NEW FEATURES OF MIRAMON V.6.1

CONTENTS

MIRAMON V. 6.1

Document editor: Xavier Pons
First version: 28-01-2008
Last modification and version: 19-03-2008 **1.8**
Where to find this document or updated versions:
www.creaf.uab.cat/MiraMon/new_note/esp/notes/Novetats_v6_eng.pdf

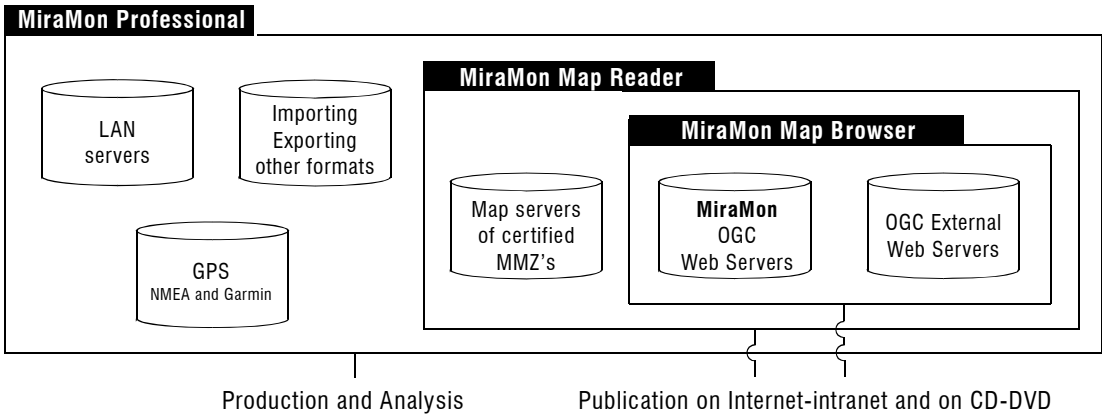
1. Introduction

This document describes the most significant new features that have been incorporated into MiraMon v.6.1 in relation to v.5.0. MiraMon is a computer program designed to be used with Cartography, Geographic Information Systems (GIS), Remote Sensing, Spatial analysis and related disciplines. This summary includes improvements made to version 5 and entirely new features that have been incorporated into v.6.1 (some of which already appeared in v.5.5 and have been consolidated in version 6). Features that were included in the document describing new features for v.5 have been deliberately omitted, but may be downloaded at http://www.creaf.uab.cat/MiraMon/new_note/esp/notes/new_v5.pdf. We strongly recommend any user who is not very familiar with v.5 to read this document. The more advanced features are described using a smaller font size as they are only of interest during a second reading or to expert users.

In this document, “MiraMon” refers to the complete MiraMon software package, whereas MM32 refers to the main module, the one that is used for displaying, querying, printing, digitizing and graphic editing, etc. “MiraMon Professional” refers to the whole MiraMon package (with license) as opposed to the “MiraMon Map Reader” and the “MiraMon Map Browsers” for the Internet.

The diagram below depicts the main ways of accessing geographic information using the different MiraMon applications. It shows that MiraMon Professional allows users to access their own datasets in corporate LAN servers (including access to such databases as MDB, Oracle, MS-SQL Server, etc), import from and export to other formats, communicate with GPS receivers in real time or at

a later date, access MMZ datasets published by other entities transparently, or browse through datasets offered by servers from any manufacturer that follows the Open Geospatial Consortium (OGC) standard. In the latter case, MiraMon servers offer useful features such as a much higher access speed, downloading of real data, etc. The application and the installation mechanism are totally compatible with all versions of Windows 32-bit, including Windows Vista.



This document does not include the minor improvements and solutions to problems that have also been developed. However, these are described in the *MiraMon versions Diary*, which is available on the Internet (see below).

2. Support for viewing and querying multiple rasters.

Before version 6 of MiraMon, the metrics which governed screen viewing were basically the result of two situations created by the presence or absence of a raster between the layers that were being viewed. In the first case the raster controlled the situation because:

- Its area limited that of the vectors.
- Pixel size conditioned the different zoom levels in such a way that zoom=1 meant that a screen pixel was the same as a raster pixel. Thus, in an image with a 20 m pixel size, when viewed at zoom=1, each screen pixel was equivalent to 20 m on the ground (taking into account, naturally, the cartographic projection being used).
- Only one raster could be open simultaneously, either in the same geographic area as the vectors and WMS layers or by enlarging it.

In the second case (exclusive presence of vectors or WMS layers), on the other hand, the situation did not result in any area limitation and the whole view of all the open layers determined the zoom level with greatest reduction (in version 5, zoom=/100) and the equivalence of the size of screen pixel.

This situation was caused by the origin of the software, which is widely used in Remote Sensing, in which the situation described above was preferred to alternatives that were aesthetically more plastic and flexible for other users but which lacked a strict control of pixel viewing quality (when the zoom interpolates the raster at all levels there is no level which permits optimum screen viewing)

or scrolling between them. Moreover, at first the memory and processing capacity of the computers that existed at that time did not provide the majority of users with an agile view of several simultaneous satellite images, each weighing dozens of Mbytes. Consequently, limitations on viewing these images did not appear to be a particularly problematic.

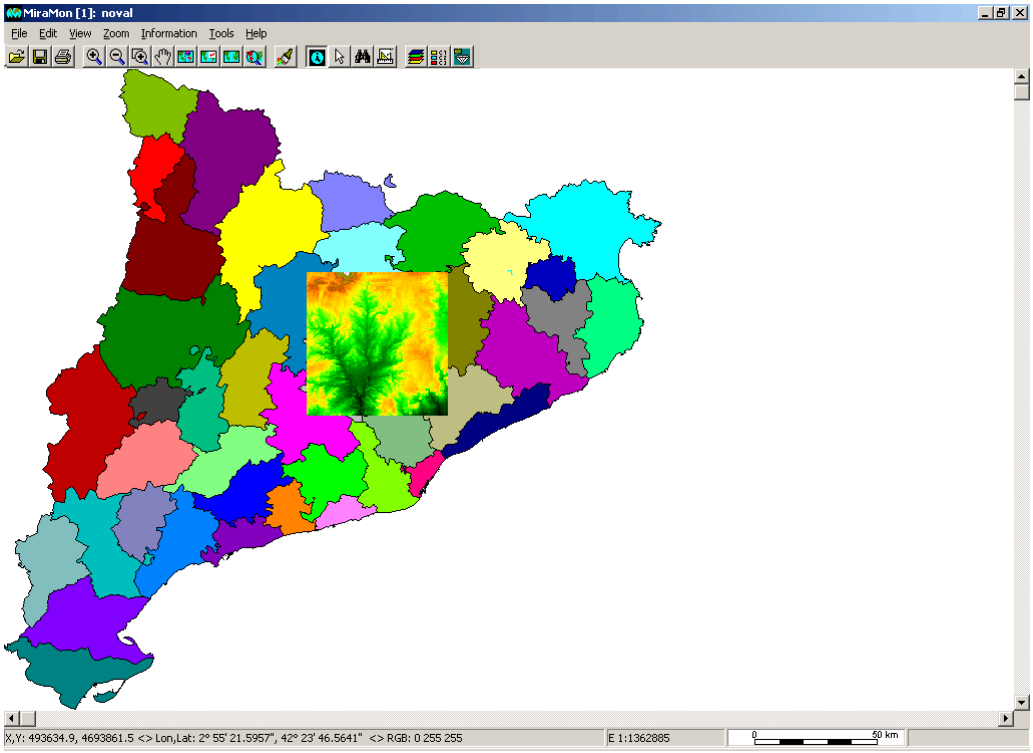
Increased computer memory and the generation and increasing availability of orthophotos resulted in the growing use of images by GIS users. Consequently, the decision was taken to incorporate more than one raster into MiraMon while at the same time preventing this from conditioning the geographic area. The challenge, however, was not to lose the possibility of maintaining the advantages of strict control on viewing, which is so highly valued in Remote Sensing, or the evaluation, in optimum conditions, of raster quality.
Despite the growing need to be able to open more than one raster, the appearance and increasing acceptance of the WMS cartographic viewing specification (mainly in Internet) by the Open Geospatial Consortium (OGC) delayed the implementation of the possibility of opening several rasters in favor of opening the door to the world of WMS, which was quickly gaining popularity. In fact, MiraMon was one of the very first desktop programs capable of accessing WMS bases on Internet, as well as local servers, showing cartography offered by WMS servers combined with other conventional, raster and vector layers with the required geodesic accuracy.

The decision to incorporate WMS was, on the one hand, the result of the growing availability of datasets for this specification, the desire to join the newly-created standard, and also the fact that it offered a better solution than opening very large rasters or many rasters simultaneously. Indeed, there is no question that the WMS presentation is a major improvement which provides a seamless solution to traditional products based on a specific sheet distribution and removes the need for the client system to have large quantities of memory (although it does require powerful servers and, above all, sufficiently fast network communications). In other words, users integrated into a relatively agile network could access enormous cartographic datasets continuously from relatively simple computers and, above all, without bothering to open one sheet or another and having to worry about whether they had enough memory for so many rasters. It was decided, therefore, that rather than allowing the user to open 2,3,4, etc. orthophotos simultaneously, it was better to be able to access hundreds or thousands of orthophotos covering a country.

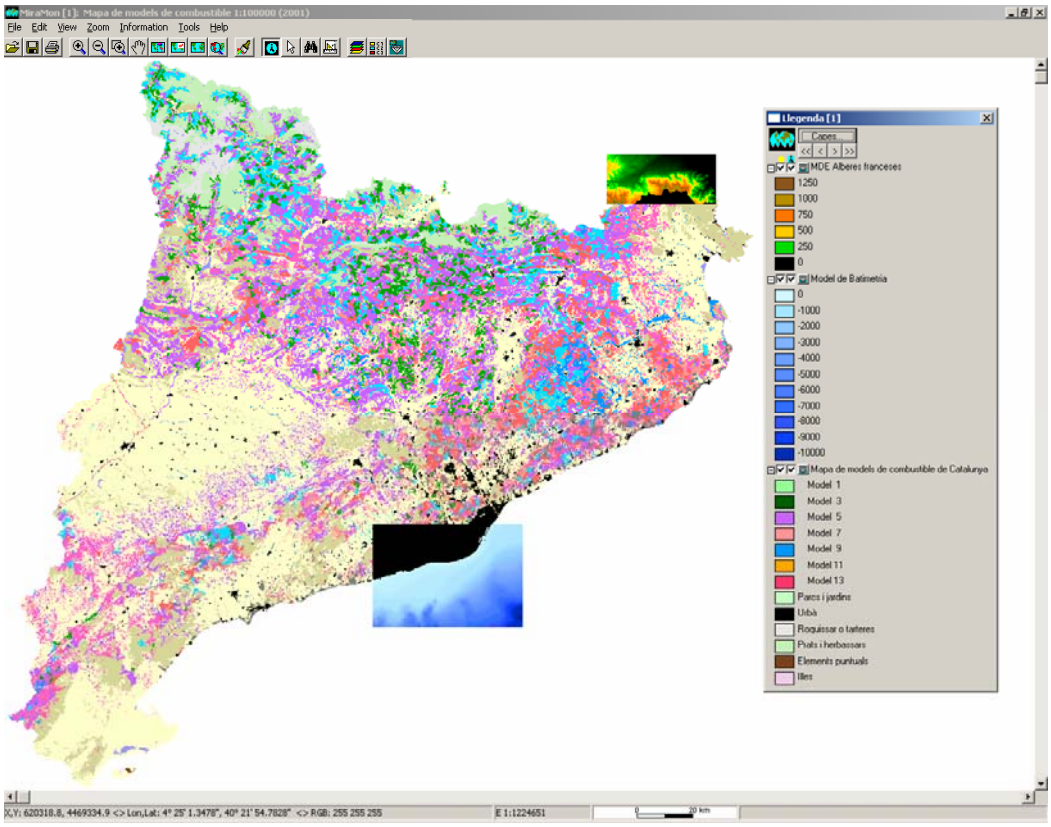
The growing availability of WMS datasets meant that MiraMon was being constantly refined by also introducing adjustments in order to ensure optimum quality and viewing in relation to the overall area offered and the native size of the pixel when the WMS base is raster at origin (this is the case, for example, for the orthophoto cover of a country). For more information about the characteristics of this format, see the relevant section of this document.

Having refined and resolved these and other features, MiraMon v.6 is finally able to open several rasters simultaneously. They can be placed in juxtaposition or overlapping and in any arrangement regardless of their area, pixel size, number of bits per pixel, etc. Moreover, **the rasters no longer determine,**

under any circumstances, the extent of the open layers which, nevertheless, can be adequately controlled using the updated “Zoom | Define new whole view...” option (see below). In addition, **the number of open rasters and their location in the overlay with regard to the vector layers and WMS is not restricted**. The illustrations below show rasters over vectors or several open rasters simultaneously.



The solution adopted allows users of Remote Sensing, who use the images mainly for scientific purposes and would not wish to lose the benefits of the former metric assignation of zoom=1, to switch easily to the classic MiraMon mode. In this descending compatibility, however, its was not considered advisable to maintain the limitation on the area covered by the **reference Raster**, as a result of which it is possible to work in a similar way to the earlier case but with the possibility of opening several rasters and with an overall coverage resulting from all the open layers (although this can easily be limited if the user so wishes).



It was decided that this change also justified a number of changes in the appearance of the MiraMon window itself: **permanent presence of the scrollbars and the appearance of a status Bar**. These changes are covered in more depth in the corresponding section below.

The number of raster types that can also be opened directly, without the need to import them, has also been increased.

Query by location and by attribute

Queries by location or attribute in the new environment with multiple raster support are performed as before when more than one vector layer was open. In other words, the query is carried out starting from the top layer (image) to the bottom layer passing through the whole structure of open layers. If a multiband raster is being used, the query by location returns the value of that pixel in each of the available and documented bands.

3. New modes of zoom management

Multiple raster support has provided the opportunity to revise zoom management. As mentioned above, it has been possible to guarantee compatibility between the classic raster-governed mode, because of the advantages it offers in terms of maximum viewing quality and ease of metric location and control in Remote Sensing tasks, and the plasticity required by many users who do not need this metric reference. As a result, two management modes have been developed in addition to a third mode which automatically switches between the other two.

Before describing the modes, it is **important** to take into account that:

- **Users who do not need to be able to change the mode of zoom management can work exclusively in Whole view mode.**
- **It is possible to switch between the zoom management modes at any moment**. This new option should therefore be seen as a powerful technical resource for advanced users, who will be able to change mode according to their needs whenever they wish.

The following modes are available:

Whole View Mode

This is the most intuitive and natural zoom mode. It is therefore the default mode and is recommended for non-specialist GIS users. In this mode the zooms are defined taking into account the fact that the whole view Zoom corresponds to the whole extent of all the layers or to what is specified in “Zoom | Define new whole view”.
The whole view zoom is set at level /100 in the zoom menu. From this point, the successive levels of enlargement are simply relative references to allow the user to adjust the zoom level. However, if you wish to know what dimension a screen pixel has in map units (usually metric), you can click on the numerical or graphic scale on the status bar. The dimension will then appear in the dialog box of the Zoom by scale.
It is possible to define the desired zoom level by means of exact scale requests using the dialog box mentioned above, but this is usually not necessary.
In the Whole view mode, the zoom has no upper limit and it is possible to enlarge it as much as you wish (to levels at which, for example, one screen pixel represents one nanometer on the ground, or even less, although this is unlikely to be necessary). The check menu that indicates the zoom level of the image revolves around the counter when the maximum level is exceeded. In other words, the system works by returning to a lower level, which allows you to continue increasing the relative increments. When “Zoom | Whole view” is selected again, the system always returns to the whole view defined at the start.
The WMS layers may sometimes not fill the whole screen when the zoom is used in Whole view mode. This is a deliberate feature and is due to the fact that when the user accesses WMS cartography, which among its capacities indicates that there is a typical rasterized pixel size and gives its value, MiraMon duly adapts the area in order to ensure **better viewing quality** as well as a **better response speed from the server**.

Raster mode

In this mode, a raster referred to as the *“**Reference raster**”* **determines zoom=1**. This is particularly useful in Remote Sensing where it is often necessary to have strict control over which pixel you are working with and to perform accurate displacement from one pixel to another as well as having an optimum view of raster quality in zoom=1.

The extent of the request for a whole view results in zooming in or zooming out depending on the size of the set of raster, vector or WMS layers that are open. This is because the whole view covers all the layers and it is no longer impossible, as occurred in earlier versions of MiraMon, to access geographic areas above or to the left of the reference raster: **the raster is no longer "boxed into" the top left-hand corner** (except when the vector layers have a minimum X that is greater and a maximum Y that is smaller than the those of the reference raster).

Moreover, if when working it is more convenient to box in the pixels of the reference raster in the top left-hand corner of the screen, this can be done by opening "Zoom | Define new whole view..." and clicking on "**Multiple of reference raster**".

The raster mode permits a maximum zoom enlargement of 15 times. If you wish to zoom in more, it is possible to switch temporarily to the Whole view mode and return to raster mode whenever you wish to do so.

When zoom management is in raster mode, maximum zoom is set at /1000. This is not available in the menu, but is designed for special cases such as when opening a raster with 0.5 m resolution covering the whole of Catalonia and a screen which does not have enough pixels for everything to fit in a /100 zoom.

To summarize, in this mode:

- Zoom=1 means that one pixel of the reference raster is one screen pixel and can be enlarged 15 times.
- Raster zoom operations which have the same pixel size as the reference raster are **faster** and of **greater visual quality**.
- During **displacement, increments are made in pixels of the reference raster** (except when zooming out, where they are carried out in the selected reduction, e.g. every 4 pixels if the zoom=/4).

Automatic mode

This mode applies the reference Raster mode if all the open rasters have the same pixel size, and applies the whole view mode if this is not the case or if there are no rasters. The advantage of this mode is that it provides maximum viewing quality if several rasters with the same pixel size are opened but with the specific Raster mode limitation of zoom x 15 when a raster is open.

Changing the zoom management Mode

The zoom management mode can be changed using the "Zoom | zoom management mode" window. When choosing the reference Raster mode, it is necessary to indicate which of the open rasters is the reference raster, that is, the raster that has the pixels that you wish to assign to the level zoom=1.
It is even possible to select the zoom management mode before opening a layer.

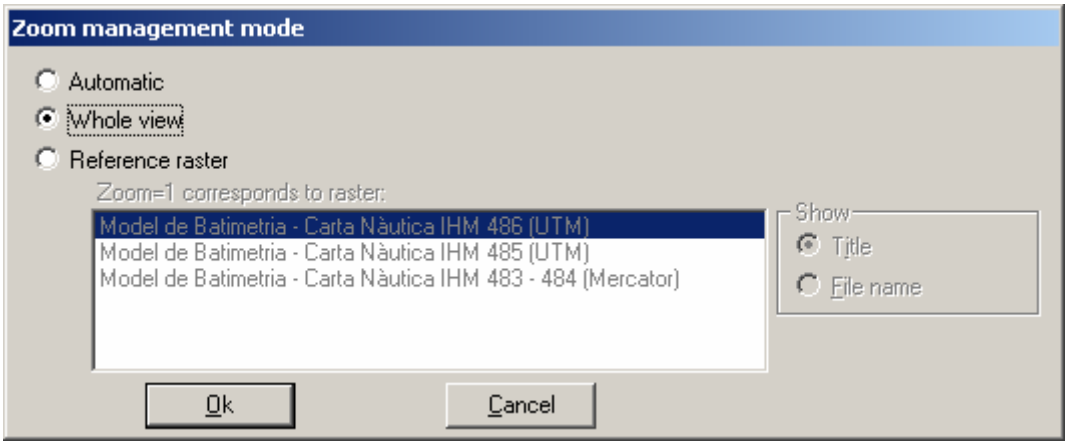
Default setting for the zoom management Mode

MiraMon's default operation is controlled by using the ZoomManageMode variable, which is started using the keyword with the same name in the [METRICS] section of MiraMon.par. This keyword offers the following options:
Whole view Mode (option 0 and default option): The zoom is managed according to the whole view of all the layers.
Reference raster Mode (option 1): the zoom is managed according to one of the open rasters, which by default is the first, regardless of whether there are other raster or vector layers, and can be changed at any moment using "Zoom | zoom management mode...".
Automatic mode (option 2): this mode switches between the other two modes, as explained above.

General recommendation for zoom management Mode

Our own practical experience over the months that the 3 zoom management modes were developed and assessed has shown that "Whole view" is the most suitable mode for the majority of users. The "Automatic" mode, on the other hand, is suitable for users who usually open only one type of raster (e.g. orthophotos of their area) and who wish to give priority to the visual quality of these open rasters, but who wish to switch to the "Whole view" mode when they simultaneously open other rasters (e.g. a digital elevation model). Finally, the "reference Raster" mode is the most suitable for users who carry out treatment of Remote Sensing images and who want the zoom and pixel-by-pixel displacement metrics of the raster they choose to determine the zoom mode, regardless of whether they open other rasters with different resolutions. It is important to remember that in each mode the geographic area is that of all the open layers (unless this is changed using "Zoom | Define new whole view...") and that the limitations of the versions of MiraMon prior to v.6 no longer apply.

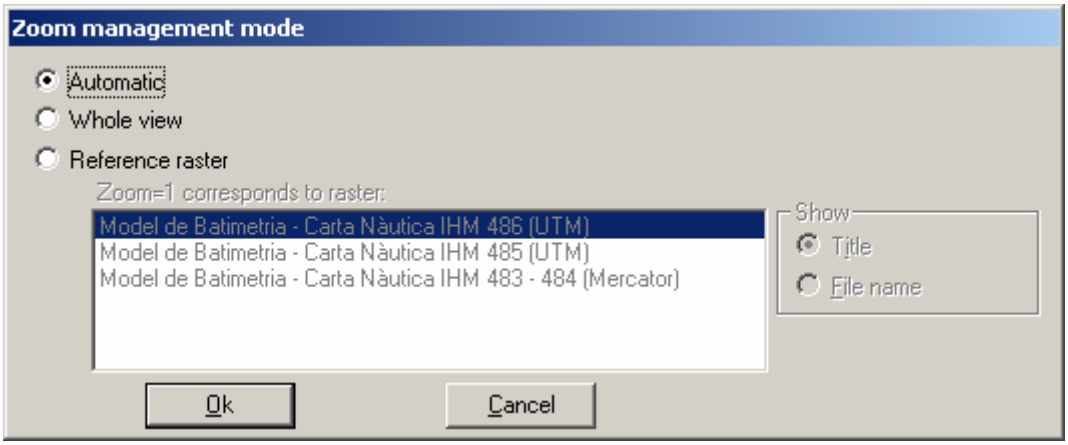
Information for users who opt to use the zoom in "Whole view" mode



This is the most intuitive management mode, but does not provide maximum quality for viewing a family of rasters with the same pixel size. If you wish to obtain maximum quality, you can easily go to "Zoom | zoom management

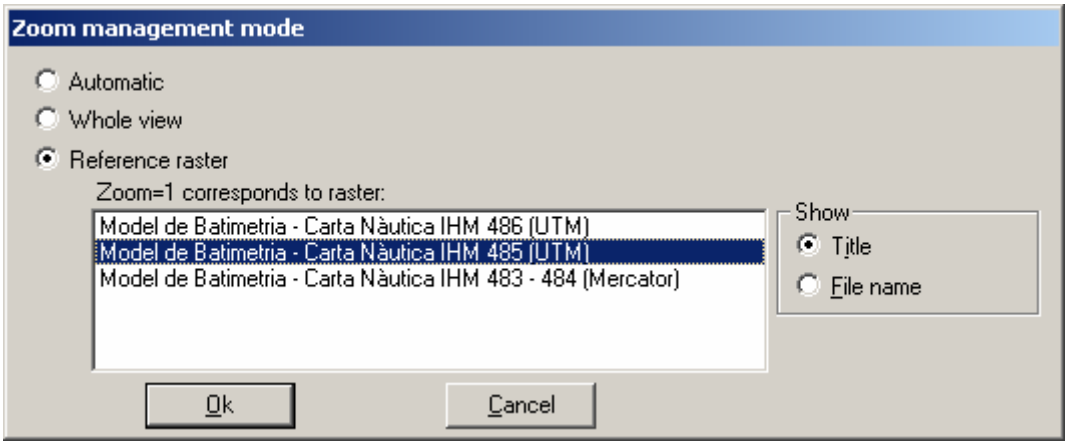
mode...” and select “raster” mode for the layers that you wish to view with maximum quality.

Information for users who opt to use the zoom in “automatic” mode



On opening one or more rasters with the same pixel size (e.g. different sheets of a cartographic series of orthophotos at a particular scale) the program runs internally in “raster” mode. This allows you to automatically obtain maximum viewing quality at zoom=1.
If other rasters with different pixel sizes are opened, the program runs internally in “whole view” mode and there is therefore no need for there to be an optimum quality for viewing any given raster. However, you can easily go to “Zoom | zoom management mode...” and select “raster” mode for a particular raster layer which you wish to view with maximum quality.

Information for users who opt to use the zoom in “raster” mode



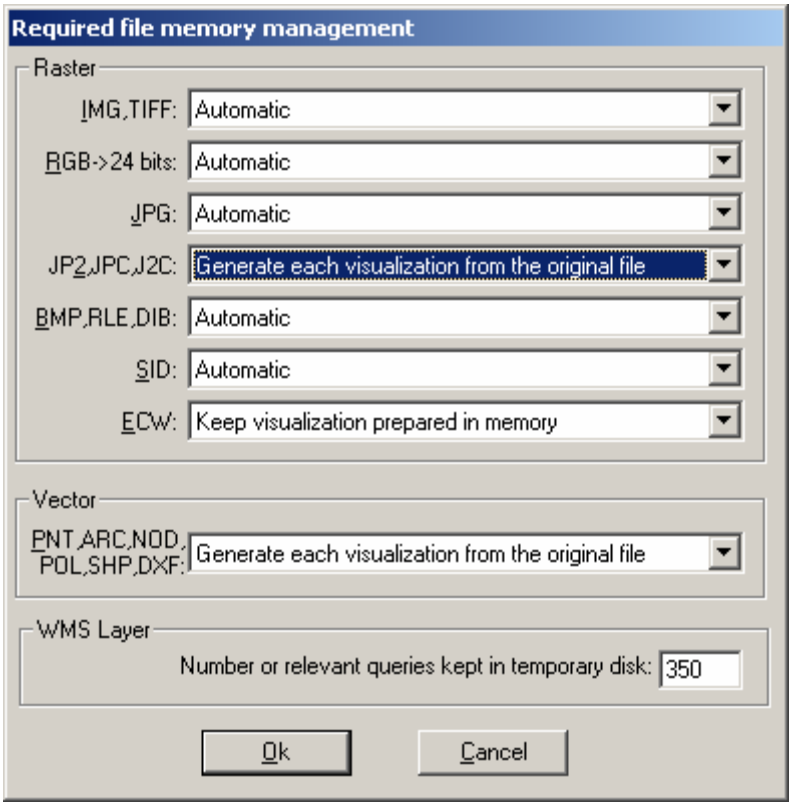
When no layer is open, if the Zoom management mode selected is “raster” and vectors are opened, the management mode switches to automatic mode because there is no raster that can operate as the reference Raster. The advantage of switching to “automatic” mode and not to “whole view” is that the desired effect is obtained when the first raster is opened: the rasters that are subsequently opened govern the zoom (provided they have the same pixel size).
Of course, if other rasters with other pixel sizes are opened when one or more rasters with the same pixel size are already open, the new “automatic” mode runs internally in “whole view” mode (because the automatic mode is only equivalent to “raster” when all the rasters have the same pixel size). If, in this situation, you wish to specify that one of the open rasters is the reference raster and governs the zoom, you can easily go to “Zoom | zoom management mode...” and select this.
For this reason, it is recommended that if the mode is “raster”, it is easier to open one of the rasters to act as the reference and not the vectors (in this way you avoid having to specify the reference Raster once again if you later open rasters with other pixel sizes).
On closing a raster that is acting as the reference Raster, if no other layers are open, the mode remains on stand-by until another raster is opened. This will become the reference Raster. If there are no other raster layers but there are WMS or vector layers, the zoom management mode switches automatically to “automatic”, as explained above. Finally, if other rasters remain open, MiraMon will assign the reference Raster function to the top raster according to the order of overlay of layers with the same pixel size. If no raster fulfils this condition, the role of reference Raster will be assigned to the top raster in the order of layer overlay.

Information for all users regarding the “Zoom | Whole view” option. 

In the Zoom | Whole view” option (or the equivalent button), MiraMon shows the corresponding whole view according to the zoom management mode that is activated. It is important to stress here that if the overall extent has changed slightly to obtain an accurate scale at any moment, this request is rendered inoperative and the overall extent reverts to the values selected as whole view. In other words, it is possible that if a scale is fixed manually, at say 1:50000, you can execute a series of “Zoom in | Zoom out” operations passing through scale “multiples” (1:100000, etc.) before reaching a whole view that displays the set of layers with an unnecessary blank area: selecting “whole view” will readjust the screen display. It should also be pointed out that returning to whole view retains those values that may have been entered in the “Zoom | New whole view...” box and that these will only be abandoned if you click on the “Restore original values” button in the same dialog box.
How to find out the exact metrics of a particular view. At any moment you can use the “ArrowCapitalLeft+F6” combination to display the main metric parameters in use. This function is designed for internal debugging, but may be useful for advanced users.

4. New management system for memory required by files

In order to improve the speed of the first loading and redraw according to the needs of the user and the configuration of his or her computer (amount of RAM memory, free space in the disc where the temporary files are stored (usually unit C), speed of Internet access, etc.), a finely-tuned configuration mechanism has been developed to manage the memory needed by the three main types of layers that can be opened: rasters, vectors and WMS layers.
The new dialog box “Display | Special | Management of memory required by files” allows you to dynamically change the memory management specifications that are described below and which can also be initially selected using MiraMon.par.



4.1. Memory management for rasters

For each type of raster supported, there are as many as four memory management modes. Not all the options are available for all file types because some may not be of any use (such as the second in the case of the classic JPEG) or may not have been programmed yet:

- **Automatic**. MiraMon determines the mode that it considers most suitable according to file type. This option is recommended for non-advanced users.
- **Generate each display from original file**. In formats in which it is possible to access specific requests for zooms and areas relatively easily (JPEG2000, SID, etc.), this option is very suitable as it occupies very little RAM memory and temporary disc. However, if the file is on an

Internet server and not on a local disc, it may be slower than other options because each zoom or displacement is processed by the server which has the file. In these cases it is possible to speed up the display process by copying the file to the local disc before opening it or by switching to one of the modes described below.

- **Keep display ready in memory**. If the user has a large amount of RAM memory and it is not necessary to wait the time necessary to read the file in the memory from the disc or the server, this mode provides faster access for each zoom or displacement operation. However, each raster that is opened in this mode occupies more memory.
- **Keep display ready on temporary disc**. If the user does not have much RAM memory, but does have a temporary disc with enough space (you can access this disc and empty it by executing the command %temp% in "Start | Execute") this option allows you to open many files and manage the redraws very easily because the temporary disc is usually located in the local unit and can be accessed quickly. However, in this mode it takes longer to load the file because of the need to prepare the data in the temporary disc. Of course, the data is erased from the temporary disc when the layer is closed or, if the session is terminated by mistake, during the "cleaning" procedure executed by the MiraMon Manager every time the program is started on another day.

The default value of this parameter for each type of raster file can be set using the keywords MANAGEMEMORYIMG=, MANAGEMEMORYJPEG=, etc. of MiraMon.par.

4.2. Memory management for vectors

Traditionally, vectors are not completely loaded into memory in order not to occupy as much of the system's memory as when large layers are loaded and also because the special structure of the MiraMon vectors permits very fast access for redraw. However, if you have located a vector layer on a slow server and network, each redraw is slowed down due to the need to access this network. It is for this reason that, during the development of v.6, a mode will be incorporated to load all the vector entities of the layer into memory.
This new option will not be necessary for opening MMZ files because as they are unzipped in the temporary disc, they can be accessed quickly.

4.3. Memory (disc) management for WMS.

If the WMS server is a slow server or the user does not have access to a fast Internet network, redraw is very slow. For this reason MiraMon can save the result of each WMS request performed in case it is required again (preview or simply by returning to the application after having switched to another application or session) so that it can be accessed quickly. If 0 is selected, no view is saved and the result is that every redraw requires a request to the servers, whether these are local or remote. This is much slower than taking advantage of the fact that you have already requested the same view.

Of course, the data are erased from the temporary disc when the layer is closed or, if the session is terminated by mistake, during the “cleaning” procedure executed by the MiraMon Manager every time the program is started on another day.

The default value for this parameter can be set using the keyword MAXNVIEWSWMSSAVED= of MiraMon.par.

5. General environment and interface

The applications and MM32 itself can be started in your preferred language by indicating /LANGUAGE= and the ISO code of the language chosen: CAT, SPA, ENG.

5.1. New menu entries

New applications have been incorporated into the menu, such as applications for importing (located at “File | Import”, e.g. ECW files) and exporting (“File | Export”, e.g. SHP files), for grouped field statistics (“Tools | Alphanumeric databases”, for modelization through combination of multiple regression and interpolation (Tools | Modelization”), for vector generalization (“Tools | cartographic generalization | vector”, for distance and route analysis (“Tools | Distance and route analysis”, etc.

New menus have been incorporated into the Tools menu, such as that for Classification of Remote Sensing images, which includes modules for non-supervised and mixed classification, as well as reference to existing modules for confusion matrices and results generalization.

An entry has been added to the “Information | Open series” menu which gives information about open cartographic series.

The number of external applications that can be incorporated into the menu has been increased to 25 (the total number of [MENU_#] and [APLIC_#] sections is limited to 35.

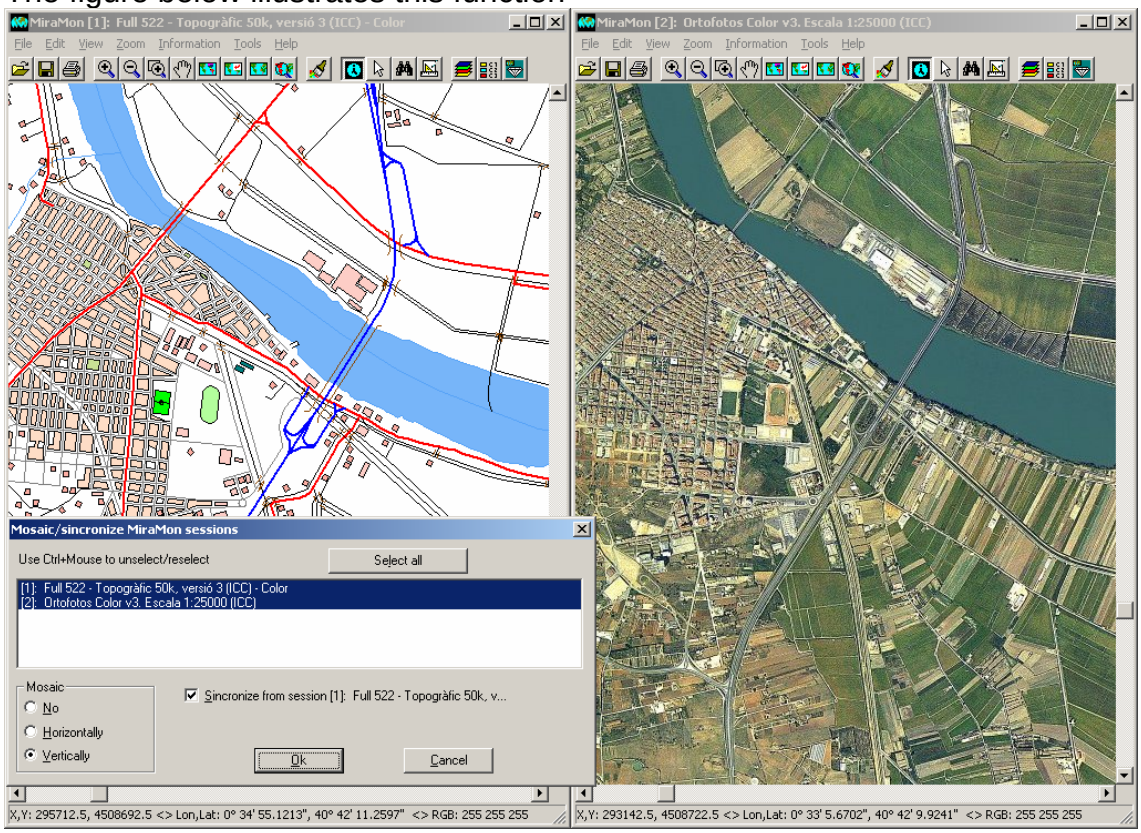
5.2. Automatic mosaic for sessions and better synchronization between them

The “Mosaic/synchronize MiraMon sessions” option has been added to the Zoom menu. This is also accessible by using the new  button located in the program’s button bar. The new option allows you to organize all the MiraMon sessions that you wish in vertical or horizontal mosaic, such as when you choose to mosaic non-minimized sessions from the Windows taskbar. Although it functions in a similar way, the new option avoids having to go to the Windows bar (which is not visible for all users and is difficult to access) and also avoids the need to minimize the unwanted sessions in the mosaic beforehand. Moreover, not all Windows users are aware of this option.

The new option also enables the user to simultaneously activate the synchronization using the same command. To prevent the legends from forming part of the mosaic, they are hidden before proceeding to the mosaic. The user may activate them in a later session using the F11 key, from the button bar or from the “Information” menu.

Finally, the box allows you to indicate synchronization but not mosaic. This is useful when you do not wish to modify the sessions to be mosaicked, or when you simply wish to work in full screen mode but with synchronization of the different sessions.

The figure below illustrates this function



Using this new feature, synchronization between sessions is metrically perfect. This means that when you instruct the program to carry out synchronization from one session, information about the aspects listed below is sent to the recipient sessions and is adopted for the duration of the synchronization:

- Horizontal Reference System (this allows you to ignore the synchronizations in the sessions that do not have the same system as the session from which the change of extent or zoom is carried out).
- Zoom management mode.
- Extent of the whole view.

So, the **session from which synchronization is commanded plays an important role regarding how synchronization is performed: it is this session which is in charge**. To show this effect, it is necessary to select a whole view in order to be able to begin to zoom in from the desired point.

However, it is possible to zoom or displacement from any of the synchronized sessions. If the zoom management mode in the section of MiraMon.par of the controlling session is a raster mode, the recipient sessions will have the same zoom x15 limitation as the raster mode. On opening a new layer or map or on closing them, or on defining a new whole view in any of the synchronized sessions, synchronization is stopped because, as the area has changed, the program does not know which session should govern the synchronization. The user must re-synchronize from the session that he or she decides now has the new extent that is to be synchronized.

On stopping the synchronization, the sessions recover the zoom and extent management mode that they had prior to synchronization.

As mentioned above, the opening of new layers or maps may change the overall extent. This causes synchronization to be stopped. You can re-synchronize later by first deactivating and then reactivating it using the "Alt+Z+M" key combination or using the new button on the button bar, indicating that you require synchronization, but not mosaic.

When an incorrect synchronization is detected, you should reactivate as indicated above. Although it is unusual for incorrect synchronization to occur, it may happen if the size of the windows is changed during synchronization. The following example reproduces a situation requiring resynchronization: two sessions that contain a map with a different extent are mosaicked, they are synchronized (the extents are the same as that of the session that controls the synchronization: controlling session), the sessions are maximized again and a whole view is requested from the controlling session. Loss of synchronization is due to the fact that the session that controls the synchronization or the mosaic with synchronization determines the metrics of all the synchronized sessions until the synchronization is stopped (or until a "new whole view", open layer, etc. operation, which also stops synchronization, is executed). If the sessions are maximized again, the controlling session now has a new window extent and can be readjusted to a whole view, but the synchronized sessions cannot modify their extent until the synchronization is stopped because the system would otherwise be difficult to control.

5.3. Optional presence of scrollbars

The left-right and up-down scrollbars are now permanently visible, even in whole view zoom mode. We have decided that, as current screens contain far more pixels than a few years ago, this does not result in an excessive loss of space and that, on the other hand, their appearance and disappearance according to the zoom level was sometimes distracting for the user (and difficult to maintain in the programming code). We are aware that this a concession to our philosophy to maintain MiraMon as an application that gives priority to "viewing territory" and therefore "seeing map" (hence our reluctance to fill it with buttons, delegating this work to the menus and floating windows), but the 16-pixel width of the bars no longer seems to represent a serious inconvenience nowadays.

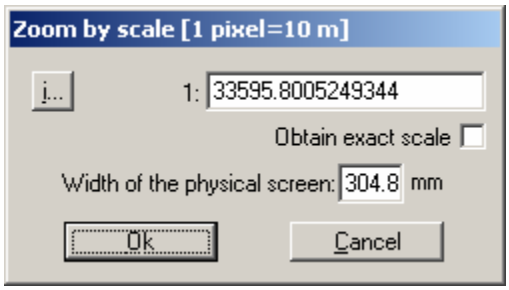
However, the “View | Environment | Show scrollbars” menu, which can be configured from the parameter file MiraMon.par (keyword `ShowBarsWindowPaint=` in the [METRICS] section, offers the possibility of working permanently without bars. This is a useful option for users who prefer to displace using the keyboard (arrow buttons for short movements, “Control+arrows” for page shifts, Control+Home or End/PgUp or PgDn to go to the ends) or using the “Displace” button (hand or pan).

5.4. Appearance of a status bar and new features of the Zoom by Scale

A status bar has been added at the bottom of every MiraMon session.
This status bar includes:

- **Cursor position**. This provides the same information as that contained in the traditional floating window which is activated with the F6 key: Column, Row (if the cursor is positioned over a raster) MapX, MapY (and its conversion to long., lat. if the Horizontal Reference System is of the Cartographic type) and color data of the point, with an indication from the index on the color palette if appropriate.
    - With more than one raster visible, the Column, Row information provided both in the F6 box and in the status bar corresponds to the raster located at that point and, if there is more than one, to the top one in the order of layer overlay. For a RGB raster, and due to the likelihood that in the future RGB combinations from different geographic areas will be supported, the coordinate is given in the R raster and, if it does not exist at that point, in the G raster, and if it does not exist at that point, in the B raster.
    - Information about cursor position is replaced by **process development information** such as the loading of a raster. Process information is therefore now provided in the status bar and not in the floating window.
    - The **complete text of cursor information can be copied to the clipboard using the Ctrl+F6 key combination** (mnemonically “Ctrl+C” to copy and F6 - the classic button of the MiraMon coordinate bar).
- The **numerical scale**
- The **graphic scale**

By clicking with the mouse on either the numerical or graphic scales, the **scale zoom** dialog box appears.



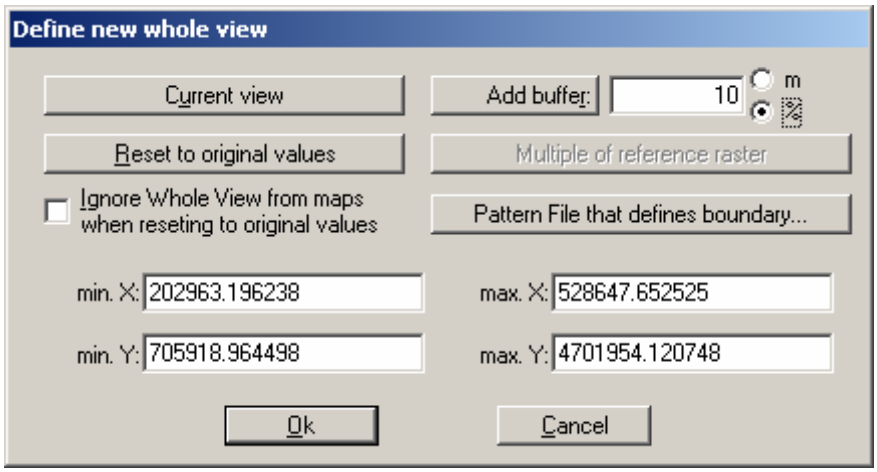
The new features in this box are:

- Indication, in the heading, of **screen pixel** size in map units. This is useful from a technical point of view. However, the "i" button is retained in case you wish to obtain the pixel size value in 15 significant figures (9 are given in the box heading; the final 0s are eliminated for clarity) or in case you wish to copy the data to the clipboard with Ctrl+C.
- New optional "Obtain exact scale" checkbox. In earlier versions, when you requested a particular scale on the screen, the program located itself at the nearest scale that allowed the whole extent to be viewed. It is now possible to indicate that you wish to see the exact scale. From this moment onwards, and until a whole view is requested ("Zoom | Whole view"), the program will move in multiples and submultiples of that scale. Of course, this option cannot be applied when zoom management mode is "Raster" (or "Automatic" but functioning as raster) because the scale is a function of multiples and submultiples of pixel size, which is what governs the zooms (and, therefore, the scales) in this mode at all times. If you click on "Obtain exact scale" while in this mode, a message will indicate this, allowing you to abandon the mode or accept an approximate zoom. In order for the exact scale to function perfectly, screen width must be correctly established, in millimeters, in the parameter file MiraMon.par or indicated in the space provided in the dialog box.

When the reference system is not cartographic (e.g. a scanned image, non-georeferenced photo, etc.), we have chosen not to show it, as the numerical or graphic scale is irrelevant. However, the possibility of clicking on the bar as a quick way of accessing the "Zoom by Scale" box has been maintained.

By default, the status bar is always visible. However, the "View | Environment | Show status bar" menu, which can be configured from MiraMon.par (keyword `StatusBarVisible=` in the [METRICS] section) offers the possibility of working permanently without the status bar. This is a useful option when you wish to gain space for the map. Although in this case the coordinates will not be visible, you can always use the floating bar that shows the scale by pressing the F6 key or from "Information | Show coordinates and colors".

5.5. Define new Whole view



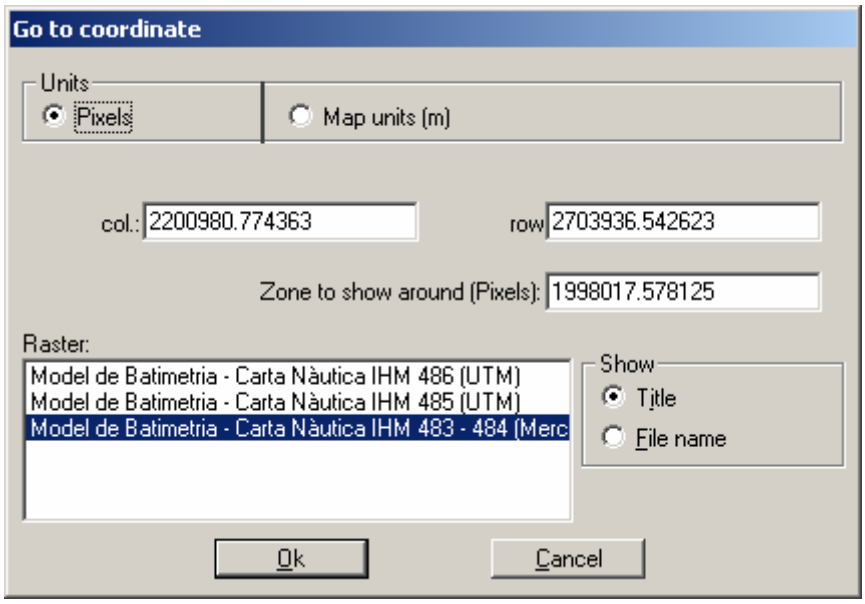
The dialog box which configures the definition of the new whole view has been expanded with the possibility to:

- Generate a space (buffer) around the extent expressed in map coordinates or in percentage of the total extent.
- Adapt the extent to multiples of the reference raster. This is a useful option if you wish to place the reference raster cells in the top left-hand corner of the screen (this is a “screen fitting” option), but it does not prevent you from seeing any part of the open layers.
- Copy the extent specification of the view of the bounding box of an external layer, which acts as a pattern file.

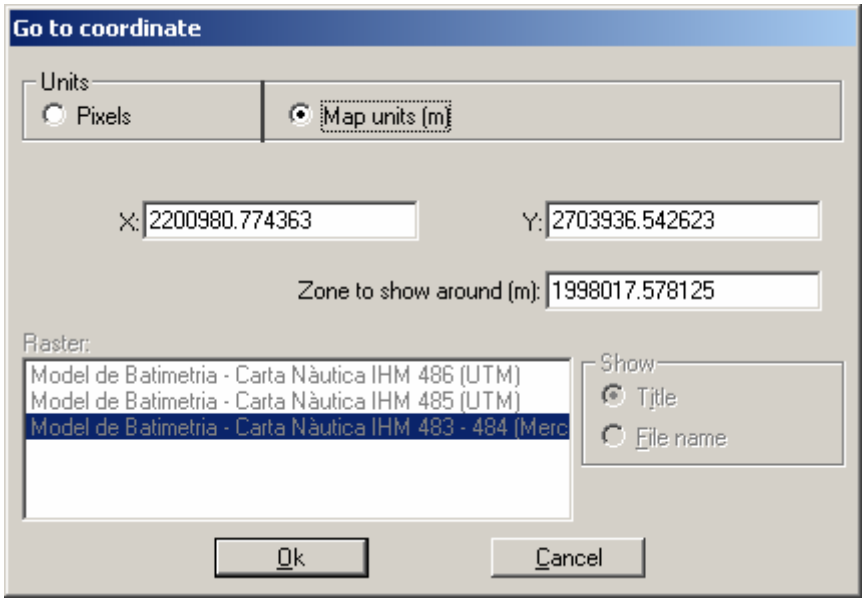
In addition, an optional checkbox called “Ignore Whole View of the maps when restoring original values” has been added. This new option is useful when you wish the recalculation of the whole view to ignore the [WHOLE_VIEW] section of the open maps, which sometimes contain additional space for aesthetic reasons or simply tend to have a greater bounding box than the total once one of the layers has been chosen.

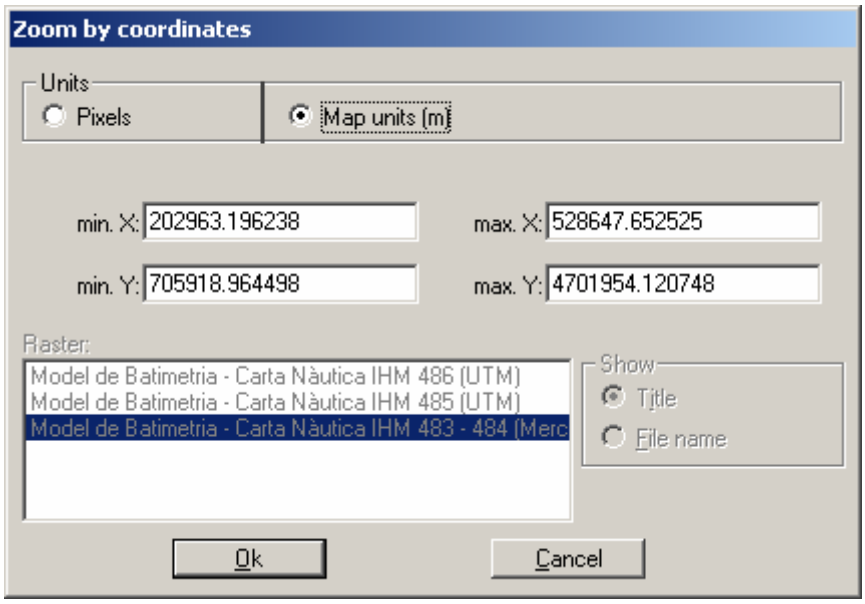
5.6. New “Zoom | Go to coordinate…” and “Zoom | Zoom by coordinates” dialog boxes

The dialog boxes for going to a particular coordinate and zoom by coordinates support the pixel coordinates which refer to any of the loaded rasters. If you wish to carry out this “search” in pixel units, it is necessary to indicate over which open raster this should be done because the metrics are defined by the characteristics of each raster.



As in earlier versions, the program also supports these zoom options defined in map units.





5.7. Redraw

The "**redraw**" function (equivalent to "View | Redraw") has been assigned to the **F5 key** in line with the Windows standard (for example, F5 updates a window of the file explorer to see if there is a new file or if one has been modified).

In addition, the **"Interrupt redraw with the Shift key (⇧ arrow) depressed" has been extended to interrupting the redraw of WMS layers**. This is useful when a WMS layer is slow to redraw and there is an open raster in the background that can be used as a reference and you do not wish to hide the layer.

5.8. Legend

24-bit raster compositions as well as BMP, TIFF, JPEG, JPEG2000, SID, ECW, etc. layers, which until now did not appear in the legend, are now present to enable you to take advantage of the useful ability to define the layer as viewable or queryable, and so that it is possible to indicate the title with which its presence can be identified in the legend during printing. The program assigns a default title derived either from the file name (or from the 3 files for an RGB composition) or, if there are metadata which display a title for the layer, from this title.

It is also possible for the legend to be, by default, always visible through the MiraMon.par keyword IgnoreShowLegend=1.

Finally, the layers that are not visible, not visible by scale or not visible by geographic bounding box are grayed out in the legend box to indicate that they will not appear in that view despite the fact that they are in the set of layers you are working with. Moreover, if there are several categories, these are not shown in the legend to avoid giving unnecessary details about the layers that cannot be seen. However, the functions associated with the layer in the legend

continue to be accessible (you can double click on the node that gives access to the menu of the layer's legend, make it visible or not, etc.). Regarding printing, the layers that are not visible, not visible by scale or not visible by geographic bounding box are not shown in the legend because we believe that it is not worth keeping them visible if they can neither be seen on the map nor are useful for accessing the functions. However, if you really wish the layer to appear in the printed legend although it is not visible, it is always possible to change this behavior using the "Legend…" button of the printing design window which now includes an optional checkbox called "Show layers that are not visible by scale or by extent". The status of this checkbox can be configured using the keyword PrntLegShwNoVisLyrs= in the [MiraMon] section of the MiraMon.par file.

6. Improvements and new features regarding types of data

6.1. Rasters

Multiband rasters, in which a single relation's file (REL) groups together several images managed jointly from the Metadata Manager, have been supported since version 5. This may be useful for remote sensing multiband images, for temporary series of changes in land use or for showing relief shading which can be queried to obtain altitude, slope, etc. In addition, the new version supports the loss of some of the bands.

Types of raster that can be opened directly:

The list of files that could formerly be opened by MiraMon has been increased to:

***.img; *.jpg; *.jp2; *.jpc; *.j2c; *.sid; *.tif; *.ecw; *.bmp; *.rle; *.dib**

This can be done directly, either from the "File | Open raster" menu or by dragging and dropping them over the application or by double clicking on the file in Explorer or launching them from the command line (MM32 NAMEFILE.EXT). MM32 continues to support **JPEG2000** and **MrSID** gray, 24-bit and multispectral (showing the desired band) files and these can now be read more quickly, as mentioned in section 4 of memory management.

MM32 makes it possible now to open TIFF (and GeoTIFF) 1 bit per pixel, 8 bits per pixel (with color palette and grayscale) and 24 bits per pixel (RGB) as well as TIFF multiband Remote Sensing images. Regarding TIFF multiband, it is also possible to select which bands to assign to each RGB component (which is useful, for example, when opening SPOT 4-band images). This can done from the "File | Open raster" menu (the TIFF files are listed in the same way as any other typology among the types supported by the pull-down menu) or by dragging and dropping the TIFF over the MM32. They can also be included in MMM and MMZ files as well as query by location and by attribute. This works in basically the same way as the TIFIMG application, apart from the fact that a definitive IMG to work with is not generated and some of the advanced options of the fine importing tools are not available. Among the new functions for reading TIFFs it is worth highlighting the support for reading and writing integer-

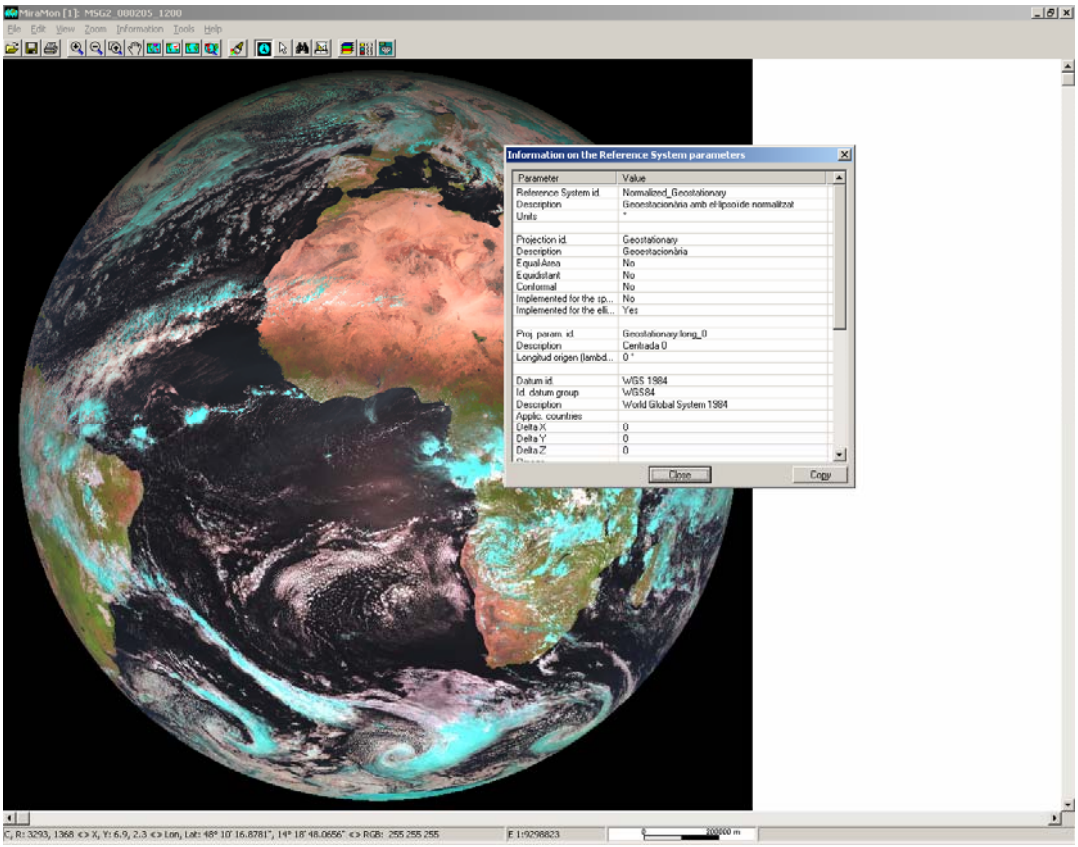
type and long-type TIFFS, the generation of an interface necessary for the “Automatic Generation of the name of output files”, which are especially useful for TIFFs that have a large number of bands, or the possibility of transforming TIFF files of up to 4 Gb both in multiband with “Tiled Image” structure and with “Planar configuration=1”. In addition, the type of data is adapted to the resulting rasters, thus avoiding the appearance of negative values in TIFFs with values that were originally unsigned: if a “char” file has values between -20 and 120, it is written as a short to preserve the negative values; if a file has values between 0 and 65536, it is written as a long to preserve the range that would not fit in an unsigned short. As no actual cases of values between 0 and 4200 million have been found, the “unsigned long” to “real” conversion has not been activated.

MM32 now also allows you to open ECW 8 bit-per-pixel files directly (grayscale) and 24 bit-per-pixel (RGB) files directly. This can be done using the “File | Open raster” menu (the ECW files are listed in the same way as any other typology among the types supported by the drop-down menu) or by dragging and dropping the ECW file over MM32. They can also be included in MMM and MMZ files as well as query by location and by attribute. This works in basically the same way as the ECWIMG application, apart from the fact that a definitive IMG to work with is not generated and some of the advanced options of the fine importing tools are not available. Opening ECW files requires the presence of the NCSEcw.dll, NCSUtil.dll y NCScnet.dll libraries in the MiraMon directory.

Types of rasters that can be imported

By import, using the “File | Import” menu:

- Popular image-processing formats, such as **PGM** or **PPM**.
- Formats of satellite image distributors, such as **CEOS** or **NDF** from **Landsat MS-TM-ETM+**, **TIFF+Dimap** from **SPOT**, **HDF** from **Terra** and **Acua** (**Aster**, **MODIS**) or **Meteosat** (**SEVIRI**) (see figure below), etc.



- Popular Remote Sensing and GIS formats such as **LAN/GIS** from **Erdas**, **GRD** from **Surfer**, **RF** from **Zebra**, **CTL** from **GrADS**, E00 from Arco/Info or RST from Idrisi (see section 11.9).

It is worth pointing out the following information in relation to the raster data import module:

- TIFFIMG allows the user to extract multiband information from TIFFs of more than three bands (e.g. SPOT images with 4 spectral bands), read *tile* models, import 16 bits-per-pixel channels, etc. Regarding GeoTIFF, reading georeferencing has been improved and, apart from the coordinates, it is now possible to recover the name of the horizontal reference system, if this file exists.
- A number of applications for reading data from Remote Sensing formats have been created, completed or consolidated. These include CEOSIMG (from ESA), NDF (NLAPS Date Format from USGS), SPOT-DIMAP (XML), QuikScat, AMSR-E (Acua satellite), etc. with particular attention to the recovery of metadata.

Formats with 24 bits-per-pixel direct type, such as JPEG, JPEG2000, TIFF, etc., allow direct viewing in 16 million colors (RGB), while other formats may, as is usual with MiraMon, be viewed with their own color palette or gray-scale as well as combined in RGB triplets to give combinations of 24 bits of visualization per pixel. Each non-RGB raster may have 1, 8, 16 or 32 bits of data per pixel. The 32-bit option is suitable for long integer numbers (of more than 2100 million values per pixel) or for real numbers with 6-7 significant digits. At the same

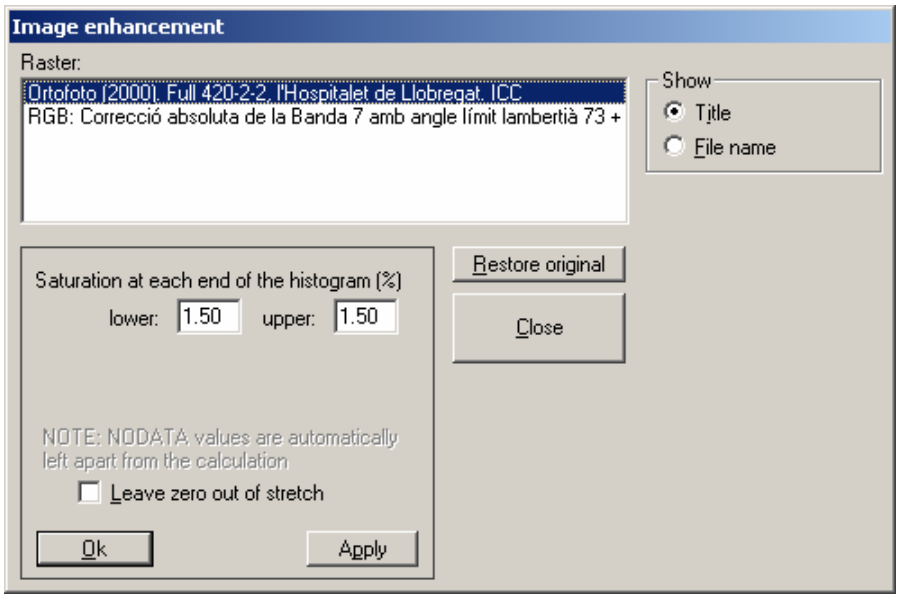
time, all the formats of more than 1 bit per pixel accept lossless compression (for lossy compression, you can opt for JPEG, JPEG2000, SID or ECW).

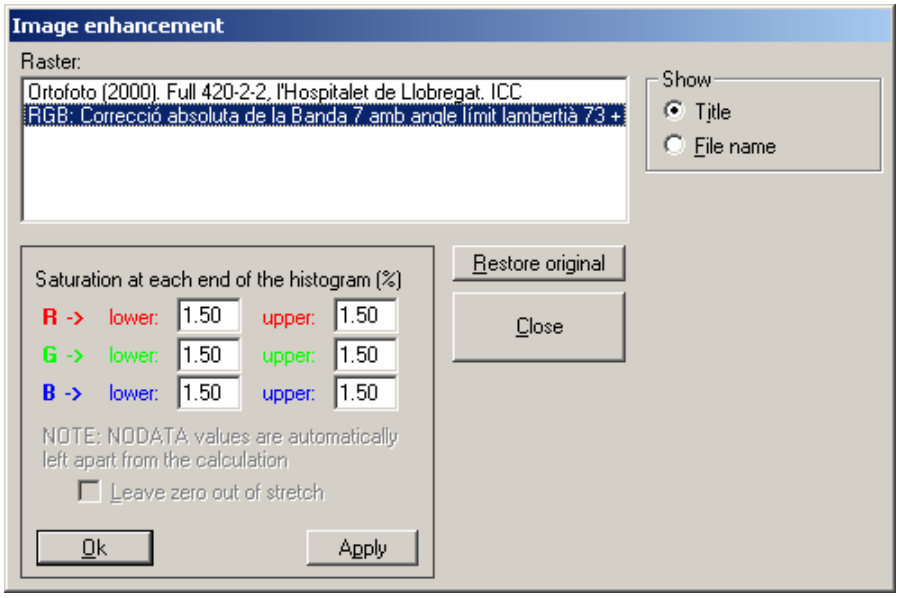
It is our intention to ensure that those formats which are most requested by users can be opened directly, without the need to import them, in order to speed up the viewing process and leaving import processes for when you wish to carry out a more stable conversion and manipulate the resulting files with the MiraMon (MSA) support applications. We will also monitor the appearance of new formats that may be worth incorporating into direct visualization or as new import tools.

The “File | Save as IMG/JPEG…” option is now labelled “File | Save raster/WMS as raster…” and also allows the user to save one of the open WMS views as a raster file whose resolution can be determined by the user.

Image improvement

Image improvement is possible for any open raster (IMG and 24-bit IMG). The “Restore original” option has disappeared from the menu and is now a button on the optimization box. From the same window you can control viewing of any open raster and perform image improvement without having to close the box. This facilitates play in fine image improvement.

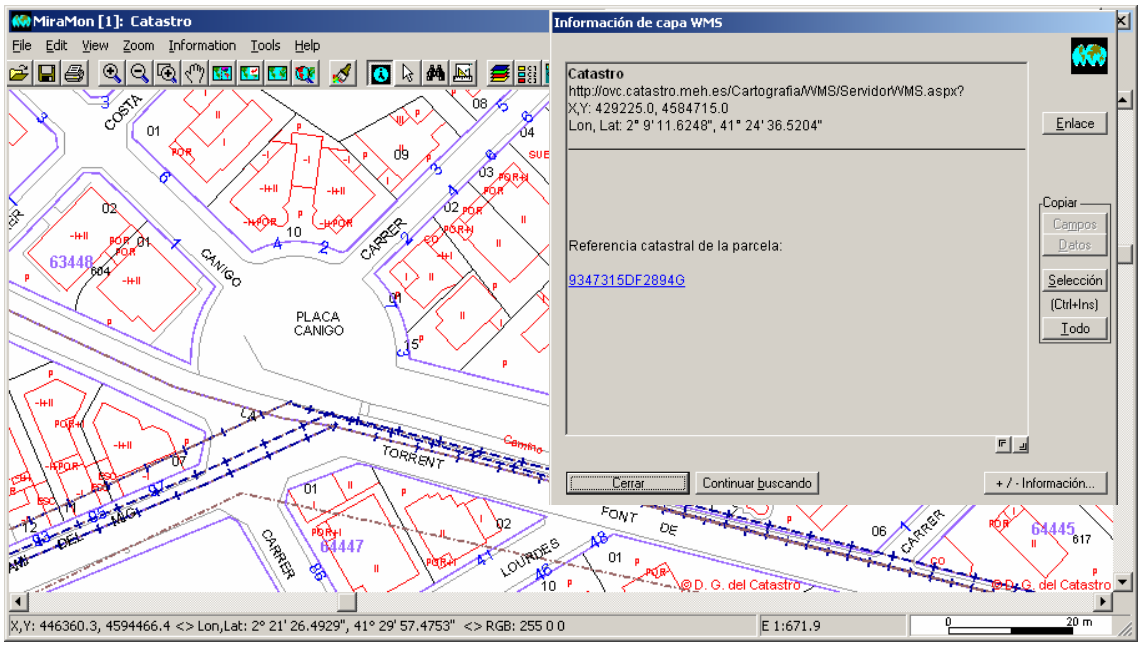


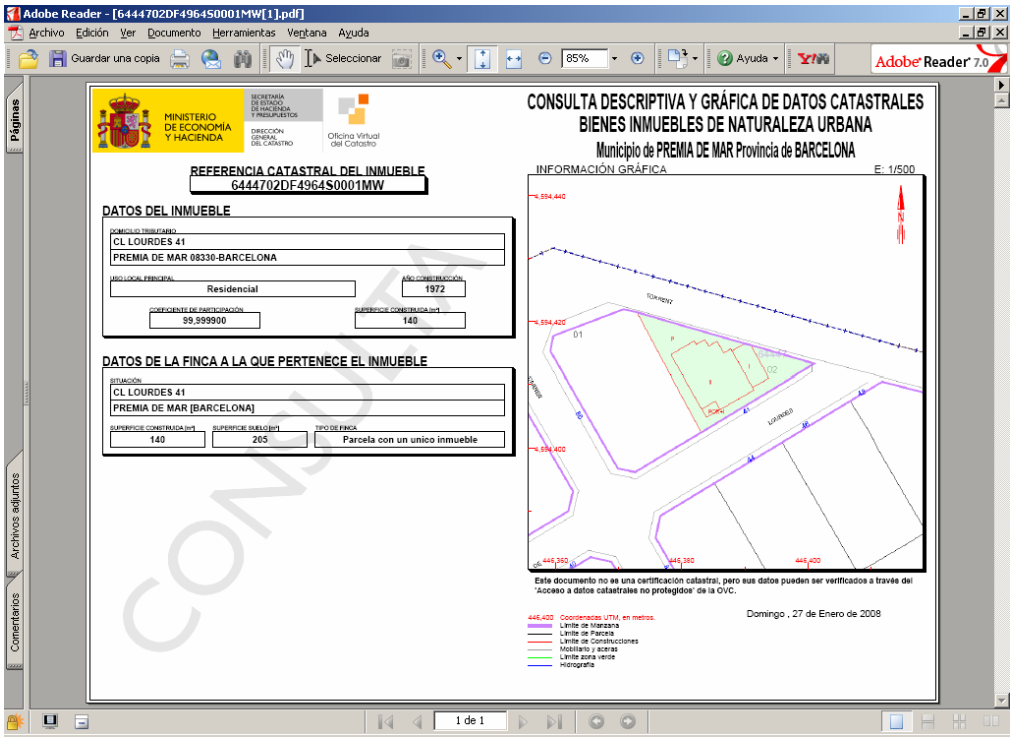


6.2. WMS layers

WMS is the acronym for Web Map Service, a specification of the Open Geospatial Consortium (OGC) designed to standardize access to cartography by means of Map Browser Systems. A client (usually an Internet browser) who has the necessary WMS browsing equipment (via Javascript, for example) may access cartographic servers which use the WMS specification, regardless of the cartographic provider and the manufacturer of the technology.

The MiraMon team constantly updates its list of WMS servers with data that may be useful for program users. By pressing the "**Update list of servers**" button when connected to Internet, you can download the updated list from the MiraMon web. As of 28-01-2008 there were over 30 servers available. The screenshot below shows a query on the server of the cadastre of the Spanish Treasury Department which allows you to open a link to the cadastral information file.





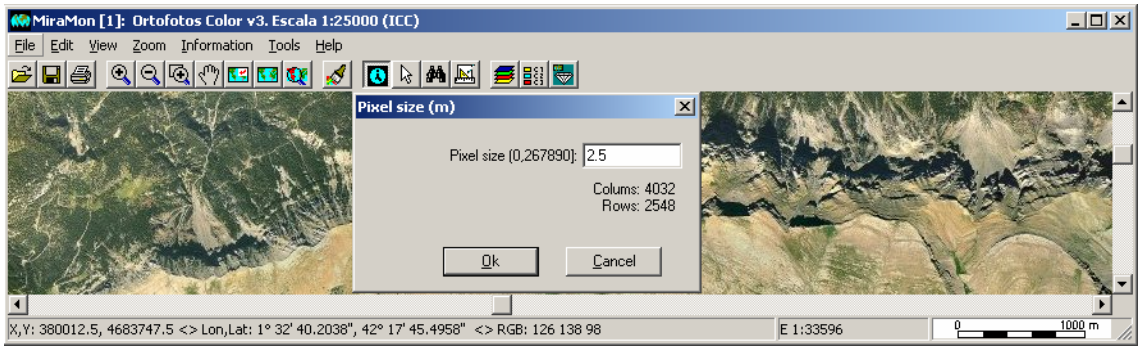
If you are aware of any particularly useful servers that could be added to the MiraMon list, please contact us at: contacte@MiraMon.uab.cat.

Improvements in WMS query interpretation

WMS response to a specific location query of ESRI servers is interpreted in a table returns.

Possibility of saving a WMS view as a raster

Using the “File | Save raster/WMS as raster” option, it is possible to save any open WMS view in either a JPEG file, a file of the same type as that used to read a WMS layer (JPEG, PNG, etc.) or in a BMP file. Until now, files saved from the WMS view had the same resolution as the current WMS view and a small metadata file was created in order to record the geographic bounding boxes, among other features. MiraMon version 6 incorporates the option of saving the current view of a WMS layer as a raster at any pixel size accepted by the server and not only at the current resolution.



At the same time, the metadata of the saved file with the WMS views are enriched with the inclusion of a process that includes information about the WMS request that generated the view.

6.3. Vectors

Types of vectors that can be opened directly

The list of vectors that could formerly be opened by MiraMon has been increased to:

***.pnt; *.arc; *.nod; *.pol; *.vec; *.shp; *.dxf**

This can be done directly, either from the “File | Open structured vector” or “File | Open unstructured vector” menus, by dragging and dropping them over the application, by double clicking on the file in the Explorer or by launching them using the command line (MM32 NAMEFILE.EXT).

It is now possible to open SHP files (shapefiles) with MM32. This is accomplished using “File | Open structured vector” (the SHP files are listed in the same way as any other typology among the types supported by the drop-down menu) or by dragging and dropping the SHP file over the MM32. The files can also be queried by location and attribute, symbolized, etc. and incorporated into MiraMon Maps. This works in basically the same way as the SHPTop application, apart from the fact that a topological structure is not generated and some of the advanced options of the fine importing tools are not available.

Another new feature is the dynamic reading of DXF files. These can be opened using the open vectors option, dragging and dropping them directly from MiraMon, etc. Open files can be queried by location and attribute, resymbolized, etc. DXF files produce a MiraMon map with the same number of layers as the

original DXF file and with fields that contain information about DXF symbolization. At the same time, if a layer had more than one basic geometric object, the corresponding files are also generated.

6.4 MiraMon maps

MiraMon maps are now given a version number. This begins with version 2.0 in order to reflect:

- The possibility of **accommodating more than one raster**.
- The possibility of **overlaying the layers in any order** regardless of whether they are of raster, WMS or vector type.

The version is recorded in the new section [VERSION] of the map with the keywords Vers=2 Subvers=0.

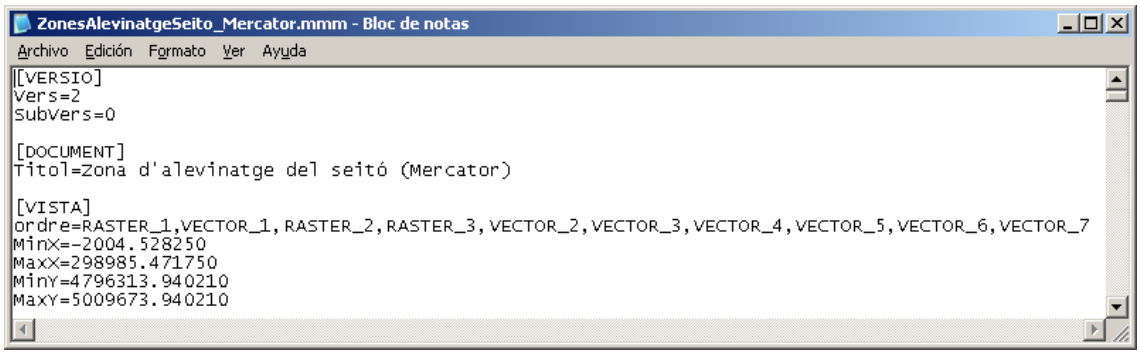
Similarly, MiraMon maps are also given a new code, Order=, in the [VIEW] section, which determines the order in which the layers appear. If the code is not there, the order is assumed to be that of versions that are earlier than MiraMon version 6.0 (rasters at the base and vectors above, WMS in the middle). Order= indicates the layer sections that should be read, starting with that which will remain “at the base”.

In addition to the obvious usefulness of the new possibility, it allows map-writing programs to exploit “unorthodox” uses which can be useful in some situations, such as “skipping” a particular section (e.g. indicating Order=RASTER_1,VECTOR_1,RASTER_RGB_3, RASTER_4).

Another option, respecting the prefixes "RASTER_", "RASTER_RGB_", "LAYER_WMS_" and "VECTOR_", is to replace the number with a name. For example, writing a particular layer in a "VECTOR_COUNTY" section. When there are no digits after the prefixes, MiraMon assumes that this is a name that the user wishes to use when saving a map layer again. However, if when loading a section during reading the programs detects that the name has already been used for another layer, the numbering criterion is adopted. Due to the fact that using non-numerical correlative values after prefixes is incompatible with earlier versions of MiraMon, and MiraMon will only search for these sections in the MMM if they appear in the “Order=” code, advanced users are advised to carefully evaluate the possibility of using section numbers which are not correlative numerical values.

The new code is not present in maps that do not contain layers. On the other hand, when reading single layers the program continues to follow the criterion of sending the rasters to the bottom because their level of opacity is usually higher. Similarly, WMS layers continue to be situated by default between rasters and vectors. The aim of this is to try to avoid the need to relocate individual layers that have just been opened in the order of layers’ overlay. The order can be managed, as always, from the order and properties of layers box and appears in the legend, screen redraw and printing option, as well as in the drop-

down layer menu with which you can carry out query by attribute or interactive query.



Moreover, MiraMon Maps (MMM) can specify the position and size of the MiraMon window on the screen in the [WHOLE_VIEW] section with the codes left, top, Width and Height. The codes of the "left" and "top" positions of the window are given in screen pixels assuming origin (0.0). The codes of the "width" and "height" sizes are given in screen pixels.

6.5. Databases

New single, dynamic table

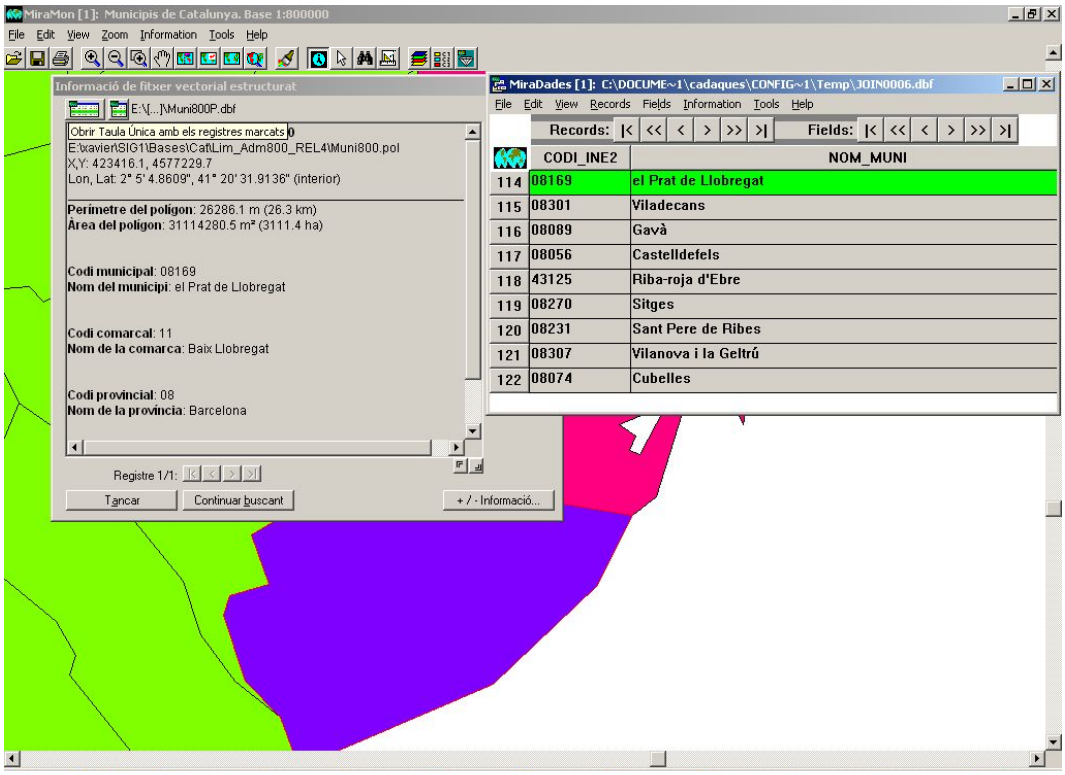
When, in an REL4 or higher vector layer, there are other alphanumeric tables related with the main table, a single table is generated automatically when a query by attribute, an interactive selection or an explicit request to view this single table with MiraDades is performed. The single table shows the different fields resulting from the display of relationships between all the tables which make up the database linked to graphic objects.

The dynamic single table is generated in the user's private temporary directory with the name JOINNameTableMain.dbf and JOINNameTableMain.rel (for example, if the main table was CountiesP.dbf, JOINCountiesP.dbf and its corresponding REL file will be generated. The single table will be erased when the layer or the MiraMon session containing it is closed. If the name is being used by a pre-existing file in the directory, the name generated is JOINNameOriginalLayer####, in which #### is a numerator which can have values between 0000 and 9999.

Please note that **it is possible to save the DBF of the single table in a different directory from the temporary one** (%temp%) by following the instructions in section 11.2, "**New features of MiraDades**".

Prevent generation of single table: the generation of a single table may take some time if there are a lot of relationships or the tables contain a lot of fields or many registers, or if there are tables located on the Internet and access to them is slow. As the features of the single table are often not needed and you may wish to prevent their generation, an optional checkbox has been included called "Generate single table" in the "Information | Advanced options" dialog box which enables you to prevent the single table being generated.

The **query by location** dialog box includes a button called “Single table” in which you can see, marked in green, the register queried in the single table resulting from the display of the database linked to the graphic objects. Please bear in mind that marking a register in green has no special effect on the register other than making it easier to locate. It is possible to unmark the register by pressing “Control+Register numerator button” or from the contextual menu in the register buttons. Displaying this single table should not involve editing its contents because the changes that are carried out have no effect on the real tables of the database to which the graphic layer is linked. However, if layers with only one linked table as the single table (the main table) and the main table are the same table, it is possible to edit the single table directly.



The Single Table button in query by attribute and interactive selection 

A “Single Table” iconic button, which is wider than that in “Main Table”, has been added to the interactive selection dialog box. This launches the dynamic single table resulting from the display of the database and shows, highlighted in yellow, the registers selected (interactively, by attribute or by layer). It should be noted that selecting a register in yellow not only helps you to locate it, but is also a true selection marker; the register is deselected by pressing “Control+Caps+Register numerator Button” (or from the contextual menu of the register button), or by selecting another unselected register by following the same procedure. If the MiraDades session was opened from MiraMon, this session will tell MiraMon to deselect or select the relevant registers. The content should not be edited from the display of this single table as any changes made will have no effect on the real tables in the database to which the graphic layer

is linked (please note that you can find out the name of the table to which the register belongs in order to edit it from GeMM and from the field statistics). However, if layers with only one linked table as the single table (the main table) and the main table are the same table, it is possible to edit the single table directly and confidently.

Small help labels ("Tooltips") appear when you place the mouse over the "Main Table", "Single Table" and "Load single table" buttons.

Important note: If the table has "to many" relationships, this should be indicated in the GeMM "Type of Relationships" specifications because if there are potential multiple registers, but in the metadata it is documented that there are only "to 1" relationships, MiraMon will ignore later occurrences.

Once the selection has been made, when saving it as VEC you can choose **any field of the database**, not only fields of the main table.

If you wish to check whether there have been changes to any table in the relationship diagram that may involve changes to the contents of the single table, it is possible to force it to be reloaded using the "**Load single table**" button in the results of queries/interactive selection dialog box, accessed with the right button from any point of the application: 

Direct opening of points located in DBF, MDB, Oracle, etc. tables

The new "Open points in table…" option, which permits the user to open a (DBF, MDB or any other ODBC route: SQLserver, Oracle, XLS, etc.) table directly, as if it were a point layer, is accessed via the "File" menu. For MDB and ODBC, it is possible to define the table by means of an SQL query. The options are basically the same as those of the BDPNT application. If you indicate which field contains the entity indicator, it is possible to maintain the link with the original table. If not, a complete import of the table's data will be carried out. It is also possible to opt to perform a (temporary) dynamic process or indicate the name of a PNT destination file in order to have the point layer permanently in PNT format. In this last case, it is also possible to link the file to the original table or import all its data.

Moreover, the function has intelligent heuristics in order to calculate which fields are more likely to contain the X and Y coordinates of each point and offer them to the user. In any of these options it is possible to save a map as well as generating an MMZ. The maps contain the link to the original table as well as the appropriate parameters (CampX, etc.) if you have opted for the (temporary) dynamic process, whereas they contain the usual vector layer parameters if you have opted for complete import.

All MMM save options and MMZ compression options for Internet distribution also function with this new option. This option also allows you to drag and drop a table onto MiraMon to open it. Despite the convenience of this option for the direct opening of points located in a table, please bear in mind that BDPNT also

allows you to structure the imported point layer topologically. This is useful when you suspect that some of the points may be duplicated in the database.

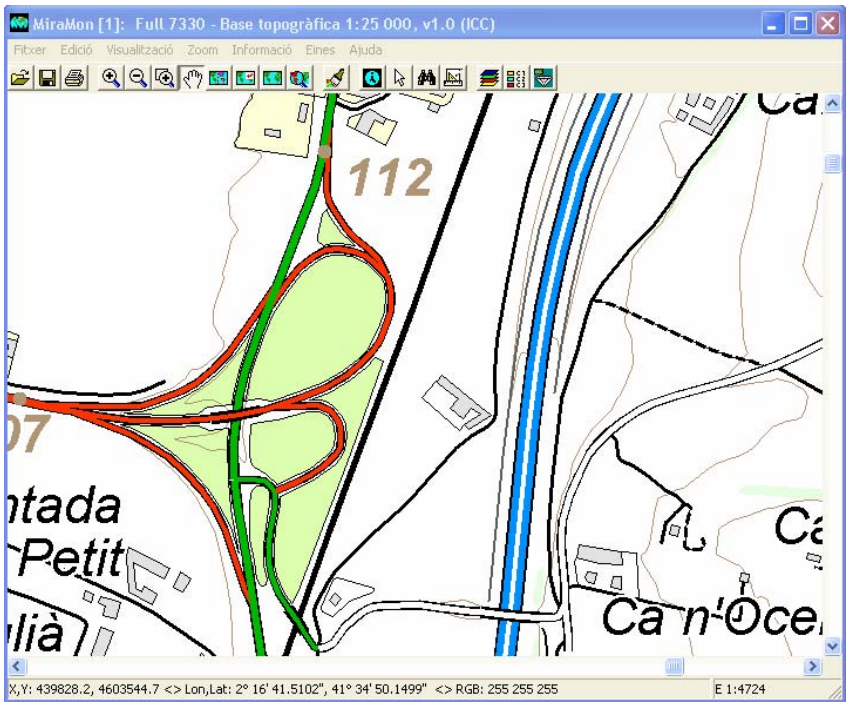
Additionally, it is possible to **directly open an REL which refers to a dbf, mdb, oracle, etc. table and define its name, metadata, etc**. The REL may be opened in command line or using the “Open points in table” option of the “File” menu as well as by dragging and dropping it onto MiraMon or as part of an MMM. This REL behaves like a distributed REL and can be used as a pattern for regularly opening a table located in a database without having to indicate the title, units of the fields, etc. each time. For more details, see the technical document:
DirectOpening_and_with_Metadata_OfPointsLocatedInTables_v6_xx.doc.

If more technical details are required, you can order the document:
DirectOpening_and_with_Metadata_OfPointsLocatedInTables_v6.doc.

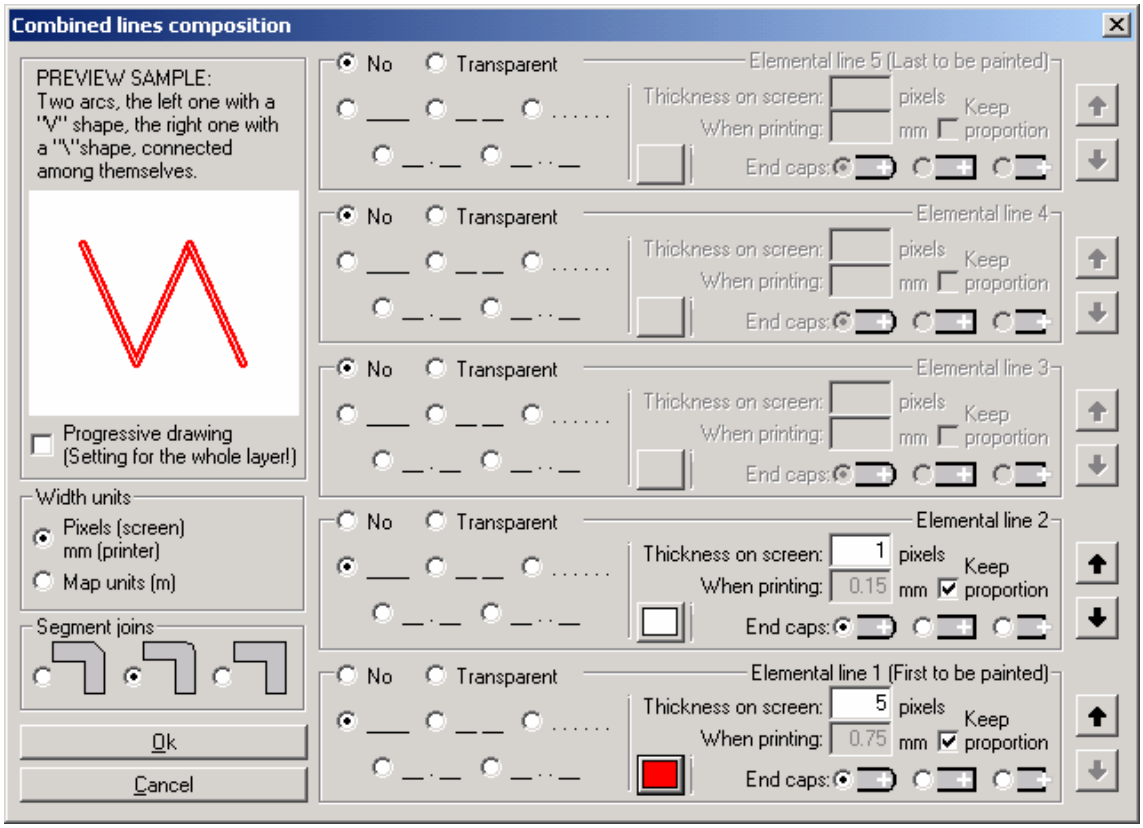
7. Symbolization and fonts

More possibilities for the assignation of symbols for point vector entities. In addition to EMF and WMF files, the program also supports JPEG, PNG, BMP and GIF files (they can even be mixed in the same symbolization table).

Combined symbolization lines and partial multiseries. The new combined symbolization lines allow to represent complex lineal elements correctly as roads, highways, etc. As it is common in MiraMon, each elemental line's width that form the combined line can be indicated in pixel units or in map units; moreover, when indicated in pixel units, printing's width is configurable in centesimal millimetric precision. Each elemental line can be configured in terms of type of line (solid, by strokes, etc.), width, color, shape on the borders, etc. The figures below show 3 examples of combined line, as well as its look in a 1:25000 map.

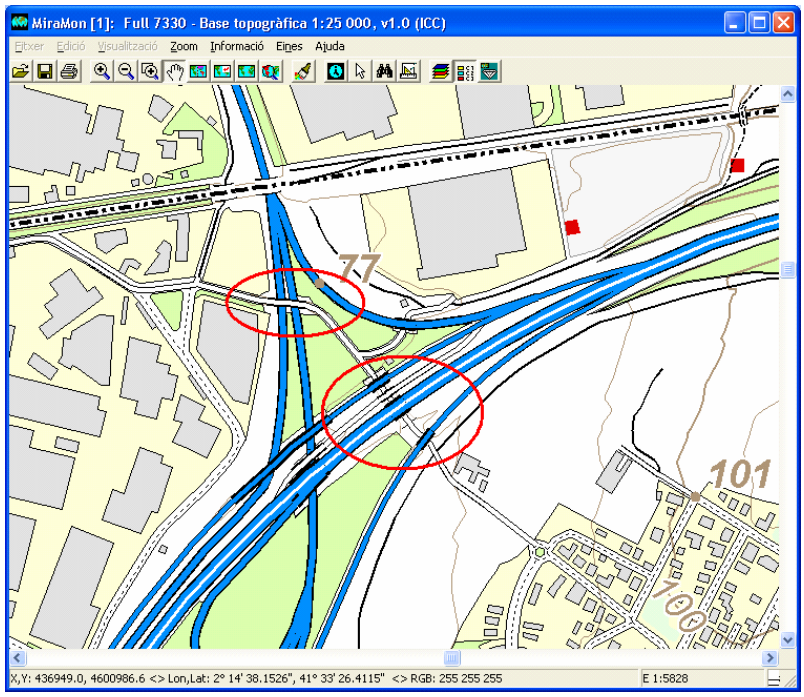


Combined lines present a wide variety of technical possibilities (can be specified as constant for a whole map, by object, by an automatic symbolization database, etc). The correspondent dial box allows to design combined line composition of interest:

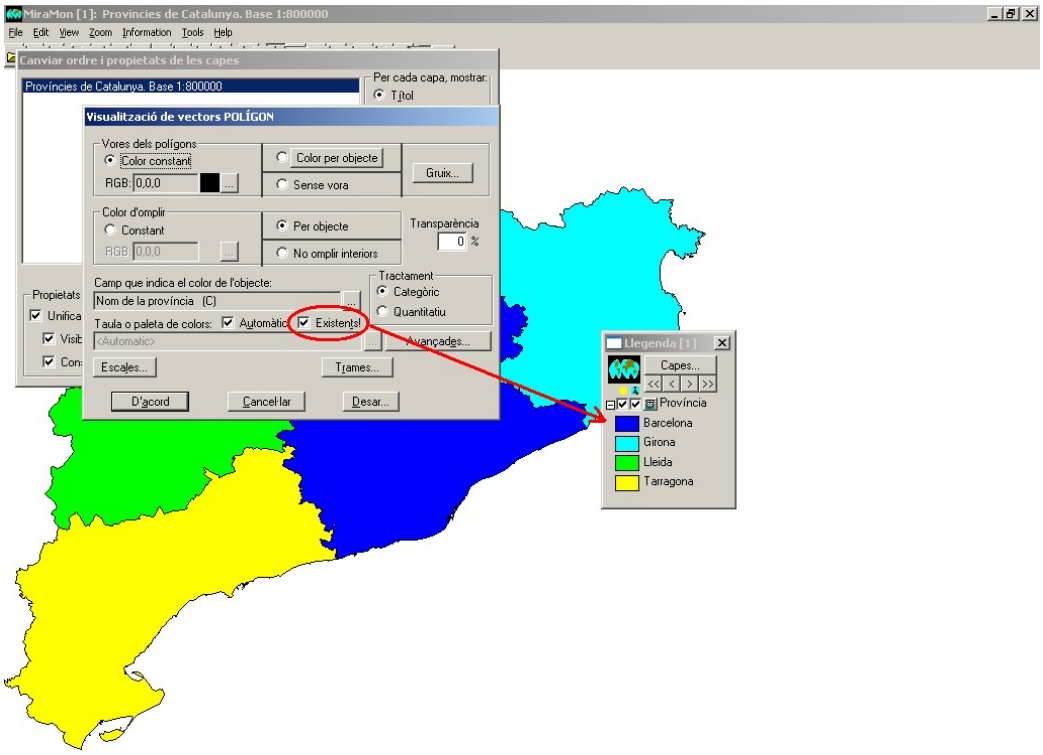


See the technical document SimbolitzacioLiniesCombinades_v6 for more details.

A specially interesting property is the use of combined lines with partial multiseries. This property makes possible, for instance, to draw a highway section over or under a road. Combined lines are perfectly compatible with the use of partial multiseries, so it's possible to get some effects as the ones shown in red on the figure below.

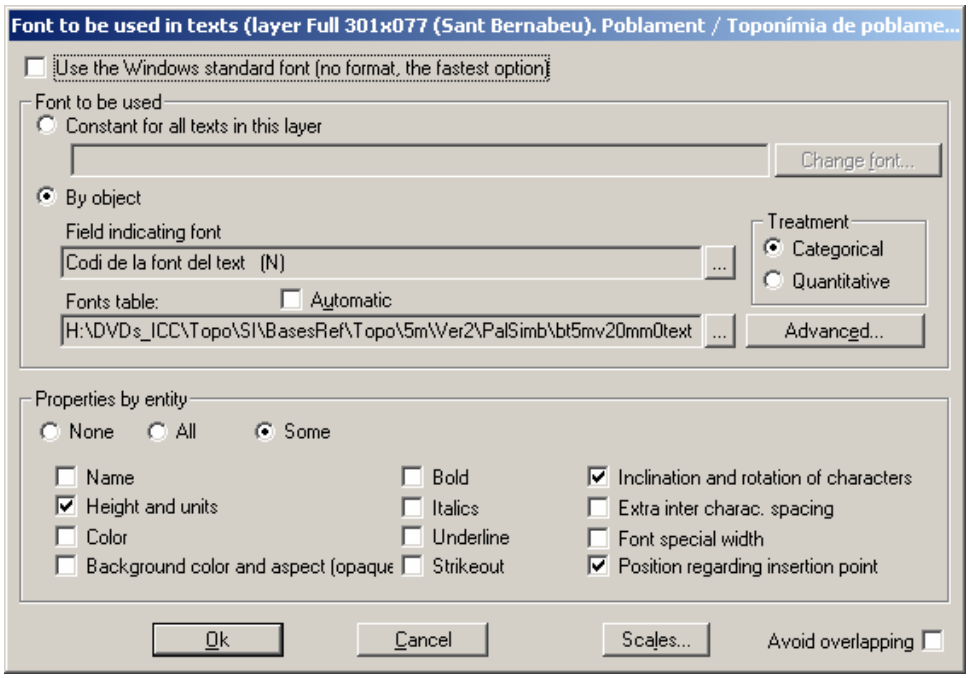


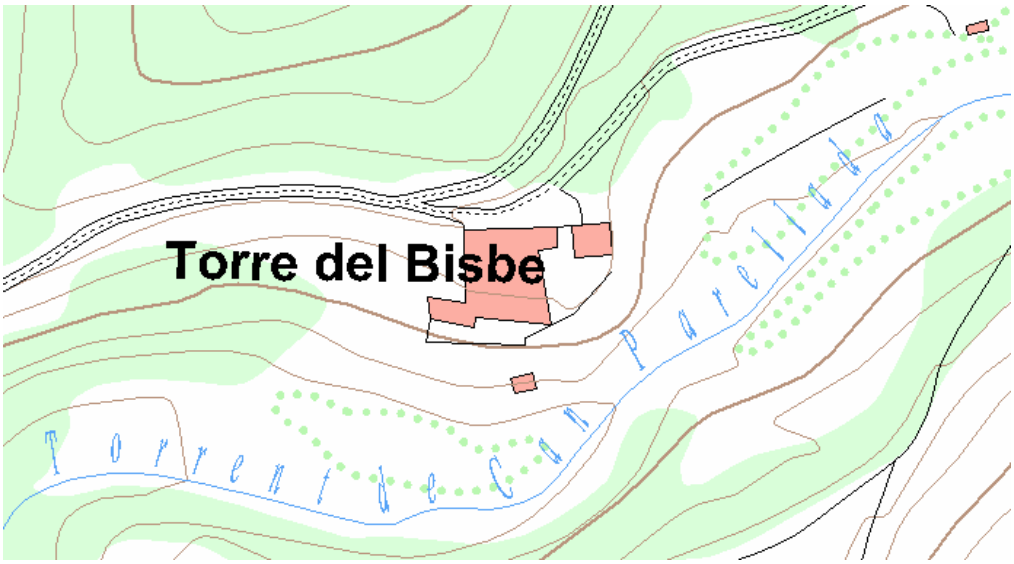
Automatic symbolization permits inclusion in the legend of existing elements only. In automatic symbolization modes you can indicate that only those that are really present in the layer are symbolized ("Existing elements" button) and not all those contained in the field with which you are symbolizing. This is especially useful when you are symbolizing with a field which contains items which in reality have never been linked with any graphic object. For example, a map of Catalonia in which the provincial code links up with a dictionary table that indicates the name of the province for each provincial code of Spain and which is symbolized by the name of the province. In this case, selecting the "Existing elements" button prevents the display in the legend of a list of many provinces which are not in fact on the map and restricts legend symbolization to the provinces which are really present.



Another typical example is to clip or select a land use layer. If you do not wish the legend to display all the elements, but only those in the subset resulting from the selection, this can also be achieved by activating the “Existing elements” button.

New possibilities for text symbolization: more accurate size, font symbolization tables for categories and properties individualized by object. Text symbolization for points supports symbolization tables as well as individualized fonts for each text. The illustrations below give an idea of the potential of this improvement:





Specification of the Properties of a font in MMM and REL files and in MiraMon symbolization tables (e.g.“/NArial/H12/B”)

As first established in version 4, a font is specified by means of a text chain which may be in an REL or an MMM file if applied to the whole layer, or in a field of a font symbolization table, etc. The format of this chain consists of a series of labels that set the different **properties of the font**. Each property is specified with a slash (/) followed by a distinctive letter (it distinguishes between upper and lower case letters) followed, if required, by the value of the property. The following is a list of the properties supported:

Properties with a compulsory value *(if these are specified, they should be followed by a value).*

- **Font name (*FaceName*): /N**. Example: /NArial. If not specified, Arial will be used.
- **Font height (*Height*): /H.** In typographic units, unless you expressly indicate that it should be in map units using /M. Examples: /H12 (height 12 points); /H200.5/M (height 200.5 map units, usually meters). If not specified, a 10-point font will be used.
- **Font color (*Color*): /C**. The R,G,B components are then specified, separated by commas. Example: /C255,0,0 (red). If not specified, BLACK will be used.
- **Font background color (*Background*): /K.** The color is specified in the same way as for /C. If not specified, WHITE will be used. If you indicate **/h** (note the lower case letter!), the background color is used for a halo and not for the whole background rectangle.
- **Font slope (*Escapement*): /E**. The font slope is then specified in degrees (to one decimal place) anticlockwise. Examples: /E450 (45º), /0225 (22.5º). If not specified, 0 (horizontal text) will be used. This property is not applied in the case of dynamic texts that label lines.
- **Rotation of each character (*Orientation*): /O**. The rotation is then specified in degrees (to one decimal place) anticlockwise. Examples: /O450 (45º), /O225 (22.5º). If not specified, 0 (unrotated letters) will be used. The same value as the slope is usually used. This property does not work with the 9x versions of Windows and is not applied in the case of dynamic texts that label lines.
- **Extra spacing between the characters of the font (*InterCharSpacing*): /i**. The extra spacing between the characters is then specified in typographical points. Example /i12. If not specified, 0 will be used (without extra spacing).
- **Extra horizontal stretching of font characters (*Width*): /x**. The horizontal stretching of the written characters is then specified. Example: /x9. If not specified, 0 (standard) will be used. For reference, in an Arial font with a height of 10, a value of 6 gives the standard result and lower values compress the font, while higher ones stretch it.
- **Distance from point of insertion: /X and /Y:** the distance between the beginning of the text and the point of insertion is then specified. The distance is expressed, by default, in pixel units. If you wish to express them in map units, you should indicate **/m**. If it is in pixel units, the positive distance is the classic one for rasters (to the right and downwards), while if it is in map units it follows the usual Cartesian criterion (to the right and upwards). Example: /X25.5 /Y33.7 /m. If not specified, 0 is used (the text begins at the XY coordinate of the point). It is possible to set only one of the two properties, in which case 0 is used for the unspecified one. See also the property /POS below. Note that the "/X" property is not the same as "/x".

Properties with optional value

- **Thickness of the font (Weight): /B**. This may be followed by a whole value if you require a value that is different from the standard one (value 700, equivalent to bold). Examples: /B (standard bold; /B400 (normal font, equivalent to not setting the /B property); /B100 (very fine font);

/B900 (very black font). These special values cannot be incorporated from the menus because the standard user does not need them. However, they can be written manually or with programs in the chain in order to show emphasis, if this is required.

Enabable properties

- **Italic font (*Italic*): /I**. Example: /I.
- **Underline font (*Underline*): /U**. Example: /U.
- **Strikeout font (Strikeout): /S**. Example: /S.

Additionally, and despite the fact that it is not a characteristic of the font itself, it is also possible to add the following property, in this case preceded by a 3-letter label.

Position of the text in relation to the point of insertion: /POS. The center position (C), or according to the eight points of the compass (N, S, E, W, NE, SE, NW and SW) are then set. These are used simply to indicate easily the usual idea that N is at the top, S at the bottom, etc. without having anything to do with the position over the Earth. If not specified, the text is written in the NE angle of the point. See also the "/X" and "/Y" properties above. This property is not applied in the case of dynamic texts that label lines and polygons.

Moreover, when the size is set in typographical units, it appears on the screen in exactly the same way as it does on paper (provided that the screen size is correctly configured).

For more details, see the technical document: **SymbolizationFonts_v6.doc**.

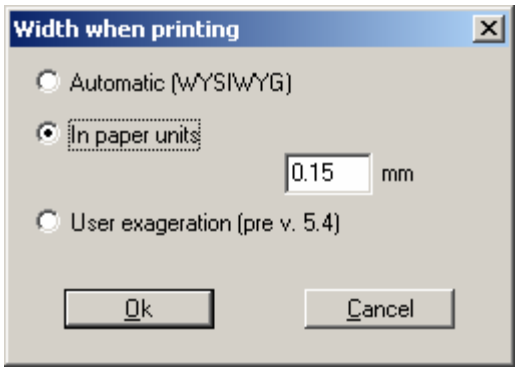
Possibility of preventing overlap of texts or point symbols. It is possible to indicate that you wish to try to avoid the overlap of text or point symbols. This is, of course, a request, but if there is a high density of texts, it may not be physically possible.

Direction of arcs or of lines in arrow form. The possibilities of the direction arrows of the arches and lines have been made more sophisticated. It is now possible to indicate their size, semiangle of opening, neck, etc. See the technical document SymbolizationArrows_v6.doc for more details.

Thickness of lines. Lines which are thicker than 2 pixels in the **output device** (screen, printer, etc.) may display any possible type (continuous, with strokes, etc.).
Symbolization of thickness for lines and polygon edges, point radius and nodes, and size of printing symbols. The following changes have been made:

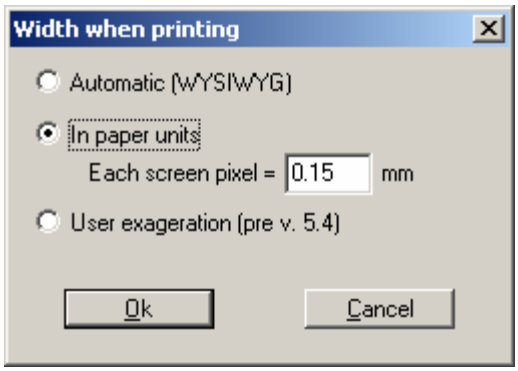
- The "pixels" text, which replaces the "map units" text that appears in the dialog boxes which specify line thickness, etc., now indicates:
  Pixels (screen)
  Pixels or mm (printer)

- Next to the box for entering the thickness, radius, etc. by field of the database or constant, there is an icon with the image of a printer that allows the user to control the settings of these parameters when printing. This button is only accessible if you have selected thickness, radius, etc. in “pixel/mm” units, but not when you have indicated that this is required in map units, as it would be meaningless in this case. The box that is opened looks like this:



and it allows you to select one of three modes:

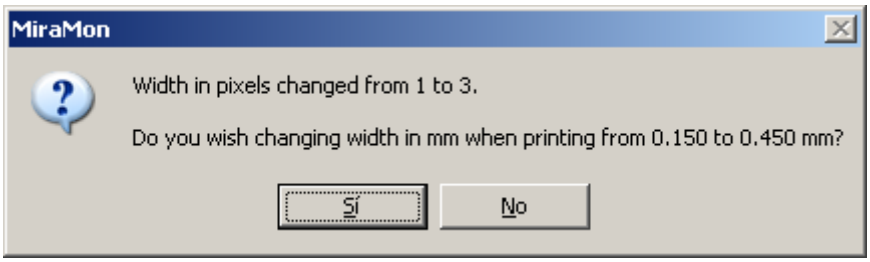
  - **Automatic (as on screen)**. In this case MiraMon will exaggerate the number of pixels used by the printer so that the element occupies the same space as on the screen. This is the new default mode and to ensure that it functions perfectly it is necessary to have established the correct screen width in mm in MiraMon.par or in the corresponding setting option (for example, in the Zoom by scale). In this situation, a WYSIWYG (What you see is what you get) environment is obtained. This mode ignores the exaggeration factor when printing from the “Map | Advanced” button. It is the default mode in point radius and nodes because a 1-radius point on the screen is usually appropriate when the printed page has similar dimensions (the point is very visible). It is also the default mode for symbol sizes because the symbol usually occupies a relatively large number of pixels and one normally wants a symbol size that is similar to that which appears on the screen.
  - **In paper units**.
    - For thickness, radius, etc., **constant**. In this case you should indicate, in mm of printed paper, the thickness, radius, etc. that you require.
    - For thickness, radius, etc., by database field. In this case you should indicate how many mm of printed paper correspond to each screen pixel. MiraMon will apply the same proportion to the different values of the database fields. This setting also appears in the dialog box:



This mode allows you, for example, to define finer lines than it is possible for the screen to display. This is not possible in automatic mode. In other words, in Automatic (WYSIWYG), when a line is printed, the printer imitates the screen thickness. However, the line may appear to be too thick from a conventional aesthetic point of view (a screen pixel usually occupies about 0.3 mm). It is for this reason that this is the default mode for line thickness and polygon edges and is set at 1 screen pixel equals 0.2 mm. This mode also ignores the exaggeration factor when printing from the "Map | Advanced" button.

- **User's former exaggeration criteria**. The program works in the same way as earlier versions up to v.5.5, applying the exaggeration factor when printing from the "Map | Advanced" button. This option is designed to obtain exactly the same printed appearance for old maps in some settings of a number of print devices, but is normally not recommended.

- When you exit from the boxes that specify thickness, radius or size of screen symbols (boxes with the  icon), MiraMon checks whether there was a ***constant*** (not for database field) value which is ***in pixels*** and has been changed. In this situation, when, moreover, the printing mode is in ***paper units (mm)***, the user usually wishes to change the mm value for printing, and for this reason MiraMon will display a message like the one below:



See the technical document SimbolizationLinesThicknesses_and_Symbols _v6.doc for more details.

Dynamic labeling of arc and line files.

Dynamic labeling of arc and VEC line files is permitted. Consequently, any field of any table of the database associated with an arc file or attribute of a VEC line file may be displayed as a text running next to the arc. This new function enables you to establish the range of scales within which to display the texts of the lines (regardless of the range of scales of the layer itself and highly recommended in order to avoid an excessive number of texts in whole views), as well as censoring excessively long texts in relation to their corresponding arc/line. This latter function allows you to indicate values such as 0% (the texts will always be drawn independently of the length of the arc), 100% (the texts will be drawn when the arc is at least as long as the text), 200% (the texts will only be drawn if the arc is at least twice as long as the text), etc. This function makes it possible to avoid the labeling of excessively short arcs and overlapping texts.

It is also possible to set a distance above or below the arc and even to have the text “pierced” by the line, as is the convention in contour lines. This parameter is also indicated in %: a 0 value indicates that the text is located above the line, 50% indicates that half a box of extra text is left separating it from the line, 100% indicates that space equivalent to a line of text is left between the text and the line, etc. If a negative value is indicated, the text moves downwards: if the value is -50% (half a text box) the text is “pierced” by the line, if it is -100% it is placed entirely below the line, etc.

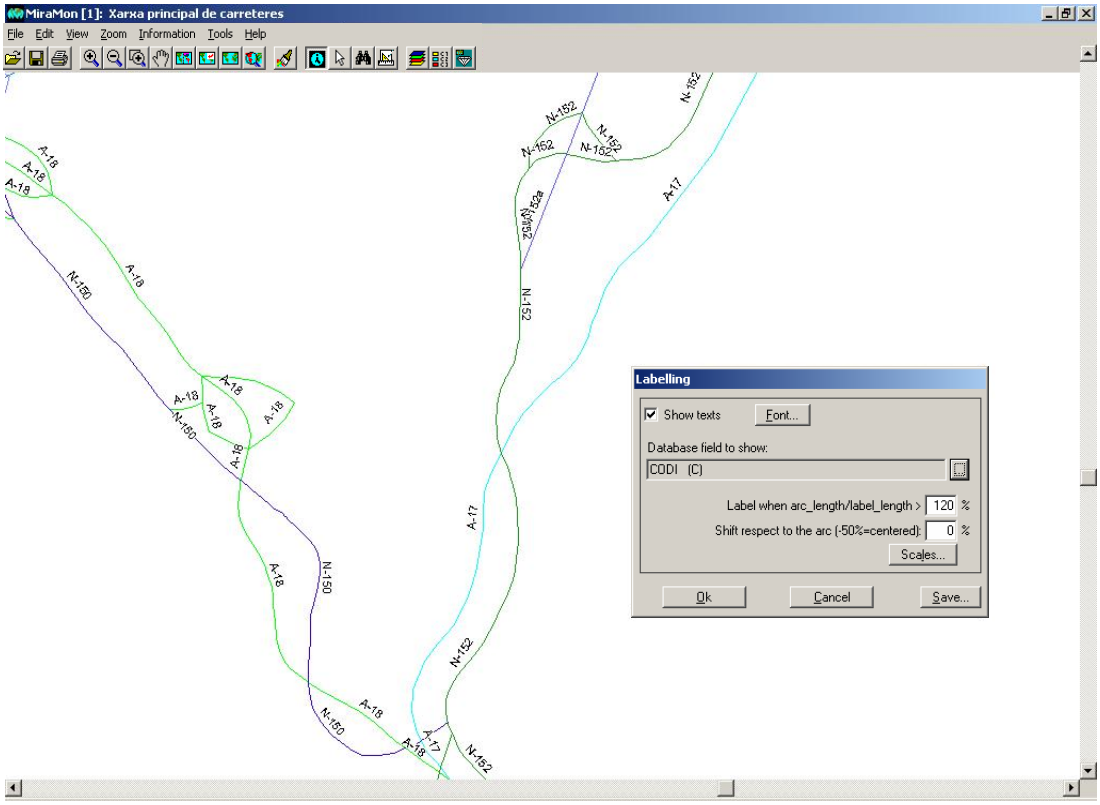
The text is oriented intuitively according to the orientation of the line itself and, if close to the edge, the program calculates whether to move it to the other side of the arc in order to make it more visible. The algorithm studies the different segments of the line and chooses the longest as the most suitable for orienting the text. However, if the segments are very short, it gradually generalizes the line until it finds a satisfactory direction. At each attempt it gives positive weighting to the centrality of the segment in the context of its arc as well as the length of the chosen segment. If an arc enters and leaves the field of view or the printing field, it is labeled every time that it appears (except when it occupies too little space and is censored by one of the other parameters).

It is possible to indicate “Avoid overlaps” so that the program tries to avoid collisions between the labels. The text can be symbolized using virtually all the properties that are applied to the points, except for slope and orientation and XY scrolling, which are meaningless for lines because they are set according to the arc/line which is to be labeled. It is important to remember that the text can be set in typographical units or in map units (the further away they are, the smaller they will appear).

Dynamic labeling of polygon files

Dynamic labeling of polygon files (both POL and VEC) is permitted. As a result, any field of any table of the database associated with a POL file or an attribute of a VEC polygon file can be displayed as text within the polygon. The new function allows you to set the range of scales within which to display the texts in the polygons (regardless of the range of scales of the layer itself, and is highly

recommended in order to avoid an excessive number of texts in whole views) as well as censoring texts which occupy too much space in relation to their corresponding polygon. This function allows you to indicate values such as 0% (the texts will always be drawn independently of polygon area), 100% (the texts will be drawn when the polygon occupies at least the same amount of space as the text), 200% (the texts will only be drawn if the polygon occupies at least twice as much space as the text), etc. This function makes it possible to prevent the labeling of excessively small polygons and overlapping texts.

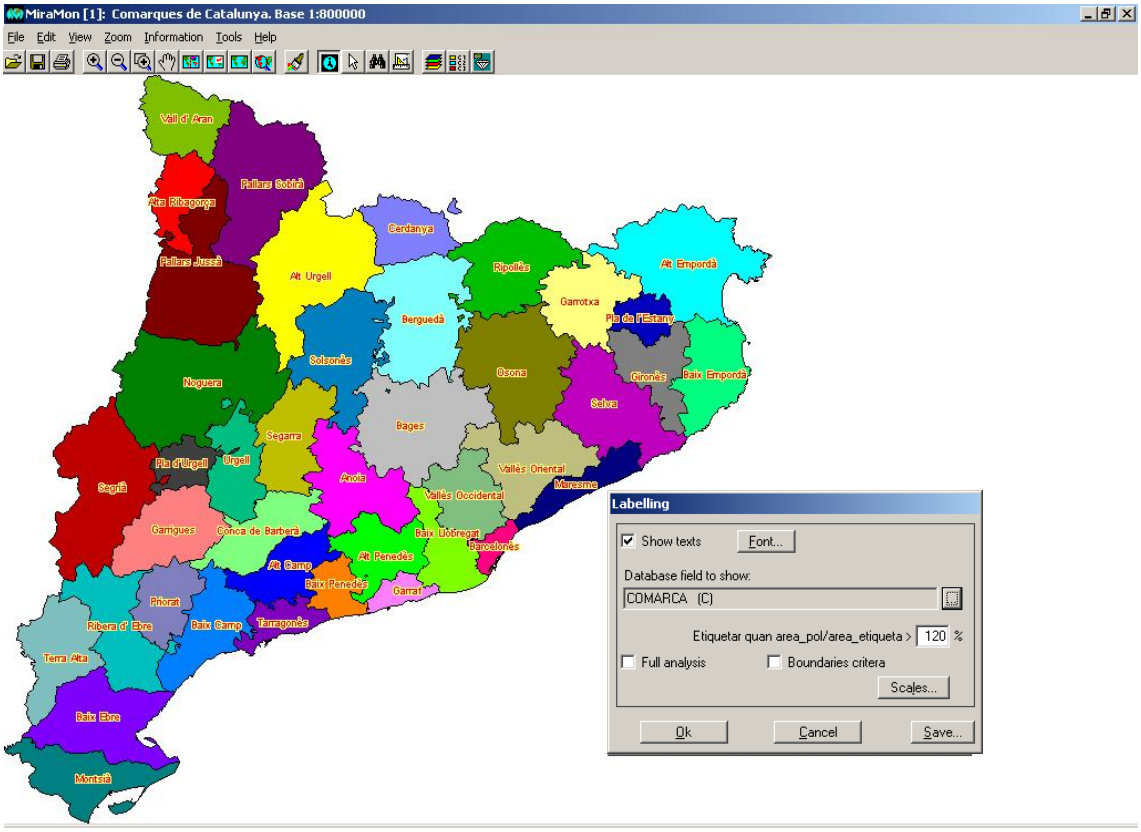


The heuristics employed for placing the labels involves a sophisticated system whose main features are:

- A large number of candidate positions are analyzed, based on the division of the bounding boxes into areas that are the same size as the label which the program attempts to locate within the polygon and each position is given a score based on how close it is to the center of the polygon and how far away it is from the polygon's internal (holes) or external edges. If you select the "Complete analysis" option, an exhaustive analysis is performed and the program finally chooses the label position that has obtained the highest score. If this option is not selected, the program will stop its search when it finds a reasonably satisfactory location, thus significantly accelerating the redraw process.
- It is possible to carry out an extra analysis which allows you to look for alternatives in the horizontal position of the label in order to avoid unsatisfactory results due to intersections with the edge in horizontally narrow areas. To activate this extra analysis, select the "Edge criteria" option.
- The labels move away from the edges of the display in order to prevent them from being cut by the viewing window.

- It is possible to indicate “Avoid overlaps” so that the program tries to prevent labels colliding with each other.
- If there are multiple islands (topological groups), the polygon with the largest area is chosen.

The text can be symbolized using practically all the properties applied to the points. It is particularly important to remember that the text can be set in typographical units or in map units (the further away they are, the smaller they will appear).



Semitransparencies: a degree (percentage) of transparency option is included in (structured or unstructured) vector symbolization.

8. Queries and selections

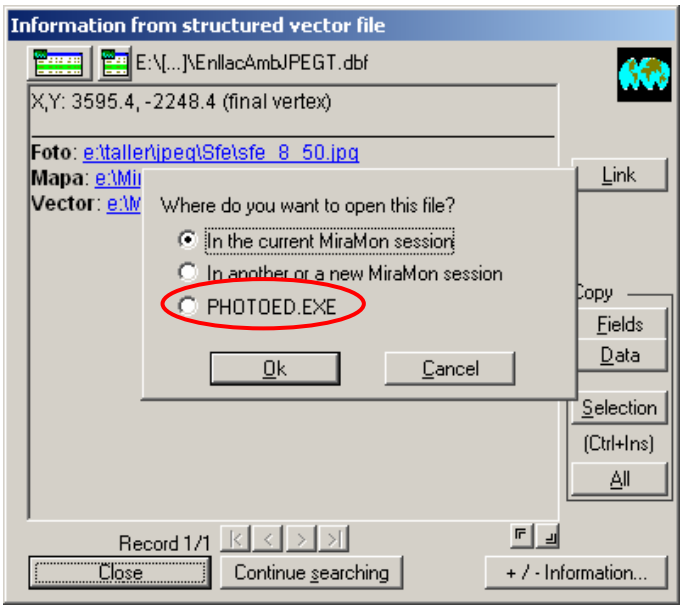
8.1.Improvements in some of the query by location functions.

When there is a single table and multiple cardinality relationships (1→many relationships) between some of the tables of the database, the single table is used to resolve the query by location. This was decided because the order in which the registers appeared in the classic case (resolve the links by “following the thread”) was not the same order as they appeared in the single table (because the algorithm that generated the registers was optimized using a different pattern). This resulted in the visual paradox of non-correspondence of the order of the registers of the same graphic object when flicking through the query by location registers and looking at the successive registers of the single table.

8.2. New features in the query by location box

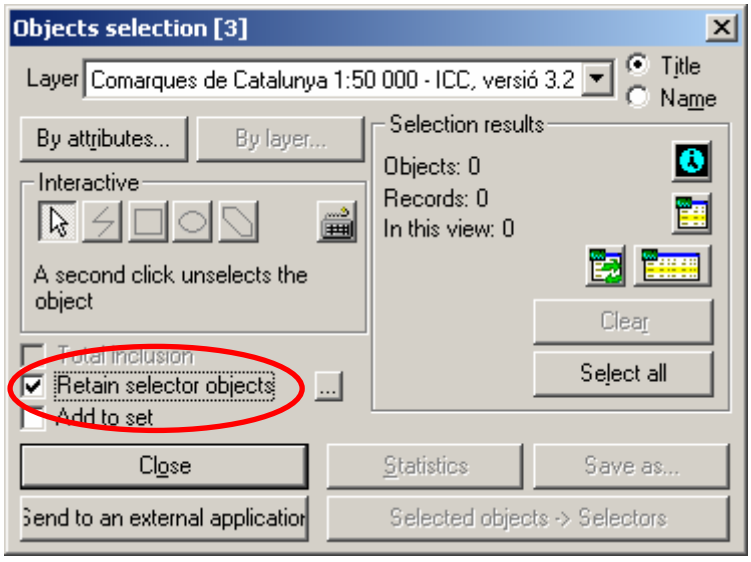
The “Main table” text button of the query by location dialog box has been replaced by an iconic button similar to the interactive selection box. Moreover, another iconic button has been added for opening the “Single Table” with the corresponding register marked in green (see illustration). When either button is pressed small help labels called “Tooltips” appear to explain how they work.

In a link to a JPEG from a query by location, a 3rd option is added to “In this MiraMon session” or “In another or a new MiraMon session”. This option corresponds to the application that, by default, opens the JPEGs, provided it is not MM32 itself.



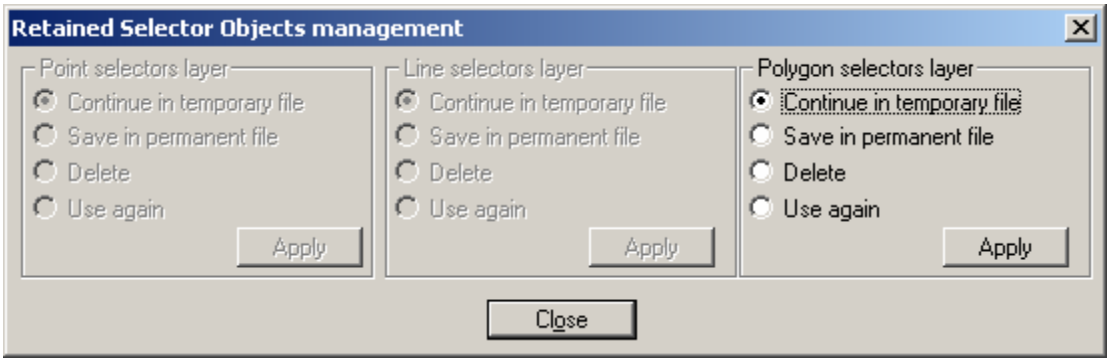
8.3. Retain selecting objects of a selective or a by attribute query

It is common to carry out an interactive selection based on a rectangle or a circle that is drawn in the digitization of a polygon that surrounds the objects that one wishes to select, etc. One sometimes does not wish this “***selecting object***” (the rectangle, circle, polygon, etc.) to disappear once the selection has been made or one may even wish to be able to use it again without having to redraw it. The interactive selection box allows you to activate the “Retain selecting objects” button which is designed to achieve this. If you activate the button, you can continue to see the selecting objects after the query and also reuse them.



When this option is activated, all the objects that function as selectors are saved to a temporary file called SelP####.vec in the case of polygonal selecting objects (rectangles, circles, polygons, etc.), SelL####.vec in the case of linear selecting objects (transects, lines, etc.) and SelT####.vec in the case of occasional selecting objects (mouse clicks, points, etc.). Note that all the objects are added to these files until you decide to eliminate them (see below for how this is done) and that objects from other layers which act as selecting layers are also added as selecting objects. The attribute given to the selecting objects is a letter (P, L or T depending on file type) followed by a whole number which, starting from 1, increases in value every time a new object is retained. The fact that the attribute is alphanumeric means that the default symbolization is categorical, which is logical for selecting objects. In the case of polygons with topological holes or with external islands (groups), the different rings receive the same attribute. This information is used by MiraMon to know that they should be treated as a single topological object.

The layers of retained selecting objects have a specific default symbolization (for example, transparent polygons with magenta edges for SelP####.vec layers, turquoise lines for SelL####.vec layers, points with a 2-pixel radius for SelT####.vec layers). These can be changed to suit the user's needs in the same way as in any vector layer.



By pressing the "…" button located next to the button that activates selecting objects a dialog box like that in the figure above will appear, consisting of three

groups, one for each selecting object. The only groups that will be active in this box will be the ones that correspond to those objects that are retained at that moment (in the example that appears in the figure above, only the selecting polygons are retained). The first three options of each group are self-explanatory and work in the same way as the equivalent objects in the box that appears when one presses the “Advanced” button of the box for measuring distances, perimeters and areas. The fourth option, “Reuse”, allows the objects to be relaunched as selecting entities against the layer chosen to receive the selections, although it is possible to retain the selecting objects used in the main interactive selection box and also those used in the “Selection by layer” box. The Management of retained Selecting Objects button can only be pressed from the main box in order to avoid an excessively crowded screen (however, as it is possible to define the selecting objects from the main dialog box by pressing the “Selected objects → Selectors” button, this option does in fact exist). For any of the four options, it is necessary to press the “Apply” button to achieve the desired effect (in fact, the first option has no other effect than to continue in the usual temporary file situation).

On exiting MiraMon, if the selecting objects have not been erased, the program asks if you wish to save them to a permanent file, in the same way as when quick measures have been carried out.

8.4. “By layer” selection

“By layer” selection has been implemented. That is, “Selection using SELECTING objects”. By layer selection works in practically the same way as interactive selection and by attribute selection, and the dialog box is therefore almost identical. However, a number of differences have been incorporated to aid visual identification: apart from the obvious fact that there is now no “By layer…” button, the expression “Selection of SELECTING objects” and not “Object selection” appears in the title of the dialog box. A further difference is that the texts on the buttons are in italics.

By layer selection consists in selecting objects from another layer (a number of municipalities, for example) to serve as selecting objects of the objects that you ultimately wish to select (forest inventory points, for example).

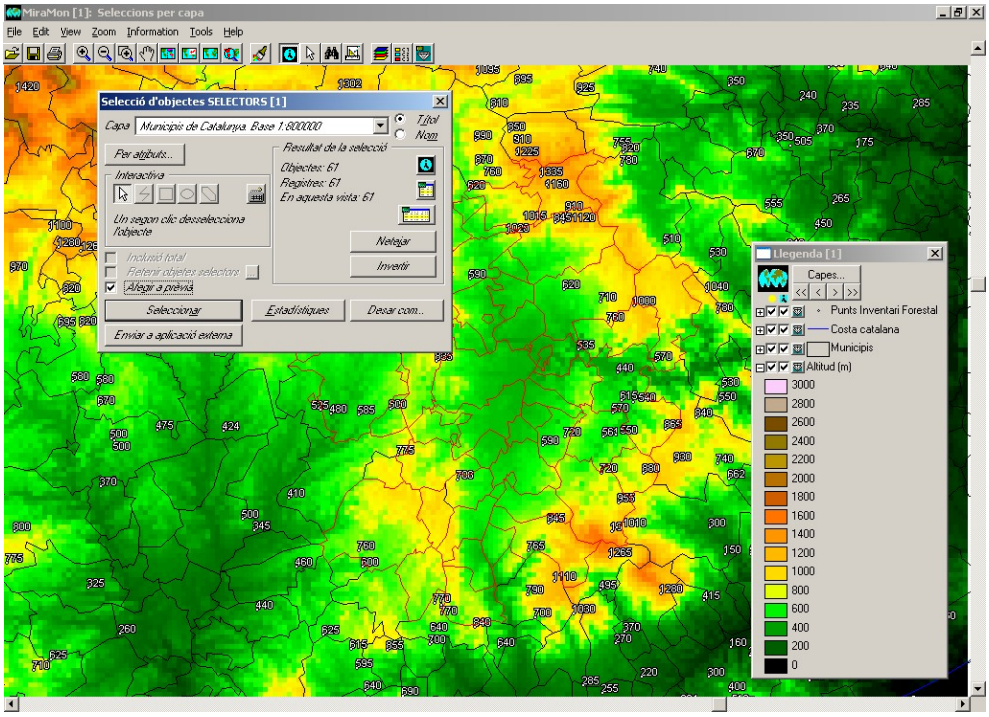
A new button, “Enter coordinates via keyboard”  , saves you from having to go to the Editing menu to find this option while you are selecting objects if you wish to enter coordinates. This button can be used in any situation: to enter one of the vertices of a selecting polygon, to enter the center of a selecting circle, etc.

Procedure:

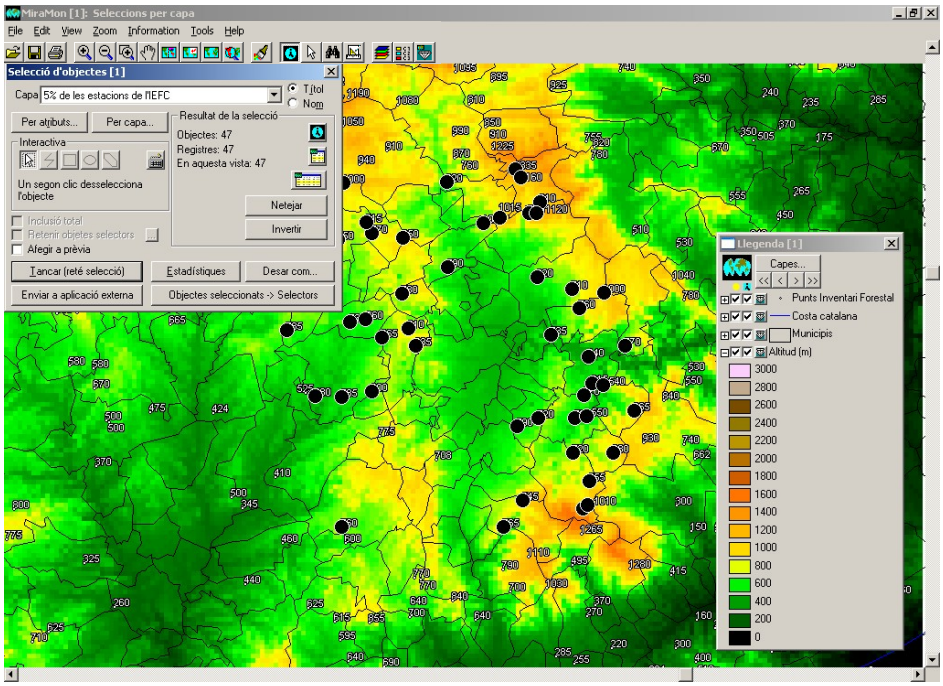
Traditional method

First of all you select, from the drop-down menu of the interactive selection dialog box, the layer that contains the objects that you are interested in (forest inventory points, for example). We will call this layer the “Layer of interest”.

Secondly, you press the “By layer…” button and the “Selection of SELECTING objects” dialog box will appear. You now choose “Selecting layer” from the drop-down menu (“Layer of interest” does not appear in the list as it makes no sense for it to select itself) followed by the selecting objects that you wish to use. This selection can be carried out interactively (e.g. by selecting the municipalities you are interested in with the mouse) or from the attributes of the selecting layer (e.g. the municipalities that belong to a given county).



When you have selected an object of the selecting layer, the “Close” button changes the text to “Select”. This happens because once the “Selection of SELECTING objects” dialog box has been closed, MiraMon will proceed to select the objects of the first layer (forest inventory points) based on the selection carried out on the second layer (municipalities) in exactly the same way as if the selecting objects had been drawn interactively in order to carry out the final selection (the inventory points will therefore be selected from within the selected municipalities). If there were other selecting objects from the first layer and the “Add to previous” button of the first dialog box was depressed, these will be saved together with the new selection.



It should be noted that the selecting layer does not have to be polygonal: a point layer acts in the same way as if one were to click with the mouse on the different points in order to carry out an interactive selection. The only difference is that when selecting points are used, a second point on the same object does not result in it being deselected; it is simply ignored (if not, when points like towns select polygons like countries, the selection of the latter would depend on there being an even number of points on the country). It should also be noted that the layer of interest may be of any kind (points, lines or polygons). Obviously, the use of some combinations may make it difficult for a selection to occur. Consequently, if you select points or lines using selecting points, the program will only select the points and lines that coincide spatially with the selecting points.

If the selecting file is polygonal, MiraMon takes its topological cover into account. That is, it supports the polygon holes correctly (excluding them from selecting area) and the external enclaves (including them in the selecting area).

Alternative method

Another function related to Selection by layer is the possibility to define a given selection as a selecting layer of a new selection. For example, you can select a particular road (interactively or by attributes) and then decide that you wish to use it to select the municipalities that it passes through. To carry out this operation you only need to press the “Selected objects → Selectors” button. A dialog box will then appear in which you can choose the layer on which to apply the selections and also a “Select” button that allows you to make the selection directly.

The “Selected objects → Selectors” button is to enable you to use the current selection as a set of selecting objects of another layer. This is useful as an alternative itinerary for selection by layer and may be more intuitive for some

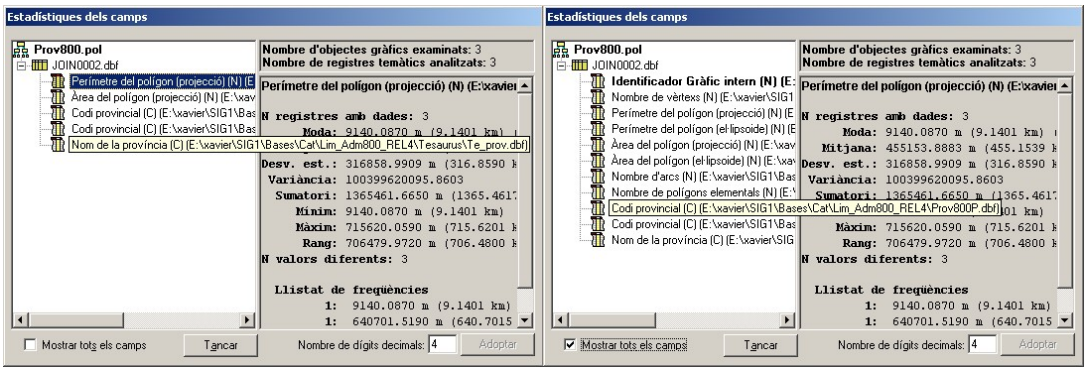
users: first of all you should make the selection of selecting objects (e.g. municipalities) from the usual interactive query dialog box. Then send the selection to the selecting objects condition ("Selected objects → Selectors" button) and make the selection. However, the "Selected objects → Selectors" button also has another very useful function; this is the ability to make **successive selections among several layers**. For example, in order to know which rivers flow through municipalities that have wastewater treatment plants equipped with a certain kind of technology. In this example, you select the plants with the technology that you are interested in, establish these plants as a layer of selecting objects, choose the municipality layer and, using the "By layer…" and "Select" buttons, produce the relevant municipalities. You then define these municipalities as a layer of selecting objects, choose the rivers layer and, using the same "By layer…" and "Select" buttons, produce the rivers you are looking for.

The "By layer…" query procedure described above usually avoids the need for a more analytical operation using the CombiCap tool that is located in the "Tools" menu. However, it should be kept in mind that CombiCap enables you to automate the process and offers inter-connected tables of resulting statistics, etc. Moreover, CombiCap performs attribute transfer between layers and fragments the objects geometrically and topologically (e.g. rivers on municipalities to produce rivers results in the rivers being fragmented according to each municipality that they flow through). These operations do not perform selection by layer. This can be summarized by saying that in the case of selection by layer, once the objects have been selected, the reason for which they were selected disappears. Finally, remember that you can use VecSelec, in the "Tools | Alphanumeric databases | Selection of a subset of vector entities…" after CombiCap in order to select those combinations you are interested in (e.g. forest inventory points located in particular municipalities) and perform new inter-connections with CombiCap.

8.5. Statistics of the single dynamic table

The statistics of the selection of vector objects are now **calculated for all fields and registers** resulting from the relationships of the alphanumeric database. Therefore, following a (interactive, by attribute, etc.) selection, you can consult the statistics resulting from each field of the database, and not only the statistics of the main table. In order to do this, the program shows the fields resulting from the display of relationships between the tables that comprise the database linked to graphic objects.

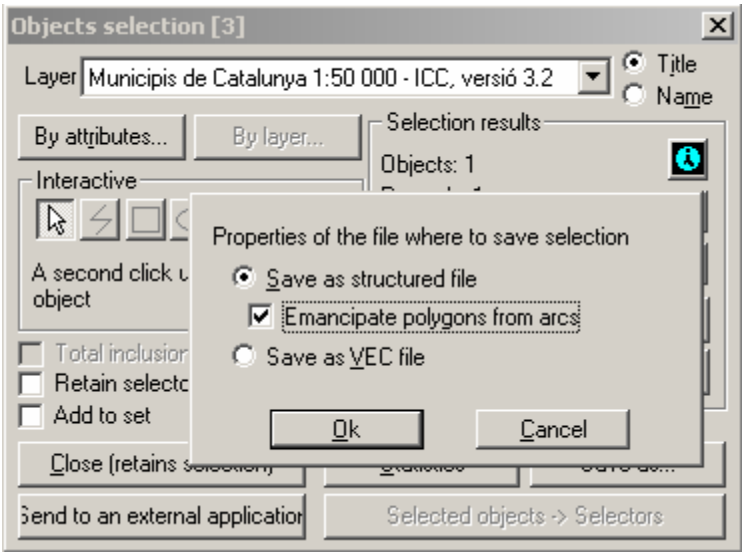
The field descriptor for this table, called the single dynamic table, is the field descriptor of the original table, followed, in brackets, by the name of the table to which it belonged, thus making it easy to know the origin of each field. Regarding the names of the fields (not the descriptors), the program tries to avoid the existence of duplicate field names in the single table. For more details about the single table, see the corresponding section.



Naturally, the single table expands all the relationships indicated in GeMM, so that when there are declared 1→many or many→many relationships, multiple registers may be generated for each graphic identifier. However, **when calculating the statistics, MiraMon excludes any repetitions that may be generated in the geometric and topological fields of the main table**, so that, for example, the area of a polygon that has been selected twice does not accumulate twice because of the presence of two forest species that were being sought. This new property (generation of single table and the resulting possibility to query all the fields) is not applied to old vector files (REL1). If you wish to convert an old layer to the modern format, this can be done using ConvRel ("Tools | File maintenance | Layer conversion…").

8.6. Emancipate the selection of a polygon layer

It is possible to generate a selection that does not depend on the arcs of the (emancipated) mother layer of polygons when you indicate, following an interactive selection or a selection by attribute, that you wish to save the selection.



9. Printing

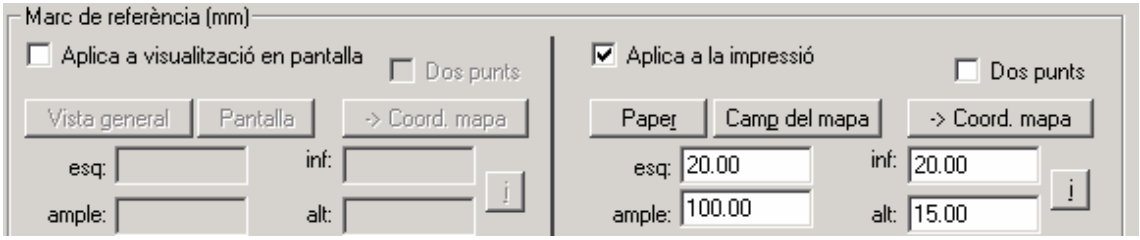
9.1. Improvements to some general functions

- A solution has been found for an unresolved problem involving printing by packages, and the different printers (SHAR-AR-M276 PS [A3 and A4], HP Color LaserJet 5500 PCL6 [A3 and A4], HP Color LaserJet 5550 PostScript [A3 and A4], HP Color LaserJet 5550DN [A3 and A4], HP Color LaserJet 4650 PCL 6 [A4], HP LaserJet 4350 PCL6 [A4], etc.) and plotters (HP DesignJet 1055CM [A0] with large and small pieces, at normal and optimum quality and when the printout is generated by the computer and by the printer), from which it was difficult to print slides, now function correctly.
- With plotters that have little memory installed (such as the HP DesignJet 1055CM), when you request A0 at the highest quality (optimum-improved), the program will indicate on the advanced properties tab that the document should be processed by the computer (and not by the printer).
- Printing of BMP (and JPG) also works correctly, although the driver may sometimes reduce the desired resolution. Using the driver of the HP DesignJet 1055CM plotter it was possible to generate a file of over 400 Mbytes (approx. 10,000 columns x 14,000 rows) normally.
- It has been confirmed that hatching also work in the printers mentioned above, even when combined with slides (hatching becomes paler).
- It has been confirmed that the EMF printouts with slides are correct.
- When printing on a BMP or a JPEG, the latter are now **georeferenced** through the corresponding B.rel and J.rel files. This allows you to reuse the MiraMon digital printouts as rasters to work with (e.g. when generating a cartographic background for visual reference).
- When defining the printout, when the reference frame and the map field coincide, it is possible to separate ("unhook") them by activating the "Use Reference Frame.Map" command.

New features in the printing boxes (in progress)

The main new features that have been developed for the **printing boxes** are as follows:

1) Any box can be applied to the printout or screen display:

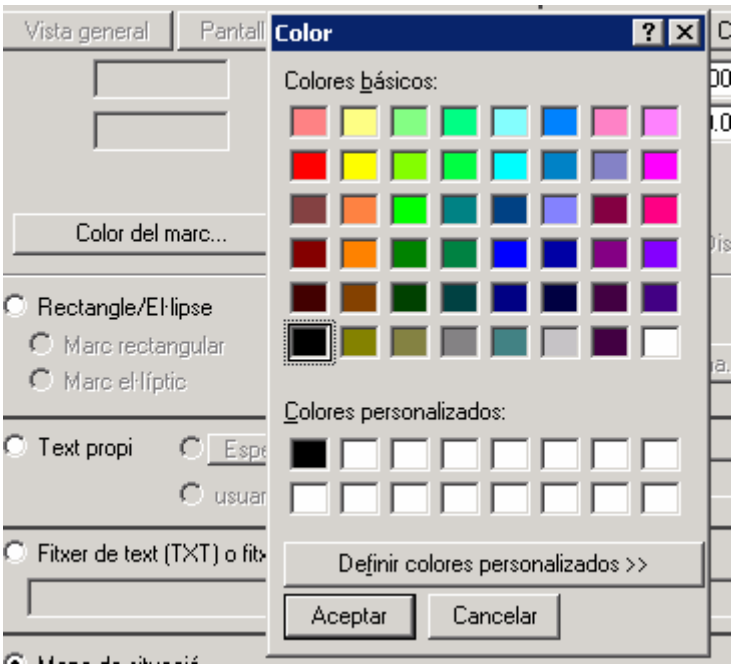


The text of the reference frame varies according to the type of box.
2) New types have been created, such as:

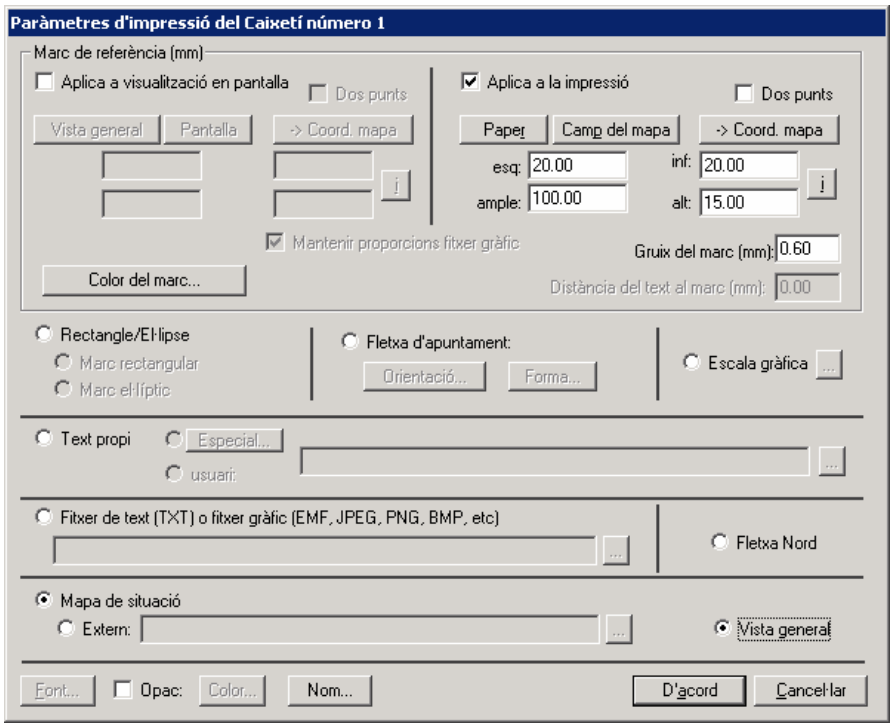
- Rectangle and ellipse (or square and circle, if a parameter that makes the program maintain proportions is forced) objects.
- Indicating arrow with options to parametize shape, color and orientation:

- Location map that can come from a (MMM or MMZ) map or be defined automatically from the whole view.

3) More options such as thickness, color, shape, etc. whenever these need to be applied.



Below is a summary of the whole interface:



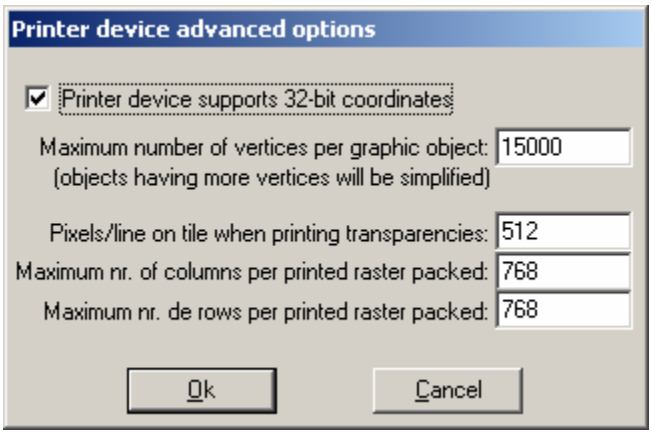
It should be noted that many descriptors will be dynamic to help the user understand the different parameters involved more fully.

9.3. Printing of line thickness

The program adopts a similar treatment for line thickness, point radius, etc. as that used in the Automatic case (as seen on screen). In this case, therefore, MiraMon will exaggerate the number of pixels used in the printer so that the element occupies the same space as on the screen. In order for it to work perfectly, you must establish the screen width correctly in mm in MiraMon.par or in the corresponding settings option (e.g. in Zoom by scale). In this situation, a WYSIWYG (What you see is what you get) environment is obtained. The exaggeration factor is ignored when the “Map | Advanced” button is pressed.

9.4. Advanced printing options

Controlling the size of the raster packages: The “Advanced printing options” dialog box, which can be accessed from the “+…” button on the printing design dialog box, has been improved with new parameters that allow you to control the maximum size of the packages that MiraMon sends to the printer when printing rasters. These parameters can also be set using the keys with the same name in the [Hardware] section of MiraMon.par



Printing vectors by packages or pieces

Printing can be carried out by “packages” or “pieces” not only for rasters, but also for vectors. This allows many printer drivers to support effects such as transparencies or hatching in complex vectors (many vertices, many internal holes, etc.). MiraMon functions using the number of pixels of the size of each piece which is placed over the printing mechanism. The default number is 1000, but this can be changed if necessary: higher values increase the printing speed but there is a higher probability that it will encounter the occasional problem involving the disappearance of transparencies or patterns; higher values than the number of pixels of the largest size of the paper (this information can be accessed using the “i” button of the map printing setting) will revert to the previous print setting, in a single piece (if you do not wish to search for the value in question, it is possible to directly use a very high value, e.g. a million). Smaller values than the default value never appear to be necessary because the size of the piece is similar to a high resolution screen, with which no problems with transparencies or patterns have been detected. It may still be useful to use the single piece for advanced uses when you require a vector result in EMF and for the vectors not to be divided. For this reason, it is suggested that you only switch to the “Printing in pieces” mode when there are hatchings or transparencies, although the user may force the new mode using the “+…” button on the printing dialog box. It is currently not possible for the user to vary pixel size, although this option may be incorporated if necessary.

Additional comments regarding printing device drivers: It is important to have the most up-to-date versions of the printer and plotter drivers. For example, the HP DesignJet 1055CM plotter with version 4.63 drivers stops printing texts, even in layers with no complexity and in A4, when it prints semitransparencies and the “Print from the printer” driver option is used. However, driver version 4.67 functions correctly.

We have also observed that some recent PostScrip plotters experience difficulties when printing semitransparencies. The manufacturers have been informed and are investigating the problem.

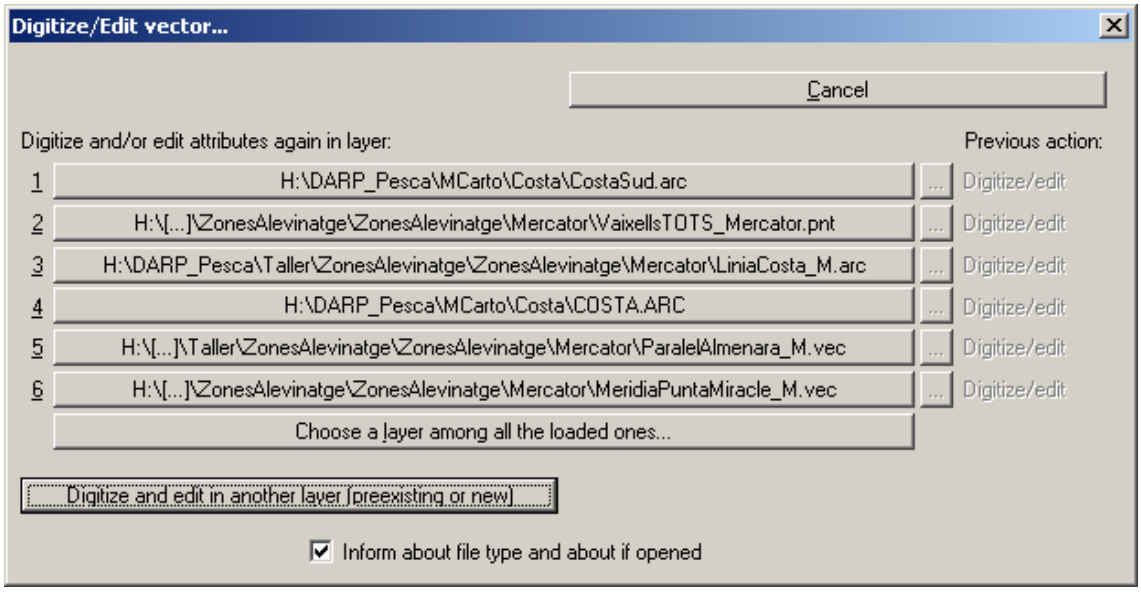
10. Vector editing and digitizing

A number of aspects have been consolidated and a variety of new ones have been improved, such as the request for confirmation before generating multi-registers in the database, the new digitizing window, the continuous connection when digitizing or the new floating digitizing tool bar.

New digitizing window

A dialog box has been created that appears when you digitize and edit a vector layer (F2) or when you edit attributes (Shift+F2). The new window offers a list of up to 6 layers that are open for digitizing in that MiraMon session (if there are fewer, the program completes the list with other open vector layers, giving preference to those located “above”). In this way, by simply clicking on it, you can once again digitize/edit on a layer in which you have already worked or on a layer that is already open.

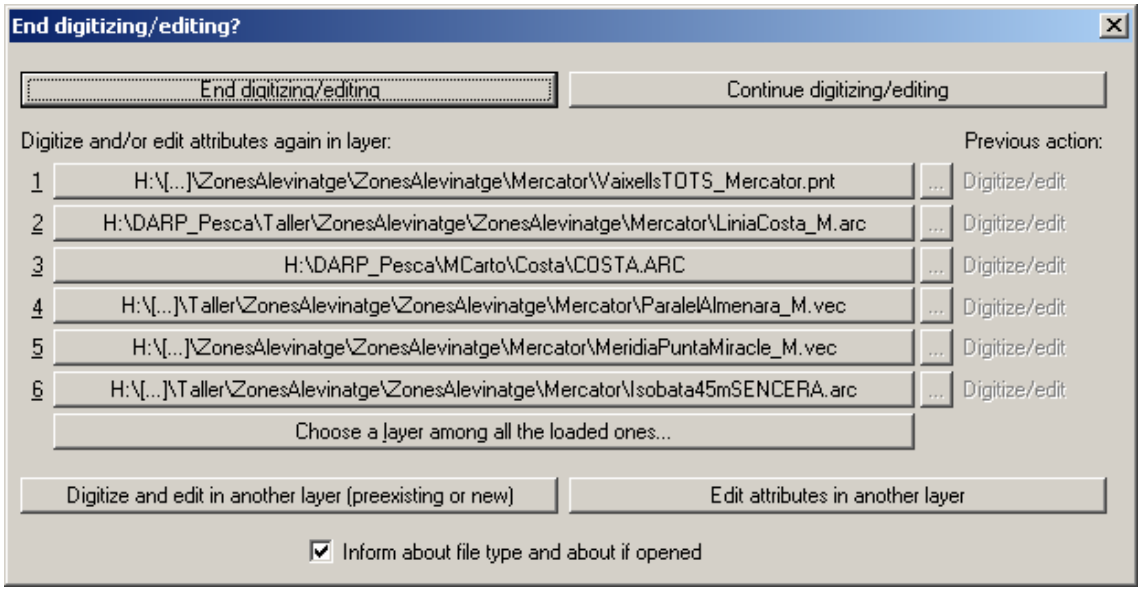
In order to access work with different files even more quickly, each layer of the list is preceded by a numerical value (from 1 to 6) that can be used as a keyboard shortcut so that you do not have to move the mouse (in other words, you can press F2+layer number and continue the task with the new layer that is being digitized or whose attributes are being edited). If the total number of layers loaded exceeds 6, the “Choose a layer from all open layers” button allows you to select from the whole list, as well as alternating between layer names and titles in order to clear up the occasional doubt. Logically, the box also offers another button which allows you to explore in order to select other, preexistent or non-preexistent layers (when editing by attribute, they must be preexistent).



The program remembers if a layer was opened in digitize/edit mode or simply in attribute editing mode, informs you of this in the list and conveniently applies it when the layer is reopened via the quick access of the new window. It is also

possible to have opened a layer using both modes, which results in both options being offered.
A "…" button allows you to eliminate from the list those layers that you do not wish to be offered as "previously digitized layer".

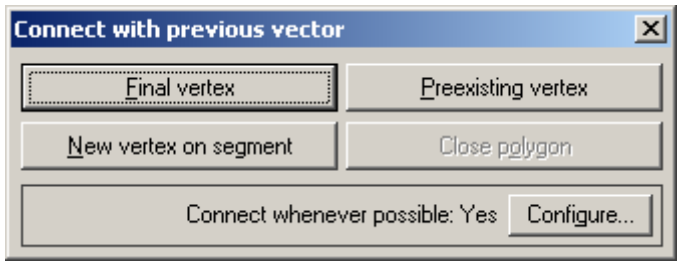
Likewise, another dialog box has been created which is similar in appearance to the previous one. This appears when the digitizing or attribute editing process is finished. The new window allows you, in addition to the usual options for confirming the completion of the editing process or its continuation, quick access options to continue the digitizing/editing or attribute editing task in a similar way to those that appear when digitizing starts.

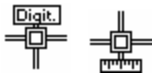


Lastly, using the new windows it is possible to deactivate the "Inform about file type and if it is already open" option. This requires even fewer mouse clicks or keystrokes in order to continue working with another layer.

Continuous connection when digitizing

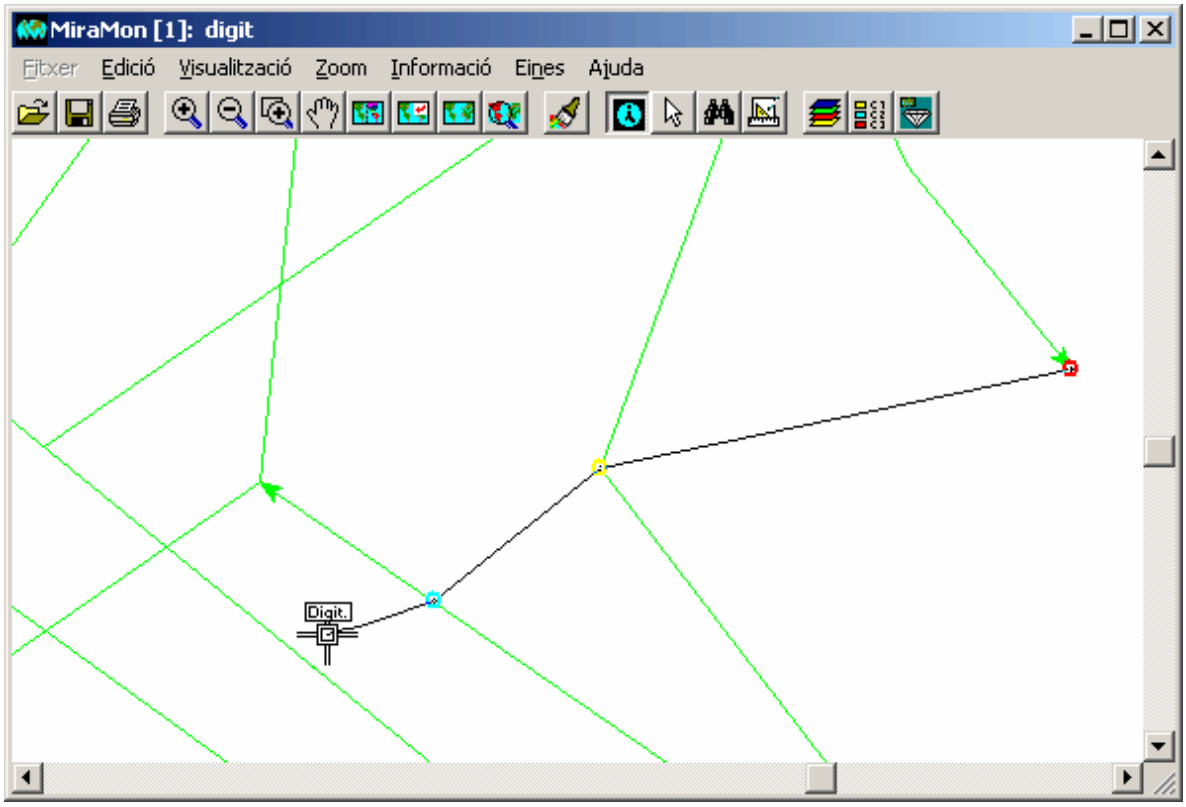
A new mode for continuous connection or "to connect whenever possible" is implemented. This new mode, configurable by default using the MiraMon.par file, try, in each digitizing click, to establish a geometric connection by exploring all the connection possibilities (final vertex, preexisting vertex and new vertex on segment) before considering that a new vertex must be created. The mode is compatible with a punctual request of just one connection possibility. In other words, although the new continuous digitizing mode is enabled, it's possible to request, for instance, just for a final vertex connection, so in the immediate subsequent click the program will only consider this connection possibility and, in case this connection is not possible, will inform it has not been possible to do. In this case, new vertex will not be created.



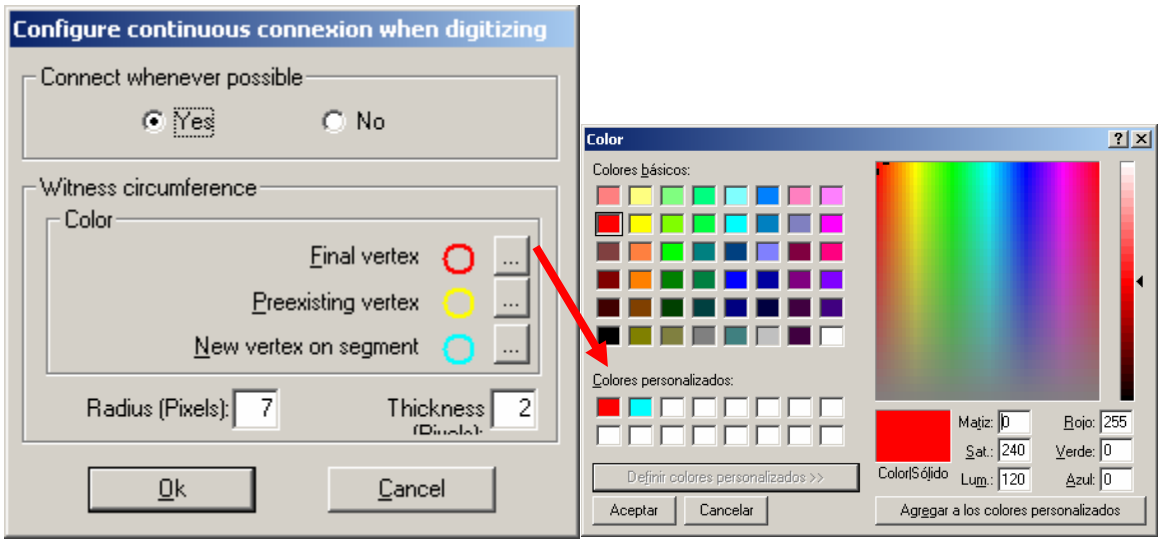
MiraMon's cursor inform, all the time, whether this new mode is enabled or not as it adopts in its central part a selector's form (square) to facilitate the visibility of the elements to be connected to: 

In this new mode, the user is informed that a connection has been made by using a small testimonial circumference where the new vertex has been generated; the circumference's colour is:

- red when the connection has been generated over a final vertex
- yellow when the connection has been generated over a preexistent vertex
- turquoise blue when a new vertex on segment has been created.



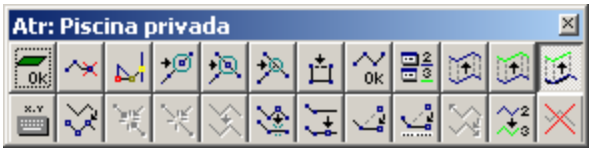
Testimonial circumferences' colour, radius and width can be modified using the dialog box appeared when clicking Configure... button (by default options can also be configured using MiraMon.par file):



See ConnexioContinuaEnDigitalitzar_v6 technical document for more details.

The new floating digitizing tool bar

A new floating digitizing tool bar is created to facilitate the use of digitizing and vectorial edition tools to those users who don't want to memorize keyboard's short cuts neither go on displaying the correspondent menu. Tooltips of the new box not only explain what each button can do, but also recall keyboard's short cuts to the user. Moreover, the tool box indicate the attribute or attributes in use (including multiregister). The box appearance can be configured using MiraMon.par file and it's managed from Edition's menu.

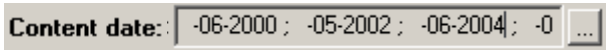


11. New applications and selecting improved features for some modules

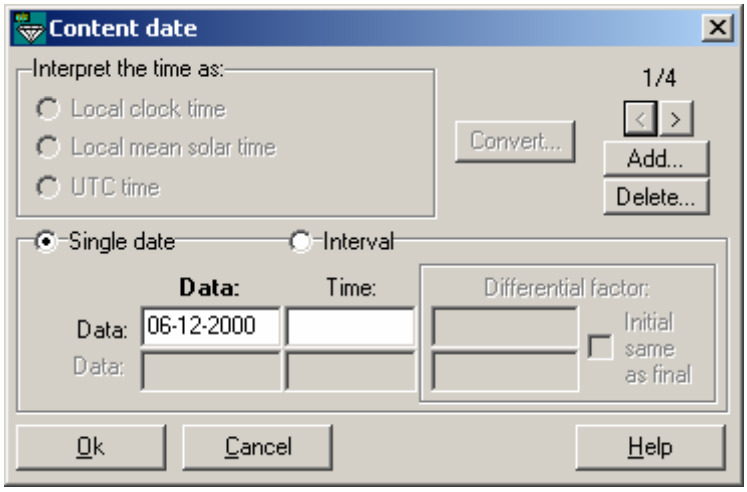
11.1. MiraMon metadata manager (GeMM)

***Enriching metadata when saving WMS views*.** A process which informs of the WMS request that has generated the saved view. It is possible to open the metadata of a GIF (G.rel), which usually result from saving a WMS view, etc.

***Possibility of saving more than one date interval for the Content Date*.** The content date has become a list so that you can define as many “packages” of dates as necessary. Each one of these packages may be a single date or an interval (always with this structure). The list of dates is displayed on the editable resource of the thematic Information tab:



and when you press the "…" button to modify it, the following dialog box appears:



The box to change a "package" has been modified in order to allow you to use two radial buttons ("Single date" and "Interval") instead of an (activated or deactivated) "Interval" button.

This modification has been implemented in order to enable the user to document, in the county distributions of the Topographic Base 1:5000 of the Cartographic Institute of Catalonia (ICC), the content dates of all the 1:5000 sheets that make up the county (the example in the figure corresponds to the county of the Alt Camp), which have different dates due to the fact that there is not a single photogrammetric flight.

***Multilingual entries*.** The number of entries that can be multilingual has been increased. These can be seen because they are accompanied by a  button (which will be grayed out if you have not defined, via the "Metadata | Information about metadata" tab, that the metadata should be multilingual). For example, the base title, the summary, the key words or the descriptor of the database fields are now multilingual.

***New "Presentation | Default symbolization" tab*.** A new tab has been added to GeMM that allows you to see the codification and version of the symbolization and the legend display contained in the REL file. On the tab there is also a button that allows you to erase the information so that you can return to the default MiraMon display itself. This information is applied to rasters and structured vectors.

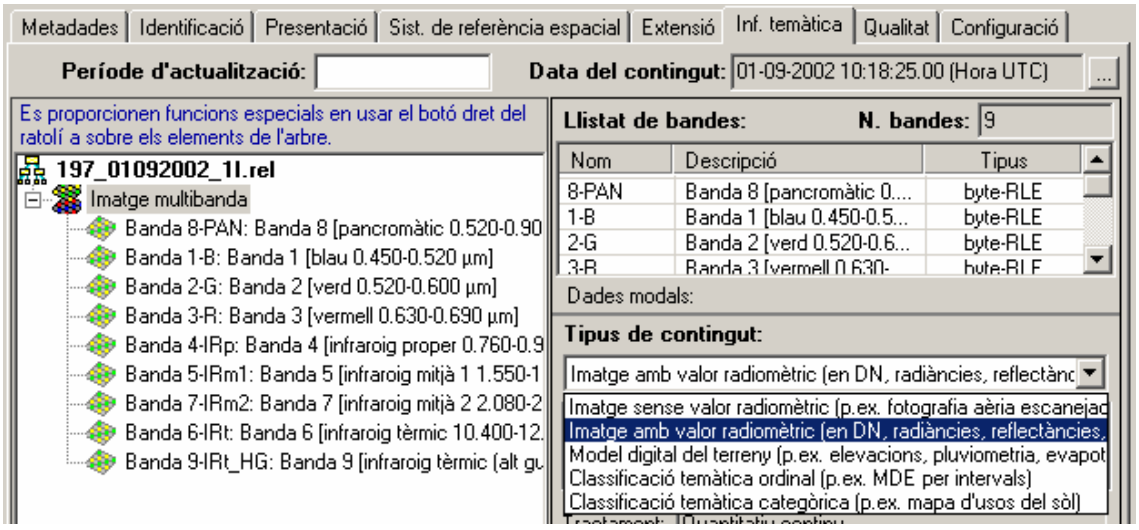
***File topology*.** GeMM shows more information, and in greater detail, about file topology (whether the topology is guaranteed or not and, in the case of polygon files, whether the file contains groups or not, and if the groups are topological). For unstructured files, the program indicates that they do not have guaranteed topology.

Metadata of cartographic series. A new model for Cartographic Series has been designed and implemented. In addition to enlarging the metadata model for the series, this allows you to define a multiseries, which is useful because it enables the display to know in which vertical order the series of a multiseries should be shown regardless of whether the first open sheet includes all the representative layers or not. The new model has been successfully applied to the series of the ICC.

In this model the GeMM function that means that when you try to modify metadata inherited from another hierarchical level the program offers to open another GeMM session with the right hierarchical level to perform editing has been generalized. At the same time, the metadata of the layer-sheet inherit the summary of the metadata (from the multiseries) and the description of the lineage and the description of the sources (from the sheet) when they do not have these entries.

More information is available on request.

***Other applications and features*.** Work has continued to allow the program to support more optional entries of the standards that have been adopted during this period given the fact that the compulsory standards are already available. Metadata visibility of the Vertical Reference System, ISO 19139 exportation following diferent profiles and the indication of rasters' type of content have been incorporated, among other features.



11.2. MiraDades

- The "File | **Save as**" option is no longer restricted to open tables via ODBC and **has been extended to DBF files**. This makes it possible to save an open DBF under another name. Apart from its usefulness as a duplicator of DBF tables, this function is necessary for some users of corporate environments who have restricted "normal" (with explorer) access to some directories (such as %TEMP%) but, on the other hand, the software is able to access them, for example in the case of the single

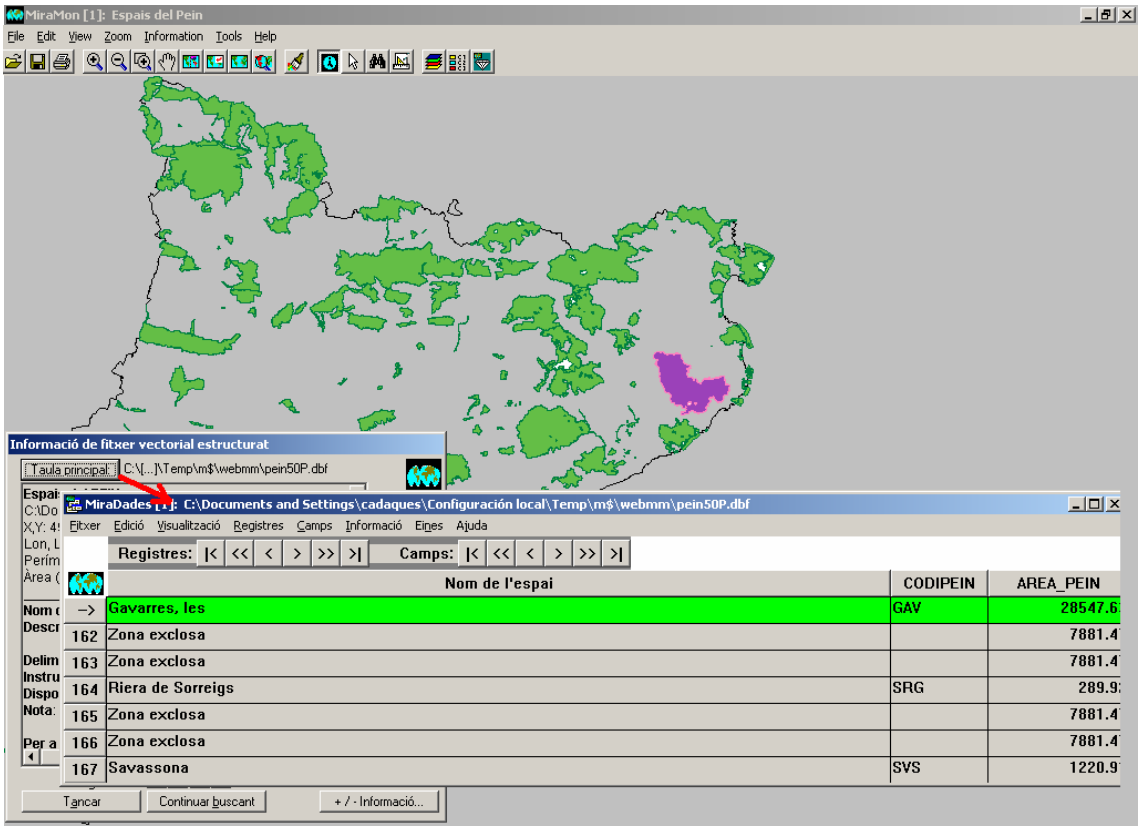
table. With the new function it is very easy to copy the single table to another location if you wish to keep it or work with it, etc.

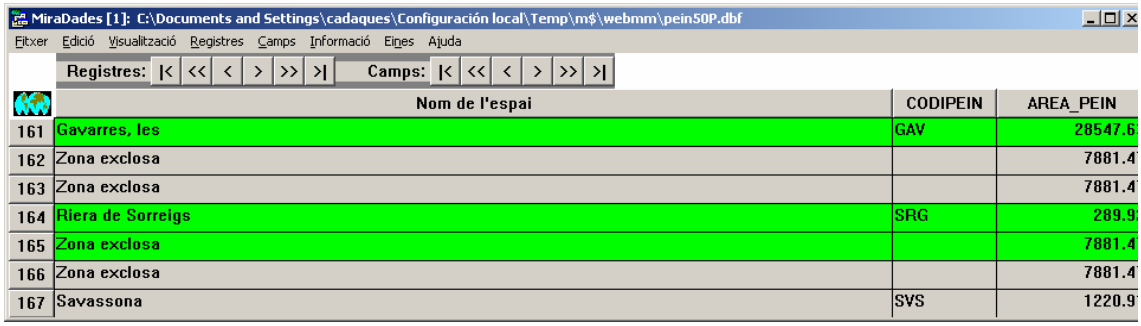
- The contextual menu of the MiraDades registers (which appears when you press the button with the register number) incorporates the possibility to mark/unmark and select/deselect registers and also remembers the keyboard shortcut.
- Marking registers. When MiraDades is opened using the MiraMon query by location button, not only does an arrow appear indicating the particular register involved, but the whole register is marked in an intense green color. If the table is placed in editing mode, the color of the marked register changes to light green so that it is similar to the gray/white of the conventional display.

  The table that is opened with this button is now linked to the MiraMon session and later marking or selecting operations will be directed to this MiraDades session.

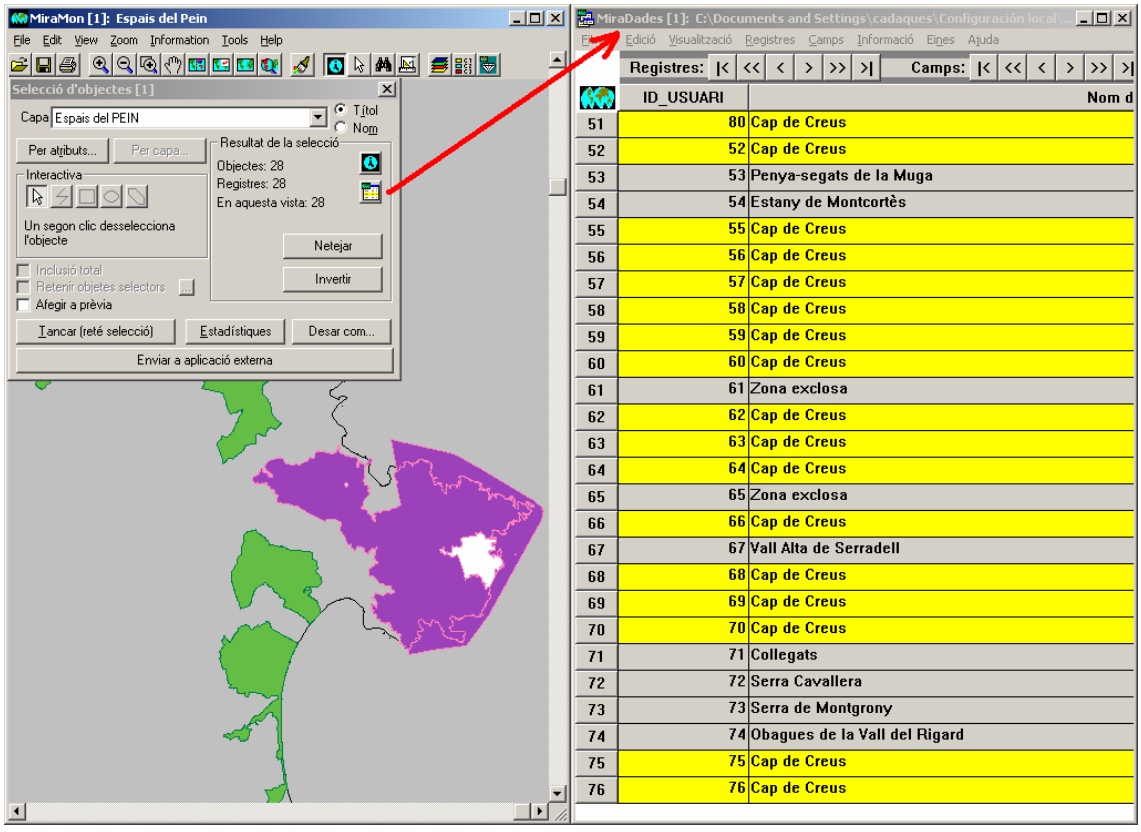
  It is possible to mark and unmark a particular register by pressing the register numerator button together with the "Control" key on the keyboard (like in Windows Explorer). Marking registers has no other use apart from making a given register easier to see. Thus it is possible to mark several registers using MiraDades by pressing Ctrl+register button.

  If MiraDades is launched in command line in order to place it in a register (/R=###, where ### is the register number), but you do not wish it to be marked in green, the parameter /no_marcar_reg should be added to the command line.





- Linking graphic and alphanumeric selection (multiple selection of registers in MiraDades). When a (by attribute or interactive) **selection** has been made in MiraMon, you can press the button and open MiraDades with the selected registers marked in an intense yellow color. The first register shown corresponds to the graphic indicator with the lowest index.



Transmission of selected/deselected objects from MiraMon to MiraDades is carried out by pressing the button and not every time the selection is changed.

If the table is placed in editing mode, the color of the marked register changes to light yellow so that it is similar to the gray/white of the conventional display. If the button is pressed without any selection having been made, this simply opens the table.

The table opened by this button is linked to the MiraMon session and later marking or selection operations will be directed to this MiraDades session. Similiarly, selection operations carried out in

MiraDades (see the paragraph below) will be directed to the MiraMon session with which it has been linked. If selections are made on another open layer using MiraMon, the program opens another MiraDades session and links it to that layer. So, it is possible to have a MiraMon session and several MiraDades sessions open at the same time, each one linked to one of the loaded layers.

Regarding the MiraDades-linked session (opened using this button or using the “Main Table” button of query by location) it is possible to select or deselect the registers that you wish using “Control+Shift+Register counter button”. The information regarding the selection is sent immediately (in this case it is not necessary to press a button to send the selection, although this could be an option if necessary) to the linked MiraMon session (which was opened by that MiraDades session), or to all the MiraMon sessions if MiraDades has been opened using Windows explorer (because in this case it does not know which MiraMon session it is linked to).

For selection from MiraDades to MiraMon, the program zooms in on the selection (this could also be made optional if necessary). A button will soon be added to the operations on registers contextual dialog box and also in the “register” menu. In the future, functions such as export, arithmetic, etc. may be programmed with this selection.

Closing MiraMon (or the layer that is linked to a particular MiraDades session) does not mean that the MiraDades session is also closed, because the program considers that the user may still wish to carry out operations on the table, although it does instruct the corresponding MiraDades session to cut off the link. Subsequently, if new selections are made in the MiraDades sessions, these are sent to all the applications and not selectively to the linked application. Similarly, when you close a table in MiraDades, the application informs the linked MiraMon session so that it can cut off the link. If new selections or markings are made in the MiraMon session and you press the button to display them in the linked table, a new linked MiraDades session will open.

- Regarding access to databases via OBDC, a number of solutions have been found in relation to Oracle when reading DSN and MMZ compression.
- Support for SQL sentences has been perfected, both in the command line and from the executing window, which now accepts consecutive calls (previously it only accepted a single call and then relaunched the application in order to carry out a second sentence). Note that in the first case it is possible to use MiraDades to carry out commands or sets of commands (e.g. in different lines of a BAT file) because the application is launched, executes the SQL sentence and then closes automatically. For example, it is possible to execute MiraD "E:\AntProject\Field measurements.mdb" /SQL="Drop table “ancient measurements", which

will eliminate the “ancient measurements" table from the “Field measurements.mdb" database.

- Similarly, MiraDades accepts calls in command line with /CREATE type sentences. For example, with an existing BD called C:/Hello.mdb, the next sentence creates a new table called ActivitiesTable, with two fields:

  MIRAD /sql="CREATE TABLE ActivitiesTable (ActivityCode VARCHAR(50), Year INT)" "C:\Hello.mdb"

  Likewise, MiraDades accepts calls in command line with /INSERT type sentences. For example, with an existing BD called C:/Hello.mdb, with a table called ActivitiesTable, the next sentence inserts the following register, with two fields:

  MIRAD /sql="INSERT INTO ActivitiesTable (ActivityCode, Year) VALUES ('08-78//12', 2005)"

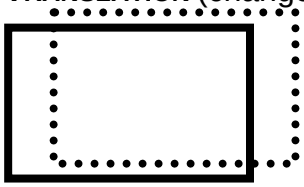
11.3. CorrGeom

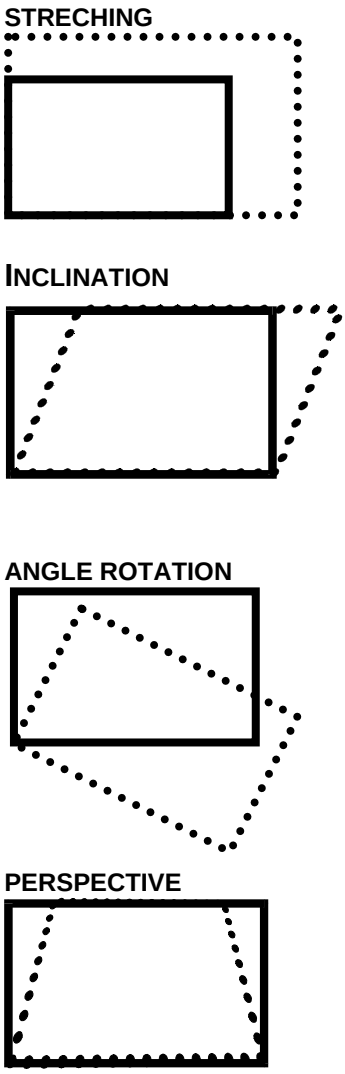
The main new features of the module are:

***Elementary geometric transformations*:** CorrGeom now allows you to carry out elementary geometric transformations such as rotations, translations, etc. without having to digitize control points. This is useful when, instead of adjusting a geometric transformation using a number of control points, you wish to indicate the parameters of the transformation because these are already known. Thus, you may have scanned a document rotated by 90º as it fits better in the scanner when rotated in this way. In this case, it will be easier to indicate that you wish to carry out a rotation of 90º to restore it to its original position and see it “naturally” (perhaps before placing control points to finely tune the geometry). A second example is when you have an old vector base, from software that stored the coordinates with single precision (such as PC-Arco/Info) and as a result the most significant figure had been truncated when it was constant for the whole layer (e.g. in Catalonia in UTM-31N it was common practice to suppress the 4 of the Y coordinate). In this way, a coordinate like 4619254.734 was written as 619254.7 and decimetric precision could be maintained. In this case, it is possible to recover the original coordinates by indicating a translation-type transformation of magnitude 4000000. Lastly, a third example would be to have a base in which the units of the horizontal reference system are km and one wishes to have them in m, for which one would apply a ratio value of 1000.

The elementary transformations implemented are:

TRANSLATION (change of origin)





Precise implementation with 2nd degree polynomials and large coordinates: CorrGeom will geometrically correct a raster or a vector, for example by using 1st and 2nd degree polynomials. In the case of the 2nd degree polynomial, the mathematical method used to adjust the coefficients was not precise enough in some cases (large coordinates like some UTMY). These imprecisions have been resolved and the 2nd degree polynomial adjustment gives much better results.

***EMANCIPA*:** This is a new mode that allows you to give the same name to the output file (in other words, it is no longer necessary to create an emancipated copy if you agree to lose the original that pointed to the arc source that was generated first).

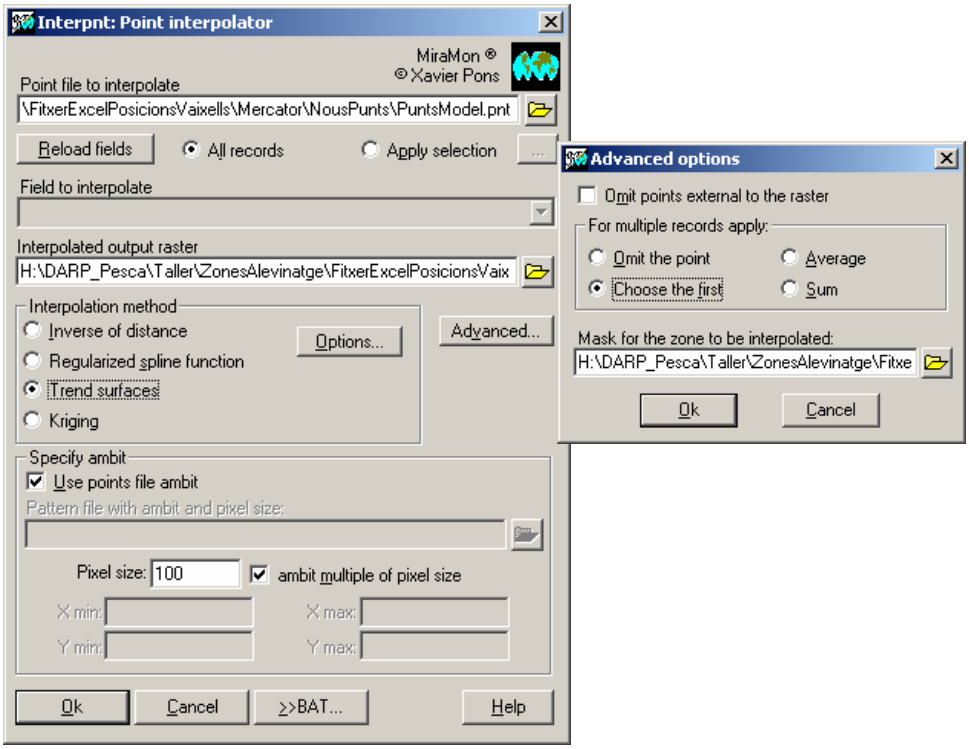
Generating orthophotos with no control point: If it is possible to know the position of the metric chamber or the attitude, you can generate orthophotos **without any control point**, which is particularly useful in chambers equipped with precise GPS and INS systems.

Generating smaller files: CorrGeom now generates highly-compressed and indexed RLE files, which are especially useful because there are often NODATA areas in the images for which compression is particularly beneficial.

11.4. InterPnt

Two new interpolation methods are implemented:

- **Trend surfaces**, which allow you to obtain a model of general trends, whether they be linear, quadratic or cubic. The program endeavors to reduce the magnitude of the coordinates before carrying out quadratic or cubic transformations which could result in a serious loss of precision.



- Kriging, which is explained in the point below.

11.5. Kriging

Kriging is a sophisticated interpolation technique for specific, irregularly-distributed data and is based on spatial statistics. Interpolation using this technique consists of two phases:

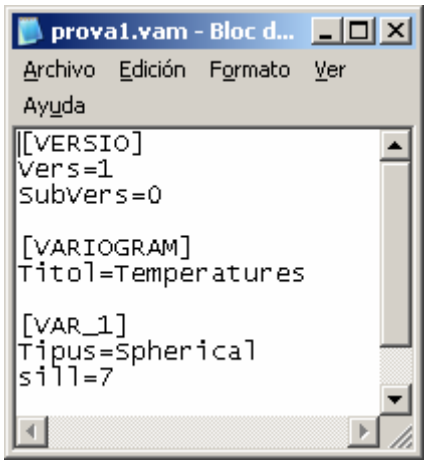
- The study of the spatial variability of the variable studied in order to adjust a function known as semivariogram (or variogram).
- The interpolation itself to generate the raster with the predicted values for the model in each point.

For this reason, kriging has been implemented in MiraMon in two modules: a new one, **Vargram**, and an existing one, **InterPNT**. This is a new option which

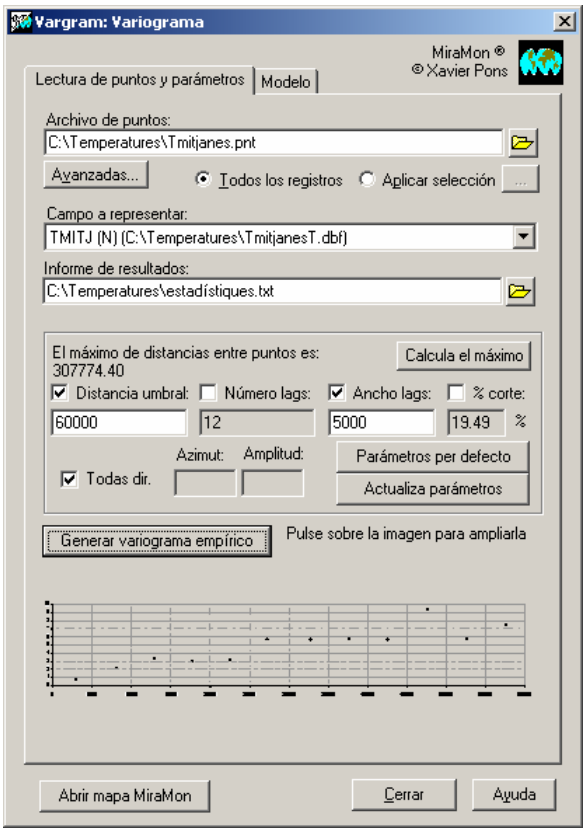
joins previously existing ones (inverse interpolation of the distance-weighted average and by splins).

Regarding the **Vargram** module, in this case it is the interface in the form of a **WVargram** dialog box which has the main utilities, while the module in command line (Vargram) has not been developed due to the interactive nature of the process (exploration and graphic display). WVargram is the tool that represents and models the so-called semivariogram, a function which represents the spatial distribution of data variability.

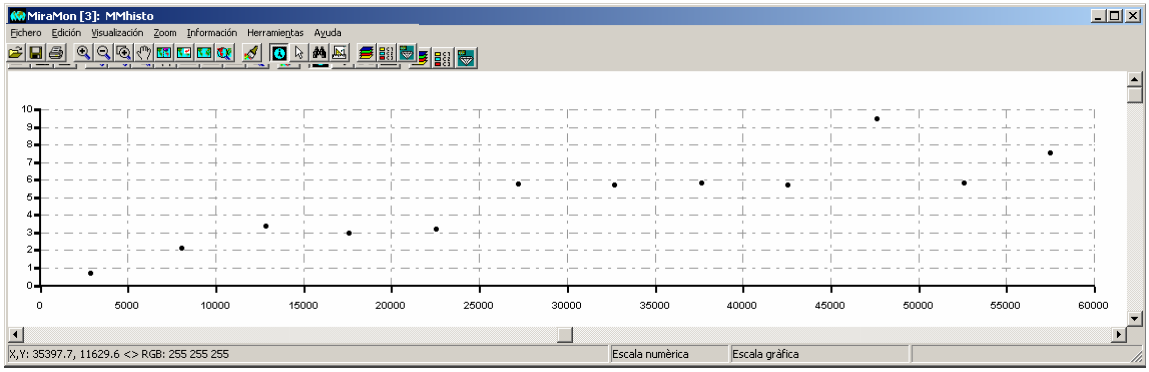
With this module, the user can construct the adjusted variogram as the sum of a number of the elementary variograms implemented: nugget, spherical, quadratic, linear, Gaussian or exponential, using the empirical variogram constructed with the data itself which will be saved as a **VAM** file.



The construction of the variogram consists of two stages organized in two tabs in the WVargram interface. The user must first fix the properties of the point sample and then define the geometric and graphic parameters of the empirical variogram.

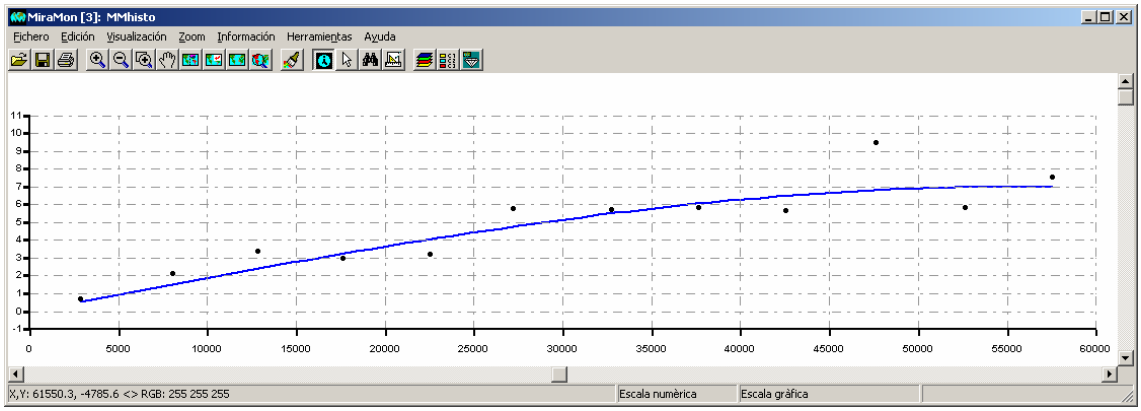


This first parametization will produce the points of the empirical variogram:



Secondly, the user chooses the elements and the parameters of each of the variograms that will make up the composite adjusted variogram. It should be noted that this requires previous theoretical and practical knowledge regarding the structure of the variogram in order to successfully find a variogram that is useful for the interpolation.

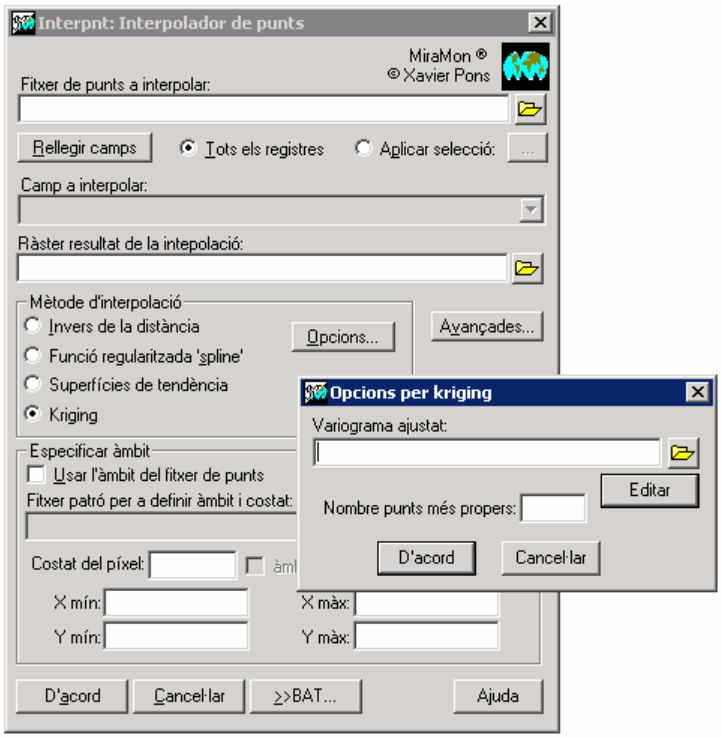
This second modeling will produce the line graph of the adjusted variogram and, if considered to be valid, it will be used by the interpolation module.



The new kriging option of InterPNT reads this VAM format and uses it to generate:

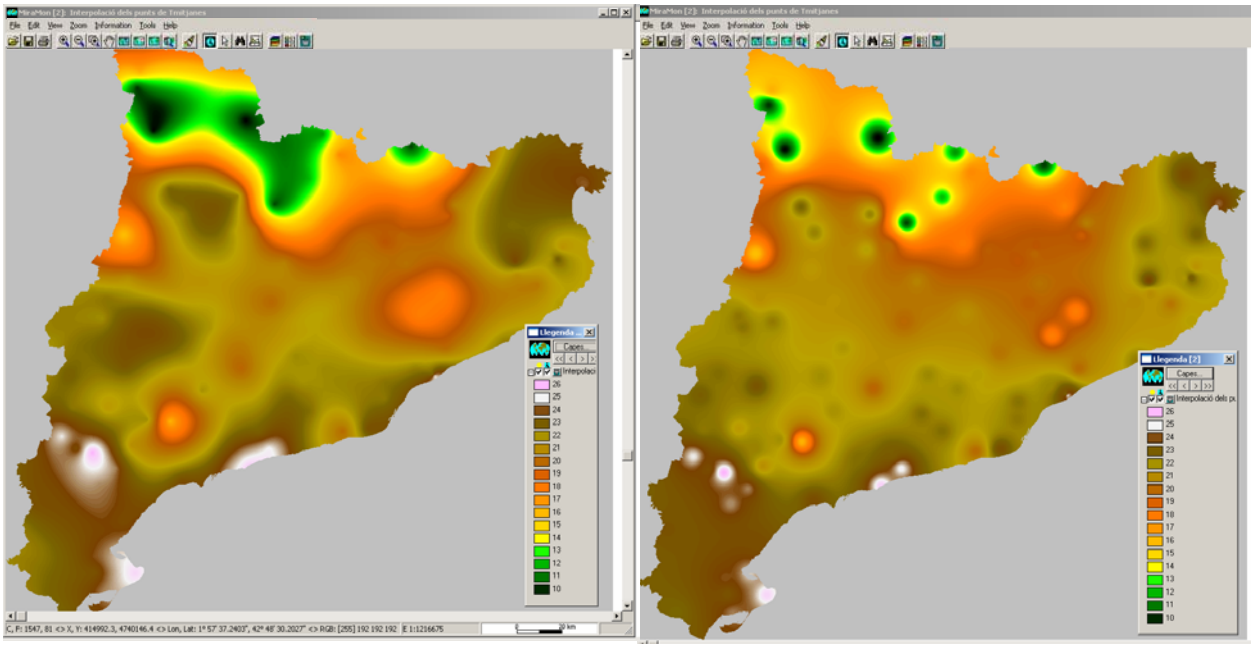
- the prediction modelling raster
- the error modelling raster

Of the different kriging models, **ordinary** kriging has been implemented in InterPNT, although it is possible to obtain **residual** kriging. In fact, it should be noted that by combining other MiraMon modules it is possible to carry out **residual** kriging with respect to a trend surface (with InterPNT itself) or with respect to a multivariant regression (RegMult module).



Although kriging is a procedure which usually requires a relatively large amount of calculation time, MiraMon has made an effort to optimize this application by considerably reducing the execution time when the user requires that the data that participate in the result of each interpolated pixel are the same (the whole set, most common case). Some programs ask for a maximum number of samples (those that are closest to the problem pixel) in order to reduce calculation time. This has also been implemented in MiraMon, although it may generate discontinuities in the result.
This effort reduces the usual enormous difference in calculation time with respect to a simpler model such as inverse distance-weighting (IDW) and, therefore, the still greater duration of the kriging execution may be a small price to pay if one compares the quality of the results which can be obtained in some cases.

Digital error model. This calculation allows you to generate, for each interpolated pixel, the error estimated by the model. The generation of this layer produces a multiband raster (interpolated band + band of errors).



Example: On the left, ordinary kriging. On the right, IDW.

NOTE: If you wish to consult some of the fundamental theoretical principles of Kriging, it is possible to request the short introductory document KrigingCourse.pdf which consists of a presentation given in an introductory course in CREAF prior to the development of the module that is described above.

11.6. Creatop

It is possible to introduce an arc file in the simplified mode of topological structuring in addition to the usual VEC file.

11.7. DGNMM

As a result of the detailed conversion of all the official cartography of Catalonia into MiraMon format, so that it can be distributed by the Cartographic Institute of Catalonia, many different aspects of this module has been perfected and made more sophisticated.

11.8. GPSMM and GarminMM

The main new features of this module are:

- It is possible to load arcs such as tracks to the GPS.
- The module continues to support new GPS models: the latest wrist-mounted GPSs also work satisfactorily.
- Improvements have been made in GPS navigation in real time by USB port: real-time navigation with a Series USB adaptor and the new GPS USB+Series (such as Garmin GPS MAP 60 CSx) both function correctly.

11.9. Mosaic and clipping

3D layers are supported in the clipping and vector mosaic functions.

11.10. Visible

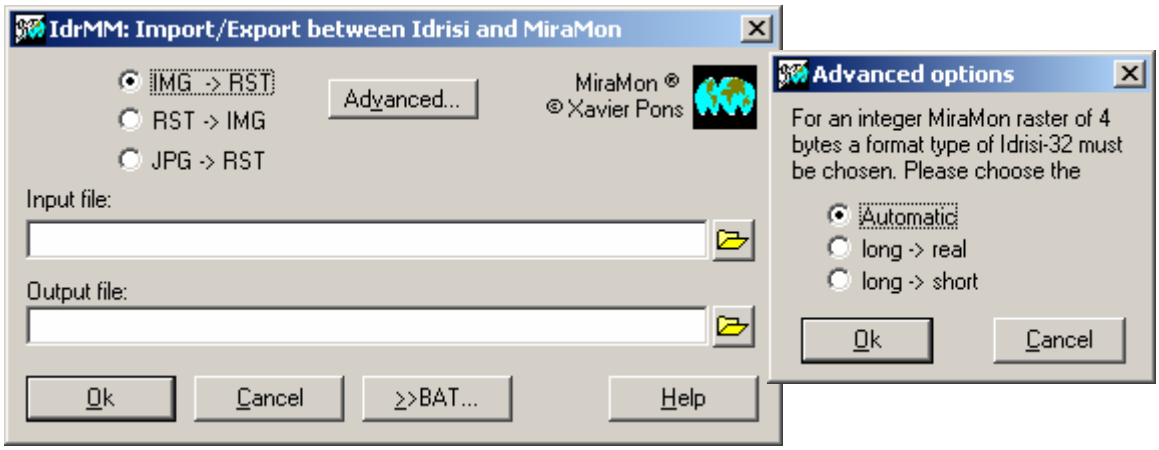
The visibility-analysis module allows you to determine **which areas are visible** from specific observation points.

11.16. UnirVEC

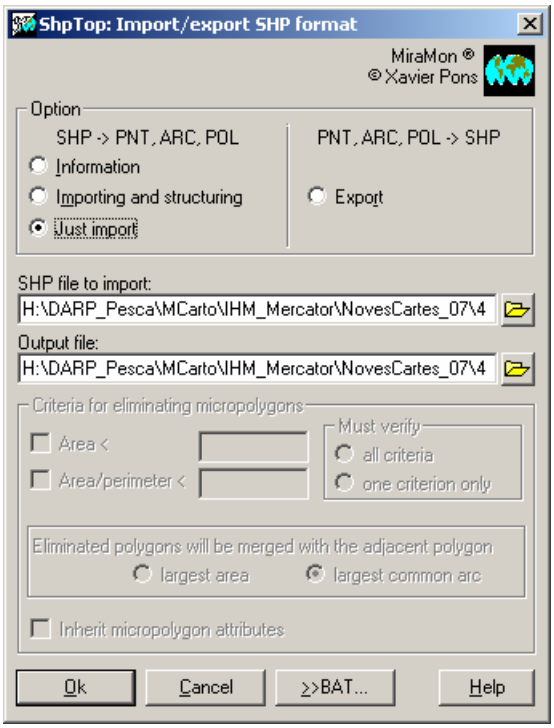
The union, without topological restructuring, of PNT, ARC, NOD and POL files has been incorporated. This new mode also supports lists of files for mass fusions. The structured destination files are marked as non-topological given the fact that no check is made of topological coherence.

11.17. IDRMM

A new module for the import and export of 32-bit Idrisi raster formats has been created. This is called IdrMM and has been incorporated into the import and export menus of MM32.



11.18. SHPTop: The export from MiraMon structured formats to SHP by ESRI format has been implemented. This enlarges the import options, with or without topological structuring, that already existed. When exporting to SHP, a specific Prj file is created for the ETRS89 Reference System that differs to the WGS84 one for the UTM projections.



11.19. DXFVEC: As a result of the implementation of dynamic reading of DXF files from Autodesk to MiraMon structured formats, several aspects of the program have been completed.

11.20. SurfMM: the export of points from VEC to DAT for Surfer allows the VEC to be 3D and in this case both the Z and the VEC attribute are treated.

11.21. RGBPal: converts between RGB formats and optimized 8-bit palettes. This is very useful for the superior compression of several different file types.

11.22. GestBD: several new options have been created for this powerful tool for managing tables and databases, such as:

- **Duplicating the register**.
- **Carrying out a ("physical") join of two tables using n links** (the old application that made the join specified in an INI file did this by means of only one link).
- Transformation from multiregister to multifield (in other words, it converts the different values in multiple registers into new fields with a similar, but not identical, name). This may be useful before exporting to programs that do not support multiregister [to define the multiregister, the repetition of the values of the key field of the table (usually the graphic field identifier for the main table of the GIS datasets) are taken as the criterion]. The fields are replicated as multiple fields as many times as the maximum number of repetitions of the values of the key field. This function is particularly useful given the complexity of the expression required to do the same in SQL. It should be noted, however, that this is usually

an unsatisfactory solution because it violates the “entity-relation” model of modern databases: the entities now have a number of properties which depend on the maximum number of registers to be created and this can, moreover, be very high in cases such as when the multiregister comes from observations carried out at fixed intervals (observations every 15’ of an automatic meteorological station) creates many empty fields in those entities which had a smaller number of multiple registers than the maximum of the table (e.g. if the multiregister represented many owners of a cadastral plot of land, and the maximum was 12 owners, the resulting table will have 12 fields, the majority of them empty). This cannot be applied to some table formats because the number of fields per table allowed is relatively small, etc.

11.23. MicroPol: a new parameter has been included that indicates that the user does not wish to maintain the registers with a percentage of partial area that is less than this value. The proportion of partial area lost is distributed among the rest of the registers that do exceed this threshold. If all the registers have a partial area that is less than that indicated, only the one with the biggest area is kept. In addition, an option has been added that allows you to condition the elimination of micropolygons next to polygon 0 depending on the arc attribute. This option is often useful for eliminating polygons abutting coastlines, but maintaining, in expectation of a mosaic with an adjoining sheet, polygons in contact with the edge of a cartographic sheet. The new possibility complements the pre-existing option of not changing the geometry of the arcs of polygon zero when the micropolygons are generalized.

11.24. Filters: the “Median” option has been incorporated as well as matrix filters of variable weights. The application offers a collection of files in INI format including the typical softening, detection and edges reinforcement, by 3x3 and 5x5 windows. It is also possible to create new files, save them, use them and modify them. In the latter case there is no limitation on size, except that it is not possible to use the interface of the 3x3 and 5x5 window, which is more pleasant for the user.

11.25. PGMIMG: this is a new module that has been incorporated into the MiraMon import and export menu and allows you to convert PGM and PPM formats to and from MiraMon. These formats, which are common in Unix environments, contain binary rasters (monoband and multiband respectively) preceded by a short text heading.

11.26. LinArc, Ciclar, AtriTop: a new option is added to LinArc to **fuse pairs of arcs that share a node** and, at the same time, ensures that no other arc coincides with the node. This operation does not carry out any topological restructuring (it does not split arcs or make connections, etc.). The most suitable operation when importing files from other formats that are limited in the number of vertices per object and are obliged to make more than one object per strip of vertices (as there is practically no limit in MiraMon to the number of vertices per arc, it is conceptually better to join contiguous arcs). In contrast, a

conventional structuring could generate unwanted intersections (e.g. roads on two levels with a level crossing). At the same time, it is possible for LinArc to **switch from ARC to ARC without structuring**. This is useful for generating files of road networks respecting the bridges which are not really connected.

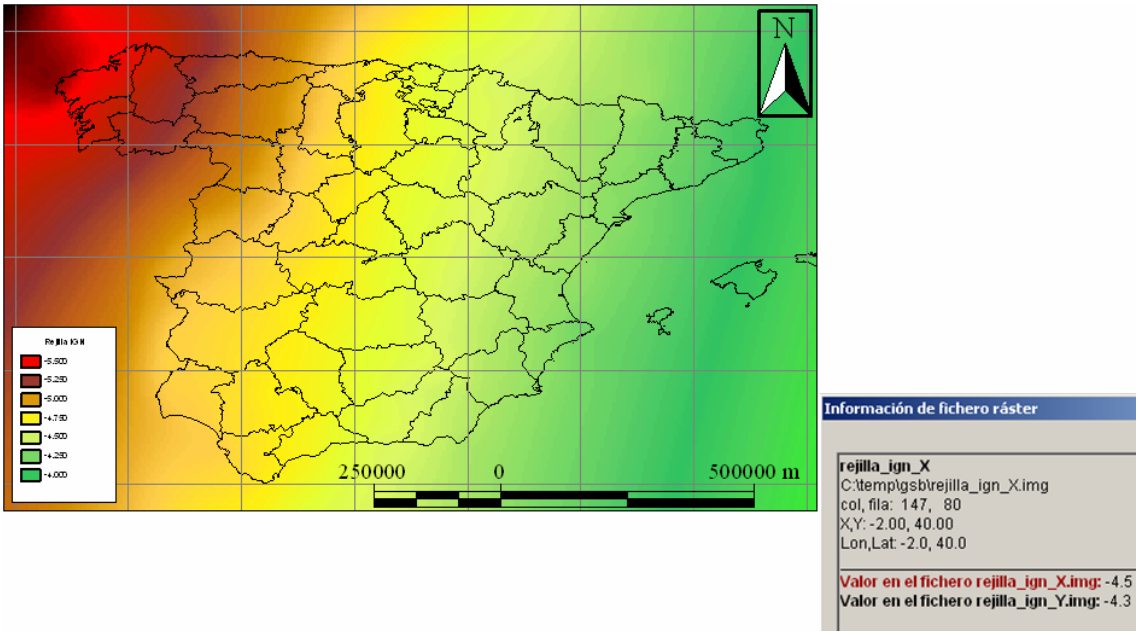
Likewise, selective recycling by node contact has been consolidated and **recycling of non-topological groups from arcs with the same attribute**, which becomes the attribute of the group of polygons and the creation of **non-topological groups with holes from topologically-structured ARC files** is supported (in other words, if a line was both the internal edge of a polygon (hole) and the external edge of another, the ARC file will have two registers); the attribute of the arc becomes the attribute of the polygon.

Lastly, AtriTop adds the option of **transference from PNT to NOD for reasons of geographic proximity**. There is an optional parameter of distance transference threshold and, in this case, it is possible for new nodes to be created if necessary (in preexistent vertices or by creating new vertices) always within this threshold. The distance between a receptor node and a pnt donor is also documented in a field of the database.

11.26. RegMult: the new MSA RegMultiple is also incorporated. This combines statistical methods (multiple regression) and spatial analysis (interpolation) for spatial modeling. The option of imposing all the independent variables (and not selecting the most informative ones) has also been added.

11.27. Routes: a new module which carries out an analysis of vector networks (graphs) according to distances, traveling time, etc.

11.28. Several improvements to geodesy: Canviprj, Calcgeo (MM32 and other indirectly involved applications). The formulae have been developed in order to implement an ellipsoid model for the Lambert-Azimuth Equal Area projection and, therefore, high-precision support for the ETRS89-LAEA system, chosen by the European Agency for low-detail cartography: "*For pan-European statistical mapping at all scales or for other purposes where true area representation is required, the ETRS89 Lambert Azimuthal Equal Area Coordinate Reference System (ETRS-LAEA) is recommended*". In this way, perfect metrics are guaranteed for everything from the projection of **Meteosat** images (see the figure in section 6.1) to rigorous **ED50->ETRS89** transformation, a new system in Spain (except the Canary Islands) (in the figure below there is a network of longitudinal movements for the transformation mentioned above, in sexagesimal degrees, and query of longitudinal/latitudinal movements).

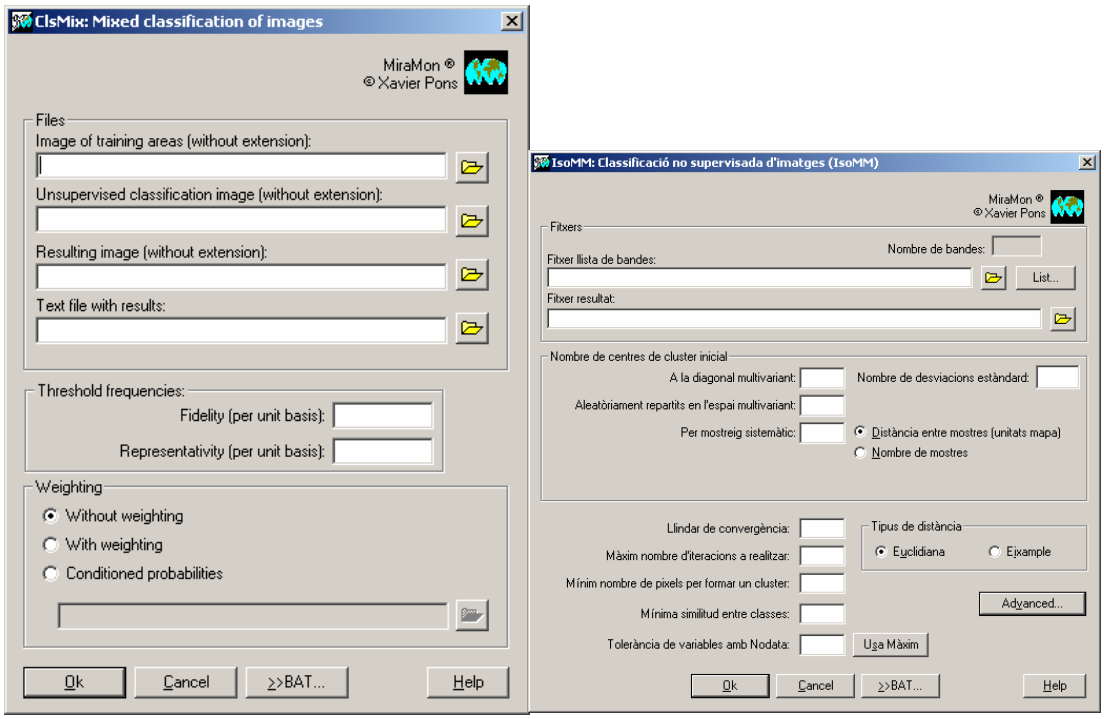


11.30. Insoldia: this is now able to give the radiation accumulated in an interval of time and not necessarily in a whole day (this had already been done in a non-accumulated instant).

11.31. RegioPNT: creation of regions from point occurrences.

11.32. Combicap: three new statistics are provided: first quartile, median and third quartile in the POL+IMG -> POL type combination with IMG statistics.

11.33. Others: a number of different improvements have been made to **CEOSIMG** for Landsat-5 (image indization, automatic palettes, better treatment of NODATA, etc.). MOSAIC fully supports 3D points. **CANVIRES** supports JPEG as well as mode for different type of byte (integer, real, long). **RETALLA** supports WMS layers. Numerous improvements have been made to **CLSMIX E ISOMM** for the classification of Remote Sensing images.



12. MiraMon Browser and Map Server

12.1. New functions, design changes and optimizations for the browser

A new “pan” or “hand” tool has been implemented that allows you to move over the browser view defining the section of the view to be moved by clicking on it and dragging it or by double-clicking. The user can establish the settings from the configuration window.

A new function has been implemented that enables you to go to a specific coordinate and display it on the map by means of a configurable symbol.

A new typical query system (by object) has been implemented which functions like a search engine and is intelligent and helps the user to assign a value to what he or she wishes to search for.

A new typical query system has been designed and implemented based on AJAX and XML technologies. This new design allows you to carry out typical queries (or by object) using several fields of the “Go to a Province” type, select a province and then “Go to County” and select one of the counties of the province and finally “Go to Municipality” and select one of the municipalities in the county.

It is now possible to define several typical query systems in a single browser by combining, for example, a search-engine type query with an initial list query.

A new type of Zoom by window has been incorporated which is activated by clicking once and dragging. It is possible to return to the old two-click style

whenever you wish as this function can be configured by the user from the browser.

It is now possible to define a number of Reference Systems and location images from the browser. This is very useful for browsers where the browsing area is very large and includes areas that are usually mapped with different reference systems and choosing only one would result in an extremely distorted view or a very different view from that to which the user is accustomed. The selection of an image and reference system will normally be done automatically and the browser will decide which layers to display and with what system. However, you will always be able to switch to manual and select the reference system you wish to use. To complete this new function, the layers can be configured in such a way that they switch on or off according to the area and the reference system. If you wish to see an example, go to http://www.opengis.uab.es/wms/europarc.

A format for queries by location (OGC-WMS GetFeatureInfo requests) based on XML and AJAX technologies and cascade servers has been designed and implemented. This makes it possible to interpret the information received and have more control over what information is to be displayed as well as standardizing the style of information from other layers. Moreover, in this way the browser's function is optimized so that the user can carry out other tasks while continuing to receive the information requested from the server.

The browser based on layers (HTML DIV) has been redesigned so that each component of the browser is now defined by a layer. It is possible to define totally static layers anchored in a particular position with a fixed size or layers with a dynamic position and size, partially anchored to other elements of the browser and which move and change in proportion to changes in the size of the browser. This allows the user to design resized images that are much more intelligent and suitable for different resolutions and screen sizes, incorporate new elements, such as logos superimposed on the view, that enable the user to personalize the browser and display information provided by the author, creator, … of the data.

A range of functions for drawing dialog boxes or information boxes incrusted in the browser (layers) have been designed. This eliminates the typical security questions which appear when you open an emergent window (blocked pop-up). The new function has been installed in:

1) the new dialog box for going to a coordinate
2) the new dialog box for queries by location, where the user can decide if he or she wants it to be incrusted in the main page or as an independent window (and switch between the two using the up and down arrow buttons in the top right-hand corner of the window).

CSS display styles have been incorporated into some elements of the browser, such as query by location, in order to make it easier to customize the browser.

Printing is now based on customizable templates with regard to both content and appearance. It is possible to define templates adapted to different types of

paper (vertical A4, horizontal A4), each one with its own elements. As a result, when printing, the user will be able to select which template he or she wants and in which of the available elements he or she wishes to display and modify the size and the position, and in some cases even the content, in the case of the title of the print-out document, for example.

The overall performance and speed of the browser have been improved by applying AJAX technologies and optimizing the programming in JavaScript. In this way, a solid base has been provided for the browsers on which new functions can be built in the future.

Another new feature is the display and query of point-type OGC-WFS objects (in GML format) using AJAX technologies. This allows you to obtain information about points contained in databases dynamically and directly.

Finally, it is possible to display and query layers of temporal series with easy and grouped control from the legend. This allows you to have a single layer in the legend and thus prevent the latter from growing indiscriminately making the legend unusable. If you would like to see an example of this feature, go to http://www.opengis.uab.es/wms/thalasa.

12.2. Improvements and new functions for the server

A rigorous review of the different versions of the OGC's WMS standard has been carried out and as a result a number of improvements and corrections have been made to the server in order to ensure stricter compliance with OGC standards, such as the negotiation of the version of the request.

A new OGC WFS (Web Feature Service) service based on GML (Geographic Markup Language) has been implemented for point layers of any database (DBF, Oracle, SQL Server, MS Access, etc.).

The syntax of WMS requests has been enlarged so that it is now possible to carry out requests in cascade from other external servers. This is very useful for avoiding security restrictions that prevent one from making requests to other servers.

The WMS parameter which gives support to the TIME dimension and makes it possible to request layers according to date and time has been implemented. This allows you to define layers made up of temporal series.

The options for layer preparation in the WMS server have been increased with the introduction of the preparation of point layers with symbols using a field and a symbology table.

A function has been developed to situate toponymy and symbols (small icons) intelligently. This means that when they are "split" by the area boundary, they move inwards or disappear completely depending on how much text/drawing is outside the requested area. This effect can be seen at

http://www.opengis.uab.es/wms/bau, where both the text layer and the symbol layers are intelligent.

13. Massive data processes and task automations

Whenever it is necessary to repeat the same process such as file import, layer analysis, etc. many times or it is necessary to regularly update the same map, it is worth saving time, avoiding errors and slips by means of batch processing.

These scripts are known in the Windows operating system (OS) as BAT files. These are text files that are edited with a notepad by entering the necessary commands and parameters so that the computer, without the user's intervention (the task can even be programmed for a specific time) or simply by starting it with a double click, executes what would otherwise have to be done manually by opening, one after the other, all the different program applications (e.g. the MiraMon MSA module) and repeating the process for each file you have or map you need.

Each version of Windows introduces new commands or enlarges its "System Prompt", also known as its "Command Line" or previously as "MS-DOS", with new parameters. MiraMon takes advantage of all these commands to leave the computer working in the background, or at night, copying, moving, erasing, creating, changing names, comparing, listing files and directories and even turning off the computer when the tasks entered on BAT have been completed.

But in addition to the large number of commands provided by the OS and the fact that it is possible to obtain them listed and described by executing the "HELP" command form the system prompt or with the help of OS, MiraMon includes a "BAT" button in most of its applications (MSA) to help us to construct these scripts. Each execution of one of these applications from the window or the dialog box corresponds to a line in the BAT with the following structure:

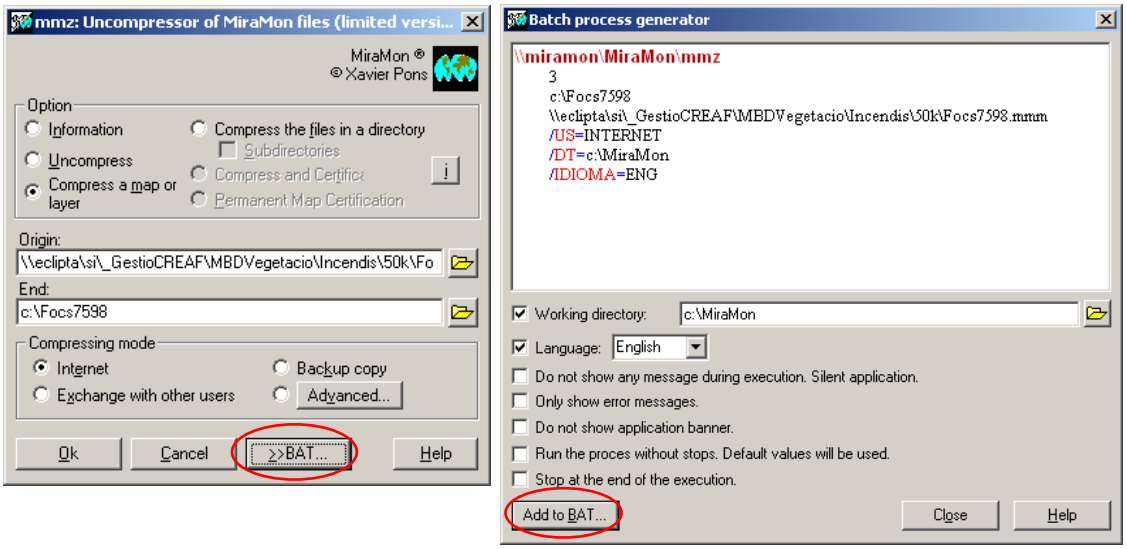
COMMAND Parameter1 Parameter2 Parameter3 [Parameter4]

For example, if you wish to compress an MMZ file:

```
MMZ 3 E:\Time\DefinitionAreaInterest "S:\Data\Definition Area Interest.MMM" /US=INTERCANVI
```

Thus, the functions that, in other programs, are implemented using a programming language and a set of libraries belonging to the program itself, are carried out in MiraMon by taking advantage of the totally standard commands of the OS, combining them with our applications and in this way creating BAT files to reduce the user's workload.

The above command was obtained using the BAT button of the MMZ application, with which it is possible to view the command and parameters to be executed from the system symbol and save it to a BAT file.



A simple example of the import/update of a point layer from an MS-Acces database in order to combine it with a municipality layer, creating new fields, concatenating (joining) and eliminating those which are not necessary to obtain a table with the desired data that can be automatically entered in an MS-Word report would appear like the example below:

Update incidents. bat

```
REM ** Creation of a point layer from an Access database **
BdPnt "Incidents2008.mdb" "Incidents2008.pnt" "/CAMPX=X Coordinate " ↵
"/CAMPY=Y Coordinate " /ODBC /TABLE=Species /DT=D:\MiraMon

REM ** Combination of incidents with town council's layer **
CombiCap "Incidents2008.pnt" "TownCouncil\c250m\towns.POL" ↵
"/FCAPA=Incidents2008Towns.pnt" /DT=d:\MiraMon

REM ** Creation BD's file **
GestBD 5 "Incidents2008TownsT.dbf" INCID_CODE "Final code" FINAL_CODE C 12 0

REM ** Code fields' link **
GestBD 10 "Incidents2008MunicipisT.dbf" FINAL_CODE INCID_CODE const(-) PROVINCIA

REM ** Export coordinates to the table **
PNTBD "Incidents2008Towns.pnt" "Incidents2008Towns.dbf" /DT=D:\MiraMon

REM ** Extract decimal numbers from coordinates **
GestBD 7 "Incidents2008Towns.dbf" MAPX "X Coordinate" COORD_X N 6 0
GestBD 7 "Incidents2008Towns.dbf" MAPY "Y Coordinate " COORD_Y N 7 0

REM ** Make a copy of the BD **
COPY /Y "Incidents2008Towns.dbf" "Reports\Incident2008_Species.dbf"

REM ** Erase unnecessary fields **
GestBD 4 "Reports\Incident2008_Species.dbf" ID_GRAPHIC
GestBD 4 "Reports\Incident2008_Species.dbf" INCID_CODE
GestBD 4 "Reports\Incident2008_Species.dbf" ID_USER
```

More advanced information is available in the MiraMon "Help" files.

- Appendices – Support Module for MiraMon
- Complementary Concepts for MiraMon – parameters in command line

in the “Help” files of the Windows OS (files or batch processing) as well as on the Internet or in any MS-DOS manual.

14. Communications between MiraMon and other applications

Among the improvements incorporated into version 6.0 is one that accepts the CLAU_SelecReg= in addition to CLAU_SelecObj= in communication commands between MiraMon and other applications. As a result, it is possible to request MM32 and MiraDades to select, based on registers (usually of the single table), what is useful when there are associated tables with multiple cardinality (1→many and many→many relationships) as this avoids uncertainty between graphic objects and registers. When CLAU_SelecReg= exists, CLAU_SelecObj= contains the same number of items. In other words, it contains repetitions of identifiers of graphic objects, if these are necessary.

15. Telematic installation and unistallation

The traditional installation of the license based on a key diskette has been replaced with a telematic installation with or without a direct connection to Internet. The new system permits installation and uninstallation and consequently updates and new licenses are handled in this way. If a user should prefer to continue using the diskette system, this will continue to be available for a limited period.

16. New ways of communicating with users

16.1 MiraMon versions diary

A website has been set up: http://www.MiraMon.uab.cat/vers_nm/ where it is possible to:

- see a complete list of new features that have been incorporated into the program since 2002.
- use registers ordered by date to see all the new features, corrections, etc. as they are added to the program.
- carry out advanced searches. For example, it is possible to search through all the registers and dates that refer to “single table” (operator “with”) in 2007 or in a particular version of the program.

16.2. MiraMon Users’ Forum (FUM)

Given the current number of users of the program, we have decided to set up a MiraMon Users’ Forum (FUM) on the Internet. In this forum, users will be able to:

- ask other users how to perform a given operation.
- discuss aspects relating to the way the program works.
- ask about the existence or availability of data that cannot be found in the IDEs.

- Enquire about jobs in specific fields.
- look for collaborators with whom to carry out specific tasks
- etc.

MiraMon users will receive a message that will invite them to participate in the forum associated with the language of their license and that will give them all the necessary indications. However, it will also be possible to subscribe to the MiraMon forums in the other languages accompanied by an automatic translation into the language of users' license.

17. Several forthcoming features for v.6

Each new version of MiraMon always incorporates a number of important new features. However, while a new version is being developed many additional improvements are made. Below is a list of those that will probably be developed in the near future. We would like to stress, however, that there will be more depending on the kind of scientific and technical advances that take place, any shortcomings detected, the specifications of standards that appear or any assignments we receive. Some of them are being developed nowadays and will be available very soon:

- **New types of predefined boxes**, especially elliptical limit or circular ones (which are useful for highlighting an area), those that contain arrows, those for situation maps, etc.
- The possibility of making the **boxes appear in the query screen** and not only in the print design screen. It will be possible to locate the boxes in the screen in window coordinates or in map coordinates.
- Support for the reading mode with no memory load in the case of ECW files and support for the pecw:// protocol.
- Support for the direct reading of OGC's **WFS** and **WCS** specifications. WCS writing is now available on some MiraMon servers, but its functions will be increased. WFS writing will be implemented.
- Support to the direct reading of **GPX** (GPS Exchange Format) and **KML** (Google Earth).
- Support to the direct reading of GPX (GPS Exchange Format) and KML
- Total support to BMP Windows and OS/2. Since the end of April, it's possible to have a BMP + B.rel. MiraMon's libraries plenty support this case as well as GeMM (for instance, files and columns are read from the top of the BMP file).
- Greater support to World files for GIF, PNG, JP2, J2C, SID i BMP (GFW, PGW, J2W, JCW, SDW and BPW).
- More support functionalities to **JPEG2000**, to be added to the current visualization, etc.
- From MM32, and while visualizing a WMS layer (read from a MiraMon's server or anyone else), it's possible to save the view with a REL5 structure. This will generate suitable data to be offered as a REL5 net resource or as a WMS server for the boundary box requested. This is a useful complement to the current option that save the WMS view with the requested resolution (not necessarily the one of screen visualization).

- **Direct digitizing with complete topological structuring**. This will complement the current PNT and ARC/NOD digitizing options without automatic intersections and will often avoid the need to use the complete topological structuring tools (LinArc, Ciclar, etc.) when the layer is being digitized. However, the structuring tools will continue to be useful for when a layer from a CAD, SHP, etc. environment is received which has no topology and it is necessary to carry out an automatic analysis of the whole file, or when you wish to carry out a mass conversion of many files.
- It's possible to **undo when digitizing**.
- Use of object selection to carry out an operation on the subset, such as changing its attributes (e.g. assigning the same owner to all the selected plots), erasing them, etc.
- The possibility of defining the **transparent color** of the raster palettes as well as defining **semitransparencies in raster layers**.
- More explicit description of raster symbolization.
- Possibility of establishing **user-defined intervals** in raster and vector symbolization as well as writing **manual descriptors**.
- **Multifield texts**, which are particularly useful when you wish to view the content of several fields at the same time.
- Symbolization of patterns in polygons and lines.
- **Floating labels** appearance (*Tooltips*) when passing by an object and stopping the cursor above, without the necessity of clicking it.
- Line symbolization based on overlay of several lines. This allows you to easily construct the appropriate appearance for freeways as roads with two or three lines, etc.
- New search options.
- Several improvements in **metadata:** new keys, extended keys [for instance rasters' Type of content key can assume five different values, instead of the primer three ones implemented (the ones that ISO indicate), as these three are insufficient to major applications; the new codes implemented at GeMM are *Image without radiometric value* (for example scanned aerial photograph exported as "Image" ISO), *Image with radiometric value* (in DN, radiance, reflectivity, etc, exported as "physical measure" ISO), *Digital Terrain Model* (for example elevation, pluviometry, evapotranspiration, pollutants, etc, exported as "physical measure" ISO), Ordinal thematic classification (for example DEM with intervals exported as "Thematic classification" ISO) and *Categorical thematic classification* (for example land use map, exported as "Thematic classification" ISO).
- **High graphical quality histograms able to be pasted to offimatic applications via portapapers**, coming from analyses and visualization of rasters and vectors.
- News and improvements in analysis tools, for instance anisotropic distances in rubbing surfaces, **image classificators**, **triangulation** as the base for generate **Thiessen polygons** or create **TINs** (nowadays are supported but not created), isolines' creation, **quartiles and other statistics**.

- Continued support and new improvements in **GPS**'s data loading and unloading as well as in dynamic reading.
- New options in polygons' **cartographic generalization**.
- Improvements at **MiraDades**, as the possibility to configure width column visualization, etc.
- New editing's environment of the **MiraMon.par** file in a dialog box.
- Improvements in **Internet servers**, as support to OGC's **TIME** dimension, **dynamic points**, etc.

18. Final note

As always, you are welcome to send any suggestions, error notifications, etc. to suport@MiraMon.uab.cat. Many thanks.