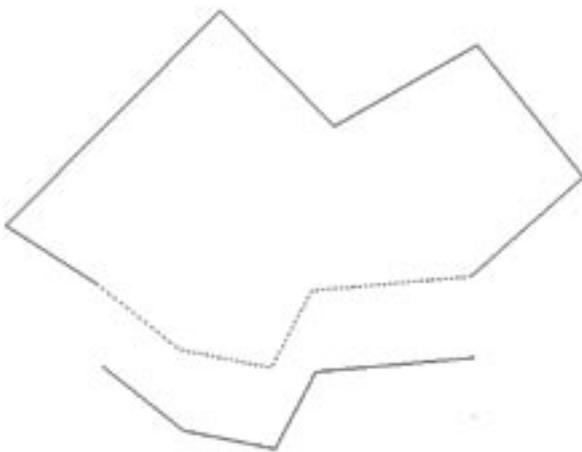


Dupliquer des parties de polyligne

Le programme ExtraitPOLY.LSP permet de dupliquer, depuis une polyligne existante, des segments ou des arcs, pour créer une nouvelle polyligne sans toutefois supprimer celle qui a servi de référence. La nouvelle polyligne prendra les mêmes propriétés et sera placée dans le même calque que celle de référence. ExtraitPOLY fonctionne aussi bien sur des polylignes optimisées ou pas, lissées ou non lissées. Par contre elle ne fonctionne pas sur des polylignes 3D.



Chargement

Pour charger le programme ExtraitPOLY servez-vous de la commande `_APLOAD` et sélectionnez le fichier ExtraitPOLY.LSP.

Utilisation

- 1 - Lancez la commande ExtraitPOLY
- 2 - Sélectionnez la polyligne
- 3 - Sélectionnez les segments ou les arcs que vous désirez extraire de la polyligne. Tous les types de sélection sont autorisés.

4 - La nouvelle polyligne se trouve être placée dans le calque : ExtraitPOLY

Le programme

; Extraction d'une partie de segments ou arcs d'une polyligne

```
(Defun c:ExtraitPoly ()
```

; Sélection de la polyligne

```
(Setq PolySelect (car (entsel "\nSélectionnez la Polyligne : ")))
```

; Mémorisation de son code DXF

```
(Setq PolySourceCode (entget PolySelect))
```

; Mémorisation du calque courant

```
(setq CalqueCourant (getvar "Clayer"))
```

; Duplication de la polyligne

```
(command "_copy" polyselect "" (list 0 0) (list 0 0))
```

; Création du calque temporaire \$\$\$CopiePoly pour y placer la polyligne dupliquée

```
(command "_layer" "_n" "$$$CopiePoly" "_f" "$$$CopiePoly" "")
```

; Transfert de la polyligne dupliquée dans le calque \$\$\$CopiePoly

```
(command "_chprop" (entlast) "" "_la" "$$$CopiePoly" "")
```

; Si le code DXF 70 de la polyligne est inférieur à 8 alors

; ce n'est pas une polyligne 3D

```
(if (< (cdr (assoc 70 polySourceCode)) 8)
```

```
(progn
```

; Création du calque temporaire \$\$\$ExtraitPoly pour pouvoir y placer tous

; les segments et arcs de la polyligne décomposée

```
(command "_layer" "_m" "$$$ExtraitPoly" "")
```

; Transfert de la polyligne dans le calque

\$\$\$ExtraitPoly

```
(command "_chprop" PolySelect "" "_la" "$$$ExtraitPoly"
"")
```

; Décomposition de la polyligne

```
(Command "_Explode" PolySelect)
```

; Sélection des segments ou arcs

```
(princ "\nSélectionnez les segments ou arcs à extraire : ")
(setq ObjAdd (ssget (list (cons 0 "LINE,ARC"))))
```

; Création d'une nouvelle polyligne contenant les segments ou arcs sélectionnés

```
(Command "_Pedit" (ssname ObjAdd 0) "_Y" "J" objadd
"" "")
```

; On crée un nouveau calque qui contiendra la nouvelle polyligne

```
(Command "_Layer" "_m" "ExtraitPoly" "")
```

; On place la nouvelle polyligne dans le calque ExtraitPoly

```
(command "_chprop" (entlast) "" "_la" "ExtraitPoly" "")
```

; On se replace sur le calque courant

```
(setvar "Clayer" CalqueCourant)
```

; On place la copie de la polyligne de base dans le calque courant

```
(command "_chprop" (ssget "x" (list (cons 8
"$$$CopiePoly"))) "" "_LA" CalqueCourant "")
```

; On efface le contenu du calque \$\$\$ExtraitPoly

```
(command "_erase" (ssget "x" (list (cons 8
"$$$ExtraitPoly"))) "")
```

; On purge les 2 calques temporairement créés

```
(command "_purge" "_la" "$$$ExtraitPoly,$$$CopiePol
y" "_n")
)
```

; Si c'est une polyligne 3D on affiche une alerte

```
(Alert "Cette polyligne 3D ne peut pas être traitée.")
)
```

```
(princ)
```

```
)
```

```
(princ "\n==> EXTRAITPOLY")
```

```
(princ)
```

Un décalage répétitif

Le programme DecaleGO permet de générer un décalage répétitif d'un objet sélectionné. Vous définissez la valeur de décalage, l'objet à décaler, le nombre de fois qu'il doit être décalé, la direction de décalage et GO !!! l'objet se décale autant de fois que vous l'avez défini. Un plus à la commande DECALER classique.

Chargement

Pour charger le programme DecaleGO servez-vous de la commande _APpload et sélectionnez le fichier DecaleGO.LSP.

Utilisation

Commande: DecaleGO

Spécifiez la distance de décalage: Entrez une valeur ou spécifiez 2 points

Choix de l'objet: Sélectionnez l'objet

Spécifiez le nombre de décalage: Entrez une valeur

Spécifiez un point sur le côté à décaler: Définissez le sens du décalage

Le programme

```
=====
```

```
; DecaleGO
```

```
; =====
```

```
(Defun c:DecaleGO ()
```

; rendre inactif l'écho des commandes

```
(setvar "Cmdecho" 0)
```

; on mémorise la valeur des accrochages aux objets

```
(setq osmode (getvar "osmode"))
```

; on interdit une valeur nulle, = 0 et négative

```
(initget 7)
```

; on définit la valeur de décalage

```
(setq LongDecale (getdist "\nSpécifiez la distance de
décalage: "))
```

; on sélectionne l'objet à décaler

```
(setq SelectObj (entsel))
```

; on mémorise le code de l'objet sélectionné

```
(setq ObjDecale (car SelectObj))
```

; on mémorise le point sélectionné

```
(setq PtSelectObj (cadr SelectObj))
```

; on interdit une valeur nulle, = 0 et négative

(initget 7)

; on définit le nombre de décalage à faire

(setq NbreDecale (getint "\nSpécifiez le nombre de décalage: "))

; on interdit une valeur nulle, = 0 et négative

(initget 7)

; on définit le côté à décaler

(setq PointCoteDecale (getpoint "\nSpécifiez un point sur le côté à décaler: " ptselectobj))

; on rend inactif le mode accrochage aux objets

(setvar "osmode" 0)

; on boucle sur le nombre de décalage

(repeat NbreDecale

; on décale l'objet sélectionné

(command "decaler" longdecale objdecale pointcotedecale ""))

; on récupère le code de l'objet qui vient de se créer lors du décalage

(setq objdecale (cdr (assoc -1 (entget (entlast))))))

; on redéfinit le côté à décaler

(setq pointcotedecale (polar pointcotedecale (angle ptselectobj pointcotedecale) longdecale))

)

; on redéfinit le mode d'accrochage aux objets comme il était au départ

(setvar "osmode" osmode)

(princ)

)

(princ "\n==> DecaleGO")

(princ)

Note :

Ce programme est donné ici comme exemple. Selon la direction du côté à décaler, il se peut que le décalage ne se réalise pas en totalité voire pas du tout. Dans ce cas, essayez de définir un point de décalage le plus perpendiculairement possible vis-à-vis du point de sélection de l'objet.

Extrait des Cahiers d'AUTOCAD

INFORMATIQUE CAO-DAO

Votre partenaire
Dominique VAQUAND sarl

BP 3 – 13430 Eyguières cedex
Tél. : 04 90 57 96 70 – Fax : 04 90 57 96 23
Email : dominique.vaquand@free.fr
Http://dominique.vaquand.free.fr

Cimmety Systems, Inc.®

AutoVue

Visualization & Collaboration™

3D

2D

EDA

Bureautique

Visualization & Annotation NATIVES

Maquettage Numérique

Supporte 450+ Formats:
AutoCAD, MicroStation, CATIA, Pro/ENGINEER, Unigraphics, HPGL, SolidWorks, SolidEdge, TIFF, Word, Excel, PowerPoint, Visio, Acrobat PDF, Mentor, Cadence, Zuken, Gerber, ORCAD, PADS, pour en citer quelques un.

Éditions Desktop ou Client-Serveur disponible sur plusieurs plate-formes

Collaboration en Temps Réel

● Télécharger votre copie d'évaluation complète et gratuite ● Essayez en ligne notre Édition Client-Serveur sur notre site Web:

www.cimmety.com

info@cimmety.com
+1 514 735 3219

AutoVue, Cimmety and Visualization & Collaboration sont des marques déposées de Cimmety Systems, Inc.. Tous les autres noms de produits sont des marques déposées de leurs propriétaires respectifs.