



by André Pascual  
<apascual/at/club-internet.fr>

*About the author:*

Originally an industrial designer, now professor for production, he teaches CAD. Computer graphics, especially 3D graphics, is one of his passions.

## QCAD: Technical drawing with Linux



*Abstract:*

**QCAD** is a 2D CAD system with which you can draw and modify plans easily.

---

*Translated to English by:*  
Katja Socher  
<katja/at/linuxfocus.org>

## General Notes

A "plan" is any precise plane representation of a real object for study or for production purposes. The dimensions of each element (entity) of which the drawing consists of, must be exact no matter what scaling is used. This differentiates a CAD program from a vectorial drawing tool such as Sketch, Illustrator or Corel Draw, which is a more or less faithful representation of reality. With CAD a plan first of all has to be exact. This is in contrast to illustrations (Drawing) where the aesthetics of a picture are of more concern.

## Installation of QCad

The version qcad-1.4.x used for this article is on the applications CD of the Redhat and Mandrake distributions as ready-made package. Other distributions surely have similar packages. You can download the newest version at <http://www.qcad.org/>. Qcad needs QT 2.2 as GUI library.

# A little bit of theory

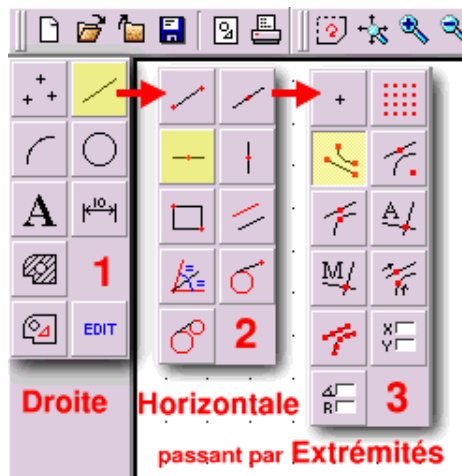
Before beginning with your first document you should have understood certain CAD concepts and definitions.

## The entities

An entity is a layout element that is "known" to the program by its form (segment, arc...), in the geometric characteristics of its position (vertical, tangential...), in its start and end positions which determine its dimension (fixed at intersections, coordinates, center...), in its attributes (color, thickness, types of characteristics) and its membership to a layer (blue print). Generally speaking to build an entity, it is necessary:

- to define your view on the working layer
- to define the attributes
- to choose the nature: straight line, segment, circle, ellipse, point, curve, hatch, text...
- to indicate the geometric framework of the construction: horizontal, oblique, concentric, vertical...
- to indicate the constraints

It results in building a virtual but exact sentence of this kind: *circle of radius X from the center passes through the endpoints of an entity which was indicated by a right click*, etc. The points will have to be indicated by a left mouse click near the desired points which have to be chosen among those that the system offers. It should be noted that the concept of fixation is found elsewhere under the name <snap> For example the sentence *horizontal straight line, which touches the outer lines of an entity* is composed with the following menus and sub-menus:



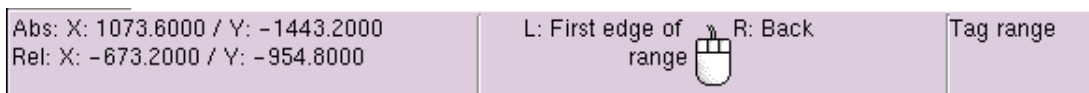
Note: That you get to the first start menu by clicking with the right mouse button onto the "paper".

## The layers

Elsewhere called levels, plans, blue prints. The layers describe in fact a virtual pile of celluloids. Each celluloid contains a part of the drawing, recognizable as a whole if you look on the pile from above, thanks to its transparency. A layer can be moved in the pile, removed (which affects only the part of the drawing that it contains), frozen or made invisible. The layer on which you work is the only active one at the moment. The operations that you carry out affect only it. When you assign attributes of color, line types or line thickness then all entities that you draw will have them by default. However you could assign blue to an entity that is on a red layer by modifying its properties. For a complex drawing you will work on one layer after the other which allows to make a certain subset visible or invisible, print only one piece, modify nothing but this.

## The status line

The status line is located in the lower part of the main window. It is not specific to CAD software programs but nevertheless essential. As a command requires several successive operations carried out in a defined order, the program shows in the status line the operations that should follow and what it expects from you and this until the end. It is therefore absolutely necessary to read the information that is displayed in that lower line if you do not want to risk that the CAD session ends with the declaration that this is a +-@-#!! program. In CAD the result is precise if the designer is working exactly and systematically.

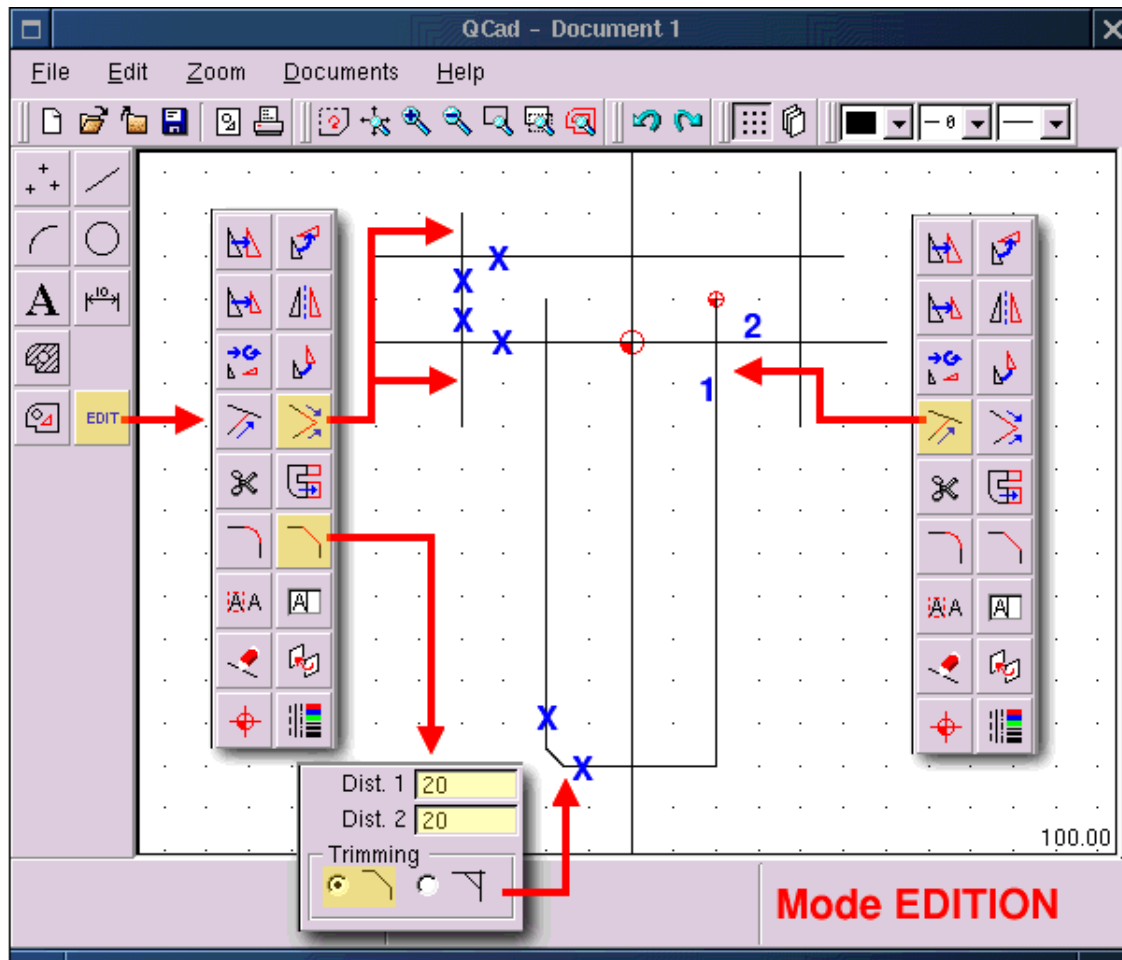


## Methods of drawing

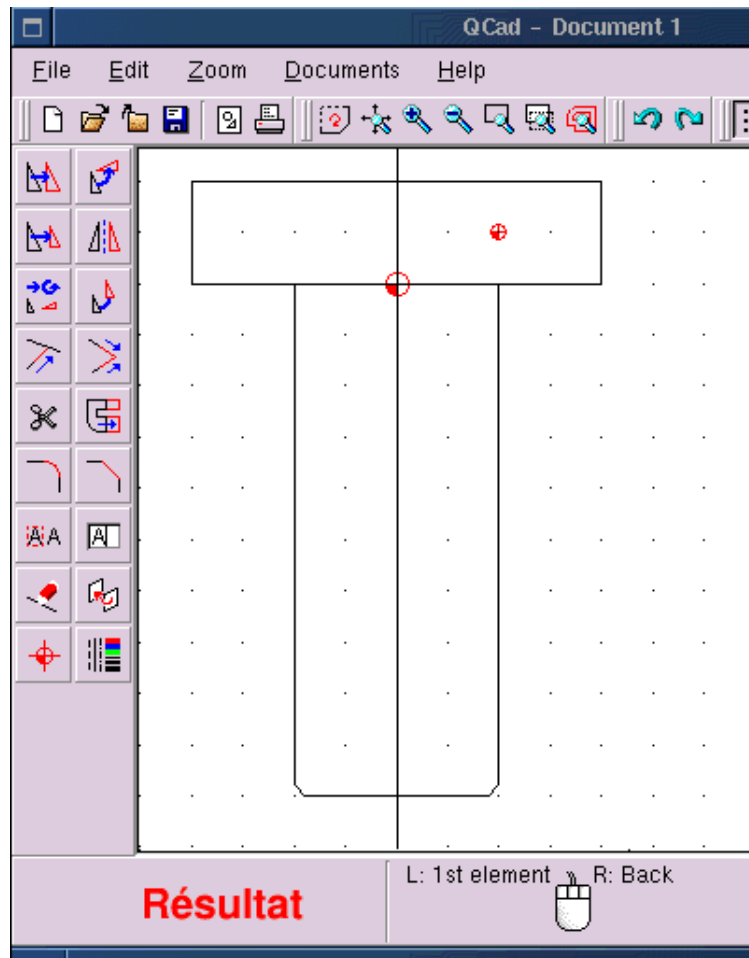
There are several ways how to do it, with at least two of them being excellent. Both use the concept of drafts based on not dimensioned (very long) straight lines but with precise relative positions (distance of one compared to the other). These straight lines, horizontal and vertical are called lines of construction in DTM or SolidWork and geometries in TSCadDraw.

The first method consists of defining a profile based on these straight lines as points of support. The exercise which we will follow in the rest of this article will demonstrate this.

The second method consists of defining a profile by adjusting the lines at fitting intersections. To do this with **Qcad** you have to right click to get back to the main menus and then choose <edit><Trim two object> then click on the line that you would like to trim (cut) next click on the line where your first line should stop. Here are 3 examples of editing objects:

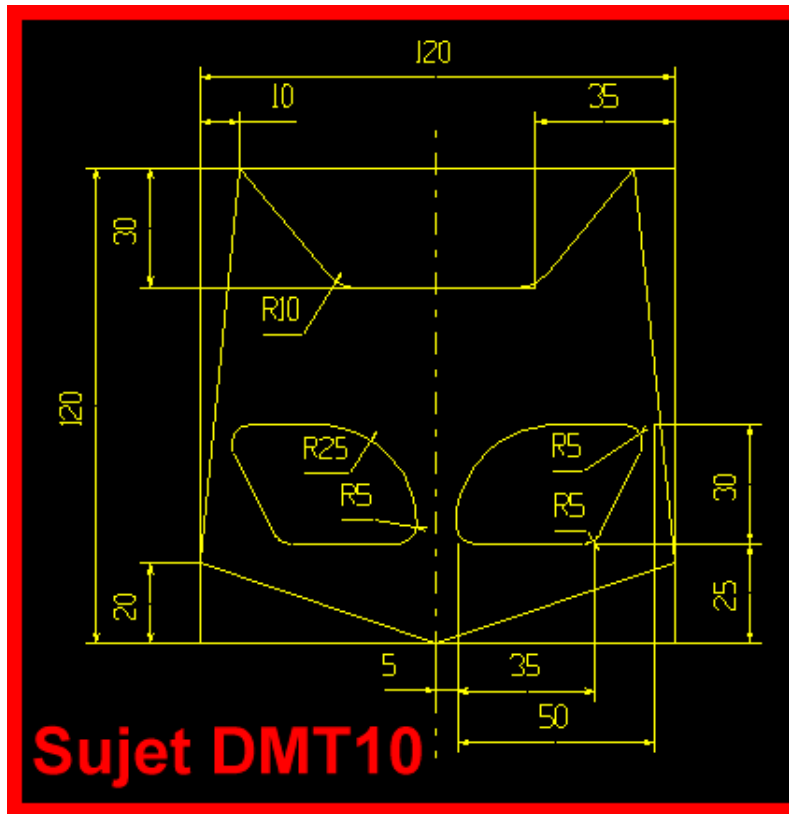


In this figure as in the following ones the yellow boxes show the selected functions, footnote: not colored by **QCAD** itself, and the blue crosses show corner points on which you can click. With the function <Trim objects> one makes an element fit to another. It is important to click first (1) on the part that you want to fit and second (2) on the entity that intersects the first. For the function <Bevel> it is important to determine the X and Y values of the edge before, no matter if it is trimmed or not and finally to click on the entity to be beveled. The steps are the same when working with intersecting lines and the function <Round>. One should also mention that QCad tries to be quicker than the user or tries to help you with your decision, in fact when a function is activated which needs the selection of a second entity to go on, then **QCAD** modifies the color of that entity which is near the pointer and indicates to you that you can select it with a left mouse click. It works the same way with the fixation points which are colored red. The right mouse click cancels an operation and allows you to go back to the main menu. The following figure shows the result of these various adjustments:



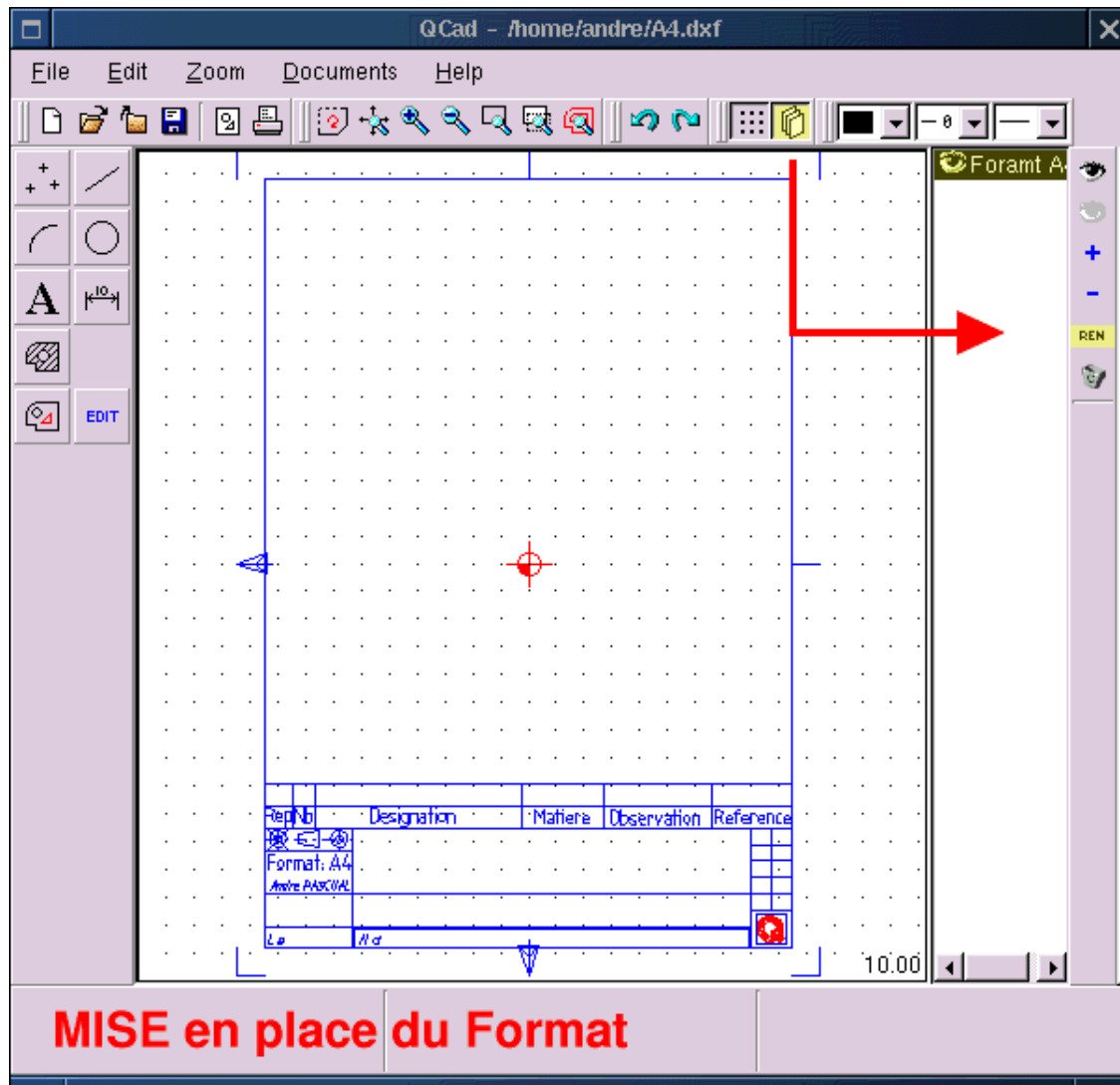
## Example application

To get to know a program there is nothing better than to use it. Let's try to draw an object which is inspired by the logo of SEV Marchal which I usually use for my beginner's courses in numerical control. With DMT 10 by Mécasoft it can be drawn in less than 5 minutes, annotations included.



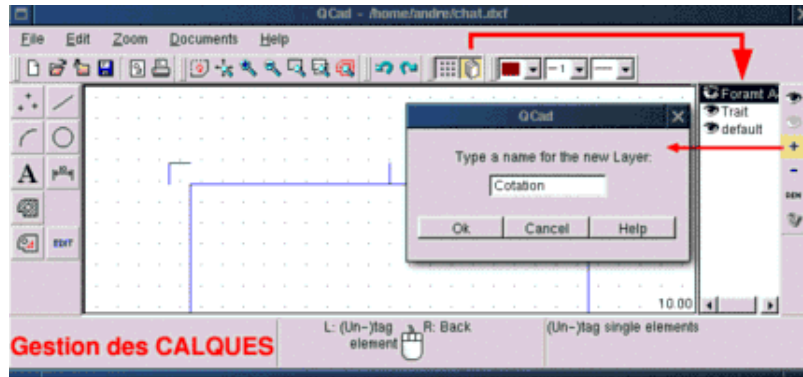
## Set up of a page format

This is not strictly necessary for the exercise but a technical drawing obeys to standards which define among other things the view and the aspect of the format (frame and data block) in which they are contained. Here I use a format coming from DMT10 transferred to DXF, the only file type that **Qcad** can read and generate which in turn guarantees the exchange in two directions with all CAD programs in the world. Once the format is loaded you see a mark off of the drawing zone with a zero reference in the middle of the area. It is recommended to delete unnecessary layers, to rename those which contain the format frame and to add those that are described in the following paragraphs.



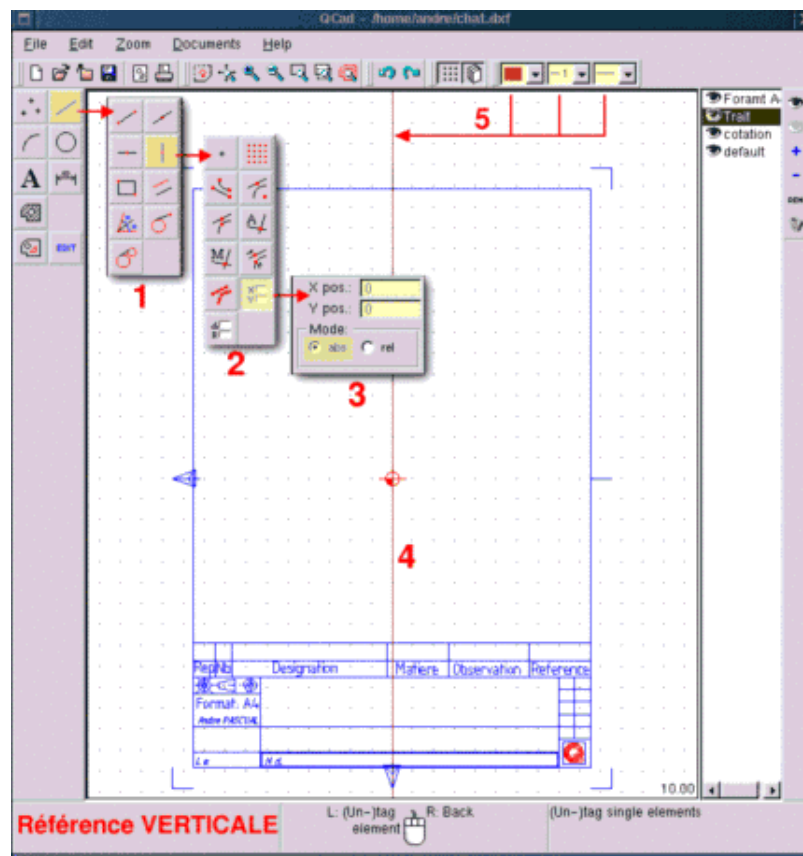
## Managing the layers

By clicking on the icon representing several piled up pages you open a side window called "Layer List". The selected blue print becomes the active blue print, it appears with intensified brightness. The eye besides the name of the layer allows to make it visible or invisible. The open eye to the very right of the layer list window makes all layers visible while the closed eye makes them invisible. The plus sign adds a layer to the list, the minus sign eliminates the selected layer, the symbol REN allows to rename the selected layer and the trash can deletes all empty layers. Now we need a layer <Format A4> that contains the frame, a layer <Trait> that contains the drawing in a front view and a layer <Annotation> that contains the annotations to the drawing.



## Defining a vertical reference

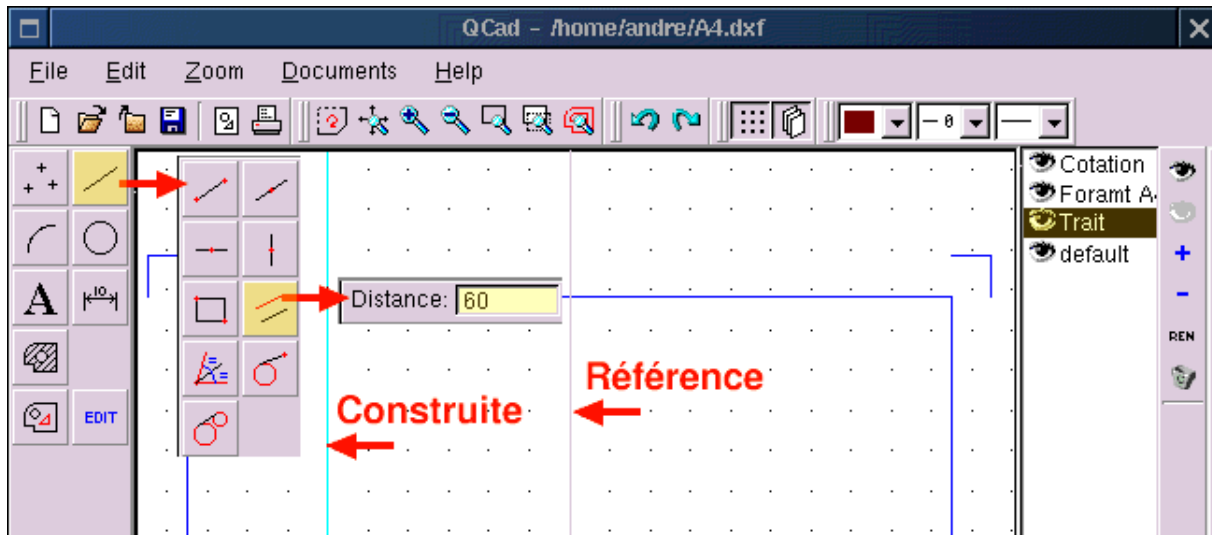
This straight line passes through the origin (zero) and allows the construction of parallels on the X-axis. Our drawing will be symmetrical to this line. At first you activate the layer <Trait>; then you choose a continuous width type with a thickness of 1 and the color red (point 5) then you construct the "sentence" <straight line><vertical(1)><passing through absolute coordinates(2)> <enter 0,0(3)> <click left(4)>





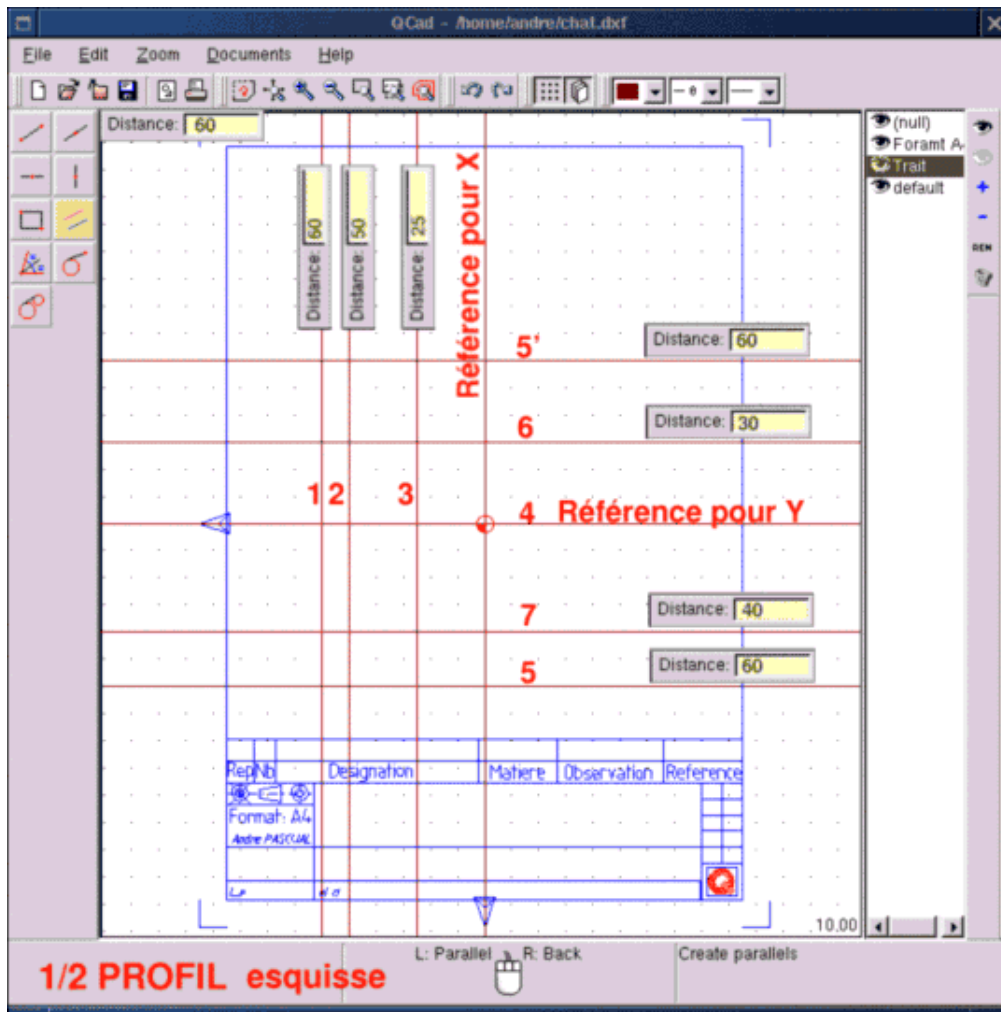
## Construct a parallel to 60

If necessary you remove the menus by several right mouse clicks and construct the following: <straight line>< Create parallels><enter 60> and approach the position marks of the reference straight line. It will become grey in turn and according to the position of the pointer **Qcad** suggests to construct the parallel either to the right or to the left of the reference. Position the pointer slightly to the left and make a left mouse click. A straight line in cyan is created.



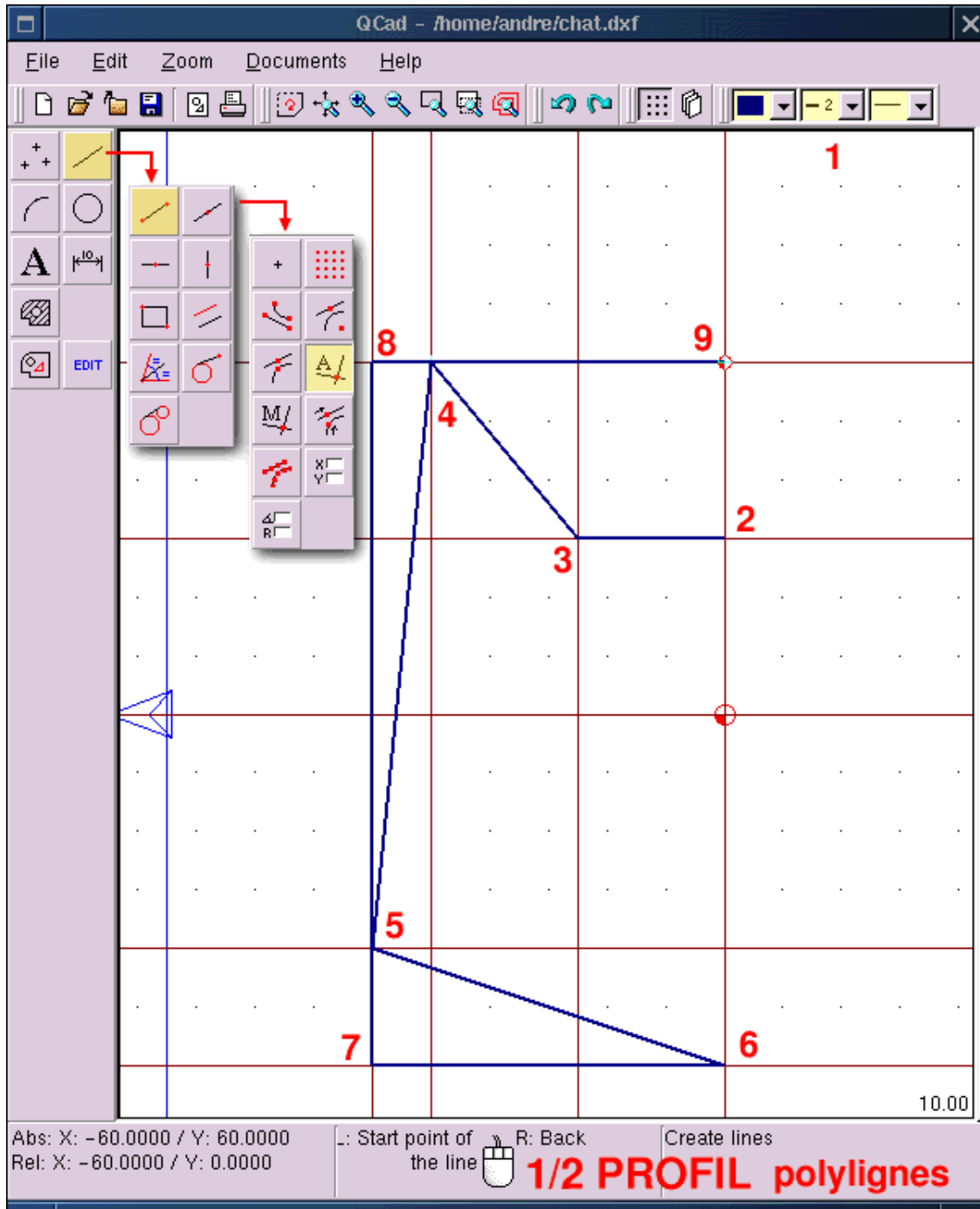
## Construct the other straight lines

Most functions of **Qcad** are repetitive, that is, they are active as long as they aren't replaced by another. Therefore <straight line><Create parallel><Distance of > is still activated. It is sufficient to replace the value 60 by 50 (Vertical 2) and to click and then to replace 50 by 25 (Vertical 3) and then to click. Go on with this for the horizontal straight lines which are defined with regards to the reference at 0 (Horizontal 4). Draw the horizontals (5) and (5') with distance 60, then (6) with distance 30 and finally (7) with distance 40.



## Construct the left half of the logo

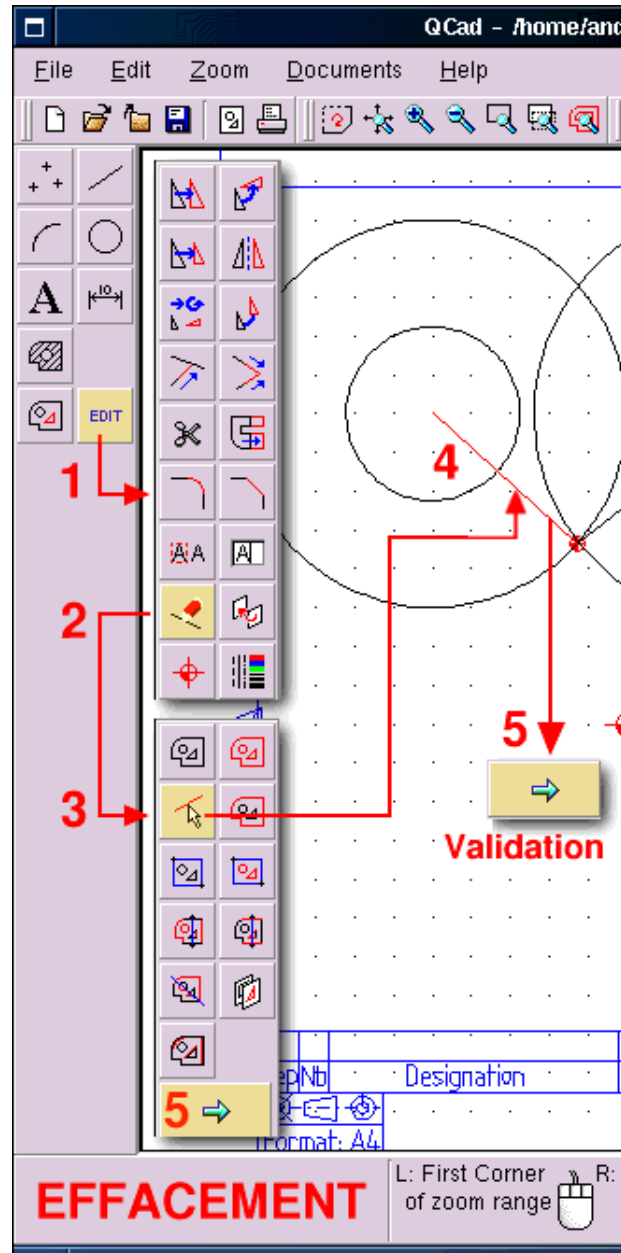
For the construction we rely here on the straight lines which we have just drawn. You have to go back to the main menu with a right mouse click and choose <lines><multiple lines (button: create lines)>< passing through the intersection (Snap automatically to..)>. From this moment on when we position the pointer near the intersection of the straight lines it will be marked with a red circle. When this intersection is suitable as the beginning of a character segment, then make a left click, move to the next intersection and make a left mouse click again. The segment is drawn. But as the function is modal this last point which is the end of the segment that we have just drawn will at the same time be the beginning of the next segment. This allows the drawing of closed contours. If you don't need it for an additional segment a right click will interrupt the active function but doesn't cancel it. So for this half side of the profile choose a continuous line type, thickness 2 and color blue like in point 1 in figure QCad10 (below). Then left click on 2, 3, 4, 5, 6, 7, 8, 9 and finally a right mouse click. The left half of the profile is ready. Easy!



## Editing the result

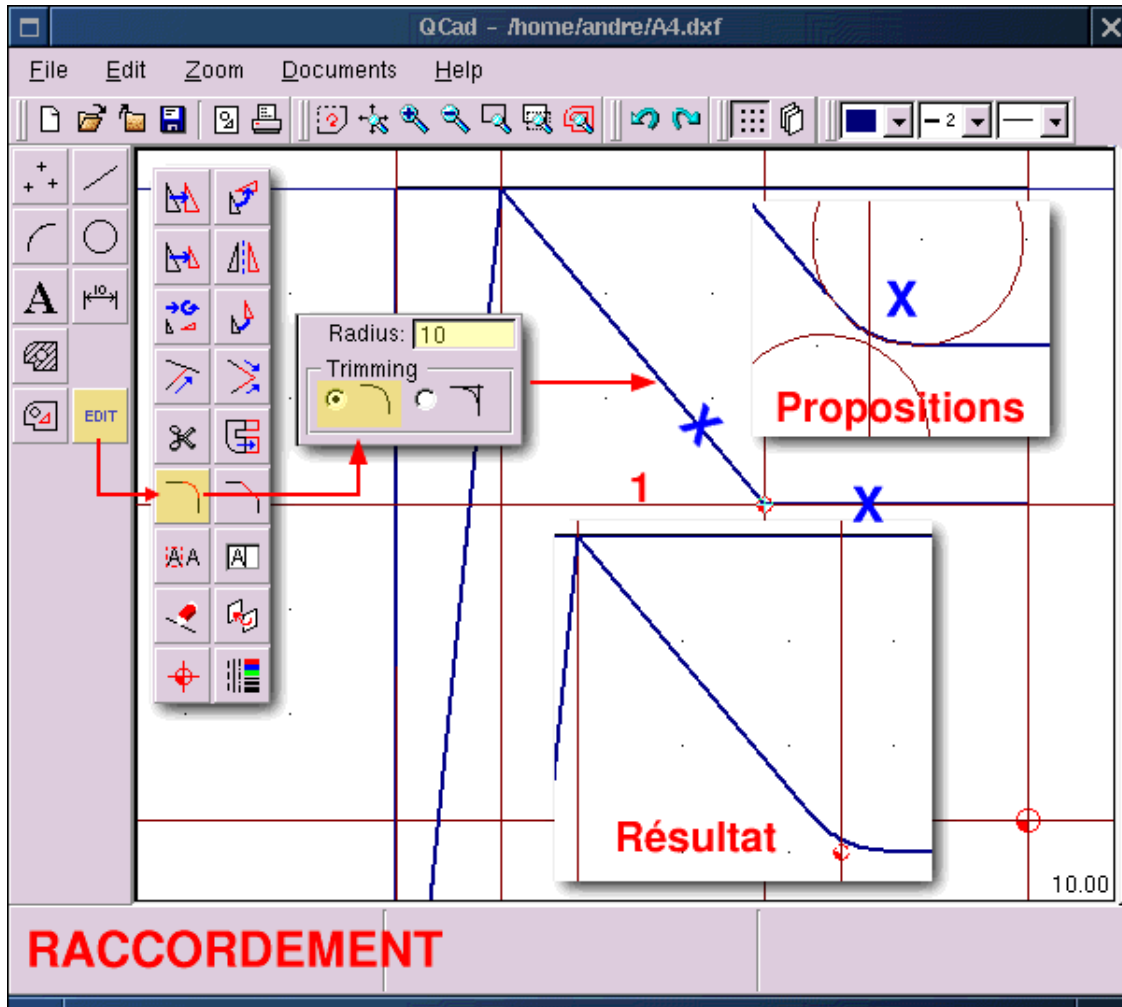
Editing is a modification of something existing. To add an adjustment or a beveling to a drawn profile or to delete a segment, that are modifications. Whatever the changes to make are there is a general approach. First you choose the function <Edit (1)> which opens a sub-menu of all possible

modifications. Select the desired function, e.g. <delete objects(2)>, which opens a sub-menu for selection: contour, all entities, tag single element... This allows you to choose the borders of the modification. If you choose for example <Tag single element(3)> design the element (4) and then acknowledge the action by a left mouse click on the arrow icon (5) then the chosen element is deleted. Please note that the function <(un-)tag single element;> is a toggle, if you click on an element then it is selected, another click and it is deselected. This allows to remove certain elements from a global selection.



## Adjusting the basis of the ear

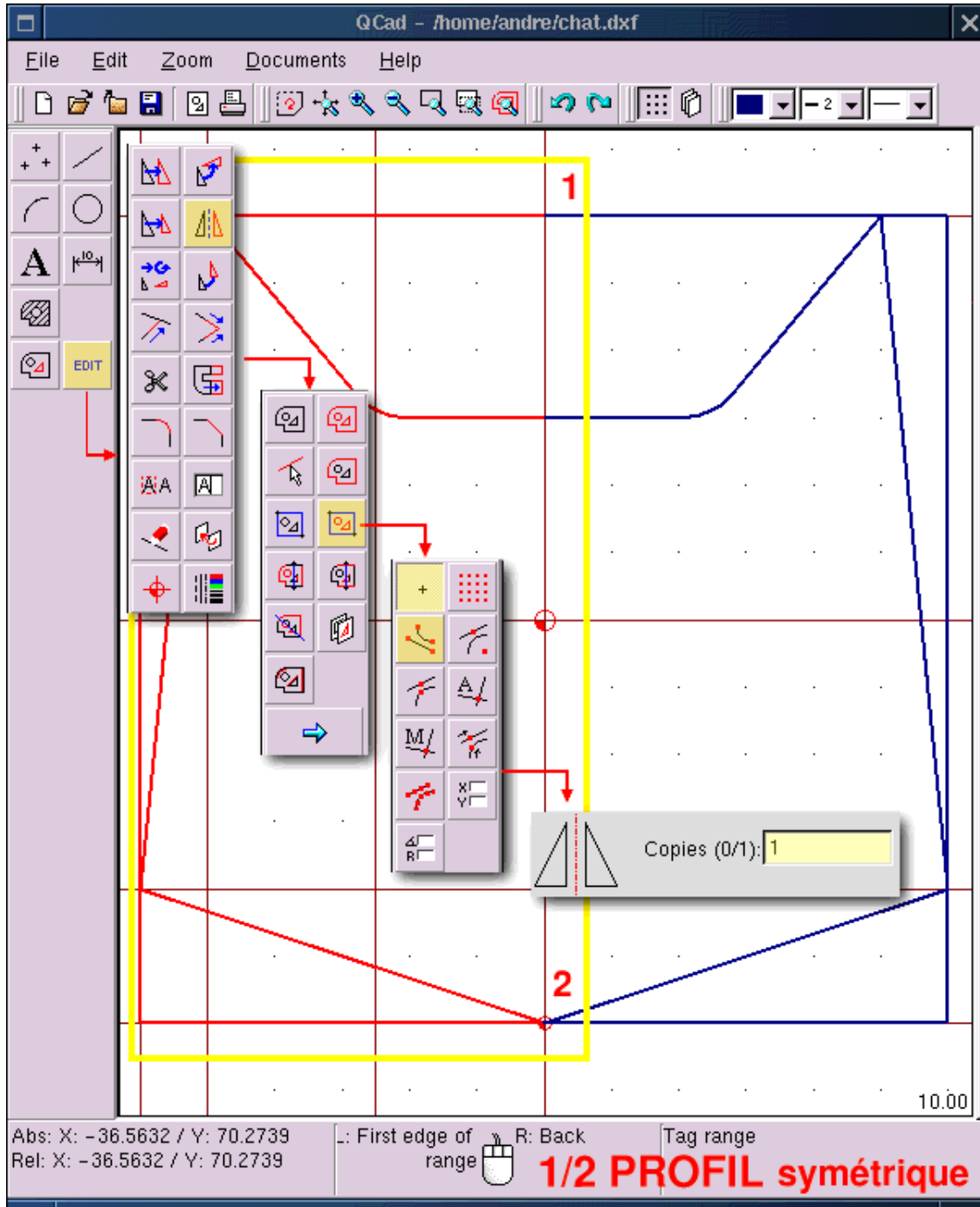
Going back to the main menu we delete the construction straight line called 1 in figure QCad12 (below) and choose <Edit><Round><Radius 10> <trimming> We determine the entities to trim, then we move the pointer near to the adjustment/link that has to be made. **Qcad** then suggests possible solutions (radius 10). If a fitting point is suggested a left mouse click keeps the linking/adjustment and trims the line.



## To construct the mirrored half of the profile

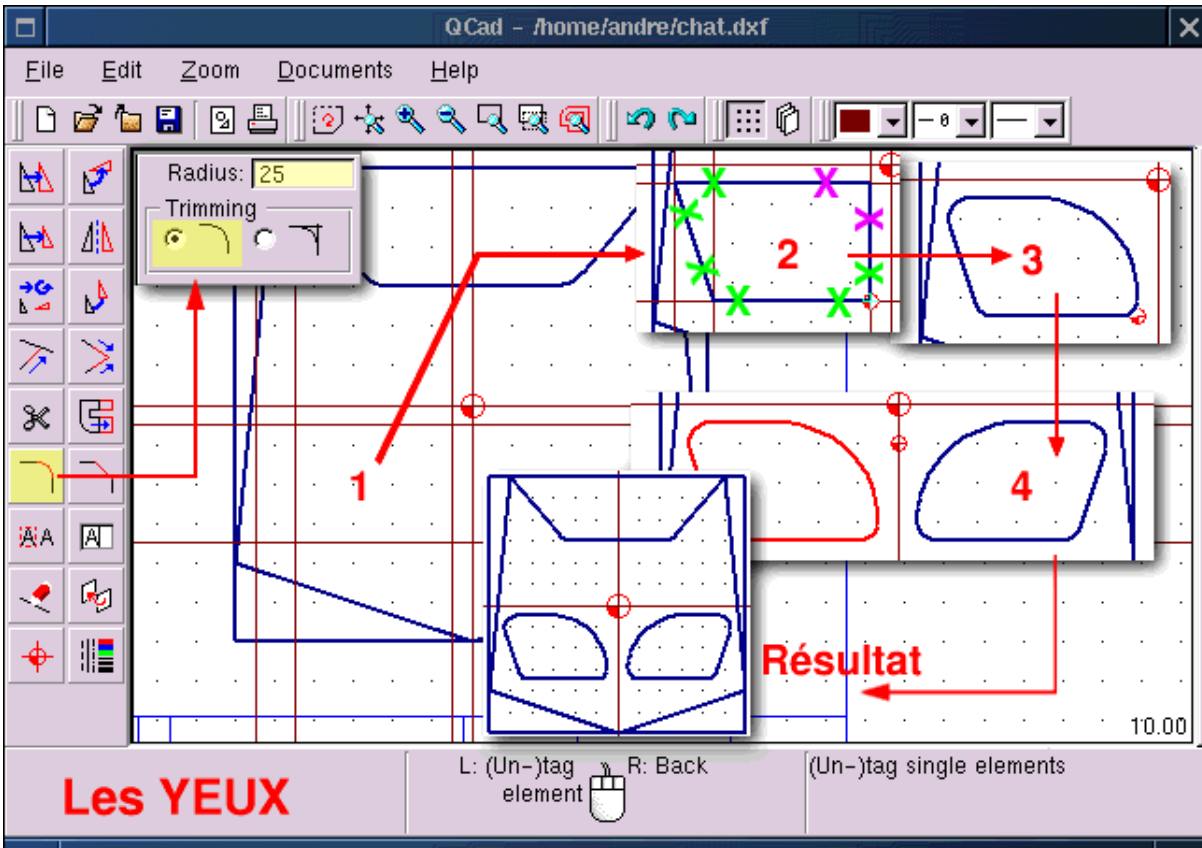
With the existing 1/2 profile it would be nonsense to draw another one, therefore it is sufficient to duplicate the first one symmetrically to obtain a complete profile. We choose <Edit><mirror objects><Tag Range><Point (Snap to nothing)> and draw a window around the 1/2 profile (yellow frame in figure QCad13). The profile is selected: it becomes red. We make a right mouse click: we get back to the selected sub-menu. We acknowledge by clicking on the arrow icon. We get back again to the

fixation sub-menu: we choose <Extremity (Snap to endpoints)> and determine the points 1 and 2 as shown in figure QCad13. A dialog box "Mirror" appears. If you type in a value of 0, the 1/2 profile is moved, if you give in a value of 1 it is duplicated. Therefore you have to give in a value of 1 and click <Ok>. And the profile is ready.



## Drawing the eyes

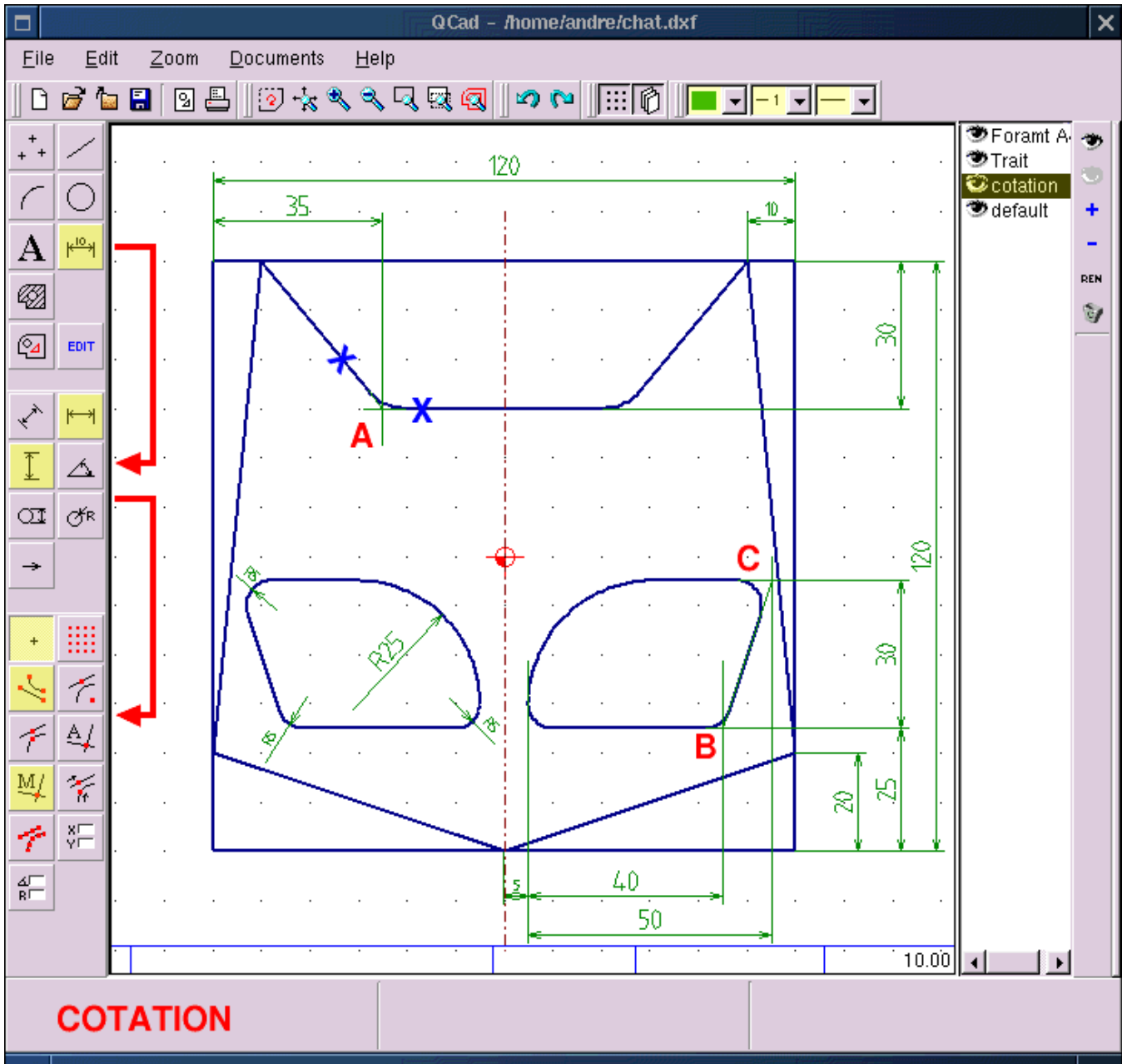
By using what we have seen already we can draw the left eye as well through the horizontal and vertical construction lines (point 1). Then you draw the profile by building upon this straight line with a polyline. You trim it with a radius 5 (green crosses) and a radius 25 (magenta cross, points 2 and 3); finally you delete the construction line and duplicate the left eye by mirroring it to the right (point 4). All necessary commands for this operation have already been explained above.



## Adding dimensions

Annotations are no strength of QCad: it is impossible to give tolerances or to write somewhere else than in the middle of the lines that indicate the dimensions. The consequence of this last point is that the size of the characters is changed depending on the available space between two reference points. This give the whole drawing a strange aspect. Well, no matter how it is, to make annotations you have: to position yourself on the annotation layer, choose a fitting line attribute, especially a thickness of 1 and a color different from the other lines, unique if possible. But this isn't obligatory. Select <Annotation (Sub-menu dimensions)><Type of annotation horizontal or vertical or radius...><Endpoints to determine the position of the construction lines or certain intersection (Snap manually to..)>< Point (Snap to nothing) for positioning the dimension> To change from one way of fixation to another you may use the short cut keys: F for <point>, E for <Extremity>, X for < automatic intersections> etc. The points A, B

and C (image below) are difficult to annotate with a dimension.

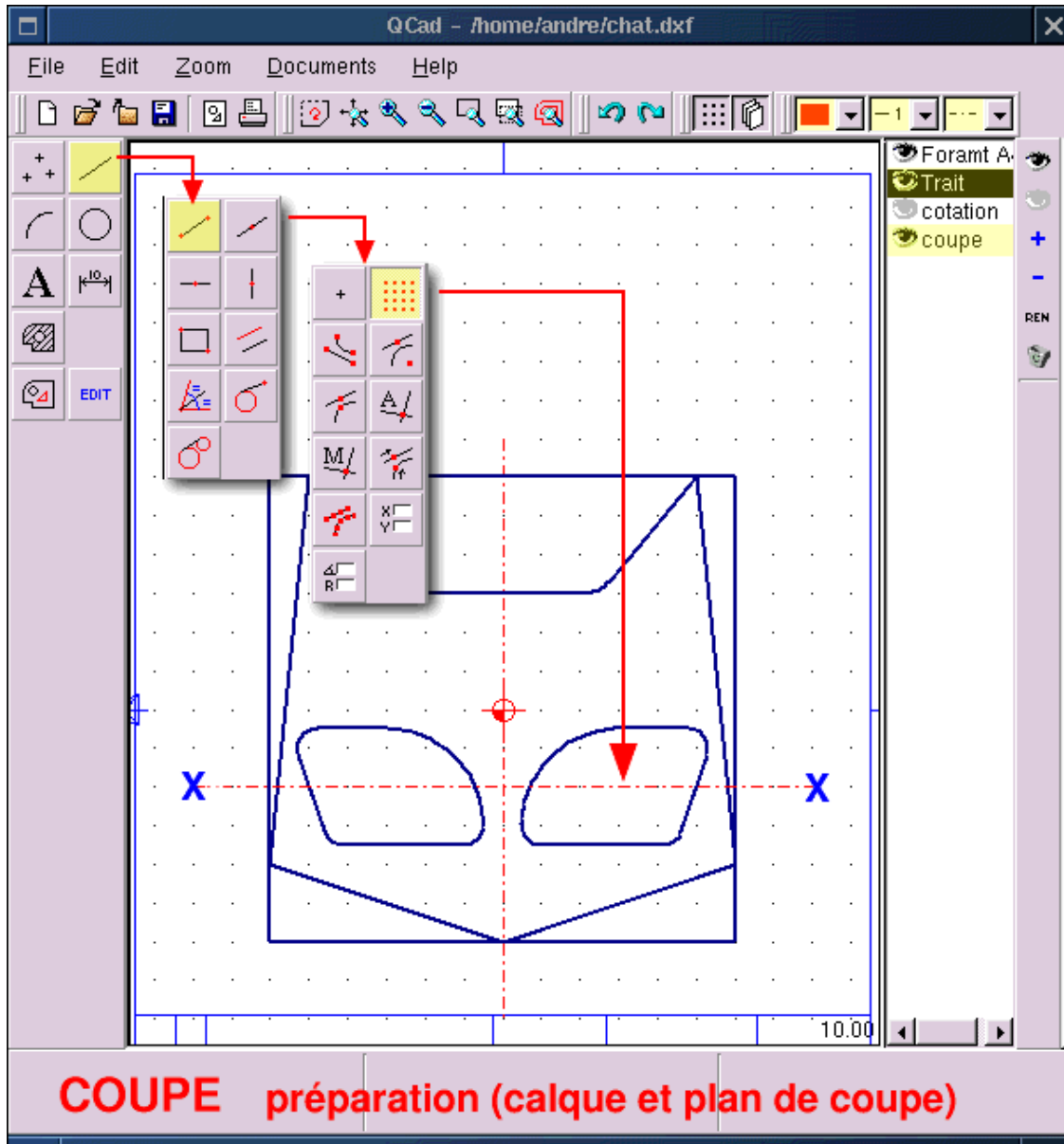


### A sectional view (cut): preparation

The representation of a three dimensional object in 2D makes it necessary to order several views according to certain drawing norms even if it only is to show the thickness of the object. Our drawing represents an object of 20mm thickness worked on in a depth of 5mm. To just say this isn't explicitly enough and a cross sectional view becomes necessary. To do this: Make the layer <Annotation>

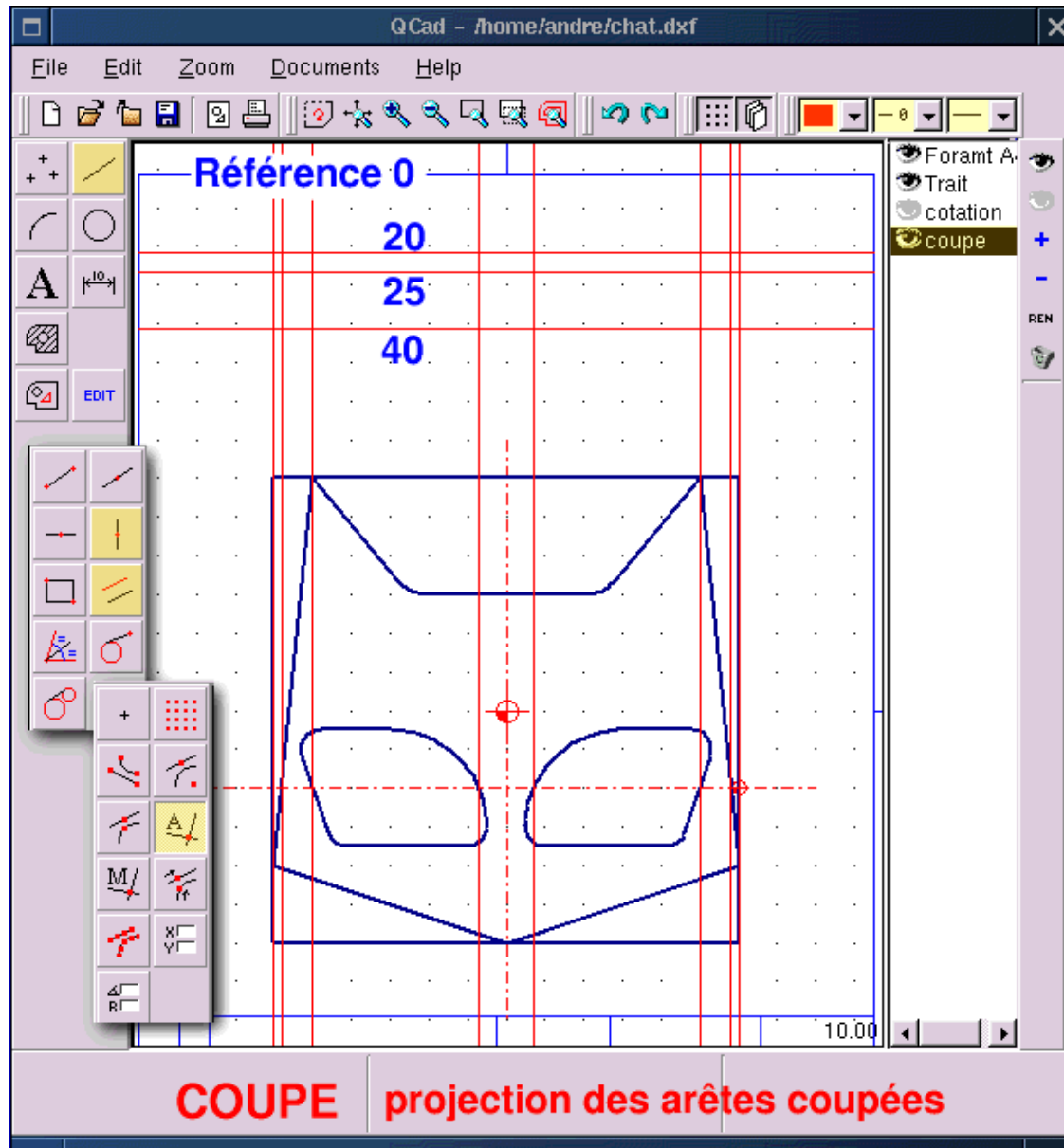


invisible with a double click on the icon with the open eye. Add a layer <cross section> with the option + in the management menu for the layer. According to the norm a cross sectional cut is indicated by a line with a points and hyphens. Activate this as a style attribute and draw a line of width 1 between the eyes of our logo (<straight line><Polyline><snap to grid points>).



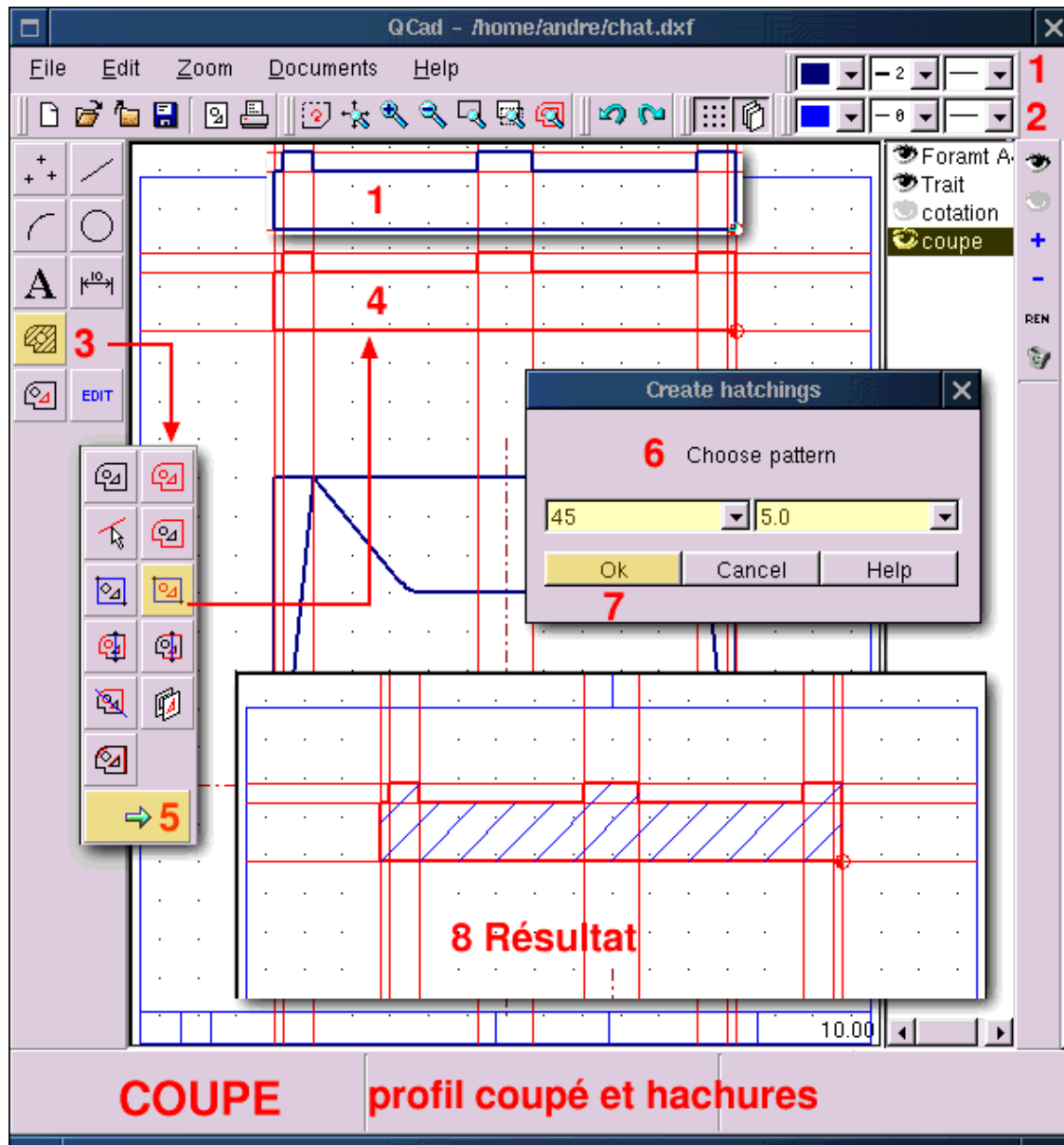
### Sketch the cross sectional view

In industrial drawing whether with pencil and paper or with CAD there must always be a correspondence between the views. The cross sectional view is a projection along the line that indicates the cross sectional cut through the object.



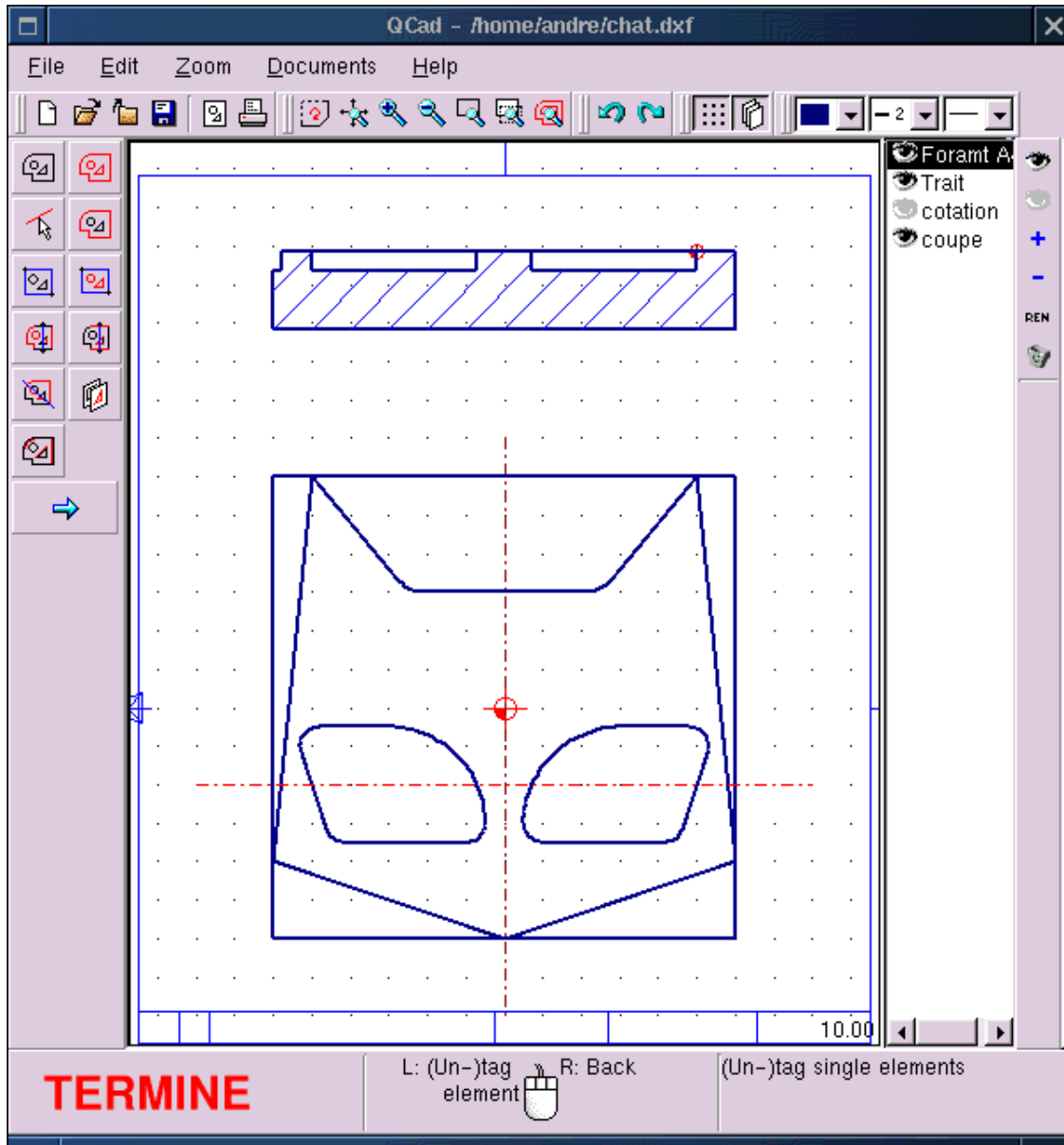
## Drawing and hatching of the cross sectional view

We use again straight construction lines to draw the cross sectional profile with a polyline (figure QCad18, below). Modify the properties of the lines for the hatch (2). Select <Create hatchings(3)> <Tag range> <Passing through the Point (Snap to nothing)> and draw a square around the cross sectional view to select the area (4). Acknowledge (5). The dialog box "Create hatchings" appears. We choose a fitting hatch parameter(6). We press OK (7) and the hatched view (8) is ready.



## And finally the finish

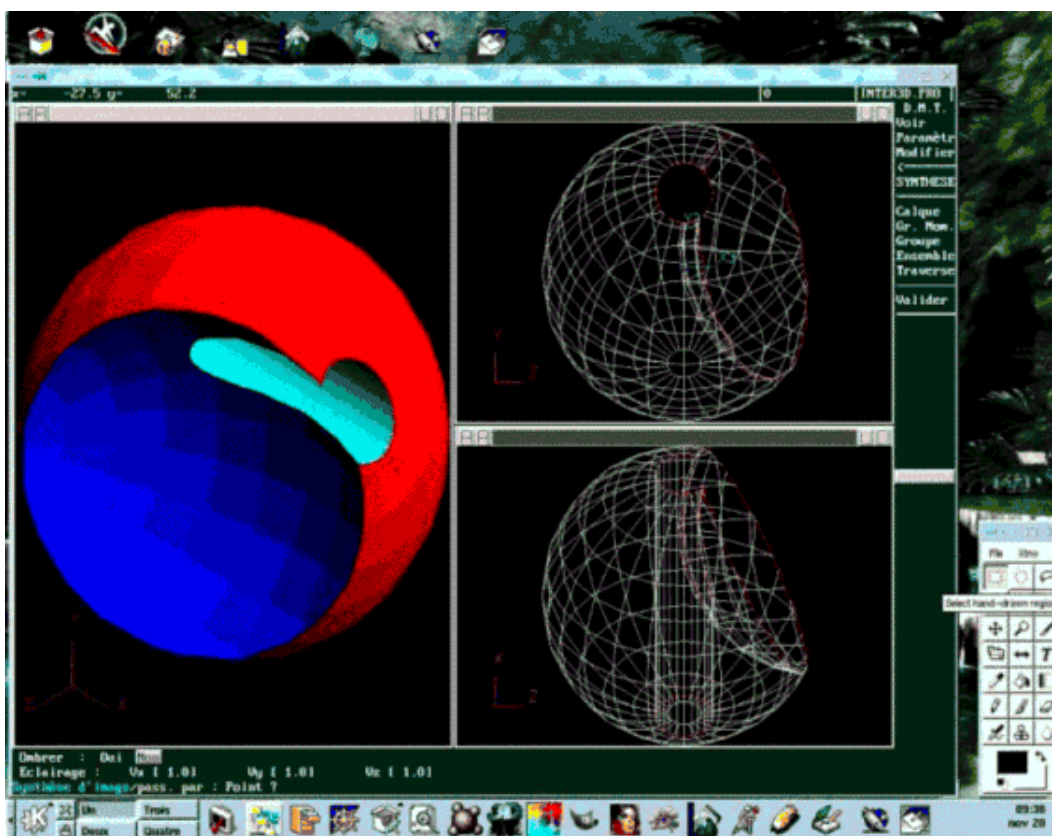
The cross sectional view as it is represented here is, according to the rules of industrial drawing, is a projection. Since obtain this view by virtually cutting our profile at the height of the eyes we have to indicate the depth of the eyes. You add this as follows <Straight polyline line (button line)><create lines> Now the drawing is ready. Only the frame of our paper (the data block) has to be filled in with text. I leave it to you to discover the < text function>.



## Conclusion

Linux CAD-Applications under the GPL aren't numerous. It is therefore appropriate to honor the authors of **Qcad** and to thank them for the useful application they give to the community. Even though it isn't replacing industrial applications of the type of Cadkey, AutoCad or DMT, it remains a good educational tool and a tool for not too complex plans. One can regret the weaknesses of the annotation function, the absence of covering/lining/boarding functions (rowness, geometrical tolerances, sectional views) and the numeric limitation of the geometric border conditions. But you can congratulate the authors for the ease of handling, for the simple and convenient user interface, the powerfulness of

linking/fitting and hatch functions, to the choice of the DXF format and not a proprietary format. Possibly as well to many other good things in **QCAD** which I have not yet discovered. **QCAD** has a help system but the documentation is English and remains therefore totally obscure to me. This proves how easy the handling of **QCAD** is. I have discovered everything by just playing around with the program. Having said that, the optics of CAD have changed dramatically in the recent years. It is less a question to produce 2D drawings to represent three-dimensional objects. But to work out a 3D model completely defined in form and dimension with the help of performant tools and volume modelers. The program generates then automatically plans, annotations and the listing for numerically controlled machines. These programs are ProEngineer, SolidConcept, Catia, Solid Edge or Think3D.... When will these tools be available to Linux? At the moment we have **QCAD** and **CAM Expert**, its commercial brother, equipped with two dimensional CAD and old MS-DOS products such as DMT10 by Mécsoft which runs perfectly in dos-emulation under Linux as you can see in the last screen shot (below).



Webpages maintained by the LinuxFocus Editor team  
 © André Pascual  
 "some rights reserved" see [linuxfocus.org/license/](http://www.LinuxFocus.org/license/)  
<http://www.LinuxFocus.org>

Translation information:

fr --> -- : André Pascual <[apascual/at/club-internet.fr](mailto:apascual/at/club-internet.fr)>  
 fr --> de: Bernhard Spanyol <[bspa/at/gmx.de](mailto:bspa/at/gmx.de)>  
 de --> en: Katja Socher <[katja/at/linuxfocus.org](mailto:katja/at/linuxfocus.org)>