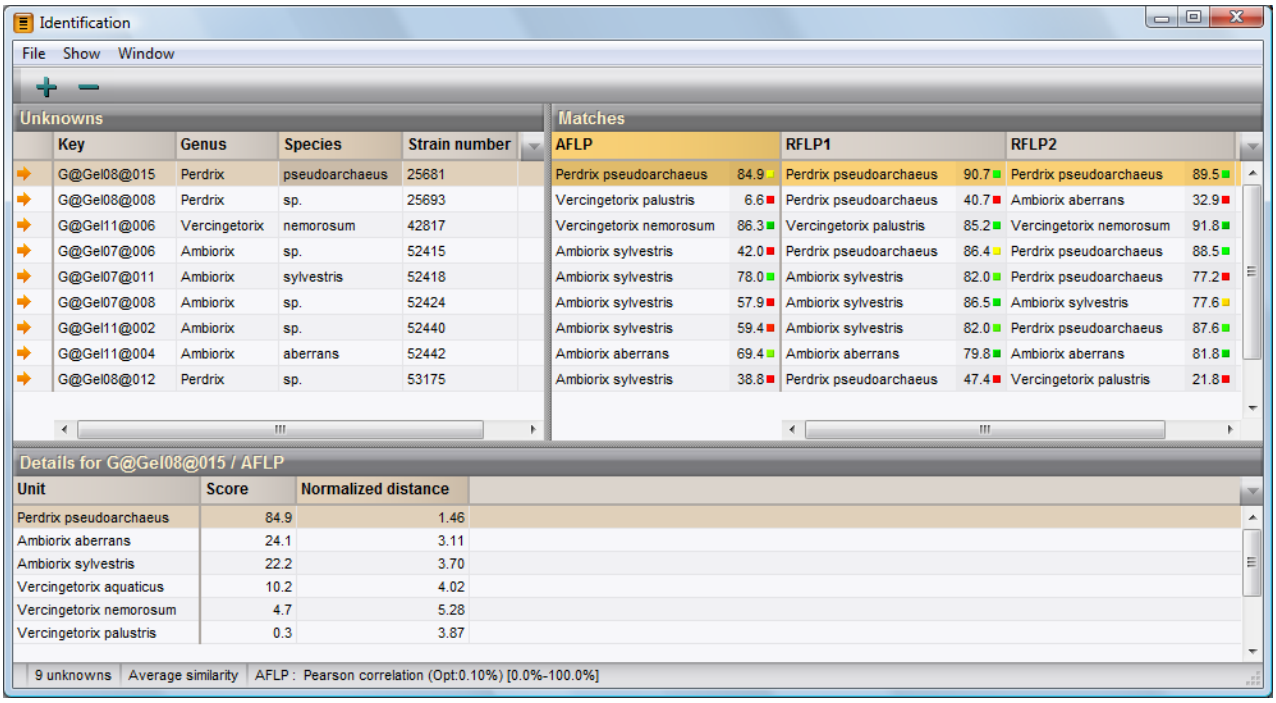


Figure 5-6. ***Identification*** **window.**

nation. See 1.6.4 for general display options of dockable panels.

You can move the separator lines between the panels to make optimal use of the display. Information fields in the *Unknowns* panel, *Matches* panel and *Details* panel can be displayed or hidden by pressing the Column properties button in the information fields header of the corresponding panels. For detailed information about the display options available for grid panels, see 1.6.6.

The columns in the *Matches* panel contain the name of the best matching library units and their identification score. The identification scores are the similarity values obtained using the coefficient which is specified in the settings of the experiment type (see Chapter 3). The normalized distances appear as colored squares next to the identification scores. They range from red (improbable identification) over orange, yellow (doubtful identification) to green (faithful identification).

5.2.2.12 Using ***Show > Show more matches*** or , the second, third, etc. best match can be shown for each unknown. To display fewer matches per unknown, select ***Show > Show less matches*** or .

The *Details* panel lists the best matching library units for the selected unknown/experiment type combination, ranked by their identification score. The normalized distance is here displayed as a number. Clicking in the *Unknowns* panel or *Matches* panel updates the *Details* panel with the information of the newly selected unknown/experiment type combination.

5.2.2.13 Double-clicking on a library unit within the *Details* panel opens an *Identification comparison* window. This window is similar to a normal *Comparison* window, listing the unknown entry and the entries of the library unit.

5.2.2.14 Export the identification overview to a text file with ***File > Export overview***, or create a detailed text report with ***File > Export details***.

For routine identification purposes, it can be useful to store the identification results for each unknown entry. Thereto, first create a dedicated field in the database:

5.2.2.15 In the *FPQuest main* window, select ***Database > Create new information field*** and name the new field e.g. 'ID result'.

5.2.2.16 Click on the 'ID result' field in the *Unknowns* panel (if the field is not displayed, press the column properties button and select it from the pull-down menu) and select ***File > Fill information field***.

5.2.3 Creating a neural network

• Theory

A neural network is a means of calculating a function of which one does not 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 slightly adjusted, 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 FPQuest, 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 5.2.1 to create a new library. To obtain a reliable neural network, each of the library units must have sufficient members, many more than just two or three. The number of entries required 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.

5.2.3.1 Double-click on a library to open it.

5.2.3.2 Select ***Experiment > Train neural network*** or  . A dialog box pops up, listing the existing neural networks for this database, if any.

5.2.3.3 To add a neural network, press **<*Add*>**.

The *Neural network training* dialog box appears, as shown in Figure 5-7.

5.2.3.4 Under ***Select experiment to be used in the neural network***, you can select the experiment type to train the neural network on.

5.2.3.5 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%.

5.2.3.6 With ***Max. number of iterations***, you can specify the maximum number of training cycles to be performed. By default this value is 20,000.

5.2.3.7 ***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.

5.2.3.8 Enter a name for the neural network under ***Neural network name***. You can use the name of the experiment type.

5.2.3.9 When all parameters are entered, press **<*Start training*>** to start the training process. Depending on the size of the library, the training process can take several

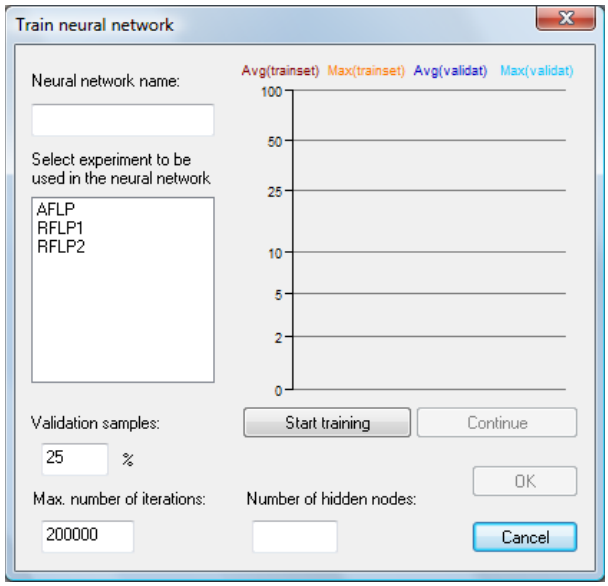


Figure 5-7. The *Neural network training* dialog box.

minutes. An animation of the progress of the training is shown in the x-t diagram (Figure 5-7).

5.2.3.10 During the training, it is possible to interrupt or abort the process by pressing **<*Stop*>**.

5.2.3.11 If you wish to resume the training process, press **<*Continue*>**. The program will continue the iteration process until the maximum number is achieved.

5.2.3.12 To save the neural network, press **<*OK*>**.

5.2.3.13 To identify database entries using a neural network, proceed as explained in 5.2.2.1 to 5.2.2.11, but choose ***Neural network*** instead. A drop-down list showing the existing neural networks will become available, allowing you to choose one of them for the identification.

6. APPENDIX

6.1 Connected database table structure

6.1.1 Introduction

In the description below, the structures of the tables required by FPQuest in a connected database are given (see Section 2.3). The tables are indicated with their default names. 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.

NOTE: A number of tables in a FPQuest connected database deal with BioNumerics experiment types such as sequence types, matrix types, trend data type, and 2D gel types. These tables are also required by FPQuest, in order to assure compatibility with BioNumerics databases and to allow migration from FPQuest to BioNumerics.

6.1.2 Table ATTACHMENTS

Contains a record for every attachment present in the database.

- KEY (VARCHAR(80))

  Key of the entry the attachment belongs to.

- IDN (VARCHAR(10))

  Identifier attachment.

- CLSS (VARCHAR(20))

  The data type of the attachment (1 = Text file, 2 = Bitmap image, 3 = HTML document, 4 = Word document, 5 = Excel document, 6 = PDF document).

- DESCRIPT (VARCHAR(80))

  The description of the attachment.

- FILENAME (VARCHAR(250))

  The path where the file is stored.

- CONTENT (CLOB)

  The content of a text file.

6.1.3 Table COMPARISONS

The table contains information about all comparisons saved in the database.

- NAME (VARCHAR(200))

  The name of the comparison.

- CMPTPE (VARCHAR(80))

  The type of comparison (comparison or library).

- CMPCLSS (VARCHAR(200))

  The comparison class.

- CMPOWNER (VARCHAR(80))

  The comparison 'owner'.

- CMPDATECREATED (VARCHAR(80))

  The date the comparison was created.

- CMPDATEMODIFIED (VARCHAR(80))

  The date of the last modifications made to the comparison.

- CMPDATA (VARCHAR(CLOB))

  Comparison data.

6.1.4 Table DBSCHEMAS

This table holds information about the software and the table structure of some installed plugins.

- NAME (VARCHAR(80))

  Name of the software and the installed plugins for which new tables were added to the table structure.

- SCHVERSION (VARCHAR(80))

  Version number of the software and the installed plugins for which new tables were added to the table structure.

- SCHDEF (CLOB)

  XML information on the installed plugins.

6.1.5 Table DBSETTINGS

This table holds the installed plugins and the active information fields of the *Database* panel.

- NAME (VARCHAR(200))

  'ActivePlugins', 'DEFAULTLEVELSETTINGS'.

- CONTENT (CLOB)

  The string of the 'ActivePlugins' holds the installed plugins, the 'DEFAULTLEVELSETTINGS' holds the active information fields of the *Database* panel.

6.1.6 Table ENLEVELS

Contains information about the levels defined in the database.

- LEVELID (NUMBER)

  Holds the levelID of the defined levels.

- LEVELNAME (VARCHAR(80))

  Name of the level.

- SETTINGS (CLOB)

  String that holds the active fields of each level.

6.1.7 Table ENRELATIONS

This table contains all entries belonging to a relation.

- RLID (NUMBER)

  The unique ID for each defined relation in the database.

- RELTYPEID (NUMBER)

  The identifier for each defined relation type.

- KEY1 (VARCHAR(80))

  The key of the entry belonging to the forward relation.

- KEY2 (VARCHAR(80))

  The key of the entry belonging to the reverse relation.

6.1.8 Table ENRELATIONTYPES

Contains information about the defined relation types.

- RELATID (NUMBER)

  The unique identifiers for each defined relation type.

- RELATFORWNAME (VARCHAR(200))

  The name of the forward relation.

- RELATBACKNAME (VARCHAR(200))

  Name of the reverse relation.

- LEVELID1 (NUMBER)

  The levelID of the forward relation.

- LEVELID2 (NUMBER)

  The levelID of the reverse relation.

- RELTYPE1 (NUMBER)

  The type of forward relation: many = 0; one = 1.

- RELTYPE2 (NUMBER)

  The type of reverse relation: many = 0; one = 1.

6.1.9 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).

- LEVELID (NUMBER)

  The levelID for each entry in the database.

Other fields: additional database information fields.

6.1.10 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.

6.1.11 Table EXPERATTACH

This table contains descriptive information for any specific key-experiment combination. For example, the error reports generated in the Spa, MLST and batch sequence assembly plugin (BioNumerics) are stored in this table.

- EXPRATTACHID (NUMBER)

  The unique identifier for each key-experiment combination.

- KEY (VARCHAR(80))

  Key of the database entry the information relates to.

- EXPERIMENT (VARCHAR(80))

  Name of the experiment the information relates to.

- NAME (VARCHAR(80))

  Names assigned to groups of key-experiment combinations.

- CONTENT (CLOB)

  Descriptive information specific for each key-experiment combination, e.g. error report.

6.1.12 Table EXPERIMENTS

This table contains a record for every experiment type present 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”, “Sequence”, “Matrix”, “Curve” or “2DGel”.

- 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).

Other fields: additional experiment type information fields.

6.1.13 Table FPRBNDCLS

This table contains a record for each band class defined in the database.

- CLSID (NUMBER)

  The unique identifier for each band class defined in the database.

- CLSEXPER (VARCHAR(80))

  Holds the name of the experiment type.

- CLSNAME (VARCHAR(80))

  Name of the band class.

- CLSPOSIT (FLOAT)

  The position (metrics) of each band class.

6.1.14 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.

- TONECURVE (VARCHAR(250))

  Contains the tone curve.

- CHPTRN (VARCHAR(250)) (only with "Fast band matching" enabled)

  Contains cached pattern information on the band positions for a fingerprint type with "Fast band matching" enabled.

Other information fields: additional information fields added in the *Fingerprint information* panel in the *Fingerprint file* window.

6.1.15 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.

Other information fields: additional Fingerprint file information fields.

6.1.16 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.

6.1.17 Table SEQTRACEFILES

This table holds information about the sequence trace files (four-channel chromatogram files from automated sequencers).

- KEY (VARCHAR(80))

  For use with the Kodon software.

- CONTIGFILE (VARCHAR(80))

  Unique ID of the contig that is associated to this sequence trace file.

- TRACEID (VARCHAR(80))

  Unique ID of the trace file.

- DATA (CLOB)

  Holds the full trace information including sequence and the chromatogram files in case the trace files are stored in the database. Otherwise, it stores a link to the path of the trace file.

- INFO (CLOB)

  Contains the full editing information of the sequence trace file.

6.1.18 Table SEQUENCES

This table holds the sequence information stored in the database. Note that the columns designed for contig files have changed with respect to earlier versions of the software.

- KEY (VARCHAR(80))

  Key of the database entry this sequence belongs to.

- EXPERIMENT (CHARCHAR(80))

  Experiment type of the sequence.

- SEQUENCE (CLOB)

  Sequence data.

- SEQUENCEQUAL

  Quality coefficient for each base in the sequence.

- CONTIGFILE (VARCHAR(80)

  Unique ID of the contig file that is associated to this sequence (if any).

- CONTIG (CLOB)

  Holds the contig sequence and its full editing history.

- CONTIGSTATUS (VARCHAR(10)

  Contains the status of the contig file, i.e. confirmed or not.

6.1.19 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.

6.1.20 Table TRENDDATA

Holds information about the trend data types.

- KEY (VARCHAR(80))

  The key of the database entry.

- EXPERIMENT (VARCHAR(80))

  Name of the trend data type.

- CURVE (VARCHAR (80))

  Name of the trend curve.

- DATA (CLOB)

  XML string that holds the data.

- PARAMS (CLOB)

  Lists the parameter(s) defined for the trend data type.

6.1.21 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).

- EXPERIMENTS:

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.

6.2 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',

`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'.

`[: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.

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

Bio-Rad Laboratories, Inc.

Life Science Group

Web site *www.bio-rad.com* ***USA*** *800 4BIORAD* ***Australia*** *61 02 9914 2800* ***Austria*** *01 877 89 01* ***Belgium*** *09 385 55 11* ***Brazil*** *55 21 3237 9400* ***Canada*** *905 712 2771* ***China*** *86 21 6426 0808* ***Czech Republic*** *420 241 430 532* ***Denmark*** *44 52 10 00* ***Finland*** *09 804 22 00* ***France*** *01 47 95 69 65* ***Germany*** *089 318 84 0* ***Greece*** *30 210 777 4396* ***Hong Kong*** *852 2789 3300* ***Hungary*** *36 1 455 8800* ***India*** *91 124 4029300* ***Israel*** *03 963 6050* ***Italy*** *39 02 216091* ***Japan*** *03 5811 6270* ***Korea*** *82 2 3473 4460* ***Mexico*** *52 555 488 7670* ***The Netherlands*** *0318 540666* ***New Zealand*** *0508 805 500* ***Norway*** *23 38 41 30* ***Poland*** *48 22 331 99 99* ***Portugal*** *351 21 472 7700* ***Russia*** *7 495 721 14 04* ***Singapore*** *65 6415 3188* ***South Africa*** *27 861 246 723* ***Spain*** *34 91 590 5200* ***Sweden*** *08 555 12700* ***Switzerland*** *061 717 95 55* ***Taiwan*** *886 2 2578 7189* ***United Kingdom*** *020 8328 2000*

10013993 Rev A US/EG *07-0921 1107 Sig 1106*