

# **QCad2X**

André PASCUAL [andre.pascual@linuxgraphic.org](mailto:andre.pascual@linuxgraphic.org)

# Table des Matières

<b>Qcad2X : faisons le point par l'exemple.....</b>	<b>1</b>
1 Se procurer le programme:.....	1
2 Conditions du test:.....	2
3 Rappels indispensables:.....	3
4 Mise en route:.....	4
5 Création du format:.....	6
6 Changer d'attributs:.....	9
7 Le vif du sujet:.....	10
8 Conclusion:.....	21

# QCad2X : faisons le point par l'exemple

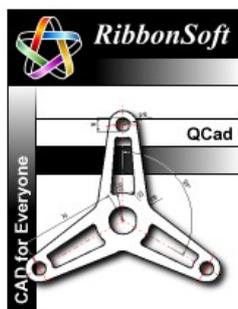
Quelques années plus tôt, **QCad 1.3.3** avait été présenté dans les pages de **Linux Magazine France**. La réalisation d'un *dessin technique* servait alors l'argumentation qui permet de conclure à la validité du programme testé, pourtant encore en cours de développement. Toutefois, les inévitables faiblesses dues à sa nouveauté demandaient à être gommées. Qu'en est-il aujourd'hui de **QCad 2.x**?

## 1 Se procurer le programme:

Il est à peu près certain que la plupart des distributions actuelles dispose d'un paquetage **QCad 1.5x** propre à une installation correcte et rapide. Mais pourquoi pas de version **2**?

C'est que depuis le passage au numéro **2**, **Andrew Mustun**, programmeur suisse auteur du projet, a décidé de changer quelque peu le mode de distribution. Les exécutables prêts à l'emploi, disponibles pour **Linux, AIX, BSD, HP-UX, Solaris, Mac OS** et **Windows**, ne sont pas gratuits. En langage courant, on peut dire que leur utilisation est soumise à l'acquisition d'un droit d'utilisation. Cependant, **QCad** est toujours un programme libre et gratuit, placé sous licence **GNU GPL**, lorsqu'il est livré sous forme de sources à compiler soi-même, sauf pour **Windows** qui reste, selon **Andrew**, une plate-forme inamicale (litote!) pour le logiciel libre. Aussi, la version **Windows** de **QCad**, qualifiée de programme propriétaire par son auteur, n'est-elle livrée que sous forme de *démo* exécutable à essayer (*Trial version*), dont il conviendra d'acquitter les droits.

Il en est d'ailleurs de même pour toutes les versions pré-compilées, dites *démos*, qui sont donc des produits commerciaux.



## NOTA:

**QCad** existe sous plusieurs formes: *Community* (sources), *Professionnelle* et *Démo*. Techniquement, il s'agit de produits identiques, disposant pleinement des mêmes fonctionnalités. La version *Professionnelle* propose un support de la part de l'auteur et, dans l'avenir, recevra des modules additionnels (info. valable au moment de l'écriture de ce texte, mais susceptible de modifications); la version *Démo* soumise à paiement de redevance s'interrompt après 10 minutes d'utilisation et nécessite un redémarrage, mais le travail effectué est sauvegardé; la version *Community* à compiler chez soi n'est aucunement limitée, mais nécessite de posséder **QT Developer**.

Certains ont pu critiquer un tel mode de distribution. Mais il est tout à fait normal qu'un programmeur cherche à tirer profit de son travail. Toute peine ne mérite-t-elle pas salaire? Et il n'y a pas incompatibilité entre **GPL** et gagner sa vie, si?

On se rendra donc sur <<http://www.ribbonsoft.com/qcad.html>> pour remplir son panier. Un paquetage **QCad** pèse entre 9 et 10 Mo, selon le système d'exploitation choisi. Il faudra compter 2 Mo supplémentaires si l'on veut télécharger les trois douzaines de fontes disponibles (d'un intérêt limité en

dessin technique: il y a même une fonte fracture, dite gothique!). Et si l'on est un adepte des symboles à insérer dans son travail, il en coûtera 12 Mo de plus pour 4500 objets, appelés *parts*, qui couvrent des besoins aussi divers que la mécanique, l'architecture, l'électronique, l'hydraulique... Attention toutefois: la prise en charge des symboles et des fontes s'effectue au lancement de **Qcad**, ce qui en ralentit l'ouverture.

L'installation d'un *rpm* sous **Linux** ne présentant aucune difficulté à partir de **Konqueror**, par exemple, il n'en sera pas fait mention. Prière de se reporter au précédents numéros de **Linux Pratique** dont plusieurs articles décrivent la manière idoine de procéder.

## 2 Conditions du test:

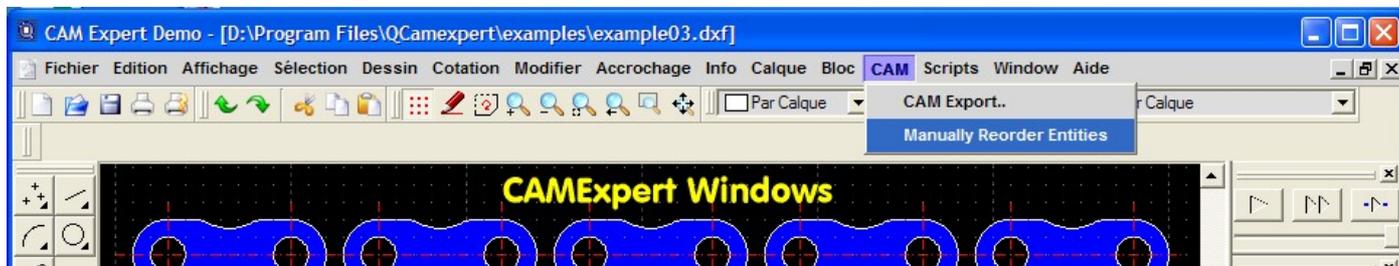
**Machine de test** : ordinateur portable Medion 7438, AMD 2200, 512 Mo de Ram, vidéo n'Vidia Geforce4 avec 64 Mo de ram dédiées.

**Système d'exploitation** : **Mandrake Linux 9.2**, avec mise à jour de **KDE** (3.2.1) et de **QT** (3.2.3) sur une partition, et **Windows XP Pro** sur une autre.

**Programmes** : **Qcad** 2.0.1.3 Professionnel sous **Linux**, **Qcad-demo** 2.0.3 et **CamExpert-demo** 2.0.3 sous **Windows**. Ces versions **Windows** n'ont eu pour but que de mettre en évidence similitudes et différences, si nécessaire, entre les produits destinés aux deux plates-formes.

### NOTA:

**CamExpert** n'est rien d'autre que **Qcad**, enrichi d'un module qui s'y intègre de manière tout à fait transparente. Sa présence n'est révélée que par une entrée supplémentaire dans la barre des menus standards: **CAM** (Computer Aided Manufacturing, qu'en français on appelle **FAO**, Fabrication Assistée par Ordinateur). Il s'agit, par cette extension logicielle, de générer du code de Commande Numérique (**G-Code**) à partir d'un profil dessiné ou récupéré en **DXF**, et de le transférer vers le **DCNC** de la machine d'usinage. Application métier commerciale, dont il ne sera pas question par la suite. (Voir Fig: "CAMExpert Windows")



## 3 Rappels indispensables:

### 3.1 Normes de dessin:

Comme par le passé, il semble logique de proposer la réalisation d'un dessin technique (familièrement: un *plan*) sous forme de pas à pas, afin de voir si **QCAD2.x** en autorise une confection aisée. Cependant, ce genre d'exercice se doit d'obéir à des règles normalisées.

Dans un système de *projections européennes*, si la *vue de face* est la vue de référence, la *vue de dessus* se situe en dessous, la *vue de dessous* se situe en dessus, la *vue de droite* est placée à gauche, et la *vue de gauche* à droite. Il n'y paraît pas, mais ceci est d'une logique rigoureuse (Voir le **Glossaire Illustré de l'Infographie**). De plus, il doit obligatoirement y avoir correspondance entre ces vues.

Selon la même normalisation, le type des traits est représentatif du type d'arête tracée.

Une arête **vue** est dessinée en *trait fort* (ou *plein*), d'épaisseur 0.6 à 0.8. Une arête **cachée** est représentée en *trait interrompu court* (familièrement: *pointillé*), d'épaisseur 0.3 ou 0.4. Les limites de matière, en *lignes brisées fines* de 0.3 ou 0.4. Les axes, les traits de coupe, les limites de déplacement... en *mixte fin*. Le hachurage et les arêtes **fictives** en *trait fin*.

Méconnaître ces conventions conduit à une lecture erronée du plan.

### 3.2 Logique DAO:

Les programmes de **DAO** (Dessin Assisté par Ordinateur, **CAD** en anglais) obéissent aux mêmes règles de construction d'une *entité* (Segment, cercle, arc...), que l'on peut résumer ainsi:

**Entité dessinée = Type + Contrainte + accrochage.**

**Type** = point, cercle, arc, segment, droite etc.,

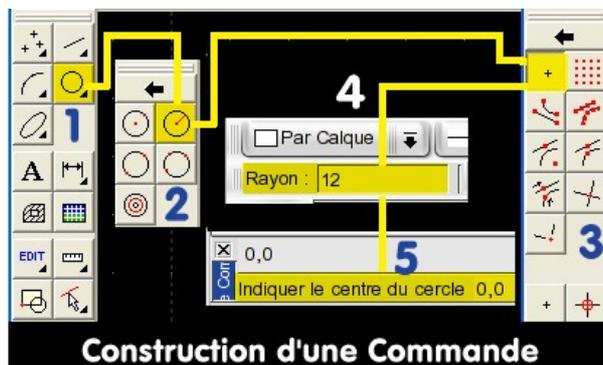
**Contrainte** = dimension, tangent, vertical, horizontal, perpendiculaire, parallèle...

**Accrochage** = coordonnées, intersection, extrémité, centre...

La méthode peut sembler rigide: elle est seulement rigoureuse et garante d'une géométrie correcte. Il n'y a pas d'à-peu-près en **DAO**. De la sorte, pour dessiner un cercle de rayon 12 mm, dont le centre se situe dans l'espace de dessin à X0, Y0, il faudra procéder ainsi dans **QCAD**:

- Pointer *Cercle* (1) dans la palette principale.
- Pointer *Cercle avec Centre et rayon* (2) dans la palette *Contraintes* qui vient d'apparaître.
- Pointer *Positionnement libre* (3) dans la palette *Accrochage*.
- Indiquer la valeur du rayon dans le *champ de saisie* en haut de l'espace de travail (4).
- Cliquer dans le *champ de saisie* de la commande en attente au bas de l'espace de travail et entrer les coordonnées (5).

Le cercle est alors dessiné. (Voir Fig: Construction d'une Commande).



**NOTA:**

La commande étant *modale*, c'est-à-dire: active tant qu'elle n'a pas été remplacée par une autre, il suffit de changer la valeur du rayon et les coordonnées du centre pour dessiner d'autres cercles, de dimensions et de positions différentes.

On remonte vers la *Palette Principale* en cliquant sur la flèche de retour arrière, ou par une succession de clics droits ou encore par la touche *Escape*. Une fonction se désactive par un clic droit.

**Très important** : toutes les commandes peuvent être passées via la barre de menus, où les fonctions exprimées littéralement sont compréhensibles sans ambiguïté, au contraire des icônes. On y aura souvent recours dans la suite de cet article.

**4 Mise en route:**

Au premier lancement, **QCAD** demande de configurer trois éléments de l'interface:

- système de mesure.
- langue de l'interface.
- langue des commandes.

On choisira *millimeter*, *french* et *french*.

Il sera toujours loisible d'y revenir par:

>Menu *Editions*> *Préférences générales*.

(Voir Fig: 1° Démarrage).





Lorsqu'il s'agit d'un *plan d'ensemble*, ce qui n'est pas le cas ici, on optera également pour un calque par pièce de l'ensemble.

Remarques :

Il existe une vieille habitude provenant d' **AutoCad** et qui se justifiait par l'utilisation, pour les sorties sur papier, d'un traceur à plumes d'épaisseur et de couleur différentes. Ainsi, un dessin d'**AutoCad** était composé d'un calque par type de trait, paramétré chacun pour une couleur et une épaisseur donnée correspondant à une plume du traceur.

**Qcad** semble avoir repris cette logique; en effet lorsqu'on charge en bibliothèque un format existant par: >Affichage >Vues >Explorateur de bibliothèques >Templates >Fr , et qu'on l'insère dans sa session de travail, celui-ci arrive nanti de plusieurs calques.

De nos jours, et pour un usage courant, avec des plans n'excédant pas le A3, la sortie papier s'effectue sur une imprimante à jet d'encre, le plus souvent en noir et blanc. La structure en *calque/type de trait* ne semble pas utile. De ce fait, nous allons créer notre *Format*, en nous affranchissant de la logique **AutoCad**.

## 5 Création du format:

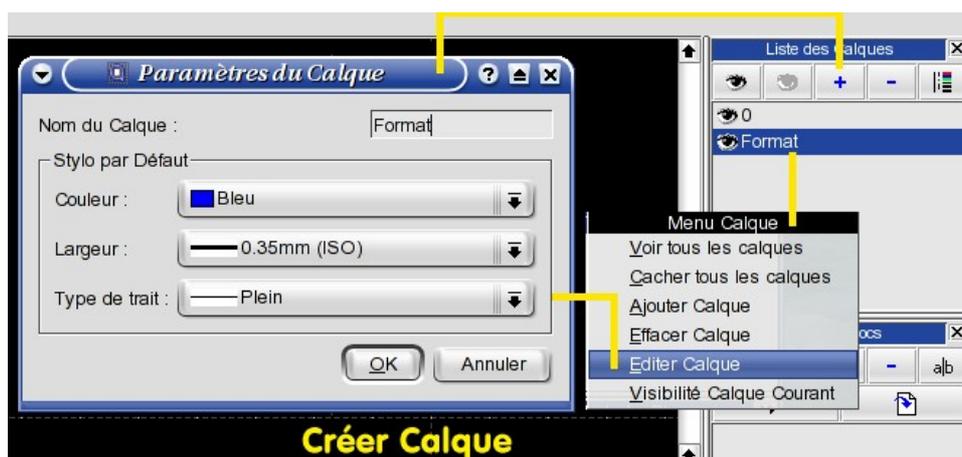
On entend par *format*, la délimitation de la feuille de dimensions normalisées (210x297 pour de l'A4), contenant un cadre en retrait de 5mm et contenant le *cartouche* destiné à recevoir la "carte d'identité du plan".

Pour ce faire:

- créer un nouveau calque en cliquant sur le "+" de la fenêtre *Liste des Calques*,
- remplacer l'appellation *noname* par *Format*,
- choisir une couleur de trait *Bleu* et une largeur de trait normalisée de 0.35 (iso).
- Valider par *OK*,

... et le calque *Format* apparaît dans la liste. On pourra toujours modifier ses attributs par un clic droit sur son nom et en choisissant >*Editer Calque*.

(Voir Fig: Créer Calque).



Nous sommes maintenant prêts à dessiner. Allons-y.

Sélectionner:

>Menu Dessin >Ligne > Rectangle.

>Indiquer premier coin> 0,0 [Entrée]

>Indiquer deuxième coin> @210,297 [Entrée] (L'@ signifie coordonnées relatives)

Un rectangle bleu d'épaisseur de trait 0.35 est dessiné.

Ensuite, pour le cadre intérieur, nous allons appeler la puissante fonction de tracé de segments (*Lignes*, dans **Qcad**) parallèles. Puissante parce que le survol par le pointeur d'un segment identifié comme segment de référence, propose de tracer un segment de longueur identique à la référence, et positionné d'un côté ou de l'autre de la référence, selon la position du pointeur. Il suffit alors de cliquer. La fonction étant *modale*, on désignera alors des références différentes pour obtenir des segments distants de la même valeur que précédemment, sauf si, entre temps, on modifie la valeur d'écartement.

>Menu Dessin >Ligne >Parallèle.

>Distance: 5

> Désigner la référence, se placer du bon côté, [ Cliquer ].

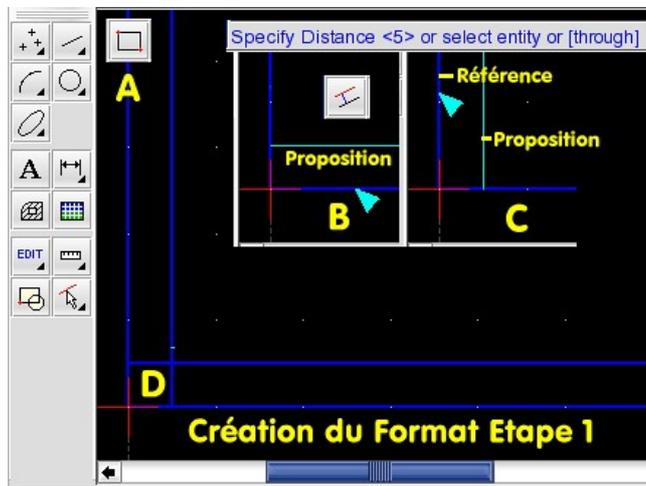
Procéder ainsi en prenant pour référence tour à tour les quatre côtés du premier rectangle.

On obtient ce que représente la figure: "Création du Format Etape 1".

– En A : premier cadre.

– En B et C, survol de référence et proposition de tracé de segments parallèles.

– En D, résultat de l'opération (Seul le coin à 0,0 est représenté).



Les délimitations de la feuille sont généralement indiquées par des "coins". On les obtiendra par découpage du premier rectangle au niveau des intersections de ses côtés avec les segments parallèles précédents.

Suivre sur la figure: "Création du Format Etape2"

>Menu Modifier > Couper> Sélectionner l'objet à couper [ le désigner d'un clic ].

> Indiquer le point de coupure > Menu Accrochage > Intersection.

Le survol du dessin propose alors les points de coupure possibles. [Cliquer ] sur le point choisi (1 sur la Figure). Le système coupe l'entité et demande de lui désigner un nouvel objet à couper. Lui faire plaisir. Continuer de la sorte autant que nécessaire, puis effacer les entités à supprimer.

Pour cela:

> *Menu Modifier* > *Supprimer* > [ Désigner les entités ] ; les segments à supprimer changent de couleur. Les supprimer avec la touche [ Suppr ] ou [ Del ], selon le clavier (2 et 3 sur la Figure), ou encore avec l'icône *Continue Action*.

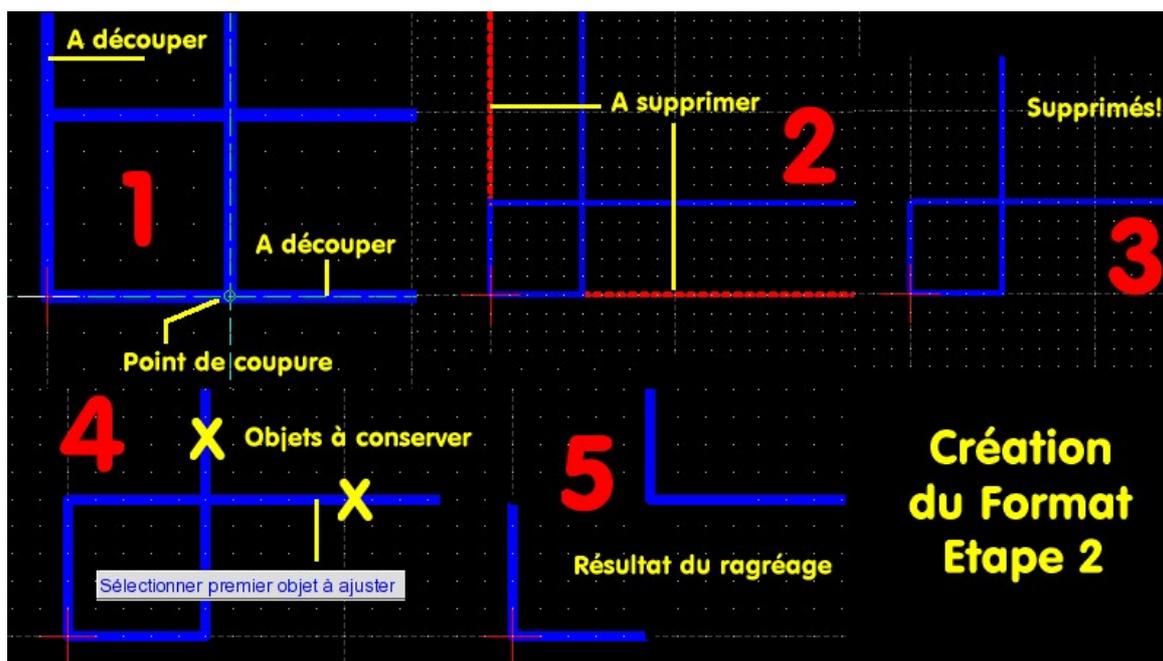
A l'issue de cette manipulation, les coins du format forment des carrés. Il faut *ragréer* (usuellement : *Trim* en anglais) les coins des segments parallèles.

### Attention :

dans le menu littéral, la fonction s'appelle *Prolonger Deux*, et dans la palette d'icônes, elle s'appelle *Ajuster/Étirer Deux*.

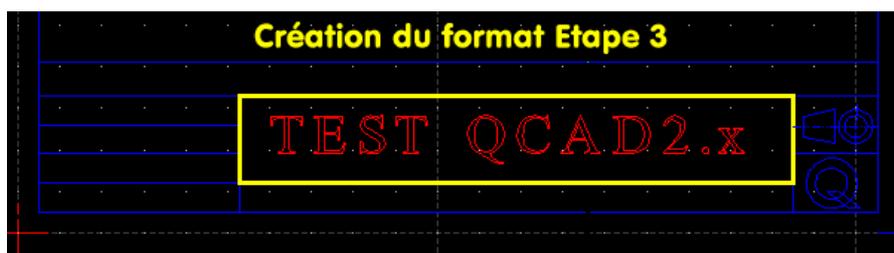
Donc:

> *Menu Modifier* > *Prolonger Deux* > [ Désigner les segments que l'on veut conserver]. L'excédent est immédiatement supprimé (4 et 5 sur la Figure).



Un format de dessin comporte un *cartouche*, zone réservée aux informations relatives au plan. La représentation de cette zone n'est pas normalisée quant à sa forme, mais quant à sa position: oui. Pour cet exercice, chacun dessinera le cartouche qui lui convient, avec les fonctions vues précédemment. Le modèle de la figure "Sujet de l'exercice" peut parfaitement convenir. Ici, nous ferons l'impasse sur le sujet, en nous contentant de celui montré par la figure: "Création du format Etape 3". Il a été construit avec la fonction segments parallèles qui ont été ensuite découpés et nettoyés. Quelques éléments, comme le symbole de *projections européennes* et une lettre "Q" fantaisiste ont été ajoutés. Le rectangle central d'épaisseur et de couleur différentes a été obtenu par une *modification d'attributs*.

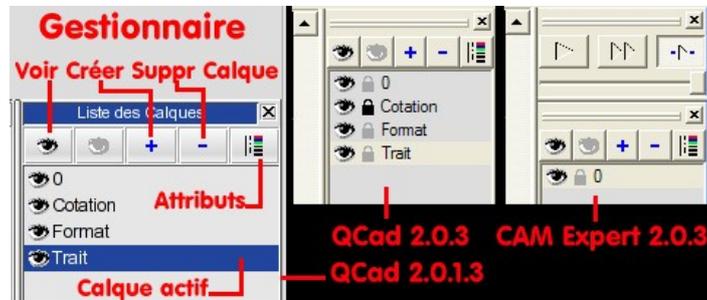
Qu'est-ce donc?





## 7 Le vif du sujet:

Le dessin proprement dit doit s'effectuer sur un autre calque, que l'on appellera *Trait*, par exemple, dont on conservera les *attributs* par défaut. Le dimensionnement du dessin (*cotation*) s'inscrira sur un calque... *Cotation*, dont on fixera la couleur sur Cyan et la largeur de trait sur 0.25. Dans la partie droite de **QCad**, le *Gestionnaire des Calques* devrait ressembler à la Figure: "Gestionnaire". On notera que dans la version 2.0.1.3 **Linux** testée, la possibilité de verrouiller les calques n'existe pas, alors qu'elle est présente dans la version 2.0.3 **Windows** (icône *cadenas*).



Habituellement, un *dessin technique* tel celui de l'exercice proposé, se commence par la mise en place des axes, puis des cercles, ensuite des segments tangents, enfin des raccords et chanfreins, et, pour terminer, de l'habillage constitué par les *hachures*, la *cotation* et autres informations. Ceci n'est cependant pas d'une rigidité absolue.

Le calque *Trait* étant actif (il suffit de le sélectionner dans le gestionnaire), faire pivoter l'ensemble de 90° pour se trouver dans la position du modèle "Sujet de l'exercice."

Pour cela, dans la palette d'icônes (ne pas négliger les *infobulles*, ni, surtout, les indications de la *ligne de commandes*):

- > *Edit* > *Rotation* > *Tout sélectionner* [l'ensemble des entités change de couleur]
  - > *Continuer Action* > *Spécifier la référence*: 0,0 [Entrée]
  - > Dans la boîte *Options de Rotation*, cocher *Supprimer Original*, entrer *Angle*: 90 > *OK*.
- L'ensemble a pivoté.

### 7.1 Tracé des axes:

#### NOTA:

Ci-dessous, les chiffres entre parenthèses () désigne l'entité créée suite à la commande décrite.

- > *Menu Dessin* > *Ligne* > *Parallèle* > *Distance*: 75 du bord inférieur du cadre (1).
- > *Distance*: 120 du bord gauche du cadre (2).
- > *Distance*: 80 de la ligne 2 (3).
- > *Distance*: 60 de la ligne 2 (4).
- > *Menu Dessin* > *Ligne* > *Angle* > Entrer *Angle*: [-10], *Longueur*: [100], *Accrochage*: [Début].
- > *Menu Accrochage* > *Intersection* de 1 et 2 (5)

A ce stade, rendre invisible le calque *Format*, en cliquant sur l'icône *oeil* en face de son nom dans le *Gestionnaire*.

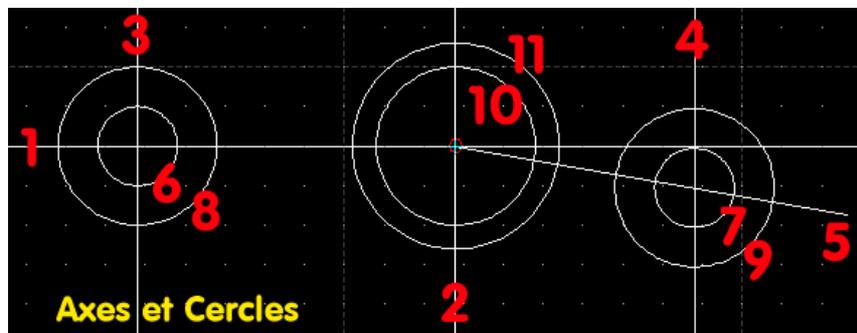
## 7.2 Tracé des cercles:

### NOTA:

La commande *Accrochage > Intersection* étant modale, et étant actuellement active, il n'est pas nécessaire de l'appeler dans les opérations qui suivent. Il suffit d'approcher le pointeur de l'intersection voulue pour que **QCad** la propose. Les intersections sont indiquées ici dessous pour préciser où se situe le centre des cercles à construire.

- > *Menu Dessin > Cercle > Centre, Rayon [10] > Intersection de 1 et 3 (6)* , puis de 4 et 5 (7).
- > *Rayon [17] > Intersection de 1 et 3 (8)*, puis de 4 et 5 (9).
- > *Rayon [20] > Intersection de 1 et 2 (10)*.
- > *Rayon [26] > Intersection de 1 et 2 (11)*.

Le travail devrait ressembler à la figure: "Axes et Cercles".



### 7.3 Ebauche du contour extérieur:

La commande *>Segment (Ligne)d'Angle tangent à >* n'existant hélas pas, il va falloir construire des directrices angulaires, puis des segments parallèles à ces directrices pour dessiner les contours obliques tangents aux cercles.

– Construire la Directrice 1 (D1):

> *Menu Dessin > Ligne > Angle: [30], Longueur: [50], Point d'accrochage: [Début] > Accrochage > Centre*  
 Désigner le centre de (6)

– Construire la Directrice 2 (D2):

> *Angle [-35] , Point d'accrochage: [Fin] >* Désigner le centre de (7)

– Construire la Directrice 3 (D3):

> *Angle: [-20] >* Désigner le centre de (10)

– Construire les segments du contour (12 à 18):

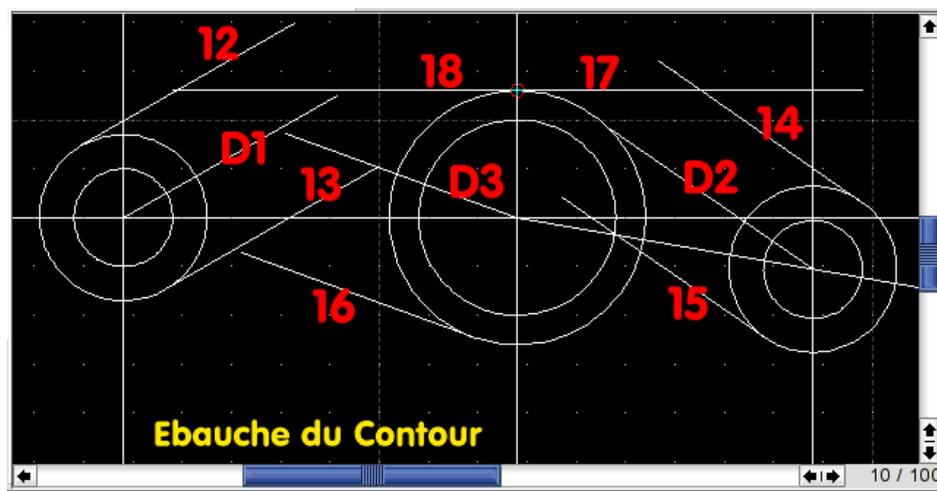
> *Menu Dessin > Ligne > Parallèle > Distance: [17],* et poser tour à tour les segments (12), (13), (14) et (15).

> *Distance: [26],* et poser le segment (16)

> *Menu Dessin > Ligne > Horizontale > Longueur: [70], Point d'accrochage: [Début] > Accrochage > Intersection* de (11) et (2), et poser le segment (17)

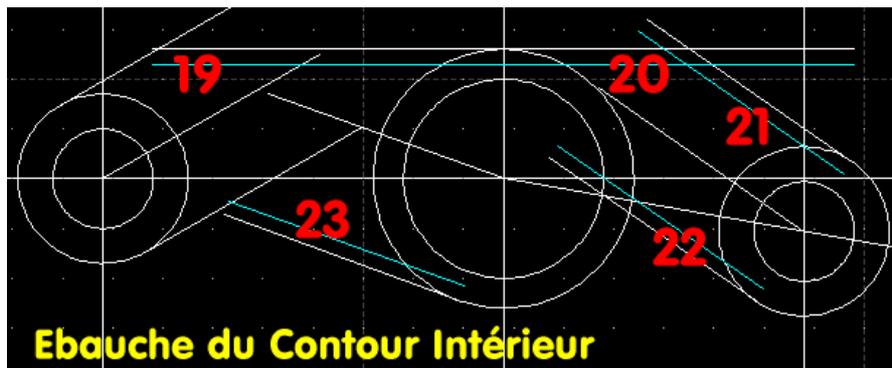
> *Point d'accrochage: [Fin],* et poser le segment (18).

Le travail devrait ressembler à la figure: "Ebauche du Contour".



#### 7.4 Ebauche du contour intérieur:

Il s'agit des deux *poches* situées de part et d'autre de l'*alésage* central de diamètre 40. Il est dit sur le "Sujet de l'exercice" que les parois minces ont une épaisseur de 3 mm et que les rayons non cotés ont pour valeur 6 mm. Pour tracer les parois minces, la fonction maintenant bien connue de *Ligne > Parallèle > Distance: [3]* sera utilisée. Il conviendra de poser cinq segments (19 à 23), représentés couleur cyan sur la figure: "Ebauche du Contour Intérieur", uniquement pour la clarté de l'exposé. *Ne changer en réalité ni de calque ni d'attributs.*



## 7.5 Finition de la vue de Face:

Le plus gros du travail est fait. Il reste maintenant trois opérations pour terminer les contours intérieurs et extérieurs.

A) Raccorder les entités:

**Qcad** dispose d'une puissante fonction de raccordement, qu'il appelle: *Arrondi*, accessible par l'icône *Edit* de la Palette Principale, ou *> Menu Modifier > Arrondi*.

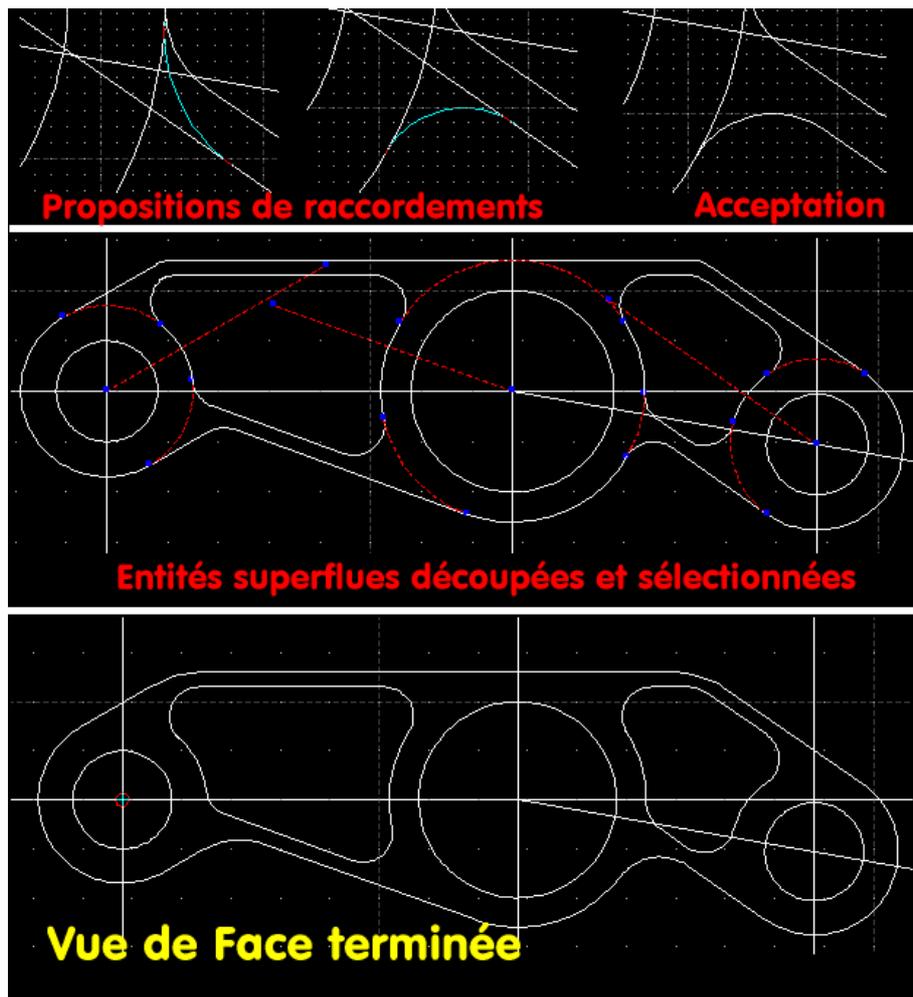
Lorsque la fonction est sélectionnée, entrer la valeur du rayon, et cocher ou décocher l'option *Ajustement*. Le fait de valider *Ajustement*, supprimera automatiquement l'excédent d'entité située "après" l'arrondi (on dit aussi: *congé*). Ragraéage très intéressant. De plus, **Qcad** propose des solutions possibles lorsque le pointeur survole des entités à raccorder. Il suffit de valider d'un clic la solution idoine qu'il proposera. La fonction étant *modale*, en changeant seulement la valeur du rayon dans le *champ de saisie*, on modifie la dimension des nouveaux congés créés.

B) Découper les entités raccordées.

Cela se fera en leur point de tangence par *>Menu Modifier > Couper > Intersection*. ( **Qcad** traite les points de tangence comme des points d'intersection).

C) Supprimer tous les éléments inutiles.

Voir illustraion de l'exposé sur la figure: "Vue de Face Terminée".



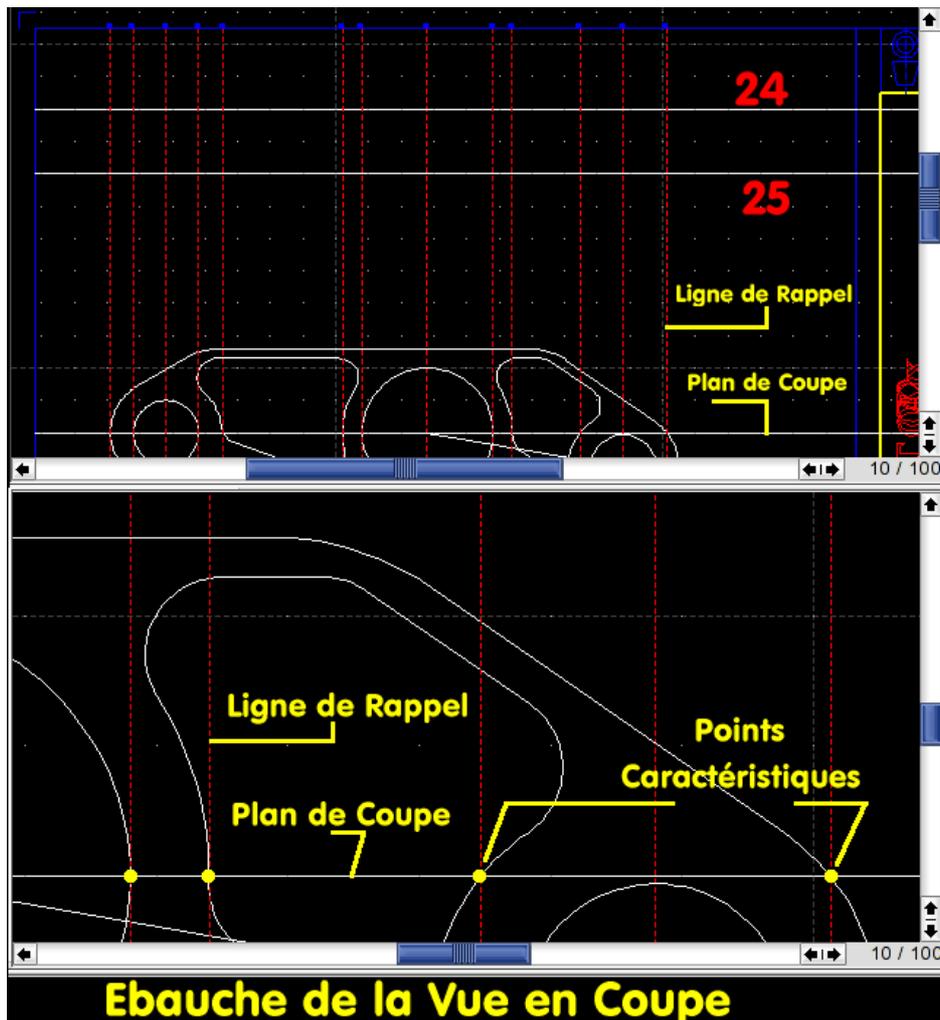
## 7.6 Construire la vue en Coupe:

Il doit y avoir correspondance entre la Vue de Face et la Vue en Coupe. Cet impératif s'obtient en traçant des *lignes de rappel* (ou: *de projection*) de tous les *points caractéristiques* contenue dans le *Plan de Coupe* ( A–A), opération ne présentant aucune difficulté.

Pour commencer, rendre de nouveau visible le calque *Format*, puis tracer une parallèle (24) distante de 25 mm du bord supérieur du cadre, et une autre (25) distante de 45. Puis, tracer les *lignes de rappel* par:

- > *Menu Dessin > Ligne > Parallèle par un point > Menu Accrochage > Intersection.*
- > Désigner le bord gauche du cadre comme référence, et ensuite les *points caractéristiques* de la vue de face contenus dans le *plan de coupe*.

Sur la figure: "Ebauche de la Vue en Coupe", les *lignes de rappel* ont été représentées en interrompus rouges, et les *points caractéristiques* sont jaunes afin de faciliter la compréhension. **Attention : QCad** ne les présentera pas ainsi.



## 7.7 Terminer la Vue en Coupe:

La Vue en en Coupe n'est constituée que de rectangles, certains hachurés (il s'agit des surfaces situées dans le *Plan de Coupe*) et d'autres non. Deux solutions sont possibles ici:

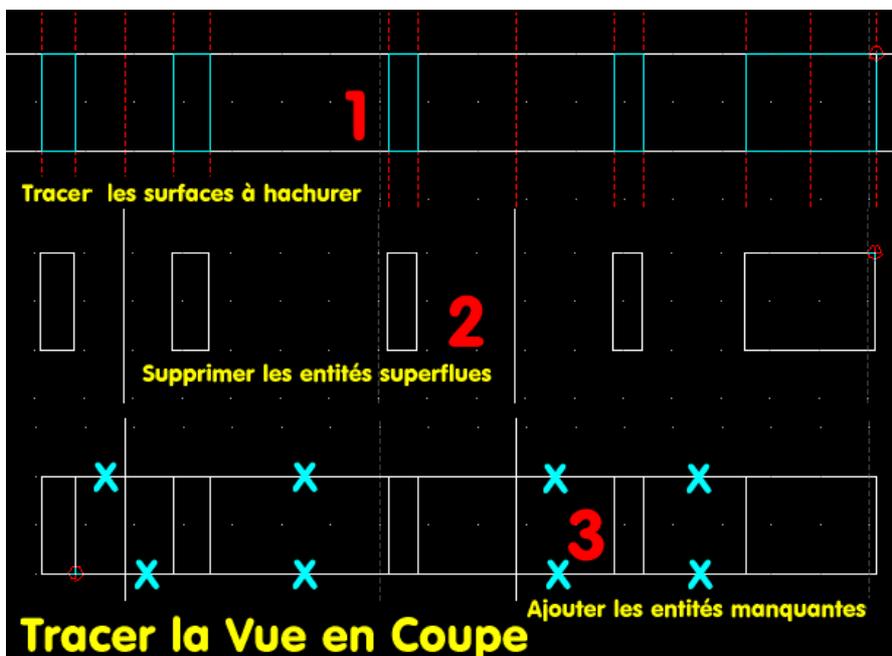
- soit découper les lignes de rappel et les lignes 24/25,
- soit tracer des rectangles en s'appuyant sur les constructions existantes, que l'on supprimera par la suite.

C'est la solution que l'on retiendra dans un but pédagogique, puisque nous avons déjà utilisé l'autre solution. Nous ne tracerons que les surfaces devant être hachurées, cela afin de faciliter l'opération de *hachurage* qui ne donne les résultats attendus, quel que soit le programme de **DAO** utilisé, que si la surface à hachurer est fermée et si les entités du contour n'en recouvrent pas d'autres. Il y a là un piège dans lequel tombent souvent les novices. Ensuite, nous supprimerons le superfétatoire et nous reconstruirons le manquant.

Donc,

- 1 Tracer les surfaces à hachurer:
  - > *Menu Dessin > Ligne > Rectangle > Menu Accrochage > Intersection.*
- 2 Supprimer les entités superflues
- 3 Ajouter les entités manquantes:
  - > *Menu Dessin > Ligne > Deux Points > Menu Accrochage > Extrémité.*

La figure: "Tracer la Vue en Coupe" montre les trois étapes décrites.



## 7.8 Changer les attributs et hachurer:

Hormis le *hachurage*, la *cotation* et les *habillages* divers, le plan est terminé du point de vue de l'*esquisse* (*Ebauche*). En donnant aux traits le type et l'épaisseur (largeur) qu'ils doivent avoir, ainsi qu'une couleur pour les différencier du premier regard, nous passons à la phase définitive de réalisation.

A) Changer les attributs des contours:

> *Menu Modifier* > *Attributs* > *Sélectionne fenêtre* > .

> Effectuer la sélection dans une *vue globale* (*Zoom auto*) en omettant les traits d'axe > *Continue Action*.

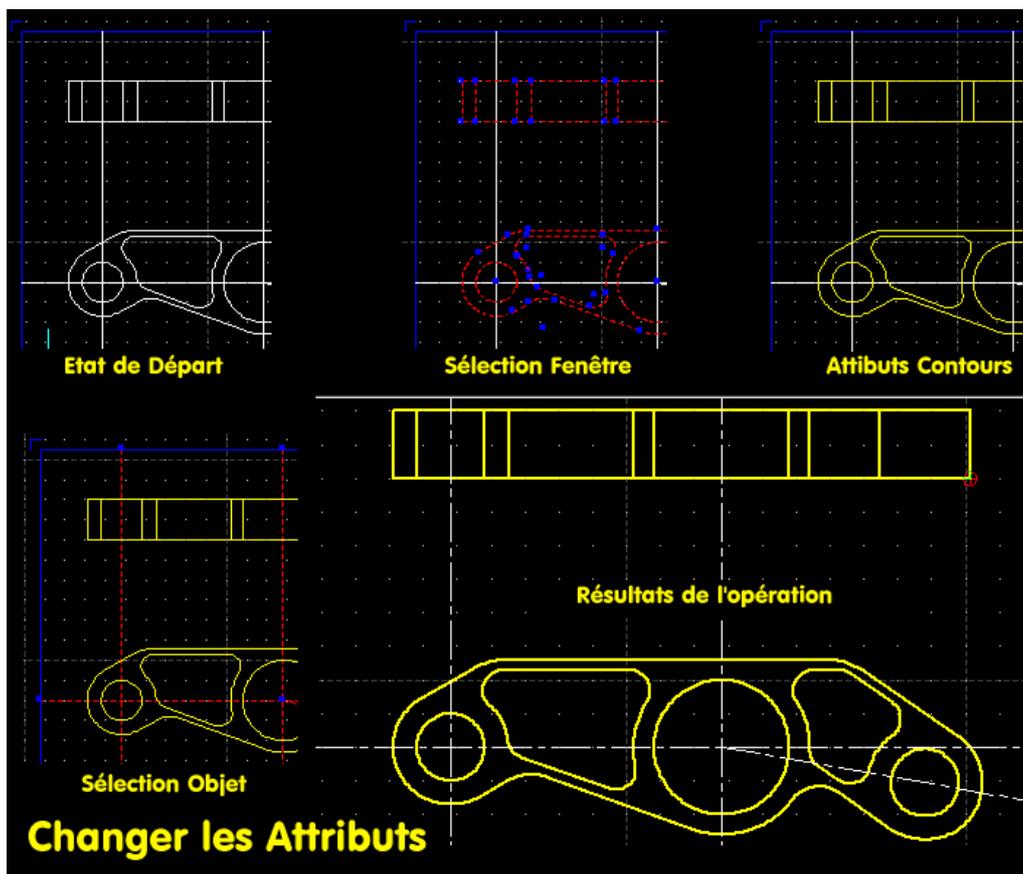
> *Couleur*: [Jaune], *Largeur*: [0.7 (iso)], *Type de Trait*: [inchangé] > *Ok*.

B) Changer les attributs des traits d'axe:

> (*De-*) *Selection Objet* > *Désigner les quatre traits d'axe* > *Continue Action*.

> *Couleur*: [Non modifié], *Largeur*: [0.25 (iso)], *Type de Trait*: [Trait d'axe] > *Ok*.

(Voir Fig: "Changer les Attributs")



C) Hachurer les surfaces coupées:

Disposer les *hachures* ne présentent pas de difficulté si l'on a pris les précautions énoncées plus avant. **Qcad** est, dans ce domaine là, un très bon outil. Plusieurs opérations logiques sont cependant nécessaires.

> *Palette Principale* > *Hachures* > *Sélectionne Fenêtre* [tracer cadre autour de la Vue en Coupe] > *Continue Action* (1)

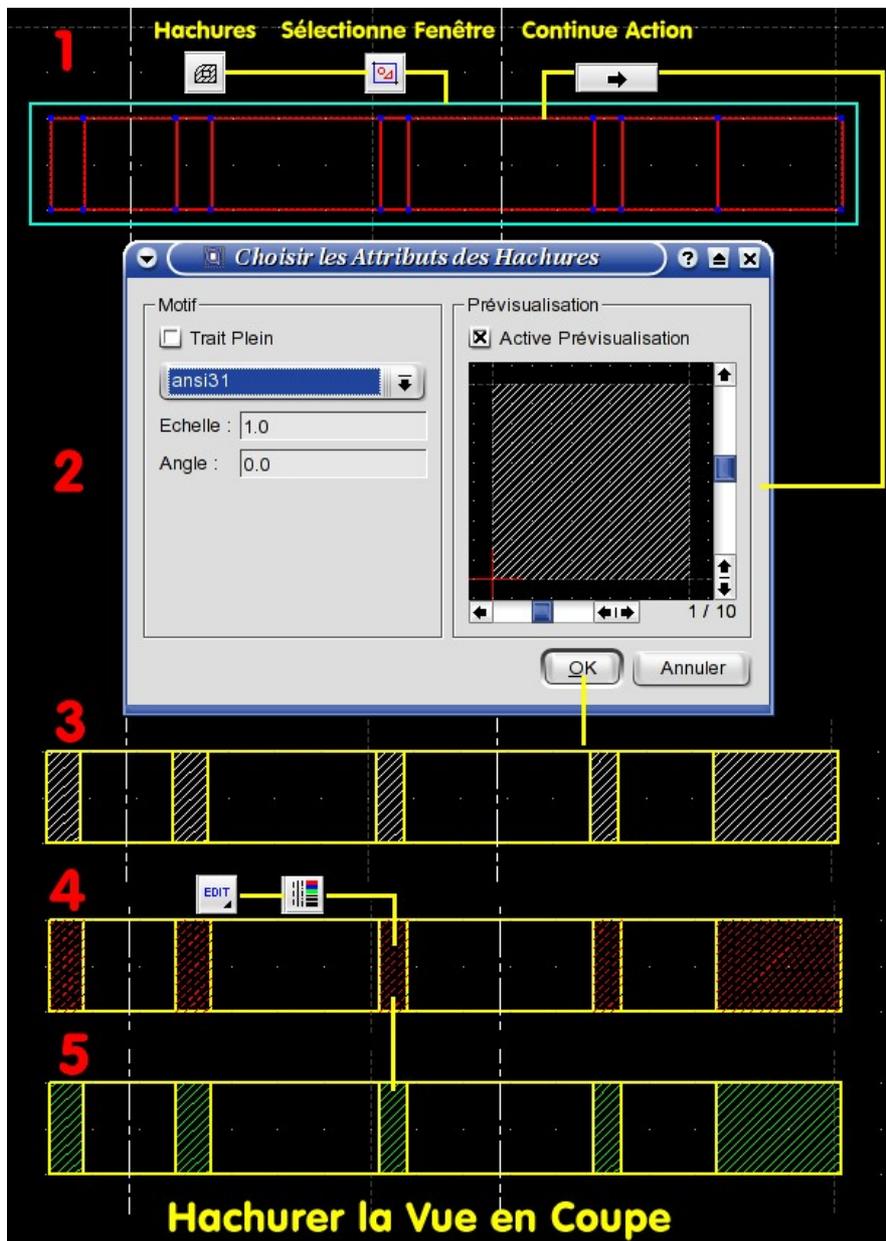
> Dans la boîte de dialogue > *Motif*: [ansi31], *Echelle*: [1], *Angle*: [0] > *Ok* (2)

> Le hachurage s'effectue (3). Comme toutes les surfaces ont été hachurées en même temps, l'ensemble des hachures est considéré comme une entité unique. Très pratique pour une modification éventuelle!

> *Palette Principale* > *Edit* > *Attributs* > *Sélection Objet* [désigner les hachures] > *Continue Action* (4).

> *Couleur*: [Vert], *Largeur*: [ 0.25 (iso)] > *Ok* (5)

(Voir résultats sur Fig: "Hachurer la Vue en Coupe", page suivante)



## 7.9 Dimensionner la pièce:

*Dimensionner, c'est-à-dire: effectuer la cotation.* Donc, il convient pour cela d'activer le calque *Cotation*, dont les attributs ont déjà été définis. Toute entité posée sur ce calque aura les attributs du calque. Avant de commencer, délimiter les traits d'axes à 5–10 mm des surfaces extérieures. Cela nécessite de tracer des parallèles provisoires distantes de 5–10 mm de ces surfaces afin de pouvoir couper les traits d'axes à leurs intersections (Peu pratique), puis de supprimer l'excédent. Cela effectué, on peut se consacrer à la *cotation* en sachant que:

- les *lignes de rappel* des cotes linéaires doivent être accrochées à des points remarquables des entités.
- les *lignes de cotes* (celles qui supportent les dimensions) devront être accrochées à la grille pour leur alignement relatif.
- la *cotation angulaire* s'effectue toujours en désignant les entités dans le *sens trigonométrique*.
- Il est parfois nécessaire d'ajouter des *lignes de rappel* pour certaines cotes angulaires (Ici: 30°, 20° et 35°)
- le paramétrage des cotes (texte, flèches) s'effectue avant ou après la cotation dans:

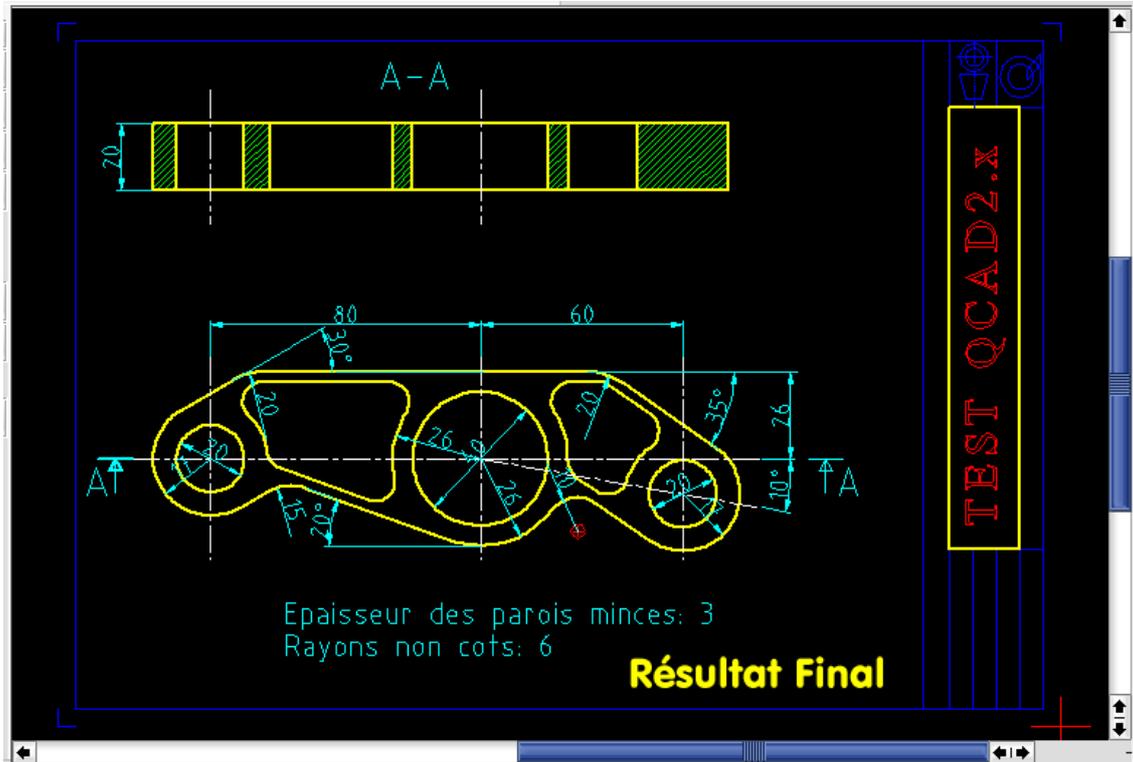
> *Edition* > *Préférences du Dessin Courant* > *Cotation* > *Hauteur de Texte*: [5], *Taille de Flèches*: [3.5] > *Ok*.

Cela étant précisé, la mise en place obéit à la démarche suivante;

> *Cotation* > *Horizontale* > *Accro. Extrémité* [1° Ligne de rappel] [2° Ligne de Rappel], *Accro. Grille* [Ligne de cote].

Il ne restera plus alors qu'à épaissir les extrémités du *Plan de Coupe*, dessiner la *flèche de Coupe* à une extrémité, la sauvegarder en *Bloc*, pour l'insérer comme *Bloc* à l'autre extrémité, écrire toutes les données textuelles et signer son oeuvre.

Arrivés à ce stade, le travail devrait ressembler à la figure: "Résultat Final", dont un examen attentif met en lumière ce qu'il y a de défectueux, faute à certaines limitations de **QCad** signalées en conclusion.



## 7-10 Imprimer le dessin:

Sur une petite A4 de bureau à jet d'encre, il convient de remettre le grand axe du format A4 en position verticale. Pour ce faire:

> *Edit* > *Rotation* > *Tout sélectionner* [l'ensemble des entités change de couleur]

> *Continuer Action* > *Spécifier la référence*: 0,0 [Entrée]

> Dans la boîte *Options de Rotation*, cocher *Supprimer Original*, entrer *Angle*: -90 > *OK*.

L'ensemble a pivoté et pourra être imprimé en *portrait*.

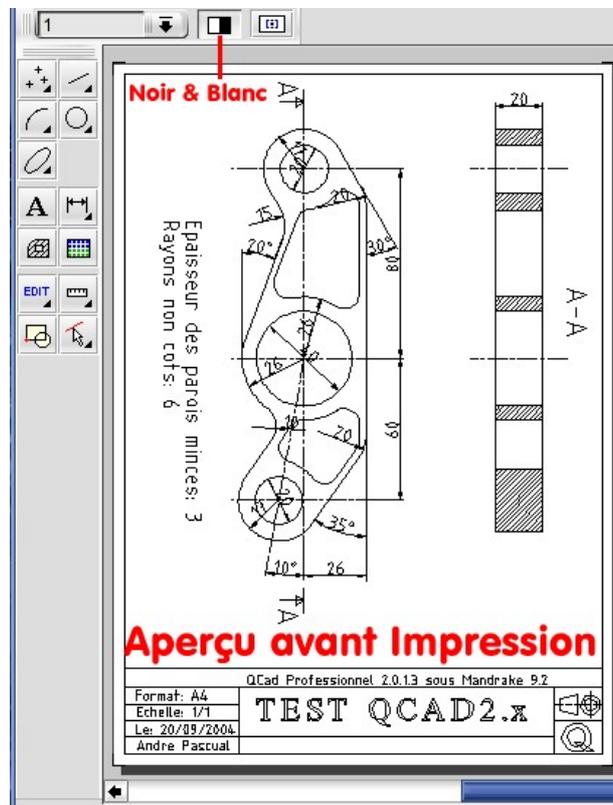
Si nous avons de la chance, aucune altération n'a eu lieu. Si, au contraire nous n'en avons pas, les hachures n'ont peut-être pas suivi le mouvement, et il faut les recréer par la méthode déjà décrite.

Tant qu'à produire un document définitif, autant remplir le cartouche, puis vérifier le résultat par:

> *Fichier* > *Aperçu avant impression* > [Noir & Blanc].

A noter que l'on peut effectuer manuellement la mise en page en déplaçant la feuille à la souris par rapport au dessin, qui, lui, reste fixe.

La figure: "Aperçu avant Impression" montre ce que chacun devrait avoir sur son écran, prêt à être imprimé, avec les types de traits choisis.



## 8 Conclusion:

**Qcad** s'est beaucoup amélioré depuis le premier test relaté dans **Linux Magazine France**. La fonction de *raccordement*, qui pêchait autrefois, est devenue un petit bijou; les propositions de solutions pour la plupart des constructions envisagées sont un régal, au même titre que la *modalité* des fonctions, qui font de **Qcad** un outil productif. Les versions **Linux** et **Windows** (et sans doute les autres aussi) sont absolument identiques. La sauvegarde du travail s'effectuant intelligemment en **DXF 2000**, lu par la plupart des programmes de **DAO**, la communication avec l'extérieur est garantie. L'impression dans un fichier en *PostScript*, facilement transformable en *PDF*, est une bonne chose. La vingtaine d'unités de mesure, les différents formats et précisions des valeurs linéaires ou angulaires, la possibilité de mesurer des longueurs, des angles, des aires, ou de recourir à des expressions mathématiques..., aussi, de même que l'importation d'*images bitmap* (une douzaine de formats) en vue de leur *vectorisation* manuelle. On appréciera également l'aide conséquente en HTML qui se lit avec un outil intégré rapide et léger, et l'on goûtera encore à leur juste valeur les bibliothèques de symboles très fournies.

On regrettera cependant:

- l'absence de droite de construction (d'esquisse ou droite infinie, selon le programme).
- le nombre limité de solutions de construction par tangence.
- la gestion fruste des calques. On aimerait que, d'une manière automatique, l'activation d'un calque protège les autres d'une altération involontaire, sans qu'il soit nécessaire de les verrouiller explicitement. De même, on aimerait que tous les calques en sous-couche soit automatiquement signalés par une couleur unique, et réservée, de toutes les entités qu'ils contiennent.
- l'impossibilité de découper une entité autrement que par rapport à une autre entité.
- l'absence du trait de coupe (mixte fin, épais aux extrémités) et de flèches de coupe associées.
- la faiblesse du système de cotation: impossibilité de poser la cote (la valeur de la dimension) ailleurs qu'au milieu de la ligne de cote, d'avoir des lignes de cotes brisées pour les rayons...
- l'absence des caractères accentués dans les fontes standard.
- la déstructuration possible des hachures lors d'un pivotement
- la disparition de la prise en charge des formats Microstation (\*.dgn) et HPGL (\*.plt, \*.hpgl, \*.inc)...

**Qcad** est fort connu dans le microcosme linuxien; son nom finira par atteindre les tympans de quelques décideurs en **CAO**, qui, dès lors, ne manqueront pas de le comparer aux autres solutions disponibles sous Linux: **VariCad V9** (2D/3D), **BricsCad V5** (2D/3D, qui se présente comme LE clone d' **AutoCad**), **Weber Synergy** (CAD/CAM,2D/3D) et, plus modestes et pas meilleurs, **BetterCad** (2D) ou **SagCad** (CAM). **Qcad**, sympathique, facile d'emploi, ne nécessitant pas un long apprentissage pour un habitué de la **DAO**, et performant pour des petits travaux, notamment pour des plans dits "d'architecte", révélera alors ce qui lui reste de chemin à parcourir pour devenir un véritable outil professionnel polyvalent.

**NOTA:**

Lors de l'écriture de ce texte, **Andrew Mustun** a posté sur Ribbonsoft la version 3.0.4 de **Qcad**. Au menu des nouveautés: corrections de bogues habituelles, compilation des exécutables avec **QT3.3**, mesures des surfaces polygonales et courbes splines, ouvertes ou fermées. Cela se traduit visuellement par une icône supplémentaire dans la Palette Principale, et ne contredit en rien le présent didacticiel. (Voir Fig: "Version 3.0.4")



**André PASCUAL**  
<<http://www.linuxgraphic.org>>  
[andre.pascual@linuxgraphic.org](mailto:andre.pascual@linuxgraphic.org)

Article publié en Janvier 2005 dans **LinuxPratique** N°28