

GUILLET Cyril  
ROLLET Samuel  
DAMOUR Olivier

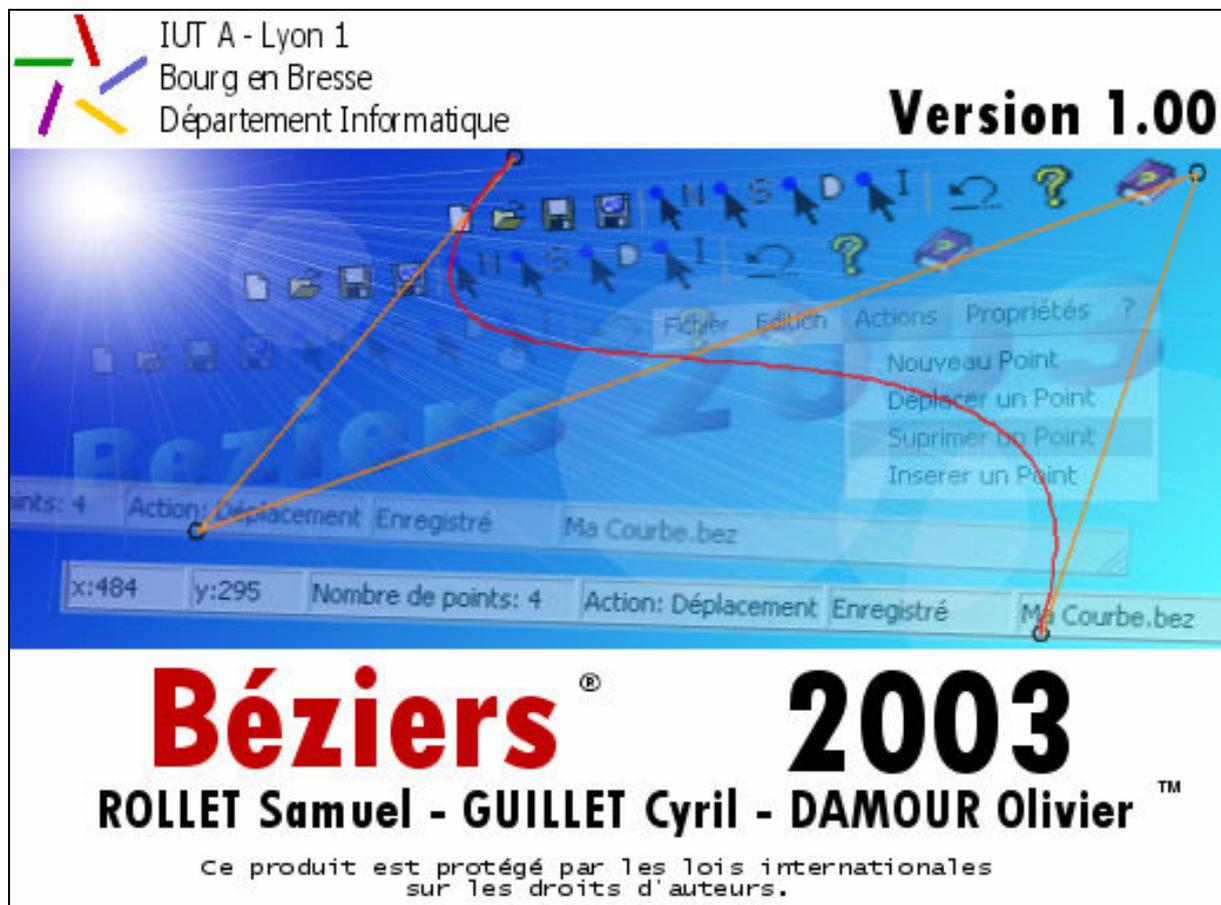


Université Claude Bernard Lyon 1  
IUT A  
Bourg-en-Bresse  
Département Informatique

1AG2

## Rapport Final

### Projet Informatique



Sujet : Courbe de Béziérs  
Logiciel : Béziérs® 2003

Pour le 28/05/003

## ***SOMMAIRE***

<u>Intro</u>		
Les Objectifs de départ et L'avancement actuel		p.03
<u>Ressources Utilisées</u>		
Ressources Humaines et Ressources Matériel		p.04
<u>Connaissances</u>		p.04
<u>Organisation</u>		
Organisation du Travail		p.05
La répartition des taches		p.05
Le Calendrier		p.07
<u>Les Difficultés du Projet et Les Problèmes Rencontrés</u>		p.08
<u>Mode d'emplois</u>		
Présentation du Logiciel		p.10
Fonctions du logiciel		p.12
<u>Conclusion &amp; améliorations possibles</u>		p.19
<u>Annexes:</u>		
Qu'est-ce qu'une courbe de Béziere		p.20
Fonction de Calcul d'une Béziere, Algorithme de De Casteljau		p.22
Visuel des différentes étapes d'avancement du projet		p.24
<u>Bibliographie</u>		p.26

## *INTRODUCTION*

Les courbes de Bézières sont des courbes particulières, définies par un nombre limité de points, les courbes s'obtenant en calculant successivement des barycentres entre chacun de ces points (retrouvez une présentation détaillée des courbes de Bézières en annexe 1). Leur principal intérêt est de pouvoir obtenir une courbe en mémorisant une faible quantité de points, les points de contrôle. Elles sont principalement utilisées dans le domaine de l'imagerie numérique, 2D et 3D, pour obtenir des formes sans angles et sur lesquelles on peut zoomer sans perte de qualité. C'est à cause de notre intérêt pour l'imagerie numérique que nous avons choisi ce sujet.

Le sujet de départ nous demandait de créer une interface graphique conviviale permettant de créer une courbe de Bézières en positionnant les points de contrôle à la souris, et proposait comme extension possible le tracé de plusieurs courbes, la modification de la courbe, et la sauvegarde des courbes tracées.

Dans notre Cahier des Charges nous avons fixé comme objectifs : « Pouvoir afficher une courbe de Bézières, la modifier en déplaçant les points, en ajoutant ou supprimant des points, l'enregistrer, et la récupérer à l'ouverture. »

Dans notre Rapport d'Analyse, nous nous étions fixé comme objectif de réaliser une interface multi documents MDI, avec un Menu et une Barre d'Outils permettant de nombreux réglages sur les caractéristiques de la courbe : précision, couleur...

Tous ces objectifs ont été atteints dans la réalisation de notre logiciel Bézières<sup>®</sup> 2003, comme le montre la description des différentes possibilités dans la partie « Mode d'Emplois p.10 » L'utilisateur peut créer plusieurs courbes simultanément dans plusieurs fenêtres. Il a la possibilité d'enregistrer ses courbes au format '.bez' et de les ré-ouvrir lors d'une autre utilisation de Bézières<sup>®</sup> 2003. L'utilisateur peut effectuer de nombreuses actions sur ses courbes : ajout, suppression, déplacement de points, insertion d'un point entre deux points déjà positionnés ; dans le cas où il effectuerait une mauvaise manipulation, il a la possibilité d'annuler la dernière action qu'il a effectué sur sa courbe. Il peut aussi modifier les propriétés de ses courbes : modifier le pas, c'est-à-dire la précision du tracé, modifier les couleurs de la courbe, des points ou du polygone de contrôle (segments reliant les points de contrôle successifs). Il peut aussi choisir d'afficher ou de masquer ce même polygone de contrôle. Enfin, il dispose d'une rubrique d'aide et d'une rubrique A Propos pour lui donner des informations supplémentaires sur le logiciel.

Comme réalisations optionnelles nous avons proposé l'enregistrement des courbes au format BMP, la possibilité d'afficher la dérivée d'une courbe de Bézières, et aussi l'affichage animé du tracé d'une courbe créée par l'utilisateur.

Nous n'avons jusqu'ici réalisé qu'une seule de ses trois options. Nous n'avons pas réussi à réaliser la sauvegarde au format BMP, car malgré de nombreuses recherches, nous n'avons pas trouvé d'informations concluantes sur l'enregistrement du contenu d'une fenêtre en image bitmap en langage C. L'affichage des dérivées n'a pas été réalisé par manque de temps. Nous avons donc réalisé l'option d'affichage animé du tracé d'une courbe, quand l'utilisateur le désire, il peut demander l'affichage animé de sa courbe, il verra alors les différents barycentres affichés et se déplacé pour donner l'image complète de sa courbe. De plus nous avons ajouté la possibilité de visualiser les coordonnées des points de contrôle et des points de la courbe dans une boîte de dialogue, offrant ainsi une option supplémentaire utile aux utilisateurs.

## ***RESSOURCES UTILISEES***

Pour la réalisation de ce projet nous avons du utiliser des ressources humaines et matérielles.

Au niveau humain, nous avons été 3 personnes, étudiants en première année en IUT informatique, à travailler sur ce projet. Le temps de travail demandé pour réaliser ce projet était d'au moins 60 heures par personnes, ce que nous avons largement dépassé, puisque le temps total que nous y avons consacré, en prenant en compte les temps de recherche d'information et les temps de développement, se situe entre 250 à 300 heures.

Au niveau matériel, nous avons principalement utilisé nos propres ordinateurs, mais aussi ceux de l'IUT. Nous avons développé Béziérs<sup>®</sup> 2003 pour et sous des systèmes d'exploitation Windows, principalement NT et XP. Le langage de programmation qui nous était imposé était le Langage C, nous avons donc respecté cette contrainte, bien que ce langage se soit avéré moins adapté à la programmation Windows que d'autres langages que nous avons étudié, comme le Visual Basic. Nous avons utilisé Borland C++ 5, logiciel de programmation en C et C++, qui c'est avéré un bon outil, nous permettant notamment de déboguer facilement notre code, et fournissant une rubrique d'aide très utile. Nous avons intégré à notre projet des bibliothèques qui nous ont permis de programmer plus simplement, notamment <windows.h> qui contient toutes les fonctions utiles à la programmation Windows.

Internet a été pour nous une source importante d'informations, aussi bien au niveau mathématique, pour nos recherches sur les courbes de Béziérs, qu'au niveau programmation, pour nos recherches d'exemples de code et d'explication du fonctionnement de certaines fonctions prédéfinies.

## ***CONNAISSANCES***

La réalisation de ce projet nous a demandé de mettre en œuvre les connaissances que nous avons acquises en cours de SE Programmation Windows (création d'une fenêtre Windows, programmation événementielle...), en cours d'Algorithmique (algorithmes de gestion des liste chaînées, de recherche, écriture d'algorithmes adaptés à notre programme), et en cours de Langage C (programmation de base en C, traduction des algorithmes...).

Mais bien entendu, ces connaissances n'ont pas été suffisantes pour mener à bien notre projet, et nous avons donc du acquérir de nouvelles compétences en mathématique et en programmation Windows.

Nous avons notamment du faire des recherches sur les courbes de Béziérs et les algorithmes de calculs des points de ces courbes par rapport aux coordonnées des points de contrôle ; il nous a fallu comprendre le fonctionnement de ses courbes, afin de pouvoir traduire en langage C l'algorithme mathématique de De Casteljaou, un des algorithmes qui permet de tracer une courbe de Béziérs d'ordre n, c'est-à-dire quelque soit le nombre de points de contrôles.

Nous avons du aussi approfondir nos connaissances dans la programmation Windows en C, notamment la programmation d'un logiciel MDI, mais aussi sur les boites de dialogue et boites de dialogues communes, ainsi que sur les interactions entre les différentes fenêtres, pour offrir aux utilisateurs de Béziérs<sup>®</sup> 2003 toutes les options que vous verrez décrites dans la partie « Mode d'Emplois p.10 ».

## **ORGANISATION**

### **Organisation du Travail**

Le développement de Béziérs® 2003 a demandé une répartition des tâches entre nous, dont vous pouvez retrouver le détail plus loin. Il a donc s'agit d'un véritable travail d'équipes, nous parlions régulièrement plusieurs fois par semaines de notre avancement respectif ainsi que des problèmes que nous rencontrions, chacun essayant d'aider les autres pour solutionner les problèmes, et précisant aux autres les contraintes qu'imposaient ses nouvelles réalisations sur la globalité du projet, pour maintenir la cohérence entre les différentes parties développées individuellement. De plus, nous avons consacré plusieurs jeudi après midi à des réunions en groupe pour décider de l'aspect visuel du projet (fenêtre simple ou MDI...), de l'orientation à donner au projet (quels sont les options que l'on choisit d'intégrer...), et réfléchir collectivement à des problèmes particuliers, notamment la manière de détecter la position ou insérer un point de contrôle entre les deux points les plus proches.

### **La répartition des tâches**

#### **- Partie Recherche d'information :**

##### **- Recherche sur les courbes de Béziérs**

Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

##### **- Recherche pour la programmation Windows <sup>tm</sup> :**

Les recherches sur ce sujet ont durées tout au long du projet, la programmation Windows en C étant assez complexe.

Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

#### **- Partie mathématique :**

##### **- Traduction en C de l'algorithme de De Casteljaou**

Responsable : ROLLET Samuel

#### **- Partie Interface :**

##### **- Réalisation de la fenêtre de base**

Responsable : GUILLET Cyril

##### **- Réalisation d'une interface multi documents**

(MDI : Multiple Document Interface Application) : pour afficher des sous fenêtres internes à la fenêtre mère, chacune avec une courbe de Béziérs propre.

Responsable : ROLLET Samuel

##### **- Réalisation d'une barre d'outils :**

Fenêtre fille consacrée à cet effet, proposant des raccourcis pour les actions les plus fréquentes.

Responsables : DAMOUR Olivier, ROLLET Samuel

**- Réalisation des Raccourcis Clavier :**

Raccourcis Clavier pour les actions les plus fréquentes :  
enregistrement, actions sur la courbe.

Responsable : DAMOUR Olivier

**- Réglage des propriétés de la courbe par l'utilisateur :**

Réglage du pas (précision).

Réglage des couleurs : de la courbe, des points de contrôle, du polygone de contrôle.

Responsable : ROLLET Samuel

**- Affichage de la Courbe :**

Affichage statique et affichage animé de la courbe.

Responsable : ROLLET Samuel

**- Programmation du menus de la Fenêtre Mère :**

Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

**- Partie Algorithmes de Modification de la courbe :**

**- Gestion des points de contrôle à la souris :**

Ajout et suppression de points d'une courbe.

Algorithme de déplacement d'un point permettant un retracé en temps réel de la courbe.

Algorithme de calcul des poignées pour l'insertion d'un point entre deux points déjà existant.

Algorithme de détection de l'appartenance d'un point à la courbe lors d'un clic.

Responsable : GUILLET Cyril

**- Partie « Edition » Annuler :**

Permet d'annuler la dernière action effectuée sur la courbe.

Responsable : DAMOUR Olivier

**- Partie de Sauvegarde :**

Enregistrement de toutes les informations utiles d'une courbe au format '.bez' (coordonnées des points, couleurs...)

Détection de la modification d'une courbe enregistrée.

Procédure de proposition de sauvegarde des courbes non enregistrées ou modifiées à leur fermetures ou à la fermeture du logiciel.

Fonction d'ouverture d'un fichier au format '.bez'.

Responsable : ROLLET Samuel

**- Partie « ? » :**

**- Aide :**

Ouverture d'un fichier d'aide expliquant le fonctionnement du logiciel à l'utilisateur.

Responsable : GUILLET Cyril

**- A Propos :**

Design de l'image (cf. p.1) et affichage de celle-ci dans une fenêtre « A Propos »

Responsable : ROLLET Samuel

**- Mise en relation des différentes parties :**

Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

**- Rédactions des Rapports :**

**- Cahier des Charges :** pour le 16/01/003

- Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

**- Rapport d'Analyse :** pour le 20/02/003

- Responsable : ROLLET Samuel

**- Rapport d'Avancement :** pour le 27/03/003

- Responsables : DAMOUR Olivier, ROLLET Samuel

**- Rapport Final :** pour le 28/05/003

- Responsables : DAMOUR Olivier, GUILLER Cyril, ROLLET Samuel

## **Le Calendrier**

Le temps de travail estimé dans le cahier des charges (60H de travail en tous) a été largement dépassé, les estimations de durées dans le Rapport d'Analyse ont été les plus réalistes étant donné le temps nécessaire pour la programmation de l'interface graphique

Le calendrier du Rapport d'Analyse prévoyait :

-Au 15 Mars : Affichage des Courbes de Béziérs dans une interface MDI, ce qui a été réalisé, mais les problèmes rencontrés avec la Toolbar (cf. Problèmes Rencontrés) ont retardé l'intégration de celle-ci au projet, l'intégration de la Toolbar sous sa nouvelle forme ayant eut lieu au 20 Mars.

Ce retard n'a pas empêché de débiter la programmation des actions à effectuer suivant les choix de l'utilisateur dans le menu, comme prévu. Certaines options de la courbe fonctionnaient au 25 Mars, le choix des différentes actions à effectuer, ainsi que l'affichage de nouveaux documents sans altérer les courbes précédentes, et l'affichage d'une fenêtre 'A propos...'. La base graphique de 'Enregistrer Sous' et 'Ouvrir' était elle aussi réalisée, prête à accueillir les fonction de sauvegarde et de récupération qui ont été développées en Avril.

Fin Avril et le mois de Mai furent consacrés à la programmation de l'option d'Insertion d'un point entre deux points existant, de l'aide, des raccourcis clavier et de l'option d'affichage.

Fin Mai fut consacré à la suppression des bugs, comme prévu initialement, ainsi qu'à la rédaction du rapport final.

Nous avons donc bien suivi le calendrier que nous nous étions fixé initialement, et avons disposé de suffisamment de temps pour développer une des options, l'affichage animé d'une courbe.

## ***LES DIFFICULTES DU PROJET & LES PROBLEMES RENCONTRES***

### **Difficultés du Projet**

Nous avons tout d'abord estimé que la difficulté de ce projet résidait dans les algorithmes de calculs relatifs aux courbes de Bézières (cf. Cahier des Charges) ; mais après quelques semaines de recherches, nous avons réussi à comprendre le fonctionnement de ses courbes, et nous avons rapidement pu calculer les coordonnées des points d'une courbe par rapports à ses points de contrôle, grâce à la traduction en C de l'algorithme de De Casteljau. Ce n'est donc pas dans la partie mathématique que nous avons rencontré les principales difficultés, mais au niveau de la programmation Windows, comme le montre les « problèmes rencontrés » exposés ci-après. En effet, le langage C ne nous a pas apparut le plus adapté à la programmation événementielle et à la programmation Windows, et il a été difficile de trouver des informations sur la programmation Windows en C.

### **Problèmes Rencontrés**

Voici les principaux problèmes que nous avons rencontrés pendant le développement, qui nous ont nécessité le plus de temps pour les résoudre, et/ou qui auraient pu nuire à la qualité de Bézières® 2003 :

#### **- Problème d’Affichage lors du Déplacement des Points**

Lors de la programmation du déplacement des points de contrôle à la souris, nous n'avons pas obtenu le résultat voulu immédiatement. En effet, la première version du Déplacement de points ne réaffichait la courbe modifiée que lorsque le déplacement du points était terminé. Dans cette première version, nous utilisions une boucle pour récupérer les coordonnées successives du point déplacé, cela empêchait le réaffichage de la courbe en temps réel.

Nous avons donc du supprimer cette boucle, et utiliser des tests nous permettant de savoir si un déplacement était en cours (test sur l'action en cours définie par l'utilisateur dans le menu Actions et sur la position du bouton gauche de la souris WM\_LBUTTONDOWN et WM\_LBUTTONDOWN), pour savoir si nous devons récupérer les coordonnées de la souris pendant son déplacement (événement WM\_MOUSEMOVE). Ces modifications ont permis un réaffichage en temps réel des courbes pendant le déplacement d'un des points de contrôle.

#### **- Problème de Scintillement de l’Ecran**

Pour garantir un affichage toujours juste de la courbe, nous avons tout d'abord opté pour un réaffichage périodique assez fréquent, et nous avons pour cela mis en place un Timer, qui recalculait la courbe et la réaffichait tout les 50 millièmes de secondes. Cette technique qui s'avérait très utile pour afficher une courbe prenant en compte les dernières modifications apportées entraîna un scintillement de la courbe à l'écran, ce qui n'était pas acceptable pour le confort des utilisateurs.

Nous avons donc du abandonner l'utilisation du Timer, pour ajouter le code nécessaire à un réaffichage seulement après chaque modification de la courbe ou de ses caractéristiques.

### **- Problèmes de Mise en Place d'une Barre d'Outils**

Pour faciliter l'utilisation de Béziers<sup>®</sup> 2003, nous avons souhaité intégrer une Barre d'Outils proposant les actions les plus courantes de notre logiciel, pour limiter les accès au Menu de la part de l'utilisateur. Nos recherches nous ont appris qu'il était nécessaire d'utiliser CreateToolBarEx () pour créer une telle Barre d'Outils. Nous avons donc effectué des tests sur une version mono fenêtre de notre logiciel, ce qu'y c'est avéré concluant. Nous avons donc commencé à développer cette Barre d'Outils, mais elle c'est avéré incompatible avec la version MDI (multi documents) de Béziers<sup>®</sup> 2003, malgré nos multiples tentatives d'en adapter le code.

Pour éviter ce problème, tout en continuant à offrir une Barre d' Outils aux utilisateurs, nous avons décidé d'opter pour une Barre d'outils flottante, en utilisant une fenêtre fille consacrée à cet effet, et proposant des boutons cliquables pour effectuer les actions courantes.

### **- Problème lors de la Maximisation d'une Fenêtre Fille**

Pendant le développement de l'interface MDI, nous avons été confrontés à un problème d'affichage des fenêtres filles à l'intérieur de la fenêtre mère. Lors de la maximisation d'une fenêtre fille, les icônes de re-dimension et de fermeture disparaissaient, ne s'affichant pas correctement à droite du menu. Il était alors impossible de réduire la fenêtre maximisée, ou de passer à une des autres fenêtres filles présentes derrière celle-ci.

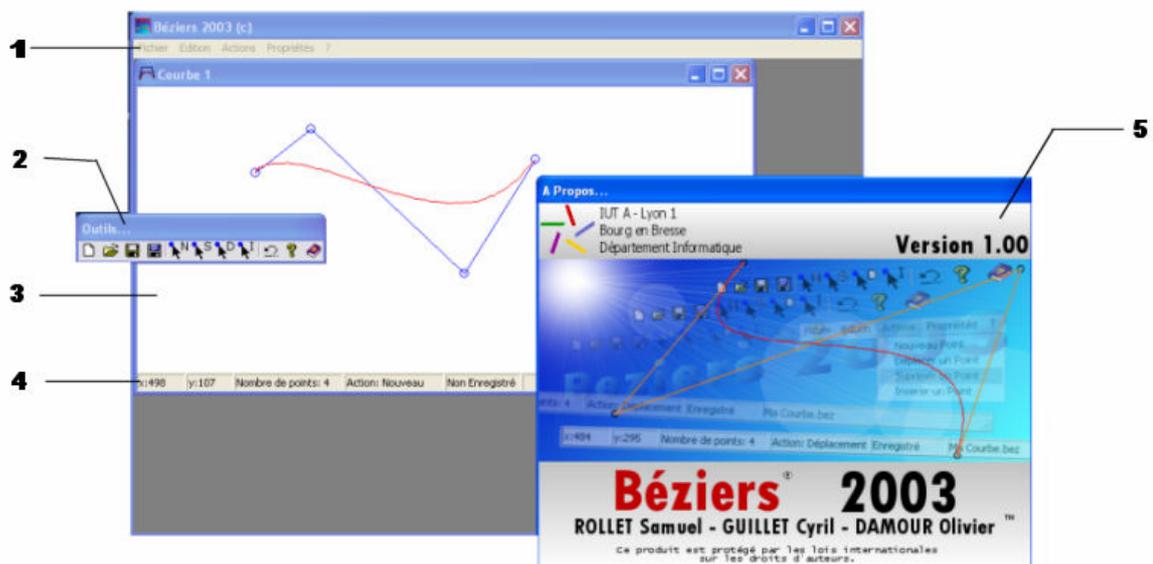
La gestion de la re-dimension des fenêtres et de l'affichage des icônes étant faite automatiquement par Windows, la solution à ce problème a été assez longue à trouver ; en effet, nous soupçonnions un mauvais paramétrage des propriétés des fenêtres fille, ce qui n'était pas le cas. La solution résidait dans la gestion faite par défaut par Windows des événements WM\_SIZE et WM\_WINDOWPOSCHANGED ; ayant programmé des actions particulières pour ces deux événements, l'appel à DefMDIChildProc() (fonctions de gestion par défaut des événements pour lesquelles des actions particulières n'ont pas été définies lors de la programmation) ne s'effectuait plus. Il nous a donc fallu ajouter un appel à cette fonction à la fin du code que nous avons défini pour ces deux événements pour permettre à Windows de gérer correctement la maximisation des fenêtres filles.

# MODE D'EMPLOIS

## I) Présentation du logiciel

### A) Présentation Générale

Béziérs<sup>®</sup> 2003 a été créé dans le but de tracer des courbes de Béziérs nous afficherons sa courbe, ses points ainsi que le polygone de contrôle qui permet de tracer la courbe.



1.1 - Schéma de présentation graphique du logiciel

1. Barre de menu où l'on retrouve les fonctions du logiciel dont Fichier, Edition, Actions, Affichage et le point d'interrogation ( ?).
2. Barre d'outils où l'on retrouve les fonctions essentielles sous forme d'icônes.
3. Zone de dessin: c'est l'endroit où s'effectue le tracer de la courbe.
4. Barre d'état permet de connaître la position du curseur, le nombre de points de la courbe, l'action en cour et enfin savoir si la courbe est enregistrer ou non.
5. Fenêtre d'état d'information ou de modification, ici c'est la fenêtré a propos.

### B) La barre d'outils

La barre d'outils permet d'accéder aux fonctions simples sans utiliser le menu. Toutes ces actions seront expliquées dans la suite de la notice.



6. Action en cours (Nouveau, Insertion, Déplacement, Suppression).

## II) Fonctions du logiciel

### A) Comment créer une courbe.

Pour créer une nouvelle courbe il faut tout d'abord ouvrir une zone de dessin puis poser des points et enfin on pourra agir sur ces points en les déplaçant, les supprimant, ou en insérant des nouveaux.

#### 1) Créer une nouvelle fenêtre de dessin

Pour créer une nouvelle fenêtre de dessin, cliquer sur **fichier** dans la barre de Menu (schéma 1.1 - 1) puis sur **Nouveau** une fenêtre s'ouvre: c'est la zone de dessin. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente une **feuille blanche** (schéma 1.2 - 3). Il existe aussi un raccourci **F1**.

#### 2) Créer des points

L'action par défaut quand on crée une zone de dessin est celle de la création de points. Pour afficher un point il faut vérifier que la partie **Action** (schéma 1.3 - 4) de la barre d'état de la zone dessin soit sur **Nouveau**.

Si il n'est pas sur nouveau il faut cliquer sur **Action** dans la barre de Menu (schéma 1.1 - 1) puis sur **Nouveau point**, on peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente un **curseur avec un N** (schéma 1.2 - 1), il y a aussi un raccourci clavier **F5**.

Pour afficher un point il faut cliquer sur la zone de dessin (la partie blanche de la fenêtre de dessin): un point apparaît.

#### 3) Déplacer un point

Pour déplacer un point, il faut cliquer sur **Action** dans la barre de Menu (schéma 1.1 - 1) puis sur **Déplacer point** on peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente un **curseur avec un D** (schéma 1.2 - 7), il y a aussi un raccourci clavier **F7**.

On pourra vérifier que l'action est active en regardant que la partie **Action** (schéma 1.3 - 4) de la barre d'état de la zone dessin soit sur **Déplacement**.

Une fois l'action active, il faut cliquer sur le point que l'on veut déplacer, un carré apparaît autour du point, ceci montre qu'il est sélectionné, on tient cliquer et on déplace la souris ce qui fait bouger le point, pour arrêter le déplacement on arrête de tenir le bouton cliquer.

#### 4) Supprimer un point

Pour supprimer un point, il faut cliquer sur **Action** dans la barre de Menu (schéma 1.1 - 1) puis sur **Supprimer point** on peut aussi utiliser la barre d'outils en cliquant

sur l'icône qui représente un **curseur avec un S** (schéma 1.2 - 9), il y a aussi un raccourci clavier **F6**.

On pourra vérifier que l'action est active en regardant que la partie **Action** (schéma 1.3 – 4) de la barre d'état de la zone dessin soit sur **Suppression**.

Une fois l'action active, il faut cliquer sur le point que l'on veut supprimer: il s'efface.

#### 5) Insérer un point

Pour insérer un point, il faut cliquer sur **Action** dans la barre de Menu (schéma 1.1 - 1) puis sur **Insérer point** on peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente un **curseur avec un I** (schéma 1.2 - 8), il y a aussi un raccourci clavier **F8**.

On pourra vérifier que l'action est active en regardant que la partie **Action** (schéma 1.3 – 4) de la barre d'état de la zone dessin soit sur **Insertion**.

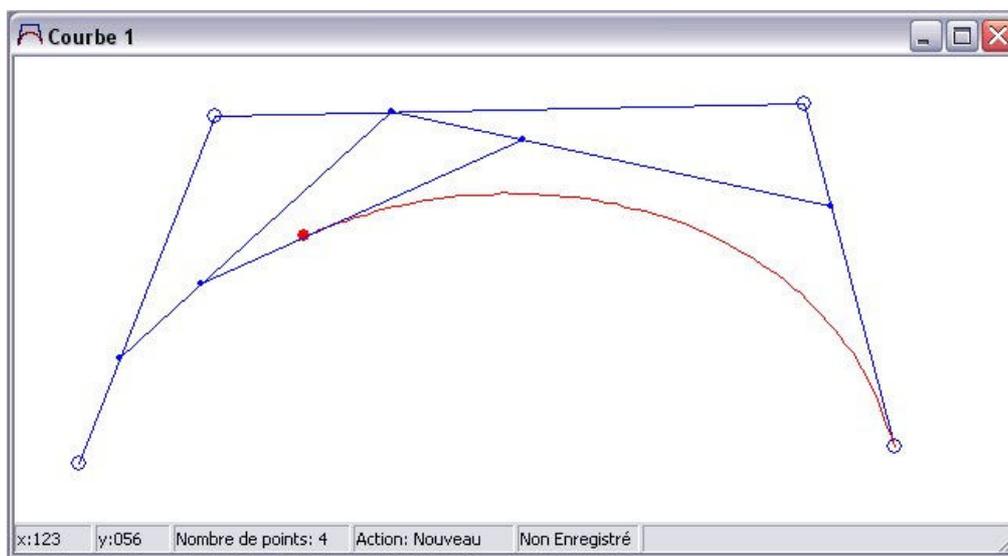
Une fois l'action active, des carrés apparaissent sur les traits qui relient les points du polygone de contrôle, il faut cliquer sur un des carrés où vous voulez insérer un point, il apparaît un point entourer d'un carré, vous pouvez déplacer ce point car après l'insertion l'action active est celle du Déplacement.

#### 6) Annuler la dernière action

Cette fonction vous permet d'annuler la dernière action effectuée.

Pour annuler la dernière action, il faut cliquer sur **Edition** dans la barre de Menu (schéma 1.1 - 1) puis sur **Annuler** on peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente **une flèche qui se retourne** (schéma 1.2 - 8), il y a aussi un raccourci clavier **F9**.

#### 7) Animation



Cette fonction permet de tracer en temps réel la courbe de Béziérs. Pour animer le tracer de la courbe, il faut tout d'abord réaliser une courbe, puis cliquer sur **Action** dans la barre de Menu (schéma 1.1 - 1) puis sur **Animation**.

## B) Fonctions liées à l'affichage de la courbe

### 1) Régler le pas



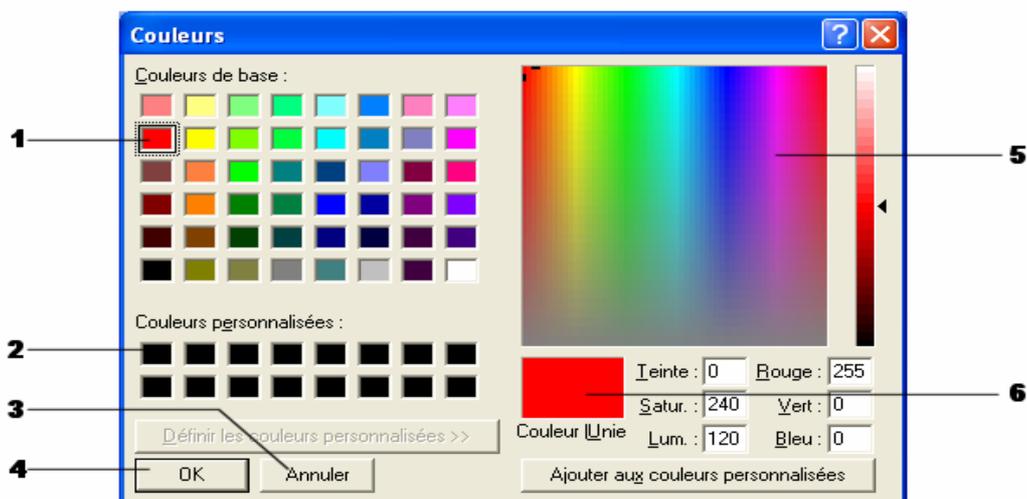
2.1 – Schéma de la boîte de réglage du pas.

1. Zone de saisie du pas compris entre 0.005 et 1.
2. Bouton d'annulation.
3. Bouton de confirmation.

Cette fonction permet de définir la précision de la courbe, plus le pas est élevé moins la courbe est lisse

Pour régler le pas, il faut cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Régler le pas**, une fenêtre apparaît. Changer la valeur de la zone de saisie (schéma 2.1 - 1) en cliquant dessus puis entrer une nouvelle valeur comprise entre 1 et 0.005, enfin confirmer avec le bouton OK (schéma 2.1 - 3).

### 2) Réglage des couleurs



2.2 – Schéma de la boîte de réglage des couleurs.

1. Couleurs préenregistrées.
2. Couleurs personnelles enregistrées.
3. Bouton d'annulation.
4. Bouton de confirmation.
5. Palette de couleurs.
6. Couleur qui a été choisie.

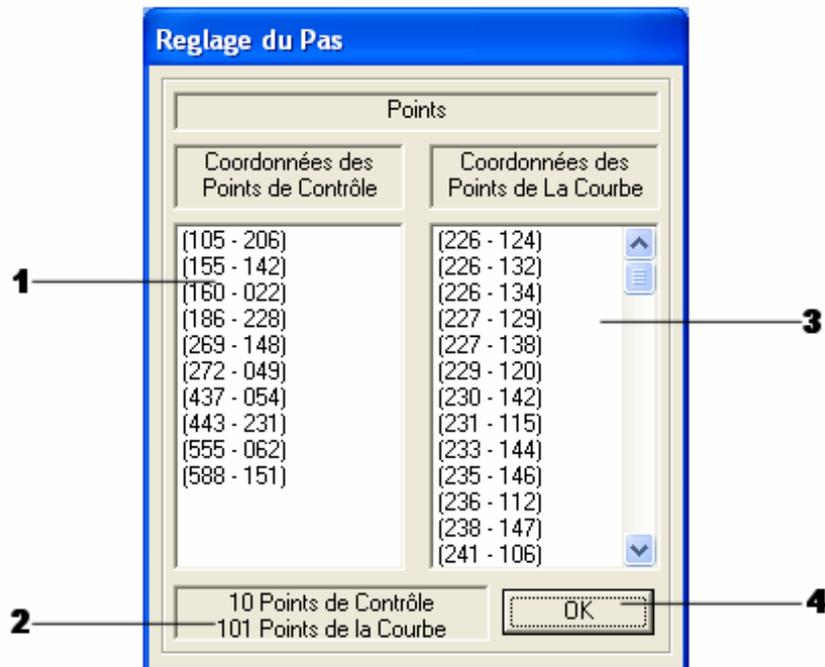
Il est possible de régler plusieurs composantes du dessin. On peut régler la couleur de la courbe, il faut cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Couleur de la courbe**. On peut aussi changer la couleur des points, il faut cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Couleur des points**. Enfin on peut changer la couleur du polygone de contrôle, il faut cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Couleur du polygone**.

Pour ces trois réglages il apparaît la boîte de réglage des couleurs (schéma 2.2). Il faut choisir une couleur parmi celles qui existe déjà (schéma 2.2 - 1) ou dans la palette (schéma 2.2 - 5), la couleur choisie est affichée (schéma 2.2 - 6) pour valider le choix de la couleur cliquer sur le bouton OK.

### 3) Affichage du polygone

Il est possible d'afficher ou de masquer le polygone de contrôle. Pour cela cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Afficher/Masquer le polygone**.

### 4) Coordonnées des points



2.3 – Schéma de la boîte des coordonnées des points

1. Liste des coordonnées des points de contrôle.

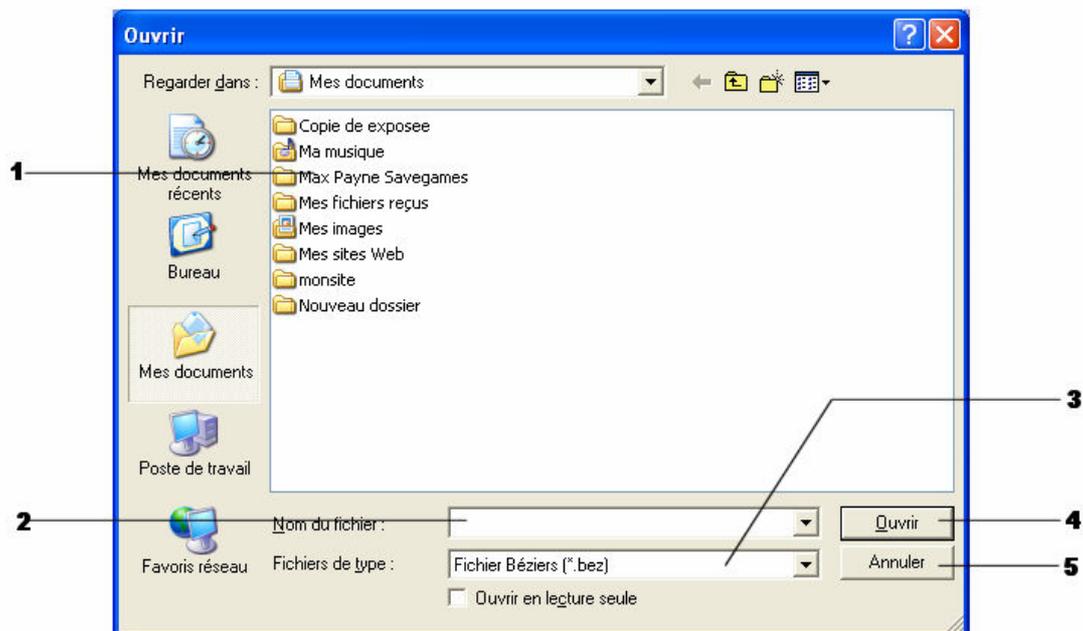
2. Informations sur le nombre de points de contrôle et de la courbe.
3. Liste des coordonnées des points de la courbe.
4. Bouton de validation.

Cette fonction permet d'afficher les coordonnées des points de contrôle et de la courbe. Pour afficher les coordonnées des points de contrôle et celles des points de la courbe, il faut cliquer sur **Propriétés** dans la barre de Menu (schéma 1.1 - 1) puis sur **Coordonnées des points**.

### C) Sauvegarde et chargement

Il est possible de sauvegarder les courbes qui ont été créées, elles sont sauvegarder au format Béziers l'extension qui correspond est **.bez** .

#### 1) Ouvrir



2.4 – Schéma de la boite d'ouverture de fichiers

1. Explorateur de fichier.
2. Zone pour taper le nom du fichier.
3. Type du fichier qui va être ouvert.
4. Bouton qui permet d'ouvrir le fichier.
5. Bouton d'annulation.

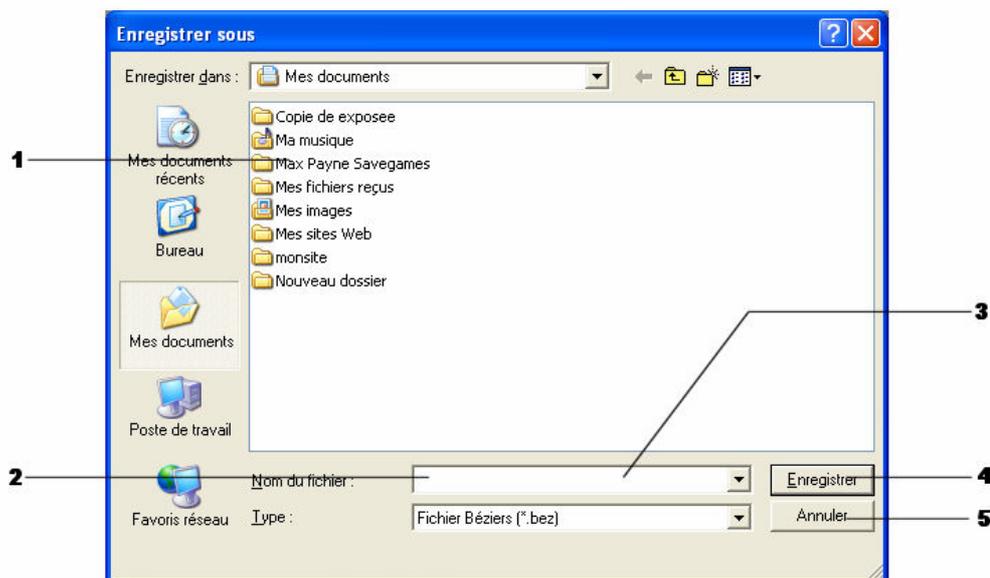
Pour ouvrir une courbe en fichier Béziers, cliquer sur **fichier** dans la barre de Menu (schéma 1.1 - 1) puis sur **Ouvrir**, une boite de dialogue apparaît. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente une **un dossier** (schéma 1.2 - 4). Il existe aussi un raccourci **F2**.

#### 2) Enregistrer

Pour enregistrer une courbe en fichier Béziérs, cliquer sur **fichier** dans la barre de Menu (schéma 1.1 - 1) puis sur **Enregistrer**, une boîte de dialogue apparaît. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente **une disquette** (schéma 1.2 - 5). Il existe aussi un raccourci **F3**.

Pour pouvoir utiliser **Enregistrer**, il faut que la courbe soit au préalable enregistrer avec un nom, avec la commande **Enregistrer sous** (ci-après). Pour savoir si la courbe est déjà enregistrée il faut regarder la barre de statut (Schéma 1.3), il est indiquier si la courbe est enregistrée ou non (Schéma 1.3 – 4), si la courbe est enregistrer son nom est indiquier (Schéma 1.3 – 5).

### 3) Enregistrer sous...



2.5 – Schéma de la boîte de sauvegarde.

1. Explorateur de fichier.
2. Zone pour taper le nom du fichier.
3. Type du fichier qui va être enregistré.
4. Bouton qui permet d'enregistrer le fichier.
5. Bouton d'annulation.

Pour enregistrer une courbe en fichier Béziérs, cliquer sur **fichier** dans la barre de Menu (schéma 1.1 - 1) puis sur **Enregistrer sous**, une boîte de dialogue apparaît. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente **une disquette** (schéma 1.2 - 2). Il existe aussi un raccourci **F4**.

## D) Autre

### 1) Aide

L'aide de Béziérs® 2003 reprend cette notice d'utilisation sous forme HTML, elle nécessite un explorateur Internet comme Internet Explorer de Microsoft ou encore Netscape.

Pour accéder à l'aide, cliquer sur le **point d'interrogation « ? »** de la barre de Menu (schéma 1.1 - 1) puis sur **Aide**. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente **un livre** (schéma 1.2 - 10). Il existe aussi un raccourci **F11**.

## 2) A propos...

A propos reprend les informations relatives au logiciel.

Pour accéder à l'aide, cliquer sur le **point d'interrogation « ? »** de la barre de Menu (schéma 1.1 - 1) puis sur **A propos...**. On peut aussi utiliser la barre d'outils en cliquant sur l'icône qui représente **un point d'interrogation** (schéma 1.2 - 11). Il existe aussi un raccourci **F10**.

## *CONCLUSION*

Comme nous l'avons montré tout au long de ce Rapport Final, nous avons réussi à atteindre les objectifs que nous nous étions fixés au début du développement de Béziérs<sup>®</sup> 2003, nous avons même réussi à développer deux des quatre options que nous avons proposé. Cependant, Béziérs<sup>®</sup> 2003 pourrait encore être amélioré, en effet, il n'est pour l'instant pas possible de tracer plusieurs courbes dans la même fenêtre, on pourrait donc envisager un système de calques pour permettre le tracé d'une courbe par calques dans la même fenêtre. De nombreuses possibilités de développement futures sont envisageables pour Béziérs<sup>®</sup> 2003, tout d'abord les deux options que nous n'avons pas pu développer, c'est-à-dire l'enregistrement au format .bmp, qui peut être utile pour pouvoir réutiliser l'image d'une courbe dans un autre logiciel ; et l'affichage des dérivées des courbes tracées. On peut aussi envisager des améliorations plus élaborées, comme l'affichage de surfaces de Béziérs, ou de courbes de Béziérs en 3D.

Ce projet nous a permis de nous familiariser avec la gestion d'un projet sur plusieurs mois, en nous poussant à nous répartir les tâches, et à fournir un travail régulier tout au long de l'année. Selon nous, aux vues du résultat obtenu, nous pensons avoir réussi à travailler en équipe en respectant un calendrier, ce qui est une bonne expérience pour nos projets futurs, notamment pour notre projet de deuxième année.

---

*DAMOUR Olivier*



*ROLLET Samuel*



*GUILLET Cyril*

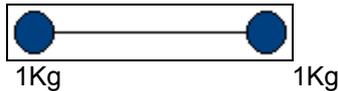


## *ANNEXE 1*

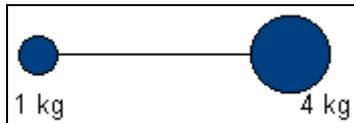
### *Qu'est-ce qu'une courbe de Béziers*

Nous allons ici expliquer les notions de barycentres et de courbes de Béziers de manière visuelle et intuitive, pour les rendre compréhensibles à des non mathématiciens.

#### **- Notion de Barycentre**

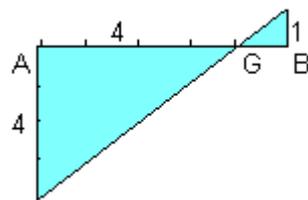


Sur une tel 'haltère', le point d'équilibre se situ au milieu du segment.



Sur celle-ci, le point d'équilibre est décalé vers la droite.

Un barycentre peut donc être considéré comme le point d'équilibre entre deux autres point auquel on attribut des poids : leur pondération.



Soit le point A avec une pondération égale à 4 et le point B avec une pondération égale à 1, on trouve le barycentre G entre (A, 4) et (B, 1) de manière visuelle, comme présenté ci-dessus.

#### **- Notion de Courbes de Béziers**

Une courbe de Béziers est une courbe définie par un nombre limité de points, les points de contrôle. C'est par construction successive de barycentres à partir de ces points de contrôles que l'on obtient un arc de courbe, dont les extrémités se situent sur le premier et le dernier point de contrôle, c'est une courbe de Béziers.

Le Pas est un élément important dans le calcul d'une courbe de Béziers, il représente la précision du tracé de la courbe, c'est la variation de la pondération de chaque points à chaque calculs.

Nous allons illustrer cette méthode de calcul dans ce qui suit.

#### **Courbe de Béziers avec deux points de contrôle A et B**

On commence par calculer le barycentre entre (A, 1) et (B, 0), on obtient le premier point de la courbe de Béziers, qui est confondu avec A.

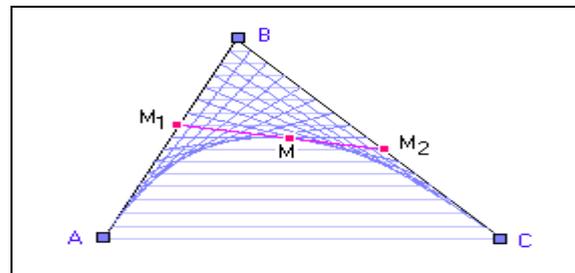
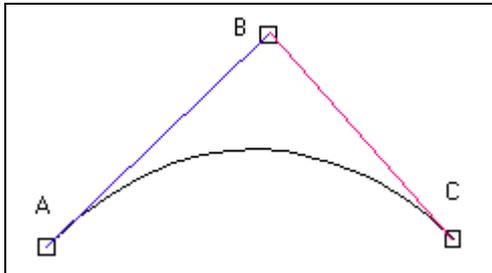
On fait ensuite varier les pondérations de A et B, on soustrait la valeur du Pas de la pondération de A, et on l'ajoute à la pondération de B. Pour un Pas égale à 0.1, on aurai donc les points (A, 0.9) (B, 0.1). On calcule ensuite le barycentre entre ces points.

On répète cette opération jusqu'à avoir  $(A, 0)$   $(B, 1)$ , et obtenir un barycentre confondu avec B.

Pour une courbe de Béziérs définie par deux points, on obtient un segment confondu avec le segment entre les deux points de contrôle.

Une telle courbe, définie par deux points, est dite courbe d'ordre 1.

### Courbe de Béziérs avec 3 points de Contrôles



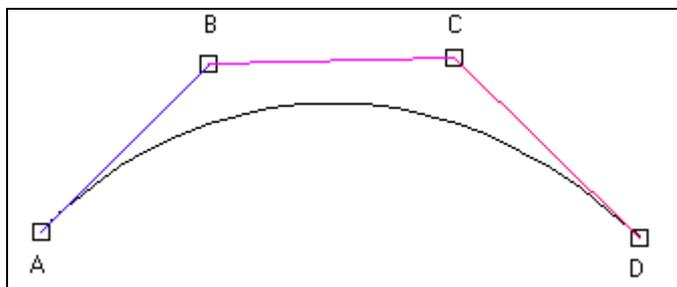
Soit une courbe définie par trois points de contrôles A, B, C.

Pour chaque couples de points successifs : A, B et B, C, on calcule les barycentres  $M_1$  de A B, et  $M_2$  de B C, depuis  $(A, 1)$   $(B, 0)$  et  $(B, 1)$   $(C, 0)$ , jusqu'à  $(A, 0)$   $(B, 1)$  et  $(B, 0)$   $(C, 1)$ , en faisant varier entre chaque calculs les pondérations des points de la valeur du pas.

Pour chaque valeur de  $M_1$   $M_2$ , on calcule leur barycentre M avec les mêmes pondérations que les couples A B, et B C.

Chaque valeur de M est un point de la courbe de Béziérs définie par 3 points. Une telle courbe est dite d'ordre 2.

### Courbe de Béziérs avec 4 points de Contrôles



Pour une courbe définie par 4 points de contrôles, in est nécessaire de rajouter une étape de calcul.

Le calcul des barycentres entre les couples A B, B C et C D, donne 3 points  $M_1$ ,  $M_2$ ,  $M_3$ . Le calcul des barycentres entre les couples  $M_1$   $M_2$ , et  $M_2$   $M_3$ , donne 2 points  $N_1$  et  $N_2$ , il faut donc calculer le barycentre M de  $N_1$   $N_2$ , pour toutes les valeurs de  $N_1$   $N_2$ .

### Généralisation

Cette construction peut être poursuivie jusqu'à n'importe quel ordre n. C'est l'Algorithme de De Casteljaou. On obtient alors une Courbe de Béziérs d'ordre n, à  $n+1$  points de contrôle. Retrouvez en Annexe 2 la traduction de l'Algorithme de De Casteljaou en langage C.

## ANNEXE 2

### *Fonction de Calcul d'une Béziere*

### *Algorithme de De Casteljaou*

**Structure** contenant toutes les informations nécessaires sur une courbe et la fenêtre qui la contient, cette structure est utilisée pour créer une liste chaînée de toutes les courbes créées successivement:

```
typedef struct _DONNEESFENETRE
{
    int numero_courbe;           //Numéro identifiant la courbe, 0 si la fenêtre a été fermée
    POINT PtControle[20];       //Tableau des coordonnées des pts de contrôle
    int NbPtControle;           //Nombre de pts de contrôle
    float pas;                   //Pas (précision) de la courbe
    int affichage_poly;         //Affichage ou non du polygone de contrôle : 0 ou 1
    HWND hwndChildWnd;          //handle de la fenêtre fille contenant la courbe
    char FileName[1000],        //Chemin du fichier ou est enregistré la courbe (si elle l'est)
        TitleName[100];         //Nom du fichier d'enregistrement
    ETAT_ENREGISTREMENT deja_enregistre; //indique si la courbe est non enregistrée,
                                    // enregistrée, ou modifiée
    COLORREF Couleur_Courbe;    //couleur de la courbe
    COLORREF Couleur_Points;    //couleur des points de contrôle
    COLORREF Couleur_Poly;      //couleur du polygone de contrôle
    HWND hStatusbar;            //Pointeur sur la statusbar de la fenêtre fille
    struct _DONNEESFENETRE *suivant;
}DONNEESFENETRE;
```

---

**Variables globales** utiles au calcul d'une courbe de Béziere.

```
int    NbPtCourbe=0;           //Nombre de points de la courbe de la fenêtre affichée
POINT PtCourbe[100000];        //Coordonnées des points de la courbe
```

---

**Procédure** de calcul des coordonnées des points d'une Béziere en fonction des coordonnées des points de contrôle, suivant l'algorithme de De Casteljaou.

```
void CalculPtCourbe(DONNEESFENETRE *fenetre_focus)
//fenetre_focus pointe sur une structure DONNEESFENETRE contenant toutes les informations sur
//la courbe contenue dans la fenêtre fille qui a le focus (qui est au dessus)
{
    int NbTours,tours,NbPtsTab,InveauPt=0,i,j;
    float pond;
    POINT pt1[1000];           //tableaux pour stoker les coordonnées des barycentres
    POINT pt2[1000];           //intermédiaires calculées pour obtenir un point

    NbTours=fenetre_focus->NbPtControle-1;
    //Il sera nécessaire d'effectuer autant de calculs successifs moins 1, que le nombre de points
    //pour obtenir les coordonnées d'une point de la courbe

    NbPtCourbe=0; //On remet la variable globale à 0

    for(pond=0,j=0;pond<=1;j++)
    //pour chaque variations de pondération, on calcule un point de la courbe
    {
        for(i=0;i<fenetre_focus->NbPtControle;i++)
```

```

//On recopie les coordonnées des pts de controle ds le tableau impaire pt1
{
    pt1[i]=fenetre_focus->PtControle[i];
}
NbPtsTab=fenetre_focus->NbPtControle;
//Le nombre de points de tab1 est égale au nombre de points recopiés
for(tours=1;tours<=NbTours;tours++)
//on effectue les calculs successif des barycentres intermédiaire
{
    for(i=0;i<NbPtsTab-1;i++)
    //On calcule les barycentres entre chaque couples de point successifs
    {
        //suivant la parité du tours, on stocke les coordonnées trouvées dans
        //le tableau correspondant
        if(tours%2==0)
        {
            pt1[i].x=((int) pt2[i].x*pond+pt2[i+1].x*(1-pond));
            pt1[i].y=((int) pt2[i].y*pond+pt2[i+1].y*(1-pond));
        }
        if(tours%2!=0)
        {
            pt2[i].x=pt1[i].x*pond+pt1[i+1].x*(1-pond);
            pt2[i].y=pt1[i].y*pond+pt1[i+1].y*(1-pond);
        }
    }
    NbPtsTab--;
    //à chaque calculs, les nombre de points trouvées diminuent de 1, jusqu'à
    //ce qu'il n'en reste plus qu'un, qui est un point de la courbe
}
//quand on a effectué tous les tours de calculs successifs, on récupère le dernier
//point calculé, qui est un point de la courbe
//suivant la parité du nombre de tours effectués, le pt à récupérer est dans le
//tableau 1 ou 2
if((tours-1)%2==0)
{
    PtCourbe[j]=pt1[0];
}
if((tours-1)%2!=0)
{
    PtCourbe[j]=pt2[0];
}
NbPtCourbe++;
//on augmente le nombre de points de la courbe

pond=pond+fenetre_focus->pas;
//on augmente la pondération pour le prochain calcul, avec la valeur du pas
//de la courbe
}
}

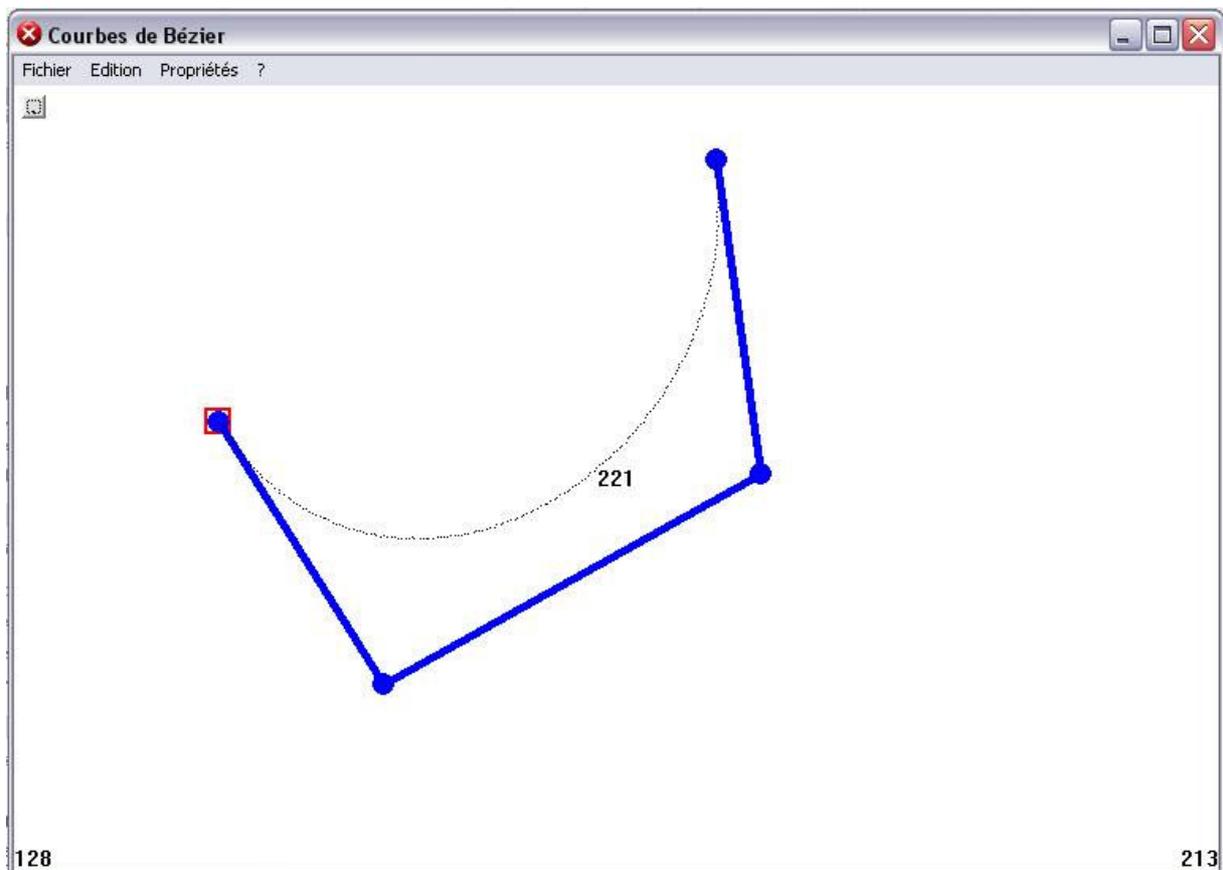
```

### ANNEXE 3

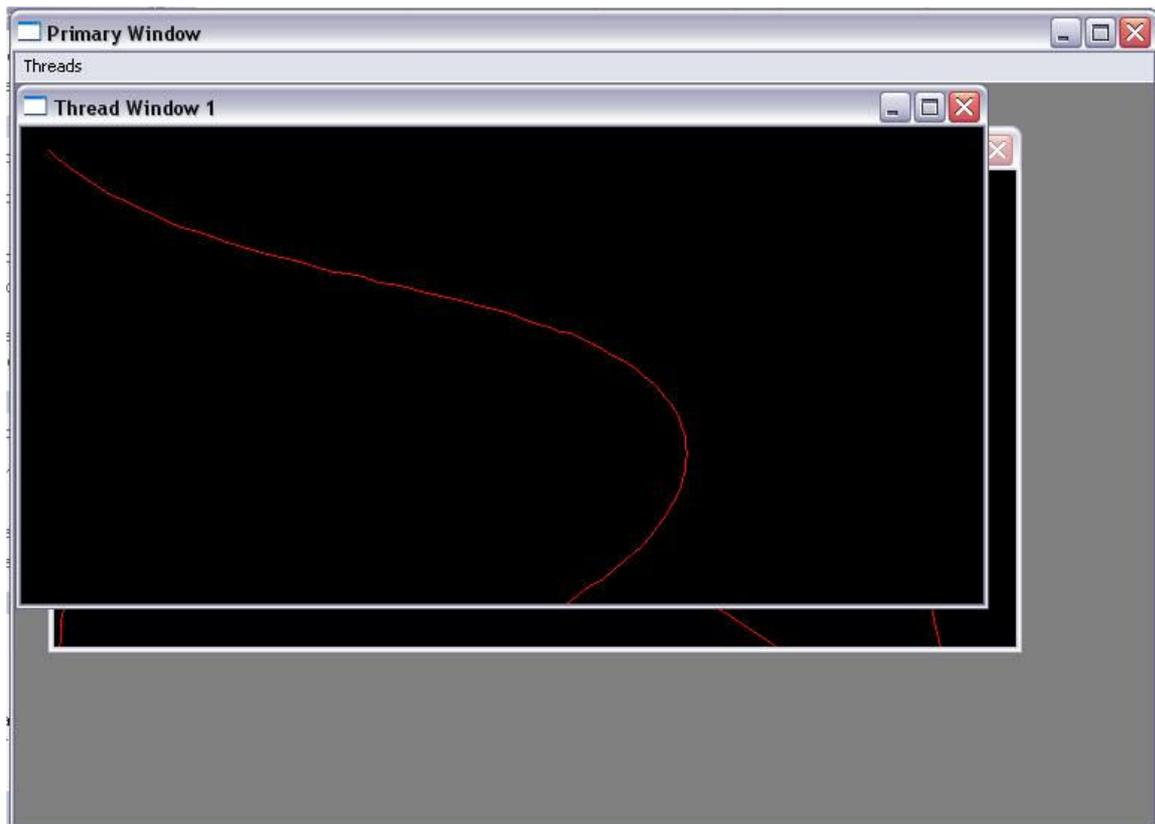
Retrouvez ici l'évolution visuelle du projet Béziérs® 2003 de ses début, jusqu'à aujourd'hui.



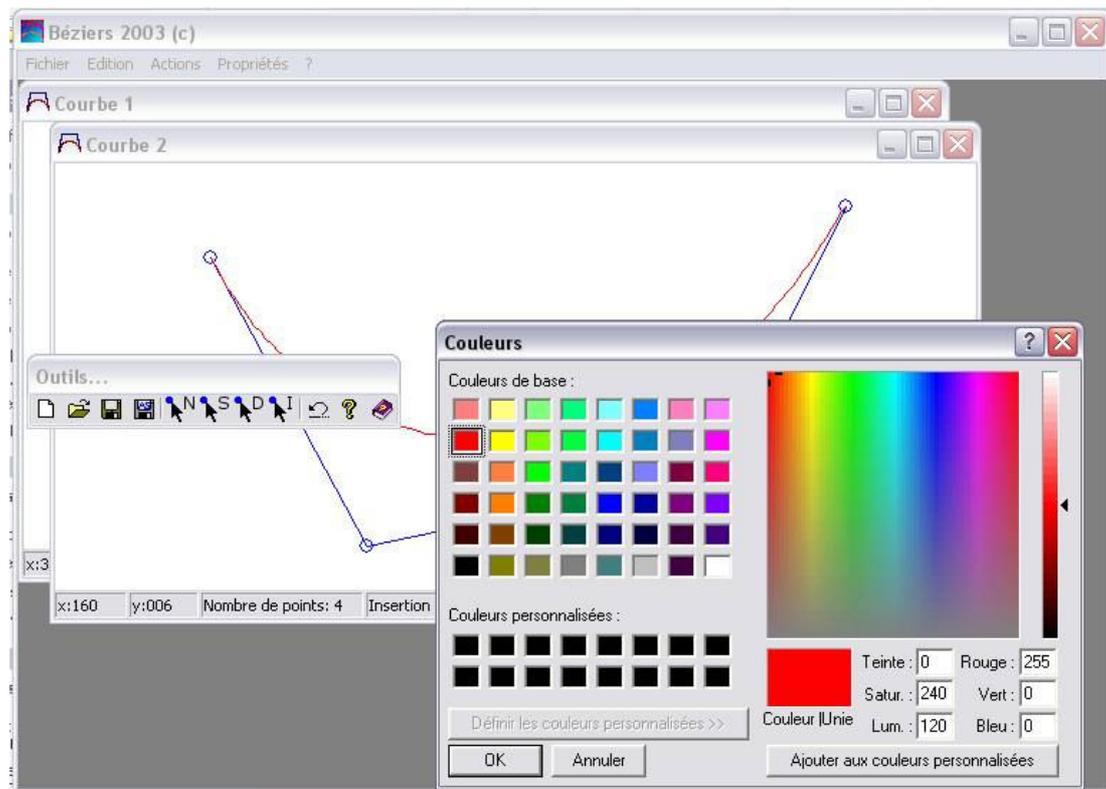
Visuel début Février 2003 - Calcule d'une Béziérs suivant des Points de contrôle.



Visuel fin Février 2003 - Premiers Tracés, et déplacement de points



Visuel en Mars 2003 – Premières Béziers en mode MDI



Visuel début Avril 2003 – De nombreuses options ont été développés après l'obtention de ce visuel

## ***BIBLIOGRAPHIE***

Voici une liste non exhaustive des livres et sites Internet qui nous ont aidé pendant notre développement.

Livres Utilisés Durant le Développement de Béziérs® 2003 :

The Nurbs Book - 2<sup>nd</sup> Edition - Les Piegl et Wayne Tiller - Ed. Springer - 1997

Le Langage C - Jean-Michel Léry - Ed. Campus Press - 2002

Langage C - Cours 1A - Véronique Deslandres

Programmer sous Microsoft Windows - 5<sup>ème</sup> Edition - Microsoft Press

Sites Internet Utilisés Durant le Développement de Béziérs® 2003 :

<http://www2.ac-lille.fr/math/> Académie de Lille Maths Sciences en LP

<http://www.programmationworld.com> Programmation World

<http://www.cppfrance.com/> CPP France