

Présentation des objets graphiques

Tour d'horizon des possibilités offertes par MATLAB



par Jérôme Briot ([Dut sur developpez.com](#))

Date de publication : 11/06/2007

Dernière mise à jour : 17/03/2009

Cet article présente les différents objets graphiques disponibles sous MATLAB permettant de réaliser des interfaces graphiques complètes et fonctionnelles.

Contenu :

- une introduction à la structure et à la gestion des objets graphiques
- une description rapide des objets conteneurs Root, Figure et Axes
- une description rapide des objets enfants de l'objet Axes
- une description rapide des objets UI

Public visé : cet article s'adresse à un public avancé maîtrisant déjà la syntaxe MATLAB.



Votre avis et vos suggestions sur cet article m'intéressent ! Alors après votre lecture, n'hésitez pas :

Avant propos.....	3
1 - Généralités.....	4
2 - Les objets conteneurs Root, Figure et Axes.....	6
2.1 - L'objet Root.....	6
2.2 - Les objets Figure.....	6
2.3 - Les objets Axes.....	6
3 - Les objets enfants de l'objet Axes.....	7
3.1 - Les objets Image.....	7
3.2 - Les objets Light.....	7
3.3 - Les objets Line.....	7
3.4 - Les objets Patch.....	8
3.5 - Les objets Rectangle.....	8
3.6 - Les objets Surface.....	8
3.7 - Les objets Text.....	9
4 - Les objets UI.....	10
4.1 - Les objets Uicontrol.....	10
4.2 - Les autres objets.....	12
Remerciements.....	15

Avant propos

Pour simplifier la lecture de cet article, l'auteur utilise une syntaxe courte et abusive pour identifier les objets graphiques.

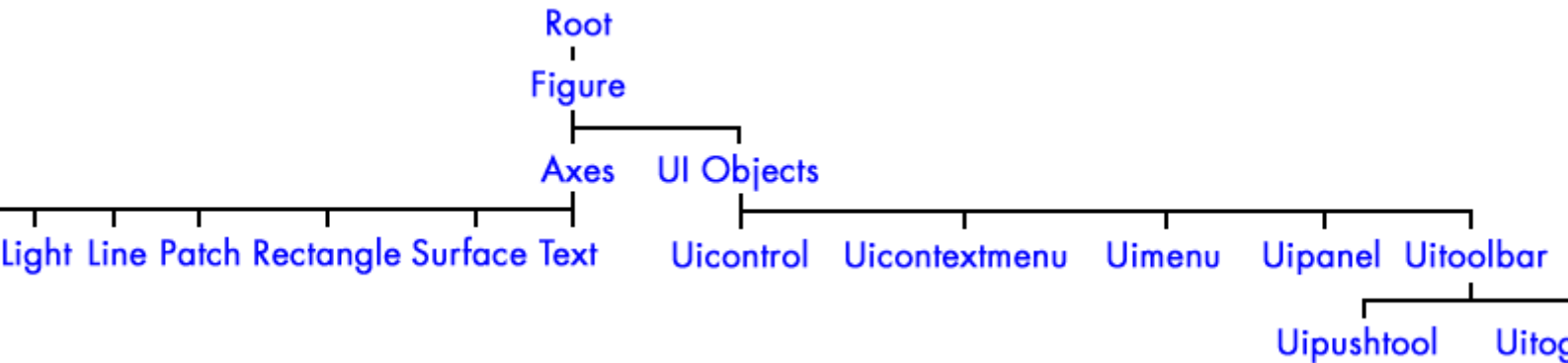
Par exemple, la syntaxe "Objet Image" sera utilisée à la place de "Objet graphique de type Image".

1 - Généralités

MATLAB est avant tout un logiciel de calcul matriciel. Il possède néanmoins une panoplie complète d'objets graphiques qui permettent, d'une part d'afficher les résultats de calculs sous des formes variées (point, courbe, surface, graphique) et d'autre part, de créer des interfaces graphiques (GUI) permettant à l'utilisateur d'interagir avec le programme.

Hierarchie :

Les objets graphiques sont disposés selon une hiérarchie parent enfant comme le montre le schéma suivant :



Identifiant :

Lorsque MATLAB crée un objet graphique, il lui attribue un identifiant (*handle* en anglais) unique sous la forme d'un nombre (entier ou réel) que l'on peut stocker dans une variable. Dans l'exemple suivant, *fig* est une variable contenant l'identifiant de l'objet Figure à sa création :

```

>> fig=figure

fig =

     1
    
```

Le programmeur gère les identifiants, soit avec la fonction GUIHANDLES, soit avec les fonctions FINDOBJ/FINDALL. Quelques identifiants particuliers peuvent être gérés avec les fonctions suivantes :

- GCA qui récupère l'identifiant de l'objet Axes courant
- GCBF qui récupère l'identifiant de l'objet Figure où se trouve l'objet graphique dont l'action est en cours
- GCBO qui récupère l'identifiant de l'objet graphique dont l'action est en cours
- GCF qui récupère l'identifiant de l'objet Figure courant
- GCO qui récupère l'identifiant de l'objet graphique courant

Propriétés :

Tous les objets graphiques possèdent des propriétés. Il est possible de configurer ces propriétés au moment de la création de l'objet en spécifiant le nom et la valeur de la propriété. Par exemple, pour créer un objet Figure dont le fond est de couleur verte :

```
fig=figure('color',[0 1 0]);
```

On peut gérer ultérieurement ces propriétés par les fonctions **GET** et **SET**, comme le montre le code suivant :

```
fig=figure;

col=get(fig,'color');
set(fig,'color',[0 1 0]);
```

Répartition des couches graphiques :

Les objets graphiques sont disposés sur différentes couches empilées les unes sur les autres en respectant l'ordre hiérarchique. L'objet Root est toujours situé en dessous, les objets Figure sur l'objet Root, les objets Axes sur leur conteneur Figure... Lorsque plusieurs objets de même niveau (exemple deux objets Axes) sont présents dans le même conteneur parent, leur ordre de visibilité est donné par leur ordre de création. Le dernier créé est au-dessus du précédent. Ce comportement peut être modifié ultérieurement en utilisant la fonction `UISTACK`. La seule exception vient des objets UI qui sont toujours sur les objets Axes à l'intérieur d'un même objet Figure.

2 - Les objets conteneurs Root, Figure et Axes

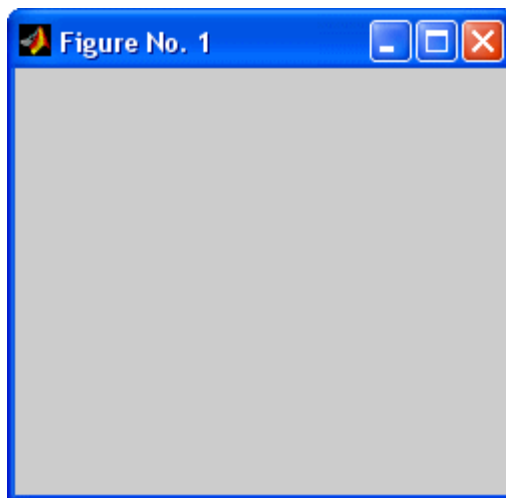
2.1 - L'objet Root

L'objet graphique Root est un objet invisible unique sans parent. Il est toujours présent et ne peut ni être créé, ni être détruit dynamiquement par l'utilisateur. Il est parent de tous les autres objets. Il représente l'écran de l'ordinateur. Il possède toujours comme identifiant le nombre 0.

La liste des propriétés de l'objet Root est disponible [ici](#).

2.2 - Les objets Figure

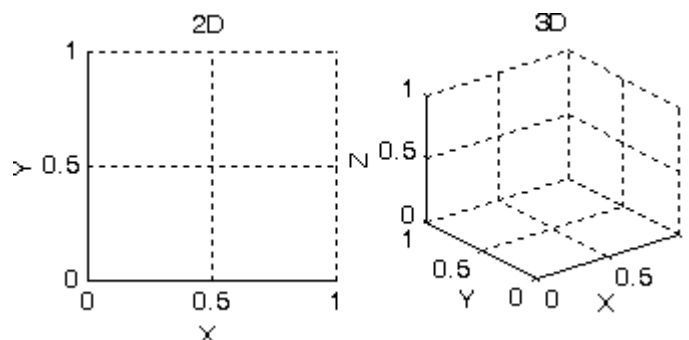
Les objets **Figure** sont créés par la fonction `FIGURE`. Ce sont les conteneurs visibles où sont disposés tous les autres objets graphiques. Plusieurs objets Figure peuvent être créés simultanément et peuvent communiquer ensemble.



La liste des propriétés des objets Figure est disponible [ici](#).

2.3 - Les objets Axes

Les objets **Axes** sont créés par la fonction `AXES`. Ce sont les zones de traçages des graphiques 2D ou 3D. Les résultats de calcul ou de simulation y sont affichés et peuvent y être manipulés (zoom, rotation...). Une Figure peut posséder plusieurs objets Axes simultanément. On utilise alors la fonction `SUBPLOT` pour gérer la position de ces objets.

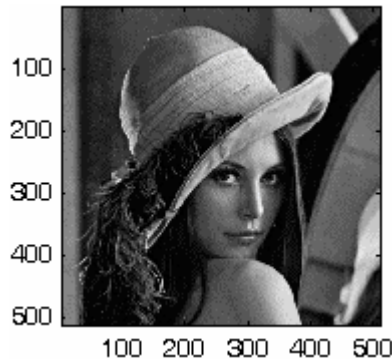


La liste des propriétés des objets Axes est disponible [ici](#).

3 - Les objets enfants de l'objet Axes

3.1 - Les objets Image

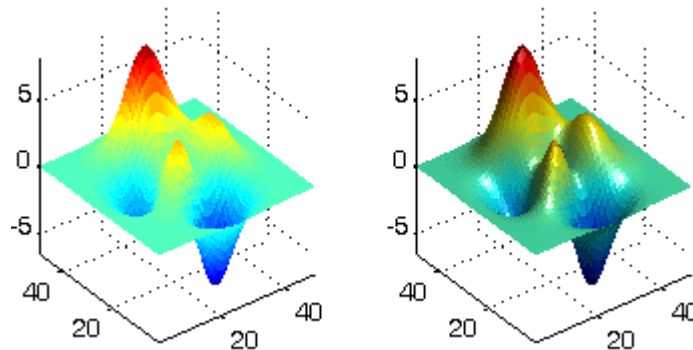
Les objets Image sont créés par les fonctions IMAGE, IMAGESC, IMSHOW... Comme leur nom l'indique, ces objets servent à afficher des images de formats standards (gif, jpeg, tiff, png, eps...) ou de type raw (matrice 2D ou 3D). Ces objets ne peuvent être affichés que dans le plan 2D de référence de Matlab. Il est impossible de manipuler ces objets en 3D.



La liste des propriétés des objets Image est disponible [ici](#).

3.2 - Les objets Light

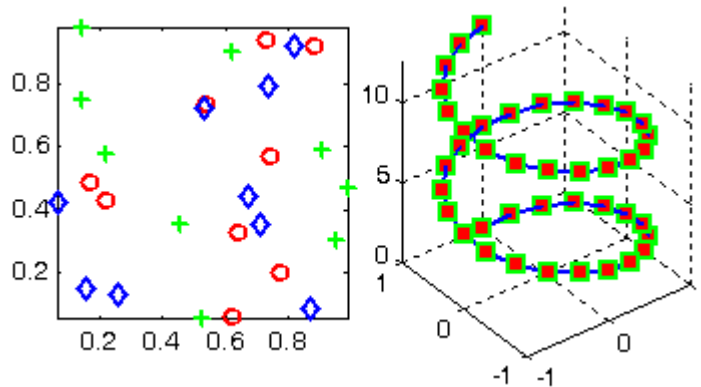
Les objets Light sont créés par la fonction LIGHT. Ils servent à mettre en valeur les objets 3D par application d'un éclairage virtuel.



La liste des propriétés des objets Light est disponible [ici](#).

3.3 - Les objets Line

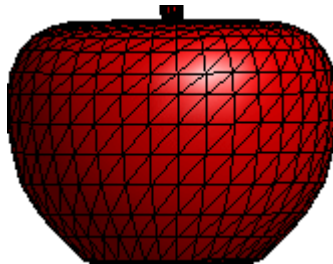
Les objets Line sont créés par les fonctions PLOT, PLOT3, LINE, STEM... Ces objets servent à la fois à afficher des nuages de points à l'aide de marqueurs ou bien des courbes (droite, polygone,...) en 2D comme en 3D.



La liste des propriétés des objets Line est disponible [ici](#).

3.4 - Les objets Patch

Les objets Patch sont créés par les fonctions PATCH, AREA, SCATTER, SCATTER3, FILL, TRISURF, TRIMESH... Ces objets sont utilisés pour afficher des éléments géométriques de définition mathématique simple (polygone, polyèdre). La flexibilité de la gestion de leur couleur et de leur transparence en font des objets très puissants.



La liste des propriétés des objets Patch est disponible [ici](#).

3.5 - Les objets Rectangle

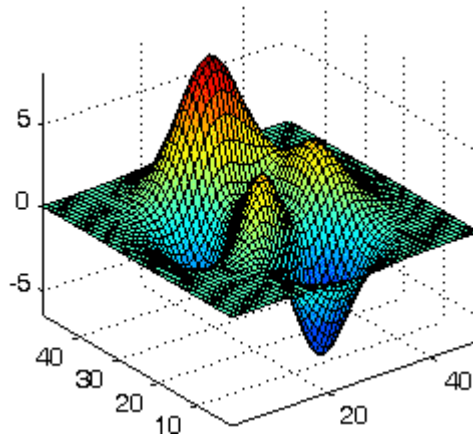
Les objets Rectangle sont créés par la fonction RECTANGLE. Ces objets sont plus basiques que les objets Patch. Ils sont utilisés pour créer des polygones 2D simples dont la forme peut être inscrite dans un rectangle (carré, rectangle, cercle, ellipse) L'autre utilité de ces objets est la possibilité d'afficher des rectangles dont les coins sont arrondis.



La liste des propriétés des objets Rectangle est disponible [ici](#).

3.6 - Les objets Surface

Les objets Surface sont créés par les fonctions SURFACE, SURF, MESH, SLICE... Ces objets sont généralement utilisés pour représenter des fonctions mathématiques en 3D du type $z=f(x,y)$



La liste des propriétés des objets Surface est disponible [ici](#).

3.7 - Les objets Text

Les objets Text sont créés par les fonctions TEXT, GTEXT, XLABEL, YLABEL, ZLABEL... Ces objets servent à afficher des zones de texte. Ils ont pour particularité d'être transparent (pas de couleur de fond). Ils sont donc très utiles pour les annotations de graphiques.

Ceci est un texte multiligne
avec des caractères spéciaux : α β γ
des **mises** en *forme différentes*
et des notations indicielles : a_0 a_1 a_2 b^0 b^1 b^2

La liste des propriétés des objets Text est disponible [ici](#).

4 - Les objets UI

Les objets UI sont des objets standards que l'on retrouve dans de nombreux langages informatiques. Ils permettent à l'utilisateur d'agir dynamiquement avec un programme à l'aide de boutons, menus, case à cocher...

4.1 - Les objets Uicontrol

Ces objets sont créés par la fonction `Uicontrol`. La propriété *Style* est utilisée pour les différencier. Le tableau suivant récapitule les différents objets Uicontrol disponibles dans MATLAB.

Nom	Désignation MATLAB	Aperçu
Case à cocher	checkbox	<input checked="" type="checkbox"/> Choix <input type="checkbox"/> Choix
Zone de texte éditable	edit	<input type="text" value="Saisie"/>
Cadre	frame	<input type="text"/>
Liste	listbox	<div style="border: 1px solid gray; padding: 2px;"> item 1 item 2 item 3 item 4 </div>
Menu déroulant	popupmenu	<input type="text" value="item 3"/> ▾ <div style="border: 1px solid gray; padding: 2px; margin-top: 5px;"> item 3 ▾ item 1 item 2 item 3 item 4 </div>
Bouton	pushbutton	<input type="button" value="OK"/>
Bouton Radio	radiobutton	<input checked="" type="radio"/> Choix <input type="radio"/> Choix
Barre de défilement	slider	<input type="range" value="50"/> <input type="range" value="50"/>
Zone de texte non éditable	text	<input type="text" value="Titre"/>
Bouton à 2 états	togglebutton	<input type="button" value="Off"/> <input type="button" value="On"/>

Tous les objets Uicontrol possèdent par défaut les mêmes propriétés. Mais certaines de ces propriétés n'affectent pas certains de ces objets. Le tableau suivant récapitule l'effet des propriétés sur chaque objet. (👍 propriété supportée, 🚫 propriété non supportée)

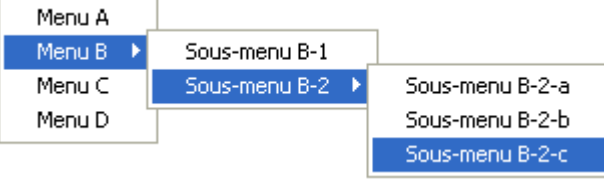
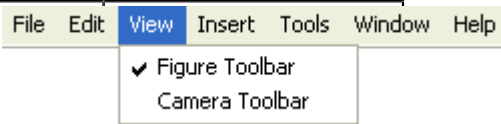
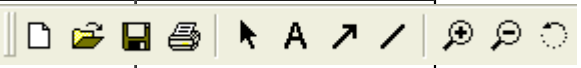


Propriété	checkbox	edit	frame	listbox	popupmenu	pushbutton	radiobutton	slider	text	togglebutton
BackgroundColor										
BeingDeleted										
BusyAction										
ButtonDownFcn										
Callback										
CData										
Children										
Clipping										
CreateFcn										
DeleteFcn										
Enable										
Extent										
FontAngle										
FontName										
FontSize										
FontUnits										
FontWeight										
ForegroundColor										
HandleVisibility										
HitTest										
Propriété	checkbox	edit	frame	listbox	popupmenu	pushbutton	radiobutton	slider	text	togglebutton

Propriété	checkbox	edit	frame	listbox	popupmenu	pushbutton	radiobutton	slider	text	togglebutton
HorizontalAlignment	👍	👍	👎	👍	👍	👍	👍	👎	👍	👍
Interruptible	👍	👍	👎	👍	👍	👍	👍	👍	👎	👍
ListboxTop	👎	👎	👎	👍	👎	👎	👎	👎	👎	👎
Max	👍	👍	👎	👍	👎	👎	👍	👍	👎	👍
Min	👍	👍	👎	👍	👎	👎	👍	👍	👎	👍
Parent	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Position	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Selected	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
SelectionHighlight	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
SliderStep	👎	👎	👎	👎	👎	👎	👎	👍	👎	👎
String	👍	👍	👎	👍	👍	👍	👍	👎	👍	👍
Style	checkbox	edit	frame	listbox	popupmenu	pushbutton	radiobutton	slider	text	togglebutton
Tag	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
TooltipString	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Type	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol	uicontrol
UIContextMenu	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Units	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
UserData	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Value	👍	👎	👎	👍	👍	👎	👍	👍	👎	👍
Visible	👍	👍	👍	👍	👍	👍	👍	👍	👍	👍
Propriété	checkbox	edit	frame	listbox	popupmenu	pushbutton	radiobutton	slider	text	togglebutton







































































La liste des propriétés des objets Uicontrol est disponible [ici](#).

4.2 - Les autres objets

Les autres objets regroupent les menus, les barres d'outils... Ils sont créés à l'aide des fonctions UIMENU, UICONTEXTMENU, UIPUSHTOOL, UITOGGLETOL, UITOOLBAR.

Nom	Désignation MATLAB	Aperçu
Menus contextuel	uicont	
Menus	uimenu	
Barres d'outils	uitool	
Bouton (barre outils)	uipushtool	
Bouton à 2 états (barre outils)	uitoggletool	

Propriété	uicontextmenu	uimenu	uipushtool	uitoggletool	uitoolbar
Accelerator					
BusyAction					
ButtonDownFcn					
CData					
Callback					
Checked					
Children					
ClickedCallback					
Clipping					
CreateFcn					
DeleteFcn					
Enable					
ForegroundColor					
HandleVisibility					
HitTest					
Interruptible					
Propriété	uicontextmenu	uimenu	uipushtool	uitoggletool	uitoolbar

Propriété	uicontextmenu	uimenu	uipushtool	uitoggletool	uitoolbar
Label					
OffCallback					
OnCallback					
Parent					
Position					
Selected					
SelectionHighlight					
Separator					
State					
Tag					
TooltipString					
Type	uicontextmenu	uimenu	uipushtool	uitoggletool	uitoolbar
UIContextMenu					
UserData					
Visible					
Propriété	uicontextmenu	uimenu	uipushtool	uitoggletool	uitoolbar

Remerciements

L'auteur tient à remercier **Lou Pitchoun** pour la correction orthographique de cet article et **Caro-Line** pour son aide pendant la rédaction de ce même article.