

XARA, lorsque les graphiques vectoriels se font libres

Charles Moir

Xara a récemment décidé de rendre le code source de Xara Xtreme disponible et libre en le distribuant sous la licence GPL. Il s'agit d'une orientation radicale pour cette société, puisque c'est la première fois qu'un produit graphique commercial de cette qualité et de cette complexité (code source supérieur à 1 million de lignes) devient libre.

Xara Xtreme est un programme de graphiques vectoriels extrêmement performant (le plus rapide du marché en réalité). Ce logiciel a été développé sur plus de 10 ans. Il a débuté avec l'ancien ordinateur Acom RISC PC (basé sur ARM) au début des années 90, puis a été porté sur Windows en 1995. Nous venons de débiter le processus de portage du logiciel vers Linux et Mac, et nous envisageons de sortir un code libre disponible sous une licence GPL.

Xara Xtreme est la toute dernière version sortie pour Windows. La version Linux du logiciel correspondant au code source libre, en cours de création, portera le nom de Xara LX.

Mémoire image et graphiques vectoriels

Il existe généralement deux classes de logiciels graphiques : les logiciels destinés aux mémoires images ou mappes de pixels (comme les photos), et ceux

destinés aux graphiques vectoriels, dont l'approche, définie de manière mathématique, est orientée objet. En règle générale, les graphiques vectoriels sont souvent associés à des graphiques très simples, de type dessins, mais les améliorations apportées aux capacités de rendu en font des graphiques riches et d'excellente qualité.

Résolution indépendante : interfaces graphiques dimensionnables

L'approche vectorielle présente un avantage considérable : la résolution indépendante des images. Il est en effet possible de rendre une image à n'importe quelle taille ou résolution, sans altérer sa définition. Bien évidemment, ceci est impossible avec les mémoires images.

Jusqu'à ce que les interfaces graphiques vectorielles se généralisent, les widgets d'affichage des systèmes d'exploitation (boutons, barres de défilement,

Sur l'auteur :

L'auteur est le fondateur et le PDG de Xara. Site Web <http://www.xaraxtreme.org>

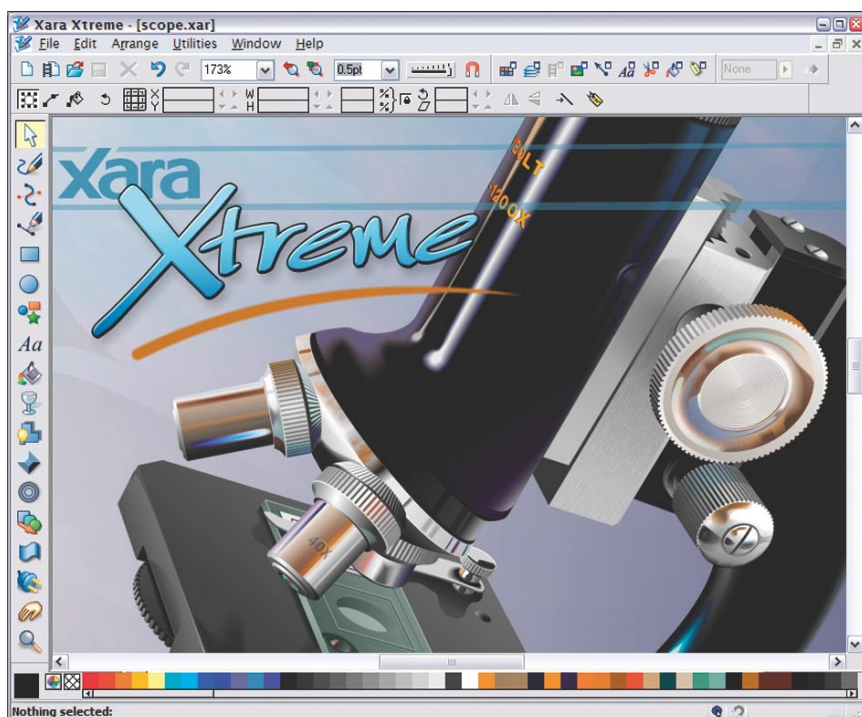


Figure 1. Xara Xtreme sous Windows affichant un dessin très détaillé d'un microscope

icônes, etc.) seront encore dessinés en mode bitmap selon des tailles spécifiques. Ainsi, plus haute est la résolution de votre bureau, plus petits seront les widgets obtenus. Avec les écrans 300 dpi LCD en cours de développement, il est désormais nécessaire d'abandonner cette approche de taille définie en mode bitmap pour adopter une nouvelle approche vectorielle entièrement dimensionnable. C'est pourquoi l'ensemble des fabricants de système d'exploitation optent désormais pour des graphiques vectoriels dans leurs interfaces graphiques : c'est le cas de Microsoft avec Avalon (désormais appelé la Windows Presentation Foundation), de Linux avec SVG/Cairo/Glitz ou d'Apple avec Quartz Extreme, pour ne nommer que les plus importants sur le marché.

Objets graphiques

Les graphiques vectoriels présentent un autre avantage qui leur permet de se différencier des graphiques en mode bitmap : l'ensemble des éléments composant votre dessin reste entièrement éditable à n'importe quel moment. Les dessins se composent de centaines, parfois de milliers d'objets indépendants, lesquels demeurent tous éditables. Là encore, ceci est impossible en mode bitmap, où chaque image est traitée

comme un tout, notamment les formats les plus répandus comme JPG ou PNG.

En quoi le logiciel Xara se distingue-t-il des autres programmes vectoriels ?

Xara comporte quelques facteurs clés à l'origine de son succès et de sa qualité.

Vitesse

Le premier de ces facteurs est la vitesse. La vitesse de Xara a été optimisée depuis le début du développement. Ce moteur de rendu est de loin le plus rapide du marché. C'est une capacité d'une grande importance pour les programmes graphiques, traditionnellement assez lents.

Utilisation simplifiée

Nous appliquons, chez Xara, une politique de simplification afin de minimiser les boîtes de dialogue, les boutons et les palettes d'outils. Ce qui permet également de réduire les supports. Xara applique plus particulièrement à ses outils une approche directe sur toile. Par exemple, alors que la plupart des programmes de dessins vous permettent d'indiquer un type de remplissage gradué au moyen d'une boîte de dialogue, (dans laquelle vous entrez des valeurs numériques ou disposez de modeste

commande de contrôle), Xara vous permet de réaliser ce type de tâches directement sur la toile, en glissant tout simplement sur l'objet où vous souhaitez voir débuter et terminer le remplissage gradué. Cette façon de travailler directement sur la toile est bien plus confortable, selon de nombreux utilisateurs, et permet de réduire le nombre de boîtes de dialogue nécessaires.

Prévisualisation directe

Xara a également lancé la nouvelle approche de prévisualisation directe, dans laquelle la plupart des commandes ne présentent plus de boutons OK auparavant nécessaires à l'application d'une modification. Désormais, le document se met à jour après avoir ajusté la commande (vous entendrez beaucoup parler de la prévisualisation directe puisque la prochaine génération de Microsoft Office va intégrer cette fonctionnalité dans ses produits). Cette prévisualisation directe s'applique également à la manipulation d'objets primaires. Il est donc possible de faire glisser des objets pour les redimensionner, les faire tourner ou les déplacer sur la page. Par ailleurs, contrairement à la plupart des autres programmes qui se contentent d'afficher les contours des objets, Xara est suffisamment rapide pour pouvoir afficher l'objet solide en question, avec les transformations alors que vous le faites glisser.

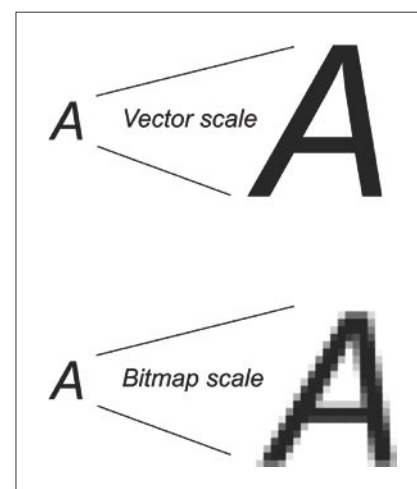


Figure 2. Au moment de dimensionner des graphiques vectoriels, tous les éléments conservent une précision de très haute qualité. À l'inverse, tenter d'agrandir des graphiques en mode bitmap ne produira jamais de résultats satisfaisants

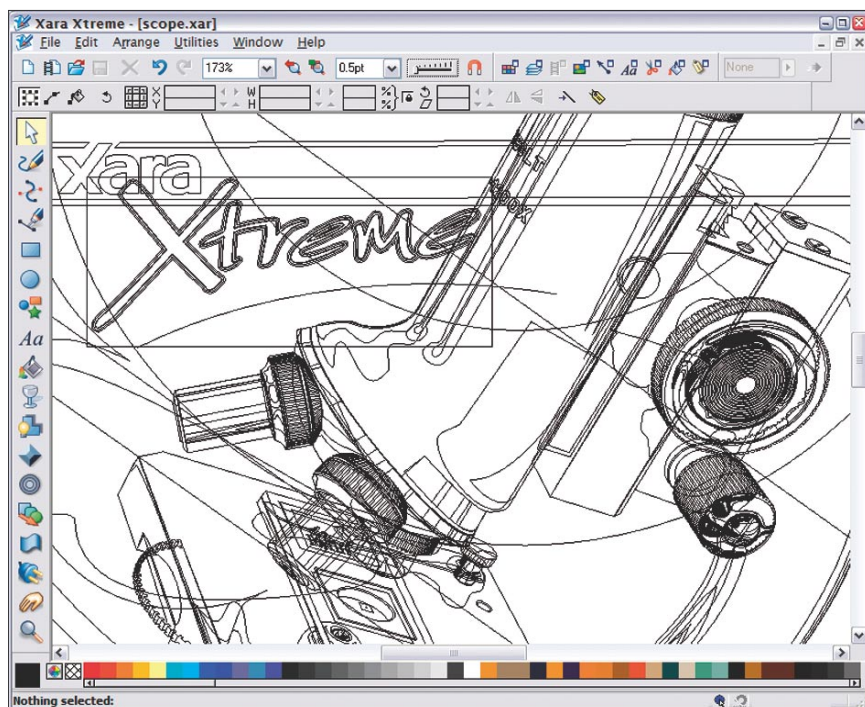


Figure 3. Même dessin exposé précédemment affichant maintenant les contours de chaque objet composant le dessin

Interface utilisateur non-modale

Autre exemple venant illustrer l'utilisation simplifiée du produit, l'utilisation de boîtes de dialogue non-modales à chaque fois que c'est possible. La boîte de dialogue intitulée *Colour Editor* peut donc rester affichée à l'écran et se mettre à jour afin d'afficher la couleur de n'importe quel objet sélectionné, tout en vous permettant d'ajuster cette couleur et poursuivre votre travail. La plupart des autres programmes ont recours à des boîtes de dialogue modales, généralement plus faciles à programmer. Ces boîtes bloquent toutefois l'accès à tous les autres composants du programme tant qu'elles restent à l'écran, et vous obligent à confirmer en premier lieu vos modifications (généralement au moyen d'un bouton OK).

Avec une prévisualisation directe et des boîtes de dialogue non-modales, vous êtes en face d'un environnement de travail extrêmement plus productif que la moyenne.

Graphiques détaillés

Les capacités graphiques proposées par un système d'exploitation ont toujours été relativement rudimentaires. Même récemment, ces dernières ne proposaient même pas de fonctionnalité anti-crênelage, de sorte que vos dessins présentaient ces fameux effets de bords *dentelés*. Là encore, Xara innove en proposant une nouvelle

technique d'anti-crênelage vectorielle. Afin de fonctionner, nous recommandons de remplacer intégralement le moteur graphique du système d'exploitation utilisé par le propre moteur de Xara, appelé CDraw.

Grâce à ce nouveau moteur de rendu que nous avons créé chez Xara, il devient alors possible d'introduire bien plus de nouveaux types de graphiques vectoriels détaillés, comprenant un large choix de

type de remplissage, la transparence vectorielle, et même la transparence graduée.

Par ailleurs, nous venons d'introduire le tout premier dégradé vectoriel et les premiers ombrages vectoriels nuancés sur le marché. Ces fonctionnalités vont permettre de créer des graphiques vectoriels plus réalistes et extrêmement détaillés, aux antipodes de cet aspect classique de *cartoon* propre aux anciennes générations de graphiques vectoriels.

Architecture de Xara

Xara est écrit en C++. Il s'agit d'un programme orienté objet capable de supporter un modèle graphique orienté objet.

Architecture orthogonale

Nous avons tenté d'élaborer une architecture capable de traiter tous les objets de manière égale. Ainsi, par exemple, il est possible d'appliquer n'importe quel attribut à tout type d'objet. Si vous décidez d'appliquer de la couleur à un objet bitmap, le programme comprend que vous souhaitez colorer tout le dessin bitmap.

Arborescence

Tous les objets composant un dessin sont stockés dans une structure en forme d'arborescence. Cette hiérarchie est parfaitement adaptée à l'organisation des objets. Le niveau supérieur débute par les chapitres et les pages, suivis des couches, puis des groupes d'objets aux niveaux

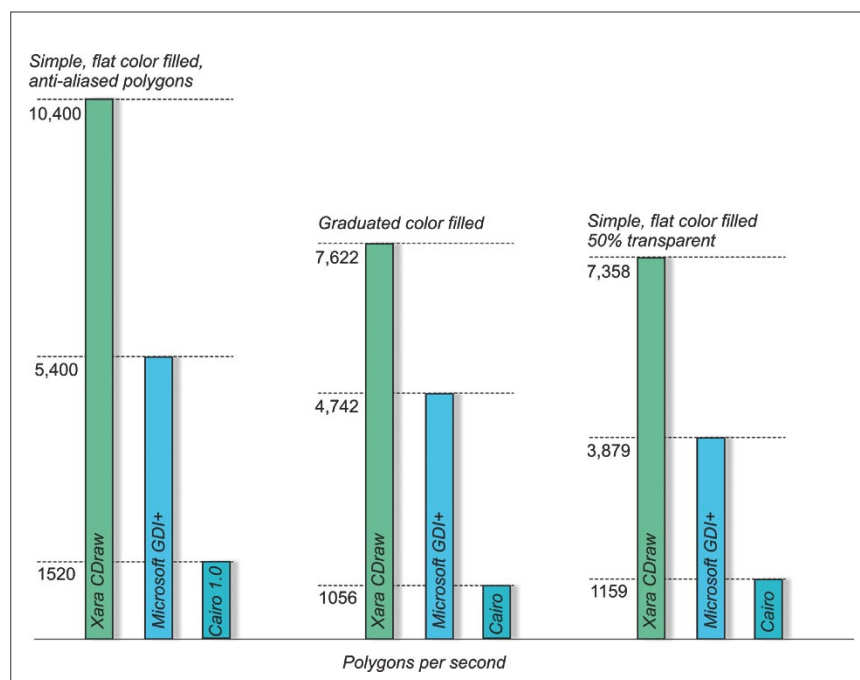


Figure 4. Comparaison des performances de CDraw de Xara, de GDI+ de Windows et de Cairo de Linux, en terme de rapidité de rendu.



Figure 5. Pas d'effet anti-crénelage (à gauche) et même caractère agrandi à 400%. Avec l'anti-crénelage (à droite), l'effet escaliers dentelés est réduit en se mélangeant à la couleur de l'arrière plan.

inférieurs. Cette arborescence contient également ce que nous appelons des *attributs*, structures de données chargées de définir l'apparence d'un objet (couleur, style de remplissage, épaisseur du tracé, valeur de dégradé, etc.).

(Remarque : vous serez sans doute surpris de trouver des références aux chapitres et aux pages dans la structure arborescente d'un programme censé n'être composé que du dessin d'une seule page. En réalité, Xara avait produit précédemment un produit micro-édition de type Quark (Impression) ainsi qu'un programme de dessin (Artworks) avec beaucoup de succès. Ce produit nouvelle génération a été conçu de sorte à pouvoir intégrer un jour les meilleures fonctionnalités de ces deux domaines que sont la micro-édition et les outils graphiques.)

Cette arborescence comporte des noeuds, tous étant dérivés de la classe principale `Node`. Chaque noeud représente un objet sur la page, y compris la page en question. Ces noeuds sont liés à tous leurs noeuds parents, frères et enfants. Naviguer dans l'arborescence est ainsi une tâche très simple et rapide.

Chaque objet est associé à un rectangle permettant de déterminer facilement et rapidement la zone de la page qu'il couvre. Cette fonction est d'autant plus importante pour pouvoir réaliser des mises à jour rapides de certaines régions de la page (au moment de défiler, ou d'éditer juste un objet, par exemple) ou détecter les objets sur lesquels l'utilisateur a cliqué.

Processus de rendu

Modèle de peinture

Presque tous les programmes de graphiques vectoriels reposent sur un modèle de rendu largement répandu de type Postscript. Son processus de rendu est parfois appelé modèle de peinture, parce que les objets sont 'peints' à l'écran, de l'arrière vers l'avant. En d'autres termes,

les dessins sont élaborés à partir de formes qui se chevauchent les unes sur les autres.

Tous les dessins vectoriels s'appuient sur un bloc de construction appelé *Bézier curve*, d'après le nom de son développeur, Pierre Bézier. Ces courbes mathématiques peuvent être 'retouchées', autrement dit un tracé épais est rendu

dans la courbe (parfois appelée chemin). Les formes de Bézier peuvent être également colorées. C'est dans ce domaine que Xara est remarquable, puisque ce programme possède une large gamme de styles de remplissage. Récemment, Adobe et d'autres producteurs ont tenté de supporter la transparence fondamentale (encore introduite par Xara il y a plus de dix ans), qui permet d'afficher selon un degré variable des objets placés sous une forme transparente. Xara va plus loin en supportant la transparence graduée, de manière à faire varier le degré de transparence au travers de la forme.

Une seule forme vectorielle peut prendre une couleur graduée selon une direction différente indépendamment de la transparence graduée.

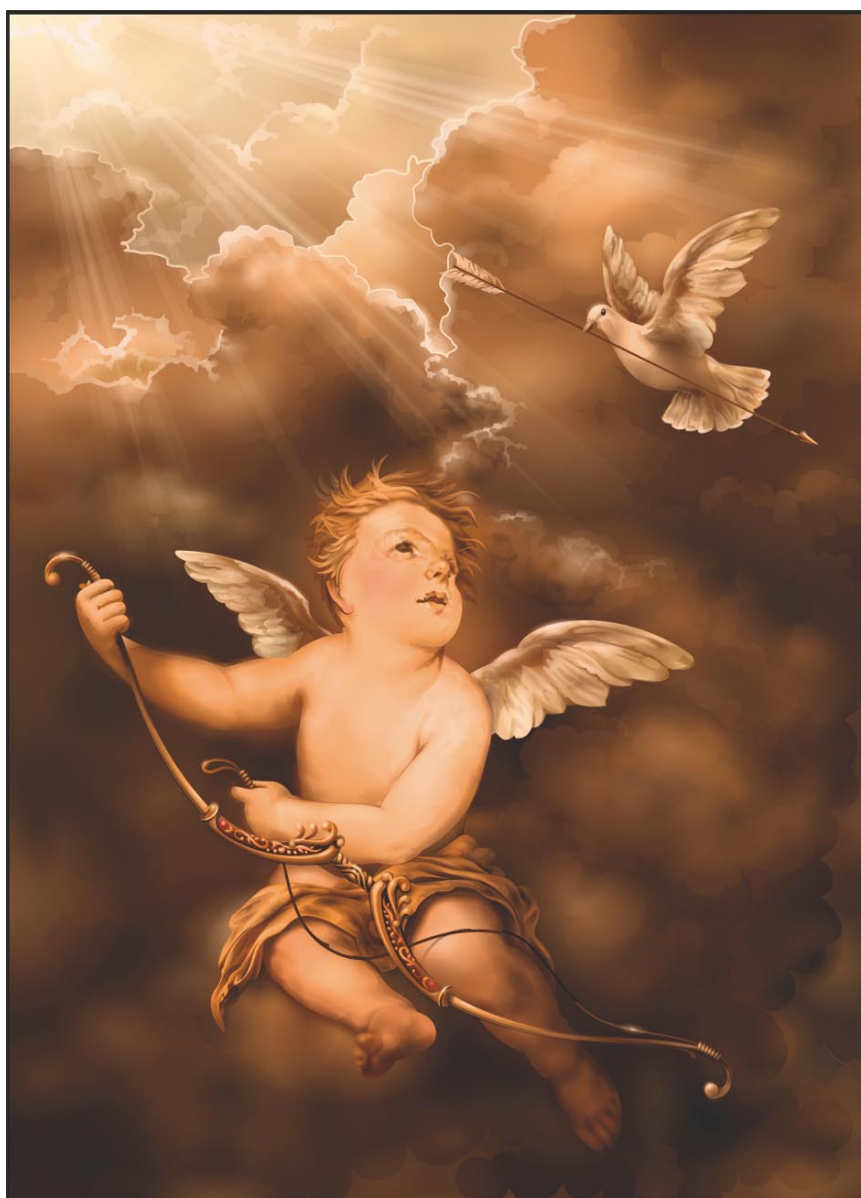


Figure 6. Cette illustration n'a pas l'air vectorielle, et pourtant !



Figure 7. Appliquer l'attribut remplissage rouge sur une forme simple produit des résultats évidents. Appliqué à un objet photo, le même attribut est compris comme contour rouge. Grâce à sa nature orthogonale, il est possible de réduire les cas spéciaux pour pouvoir utiliser la même interface utilisateur sur tous les types d'objets.

Compilation des dessins

Le processus de rendu analyse l'arborescence, en compilant les attributs appropriés pour chaque objet, pour rendre cet objet en bitmap hors écran. Le rendu des objets est réalisé par le moteur de rendu du niveau inférieur, CDraw. Toutes les demies secondes, l'image hors écran est copiée à l'écran. Ce processus peut être interrompu, afin de permettre à l'utilisateur de poursuivre son travail sur le document et réaliser des ajustements au moment du rendu, sans avoir à attendre la fin du processus de rendu (bien qu'en pratique, le rendu est si rapide que vous n'avez pas à attendre, ou très rarement).

CDraw

Tous les fabricants de systèmes d'exploitation se sont mis à proposer des primitives graphiques assez similaires à celles de Postscript dans leurs couches graphiques natives. C'est le cas, par exemple de GDI+ sous Windows (et maintenant d'Avalon, dénommé Windows Presentation Foundation pour Microsoft Vista), de Cairo sous Linux et de Quartz sous Mac. Toutes ces primitives supportent des courbes très semblables aux courbes de Bézier élémentaires, le retouchage de chemin et deux types de remplissages de couleur gradués (linéaire et circulaire), une fonctionnalité de transparence fondamentale, et même l'anti-crénelage, depuis peu. Toutefois, la performance de

ces moteurs de rendu laisse encore beaucoup à désirer. Comme nous venons de le mentionner, le moteur Xara propose un choix bien plus fourni en terme de types de remplissage et de transparence, ainsi que de nombreuses autres fonctionnalités comme le dégradé et les ombrages nuancés vectoriels.

Le rendu des formes reste le principal écueil de performance. C'est pourquoi le

moteur CDraw a été originellement rédigé dans un langage d'assemblage par souci d'efficacité. Puis au cours des années, il a été réécrit pour être porté sur les plateformes C. Ce moteur de rendu compose essentiellement de nouvelles formes en bitmap hors écran, en appliquant le niveau d'anti-crénelage nécessaire aux bords des objets, tout en mélangeant les objets inférieurs en cas de transparence.

Images bitmaps

Bien que Xara Xtreme soit à l'origine un programme de graphiques vectoriels, il propose également un support pour les tâches sur les images bitmaps. Au niveau le plus élémentaire, ce logiciel traite les objets bitmaps sur la page comme n'importe quel objet (vecteur). Il est également possible d'utiliser une image pour remplir n'importe quelle forme de Bézier (un remplissage bitmap peut être carrelé, redimensionné, tourné et placé de travers).

Par ailleurs, bien que Xara Xtreme ne soit pas un éditeur de pixels, il peut intégrer un outil d'édition de photo indépendant appelé Xara Picture Editor. D'un double clic sur une photo, celle-ci est passée dans XPE, capable de supporter un large choix de fonctionnalités d'édition de photos simples, tels que la luminosité, la couleur, la saturation, les effets de flou, la définition, etc. Une fois les modifications achevées, l'image bitmap traitée

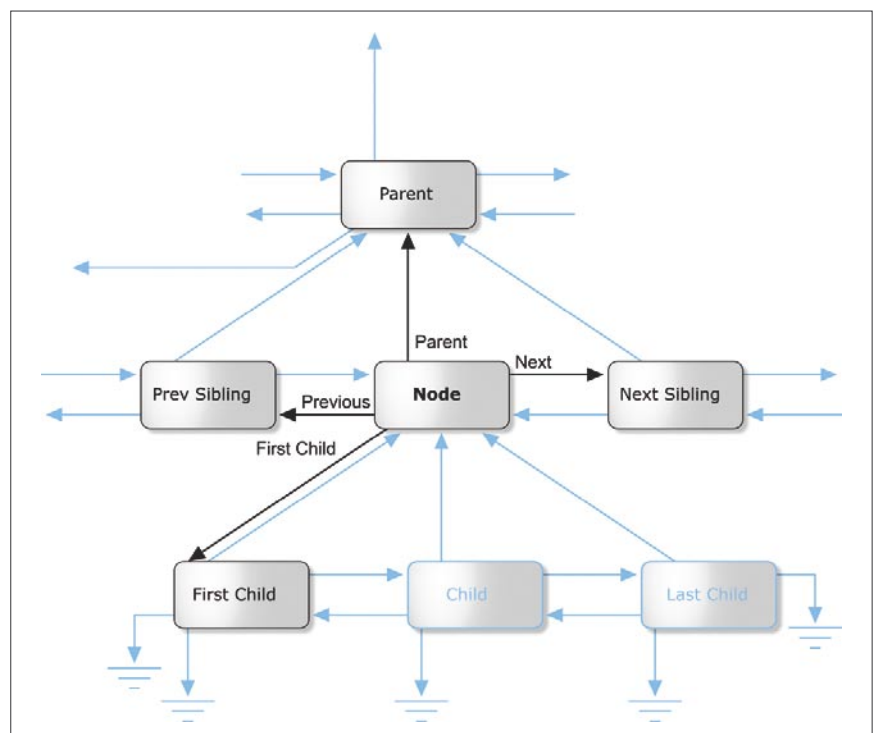


Figure 8. Diagramme de la structure arborescente d'un document Xara.

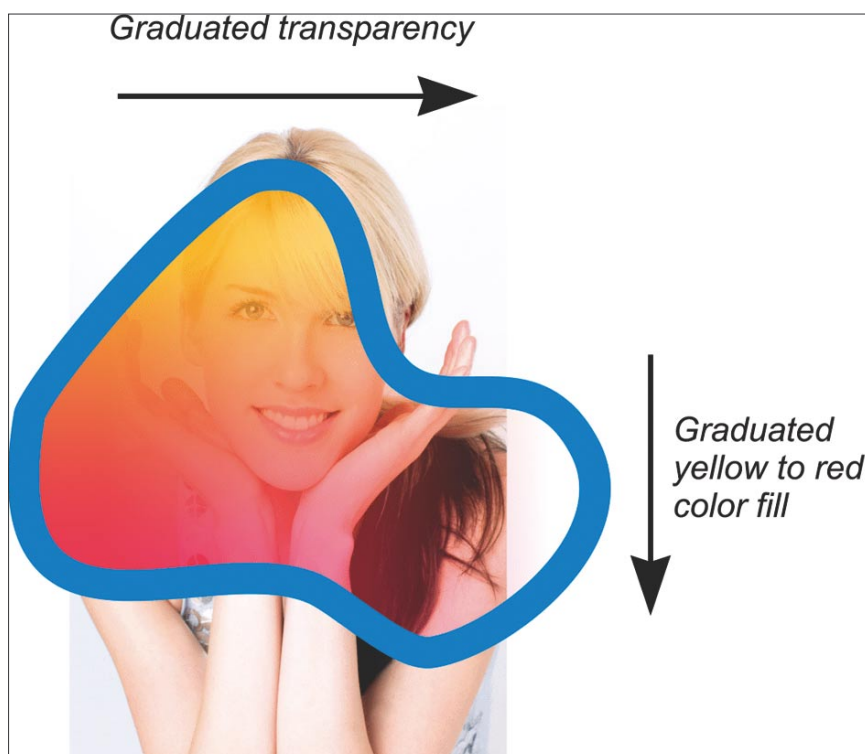


Figure 9. Simple objet vectoriel placé sur une photo, doté d'un contour bleu opaque épais retouché, d'un remplissage gradué jaune-rouge, et d'un effet de transparence graduée.

puis retournée à Xara, ainsi qu'une liste d'opérations d'édition stockée dans un fichier XML.

Plutôt inhabituel chez un éditeur d'images, XPE est capable de mémoriser l'ensemble des fonctions de traitement d'images bitmaps dans cette *liste d'édition XML*, stockée dans l'arborescence de Xara avec l'image bitmap *originale*. Xara peut ainsi reconstruire la photo traitée à partir de l'image originale, ce qui présente certains avantages incontestables.

Edition de photo dans la mémoire zéro

Lorsqu'une image JPEG est posée sur la page, Xara Xtreme stocke le fichier original compressé JPEG dans son arborescence principale. Ce fichier est ensuite décompressé afin d'être affiché à l'écran, mais seule l'image JPEG originale est conservée dans l'arborescence puis stockée sur le disque. Lorsque l'utilisateur applique divers effets de traitement d'image, au moyen de XPE, la liste d'édition XML est alors stockée en tant qu'attribut de l'objet JPEG. Puis, lorsque l'utilisateur charge le fichier Xara, le fichier JPEG est passé dans XPE avec la liste d'édition XML. L'image est quant à elle retraitée

puis retournée à Xara pour être affichée à l'écran.

De même, si un utilisateur applique n'importe quel attribut normal de Xara, comme la couleur ou le dégradé, ils seront appliqués à la volée. Par ailleurs, si l'utilisateur copie une image sur la page, Xara ne copie pas l'image bitmap originale, mais se contente de référencer l'image originale avec tous ses attributs modifiés, comme la taille, la position, la rotation, la couleur, etc...

Combinées, toutes ces techniques permettent d'économiser beaucoup de mémoire. Ainsi, par exemple, l'illustration suivante composée de 4 copies de photos JPEG haute résolution nécessite 1/20ème de l'espace du fichier requis par d'autres logiciels graphiques.

La photo originale JPEG d'un Mégaoctet représente une image RGB de six Mégaoctets, une fois décompressée. Chaque copie exige généralement six Mégaoctets supplémentaires de mémoire et d'espace disque. Sous XARA, ces 4 copies d'images, agrandies, recadrées, colorées et dégradées (l'oeil agrandi est doté d'un bord dégradé) sont toutes rendues à partir du fichier JPEG original compressé. Au moment de sauvegarder le document, seul le fichier JPEG original compressé est intégré dans un fichier natif *.xar*. Lorsque le fichier est réouvert, toutes les copies sont alors recréées à partir de l'image JPEG originale intégrée. Ainsi, contrairement aux autres logiciels graphiques produisant un fichier de plus de 20 Mégaoctets, le fichier natif *.xar* ne pèse qu'un Mégaoctet, et n'exige aucune recompression des données image.

Dans la mesure où l'arborescence Xara stocke une liste d'instructions lui permettant de recréer tous les objets sur la page, sans avoir à stocker l'image bitmap en question, n'importe quelle opération peut être modifiée à tout moment. Toutes les étapes et tous les objets demeurent des objets éditables autonomes. Comme tout utilisateur de Photoshop peut en témoigner, ce processus copiant des images dans de nouvelles couches, appliquant des



Figure 10. Toutes les images exposées ci-dessus ont été créées à partir de l'unique image JPEG originale stockée au format original compressé dans le fichier natif *.xar*. Ce fichier ne pèse donc qu'un Mégaoctet au lieu de 20, ou plus

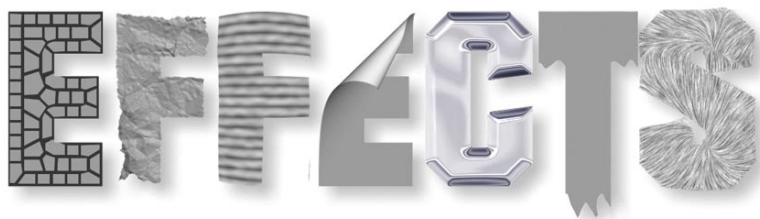


Figure 11. Comme ces effets sont stockés sous forme de commande dans la liste d'édition, un fichier de ce type doté d'effets de traitement d'images bitmaps haute résolution appliqué à un texte, n'exige que peu de place

effets, et mélangeant les résultats retournés sur l'image originale est assez répandu dans les tâches de composition graphique des photos. Par ailleurs, Xara supporte de nombreux plugins de Photoshop susceptibles de tirer profit de son approche orientée objet sans utiliser trop de mémoire.

Xara est suffisamment rapide pour pouvoir reconstruire l'image finale lors du chargement, ou dans certains cas à la volée lors de la durée du rendu, afin d'accélérer le processus.

Rendu selon Xara : sans perte et éditable

L'approche Xara se caractérise également par les deux avantages suivants :

1. JPEG est un format à perte. Vous ne voulez sans doute pas recompresser des images plus de fois que nécessaire. Ainsi, avec des outils d'édition de photos traditionnels, vous devez soit travailler avec les données brutes de l'image (ce qui prend beaucoup de mémoire et d'espace disque), soit perdre en qualité en sauvegardant les fichiers au format JPEG compressé, puis en les décompressant à chaque ouverture. Dans la mesure où Xara reconstruit toujours l'image au moyen de sa liste d'édition, vous ne serez plus confronté à ce type de problèmes.
2. Il peut vous arriver de changer d'avis et de vouloir éditer un effet appliqué antérieurement sans aucune perte. Si vous décidez par exemple d'ajuster le niveau de brillance ou d'histogramme, vous pouvez, à n'importe quel moment, retourner dans la liste et modifier ces valeurs ou les supprimer complètement. Xara se chargera de reconstruire une nouvelle image à partir de l'originale.

Effets bitmaps, effets directs

Xara Xtreme supporte une architecture plugin afin de pouvoir appliquer des effets bitmaps aux objets vectoriels. Xara supporte à la fois des plugins Photoshop et Xara. Ces effets sont appelés effets directs, dans la mesure où il est toujours possible d'éditer des objets vectoriels après avoir appliqué l'effet en question ; Xara se charge en effet d'appliquer de nouveau l'effet. Si l'utilisateur modifie une forme vectorielle avec l'application d'un effet Photoshop, cette forme est alors reconvertie en bitmap, passée dans le plugin Photoshop afin de réappliquer l'effet, et enfin l'image bitmap est retournée à Xara pour être affichée à l'écran.

Pour ce faire, Xara a recours aux plugins Photoshop dénommés *scriptable plugins*, capables de supporter une interface de programmation chargée de stocker les valeurs requises pour appliquer de nouveau l'effet. Ces valeurs sont stockées dans un objet attribut attaché à l'objet vectoriel de la principale structure arborescente. Les plugins bitmap Xara sont assez similaires, si ce n'est qu'ils supportent une interface utilisateur interactive, de manière à pouvoir mettre à jour l'écran en temps réel une fois l'effet appliqué. Les effets directs peuvent être appliqués aux objets vectoriels et bitmaps. Les effets bitmaps de ce genre permettent de réaliser des effets de flou, et bien d'autres encore.

Support multi-plateforme

Créer des applications multi-plateforme est une tâche difficile. Non seulement les langages C++ et ceux des compilateurs diffèrent légèrement, mais l'interface du système d'exploitation hôte est généralement complètement différente.

Lorsque Xara fut conçu, il était prévu de porter le logiciel sur plusieurs

plateformes. Nous avons donc séparé l'ensemble des éléments du noyau (code indépendant de la plateforme utilisée appelé *kernel*, et code indépendant de l'interface utilisateur et de celle du système d'exploitation, appelé couche *Oil*, pour *Operating Interface layer*). Le code capable de communiquer avec le système d'exploitation hôte ne peut exister que dans la couche *Oil*. Initialement, il était prévu de rédiger un code propre à la plateforme utilisée, et ce pour chaque plateforme existante, ce qui n'est plus nécessaire désormais grâce aux bibliothèques d'interfaces utilisateur multi-plateforme comme Qt de Trolltech et wxWidgets.

Nous avons donc opté pour le package libre wxWidgets (<http://www.wxwidgets.org/>), capable de supporter les plateformes Windows, Mac et Linux avec une interface de programmation commune. Ce package utilise l'ensemble de widgets natifs paramétré sur chaque plateforme, afin de permettre aux utilisateurs de créer une interface graphique détaillée et native dotée d'un ensemble de widgets personnalisables sur toutes ces plateformes.

Modèle libre pour Xara

Xara a récemment décidé de rendre le code source de Xara Xtreme disponible et libre en le distribuant sous la licence GPL. Il s'agit d'une orientation radicale pour notre société, puisque c'est la première fois qu'un produit graphique commercial de cette qualité et de cette complexité (code source supérieur à 1 million de lignes) devient libre. Les jour-

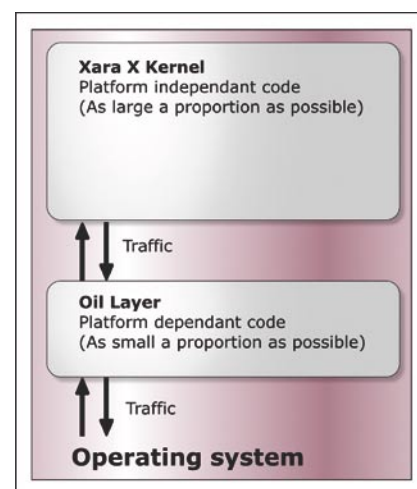


Figure 12. La couche *Oil* contient le code propre à l'interface utilisée afin de pouvoir travailler sur le système d'exploitation choisi

nalistes nous posent à ce sujets deux questions simples : pourquoi devenir un projet libre, et comment rentabiliser un programme gratuit.

Relativement petite, la société Xara dispose d'une technologie importante et d'un produit de qualité, dépourvue toutefois de ressources financières nécessaires à concurrencer la distribution ou les ressources techniques d'Adobe ou de Microsoft (qui a récemment déclaré vouloir pénétrer sur le même marché). Ainsi, afin d'attirer plus de développeurs et d'utilisateurs, nous avons pris le parti de distribuer notre produit librement, de manière à permettre à n'importe qui de travailler dessus, voire d'y développer ses propres fonctionnalités, si nécessaire. Cette décision traduit également notre intention de créer des variantes Linux et Mac de notre programme, ce que de nombreux utilisateurs nous demandaient depuis des années. Nous ne pourrions certainement pas justifier le développement d'une version de notre produit pour Mac ou Linux, seul, mais avec l'aide de la communauté Open Source, notre projet devient possible. Nous espérons en retour attirer plus de développeurs, afin de créer ensemble une solution capable de concurrencer de manière efficace des géants comme Adobe ou Microsoft.

Distribuer notre produit librement nous permettrait également de pérenniser l'avenir du logiciel. Nous ne pourrions pas revenir sur notre décision une fois le code révélé. Mais, tant que les utilisateurs nous manifesteront un certain intérêt pour développer le produit, cette application restera dans la course. Comme les utilisateurs de Freehand de Macromedia l'ont amèrement remarqué, les produits commerciaux, même populaires et largement répandus, ne présentent pas les mêmes garanties. De nombreux utilisateurs professionnels de Freehand doivent désormais trouver de nouvelles solutions censées remplacer leur produit préféré, depuis l'acquisition par un de ses concurrents de la société qui distribuait Freehand.

Comment rentabiliser un programme distribué librement ?

Nous continuerons à vendre des versions pour la plateforme Windows, ainsi que d'autres logiciels également. Nous espérons gagner la confiance de nombreux nouveaux utilisateurs avec le

projet libre Xara Xtreme, tout en engrenant des revenus complémentaires sur les activités de support, comme la vente d'ajouts ou d'éléments optionnels comme des CD et des manuels imprimés. Par ailleurs, nous espérons vendre des versions premium contenant des ajouts tiers dont le code source reste privé (comme par exemple, les bibliothèques tierces de couleurs Pantone, ou des plug-ins Photoshop tiers impossibles à distribuer librement, vendues à un certain prix), et bien sûr des supports personnalisés. Si nous pouvions établir une base d'utilisateurs suffisamment importante (le marché Linux manque en effet de programmes de dessins d'excellente qualité, stables ou 'finis'), seul un pourcentage relativement modeste d'utilisateurs désireux de nous acheter quelque chose suffirait à faire fonctionner notre modèle de distribution.

Enfin, comme la licence GPL interdit la distribution de Xara Xtreme ou de n'importe quel de ses composants sous forme de produit commercial au code source fermé, nous vendrons notre produit sous d'autres licences complémentaires autorisant l'intégration de notre programme dans des produits commerciaux.

Ce modèle économique est assez classique sur la plateforme Linux. De nombreuses sociétés de logiciels ont d'ailleurs réussi à vivre grâce à leurs produits gratuits et libres.

Accès au code source : comment nous aider

Vous pouvez consulter le site Web <http://www.xaraxtreme.org> pour plus de détails sur notre projet libre, et y télécharger la démonstration des binaires Linux et Mac. Le code source sort par étape, et devrait être disponible sur ce même site avant la publication du présent article.

Le soutien et l'assistance de développeurs expérimentés en C++ seront les bienvenus et nous permettrait de concrétiser notre vision du projet. Les logiciels graphiques de bureau sont tout aussi fondamentaux que les traitements de textes, les emails, ou les feuilles de calculs. Pour chacun de ces domaines, il existe généralement un ou deux acteurs majeurs, et nous espérons que Xara Xtreme puisse se hisser à son tour à ce rang. Si vous souhaitez prendre

part au développement de ce produit, vous pouvez consulter notre site Web et voir comment vous pourriez nous aider. Porter notre produit sur Linux et Mac va vraisemblablement prendre beaucoup de temps (selon le nombre de développeurs volontaires désireux de participer).

Résumé

Nous avons présenté dans le présent article les aptitudes et concepts fondamentaux du logiciel Xara. La distinction généralement opérée entre les programmes consacrés au traitement des images bitmaps et ceux consacrés aux graphiques vectoriels semble bien artificielle. Xara Xtreme gomme ces différences, et il semble possible, très prochainement, de créer des fonctionnalités d'édition de pixels reposant sur une approche vectorielle. Presque tous les outils classiques de type Photoshop pourraient donc être utilisés sans distinction sur des images bitmaps et vectorielles. Notre objectif et notre ambition ont toujours été très simples : créer la solution graphique la plus puissante du marché. ■