

# Interface utilisateur du package `pst-uml`

Maurice DIAMANTINI\*(email : `diam@ensta.fr`)

modif package : 27/08/06

modif doc : 27/08/06

44 pages compilées le 31 août 2006 à 10h28mn.

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Présentation . . . . .	1
1.2	Dépendances . . . . .	2
<b>2</b>	<b>Les macros de création d'objet</b>	<b>2</b>
2.1	<code>\umlClass</code> . . . . .	2
2.2	<code>\umlActor</code> . . . . .	4
2.3	<code>\umlState</code> , <code>\umlPutStateIn</code> et <code>\umlPutStateOut</code> . . . . .	6
2.4	<code>\umlCase</code> et <code>\umlCasePut</code> . . . . .	8
2.5	<code>\umlNote</code> . . . . .	9
<b>3</b>	<b>Les commandes auxiliaires de <code>pst-uml</code></b>	<b>9</b>
3.1	<code>\umlStack</code> . . . . .	10
<b>4</b>	<b>Positionnement des objets</b>	<b>11</b>
4.1	Utilisation de <code>psgraphics</code> et <code>\rput</code> . . . . .	11
4.2	Utilisation de <code>psmatrix</code> . . . . .	12
<b>5</b>	<b>Utilisation des commandes de placement relatif</b>	<b>13</b>
<b>6</b>	<b>Les connecteurs orthogonaux</b>	<b>15</b>
<b>7</b>	<b>Essais d'icônes "flèches" sur des courbes</b>	<b>15</b>
<b>8</b>	<b>Principales commandes PSTricks utilisées (A FAIRE)</b>	<b>16</b>
<b>A</b>	<b>Exemple de diagramme de classe</b>	<b>17</b>
<b>B</b>	<b>Exemple de diagramme des cas d'utilisation</b>	<b>21</b>
<b>C</b>	<b>Exemple de diagramme de séquences</b>	<b>23</b>
<b>D</b>	<b>Exemple de diagramme d'états</b>	<b>27</b>
<b>E</b>	<b>Listing du package <code>pst-uml.sty</code></b>	<b>31</b>

---

\*avec l'aide précieuse de Denis GIROU

# 1 Introduction

## 1.1 Présentation

Ce document présente l'utilisation d'un certain nombre de macro destinées à faciliter le dessin de diagramme de type UML (Unified Modeling Language) en utilisant le package `PSTricks` de Van Zandt TIMOTHY, dont l'usage est supposé connu.

Ce package est en cours de développement et n'est diffusé pour l'instant que à titre d'information. En particulier, la *stabilité* des noms de commandes proposées, leurs paramètres ainsi que leur comportement même *ne sont pas garantis*. Il en est de même que le nom et la fonction des options.

Dans le cas de diagrammes UML complexes, on peut augmenter la lisibilité du source, et donc faciliter les modifications du diagrammes en cours d'élaboration en divisant le travail en trois phases séparées :

1. définir graphiquement le dessin des différents éléments du schémas, en particulier pour les objets complexes (nombreux paramètres ou text important, ...);
2. placer<sup>1</sup> ces boites grâce à `psmatrix` ou à des `\rput` dans un environnement `pspicture` Les objets positionnés sont alors associé à un nom de node (au sens `PSTricks`). On peut également dans cette phase placer les labels ou autres dessins complémentaires liés aux graphismes des objets eux-mêmes (et non des connecteurs entre objets);
3. router<sup>1</sup> ces boites à l'aide des différentes commandes de connexion et coller les labels ou autres fleches UML.

Les macro proposées tendent à suivre les conventions `PSTricks` et `graphicx` pour le passage des options : à savoir l'utilisation de couple de la forme `Key=value`. Cette fonctionnalité nécessite l'utilisation du package `pst-key` qui est proposée dans la distribution de `pst-uml` (en attendant une diffusion plus générale). Un des intérêts du package `pst-key` est de pouvoir rajouter des options sans changer la syntaxe de la commande de base. D'une manière générales, le nombre d'options de toutes les commandes complexes est appelé à croitre.

Voici la syntaxe générale des commandes proposées :

```
\umlCommand[option1=value1,option2=value2,...]{param1}{param2}...
```

La valeur par défaut de chacune de ces options peut être redéfinie par la commande `\psset{...}` (comme toutes toutes les options de `PSTricks`).

## 1.2 Dépendances

Les packages suivants sont appelés par `pst-uml` (par la commande `\RequirePackage`):

- `pstricks`
- `pst-node`
- `pst-tree`
- `multido`
- `calc`
- `ifthenelse`
- `pst-xkey.tex` (par `\input` car en `TEX`)

---

<sup>1</sup>désolé : déformation professionnelle !

## 2 Les macros de création d'objet

### 2.1 `\umlClass`

#### Syntaxe

```
\umlClass[keyOptions]{title}{body}
```

Permet de dessiner une classe.

#### Paramètres

**title** : nom de la classe, celui-ci sera centré et écrit en gras. Il peut tenir sur plusieurs lignes.

**body** : corps de la classe proprement dit. En particulier contient les attributs et méthodes justifiées à gauche. Les différentes lignes sont séparées par `\\` et éventuellement des `\hline`.

#### Options

`umlDoubleRuleSep (dim : 2mm)` : distance<sup>2</sup> entre deux `\hline` (dans la zone **title** ou **body**).

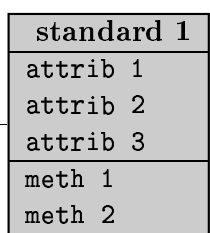
`umlShadow (boolean : true)` : si vrai, un cadre ombré de fond gris entoure la classe sinon, un cadre simple sur fond blanc est utilisé.

`umlParameter (string : "")` : si non vide, dessine un rectangle en pointillé contenant la valeur sur le coin supérieur droit de la classe (utilisé pour les classes paramétriques ou "Templates" en C++).

#### Exemples 1

Les illustrations suivantes présente une utilisation directe (i.e. sans utilisation dans une boîte ou par une commande) de la commande de création d'une classe. Ces classes sont entourées par un tiret pour montrer le positionnement vertical par rapport à la ligne de base.

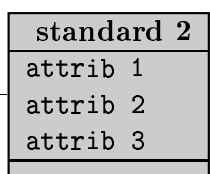
Exemples de classes standards



```

1  --\umlClass{standard 1}{%
2      attrib 1 \\
3      attrib 2 \\
4      attrib 3 \\
5          \hline
6      meth 1 \\
7      meth 2
8  }--

```

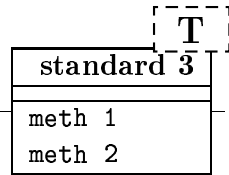


```

1  --\umlClass{standard 2}{%
2      attrib 1 \\
3      attrib 2 \\
4      attrib 3 \\
5          \hline
6          \hline
7  }--

```

<sup>2</sup>Comme pour toutes les dimensions passées à `PSTricks`, la spécification de l'unité utilisée est facultative et vaut `cm` par défaut, contrairement à `LATEX` où il est nécessaire de préciser une unité (même pour `0pt!`).



```

1  --%
2  \umlClass[umlShadow=false,umlParameter=\ T\ ]{%
3      standard 3%
4  }{%
5      \hline
6      meth 1 \\
7      meth 2
8  }--

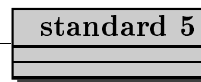
```



```

1  --\umlClass{standard 4}{}--

```



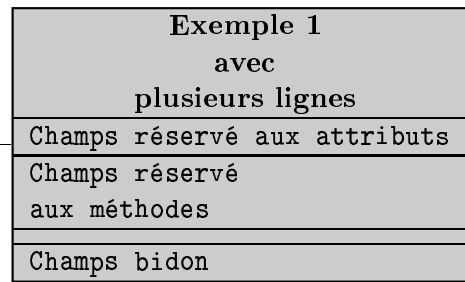
```

1  --\umlClass{standard 5}{%
2      \hline
3      \hline
4  }--

```

### Exemples 2

On peut mettre plusieurs sauts de ligne dans chaque paramètre de la commande. Pour le premier paramètre : toutes les lignes sont centrées et en gras :



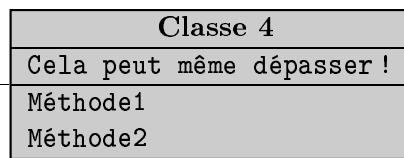
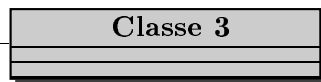
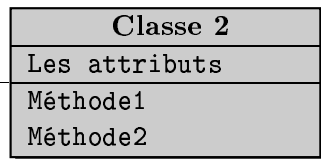
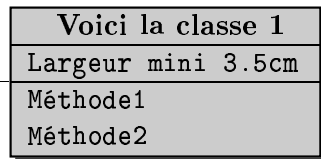
```

1  --%
2  \umlClass{Exemple 1 \\ avec \\
3      plusieurs lignes}{%
4      Champs r\`eserv\`e aux attributs\\
5      \hline
6      Champs r\`eserv\`e\\ aux m\`ethodes\\
7      \hline\hline
8      Champs bidon%
9  }--

```

### Exemples 3

On peut souhaiter imposer une largeur minimale à une classe, par exemple pour donner à plusieurs classes une apparence identique. On met alors le titre dans une `\makebox` (s'il n'y a qu'une seule ligne).



```

1  --%
2  \umlClass{\makebox[3.5cm]{Voici la classe 1}}{%
3    Largeur mini 3.5cm \\ \hline
4    M\'ethode1\\
5    M\'ethode2%
6  }--
7
8  \vspace{10mm}
9  --%
10 \umlClass{\makebox[3.5cm]{Classe 2}}{%
11   Les attributs \\ \hline
12   M\'ethode1\\
13   M\'ethode2%
14 }--
15
16 \vspace{10mm}
17 --%
18 \umlClass{\makebox[3.5cm]{Classe 3}}{%
19   \hline \hline
20 }--
21
22 \vspace{10mm}
23 --%
24 \umlClass{\makebox[3.5cm]{Classe 4}}{%
25   Cela peut m\^eme d\'epasser ! \\ \hline
26   M\'ethode1\\
27   M\'ethode2%
28 }--
29

```

Bug -

À faire -

## 2.2 \umlActor

### Syntaxe

```
\umlClass[keyOptions]{title}
```

Permet de dessiner un acteur représenté par un bonhomme avec un nom. Le titre du bonhomme est vu comme une boîte vide par  $\text{\LaTeX}$ , cela rend les liens dans les diagrammes des cas d'utilisation (qui sont la principale utilisation des Acteurs) plus agréable (les flèches sont positionnées par rapport au bonhomme et non pas au texte qui peut occuper plusieurs lignes).

### Paramètres (un seul)

**title** : nom de l'acteur. Il peut contenir plusieurs lignes séparées par `\\` et éventuellement des `\hline`

### Options

`umlActorLineWidth` (dimension : **0.6mm**) : Epaisseur de ligne pour le dessin de l'acteur.

`unit` (**dimension : 1 ou 1cm**) : option standard de `PSTricks` permet de modifier la taille de l'Acteur. Les options `xunit`, `yunit` sont également utilisable mais déconseillés.

D'autres option standard à `PSTricks` telle que `unit` (ainsi que `xunit` et `yunit`) peuvent être utilisées.

### Bug

Un agrandissement par un `yunit > 1`, a pour effet de décaler la tête vers le haut : il faut donc utiliser la commande `\resizebox` pour assurer un agrandissement correct.

L'utilisation de `unit` pour le changement de taille semble ne pas modifier pas la taille de la boîte crée : d'où l'encombre pour les petites taille (on compense par un `\resizebox{}[] [] {}` approprié).

### À faire

Option : pour concerver la taille de la boîte correspondant au Titre de l'acteur.

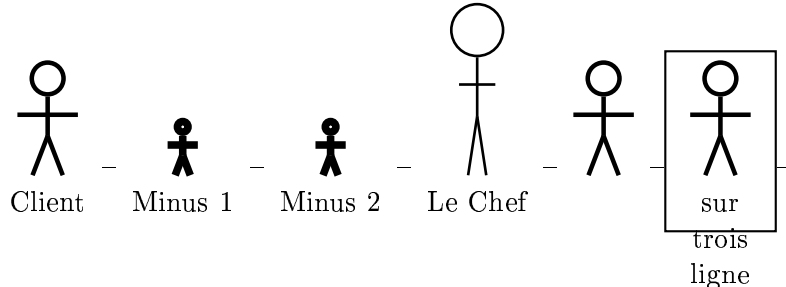
### Exemples 1

Ces commandes étant courtes, on peut les créer au moment même de placement qui se fait ici dans le corps du texte dans cet exemple (en mode LR au sens  $\LaTeX$ ).

```

1 \umlActor{Client}--%
2 \umlActor[unit=0.5,umlActorLineWidth=1mm]{Minus 1}--%
3 \umlActor[unit=0.5,umlActorLineWidth=1mm]{Minus 2}--%
4 \umlActor[xunit=0.6,yunit=1.5,umlActorLineWidth=1pt]{Le Chef}--%
5 \umlActor{}--%
6 \psframebox{\umlActor{sur\\trois\\ligne}}--%

```



### Exemples 2

On peut séparer la création et le dessin du placement. Le placement se fait également dans le texte même.

```

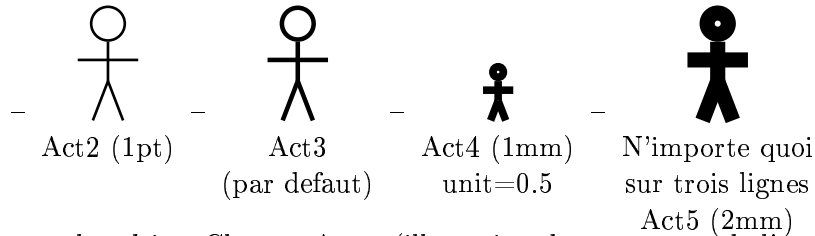
1 % Actor2 :
2 \newcommand{\drawActorii}{%
3 \umlActor[umlActorLineWidth=1pt]{Act2 (1pt)}}
4 % Actor3 :
5 \newcommand{\drawActoriii}{%
6 \umlActor{Act3 \\ (par default)}}
7 % Actor4 :
8 \newcommand{\drawActoriv}{%
9 \umlActor[umlActorLineWidth=1mm,unit=0.5]{Act4 (1mm)\unit=0.5}}
10 % Actor5 : (changement local d'une valeur par default)
11 \newcommand{\drawActorv}{%
12 \umlActor[umlActorLineWidth=2mm]{%
13 N'importe quoi\\sur trois lignes\\Act5 (2mm)}}
14 %
15 % utilisation des acteurs precedements definis :

```

```

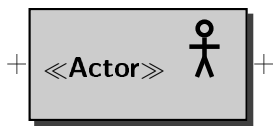
16 --\drawActorii%
17 --\drawActoriii%
18 --\drawActoriv%
19 --\drawActortv%

```



### Exemples 3

On peut composer les objets Class et Actor (illustration du stéréotype de l'acteur).



```

1 +%
2 \umlClass{\umlStereoType{Actor} %
3   \umlActor [unit=0.5] {}}%
4 }+%

```



```

1 +%
2 \umlClass{<<Actor>> %
3   %\raisebox{<raisedlift>}[<height>][<depth>]{<text>}
4   \raisebox{-0.3\height}[3ex][2ex]{%
5     \umlActor [unit=0.5] {}}%
6   }%
7 }{%
8 }+%

```

## 2.3 \umlState, \umlPutStateIn et \umlPutStateOut

### Syntaxe

Les commandes suivantes permettent de dessiner à l'emplacement courant respectivement un état, ou les pseudo-états d'entrée et de sortie :

```

\umlState[keyOptions]{title}{body}
\umlStateIn
\umlStateOut

```

À l'usage, les commandes de dessin de pseudo-état décrites ci-dessus sont avantageusement remplacées par les deux commandes suivantes :

```

\umlPutStateIn{coord}{nodeName}
\umlPutStateOut{coord}{nodeName}

```

Elles permettent de dessiner et surtout de placer et connecter les pseudo-états d'entrée et de sortie. Ces pseudo-états sont mis dans un \cnode au moment de leur création, ce qui permet aux connexions de toucher ces états même si le connecteur arrive à 45 degrés ; ce qui n'est pas le cas si on met la boîte (donc de forme carrée) créée par \umlStateIn *a posteriori* dans un \cnode !

### Paramètres

**title** : nom de l'état, peut-être vide. Celui-ci sera centré et écrit en gras. Il peut tenir sur plusieurs lignes (toutes seront en gras).

**body** : corps de l'état, peut-être vide. Il peut contenir plusieurs lignes séparées par `\\` et éventuellement des `\hline`. Il peut également contenir des environnements `pspicture` ou d'autres `\umlState`.

### options

**umlStateWidth (dimension: 0)** : (NON IMPLÉMENTÉ) largeur de l'état. La valeur 0 représente la dimension naturelle de cette boîte.


**umlStateHeight (dimension: 0)** : (NON IMPLÉMENTÉ) hauteur du corps de l'état l'état. La valeur 0 représente la dimension naturelle de cette boîte.


### Bug

#### À faire

#### Exemples 1

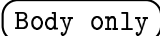

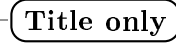
Les états d'entrée et de sortie

+  + : Input State

+  + : Output State

```
1 +\umlStateIn+ : Input State
2
3 +\umlStateOut+ : Output State
```

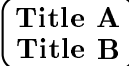
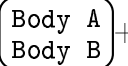
Exemples simples avec champs vides ou non

+  +  + 

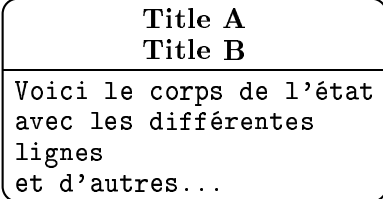
```
1 +%
2 \umlState{}{Body only}+%
3 \umlState{}{}+% Vide
4 \umlState{Title only}{}+
```

#### Exemples 2

On peut mettre titre ou corps sur plusieurs lignes. Toutes les lignes du titre sont en gras :

+  +  +

```
1 +%
2 \umlState{Title A\\Title B}{}+%
3 \umlState{}{Body A \\ Body B}+%
```

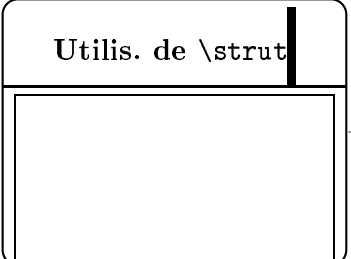
+  +

```
1 +\umlState{Title A\\ Title B}{%
2   Voici le corps de l'état \\
3   avec les diff\'erentes\\
4   lignes\\
5   et d'autres...%
6 }+%
```

#### Exemples 3

Il y a différents moyens pour réserver une place blanche de taille arbitraire (en attendant l'implantation d'options spécialisées pour ce besoin).

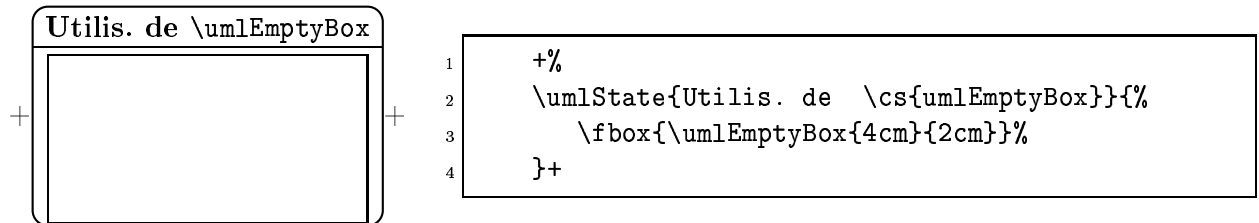
On peut rajouter un `strut` (règle invisible) pour agrandir un des champs, ainsi que pour créer un champ vide. La `\fbox` n'est là que pour indiquer la place qu'occupe les `struts`, de même l'épaisseur du `\strut` de 1 cm (au lieu de 0 cm normalement) permet de le visualiser :

+  +

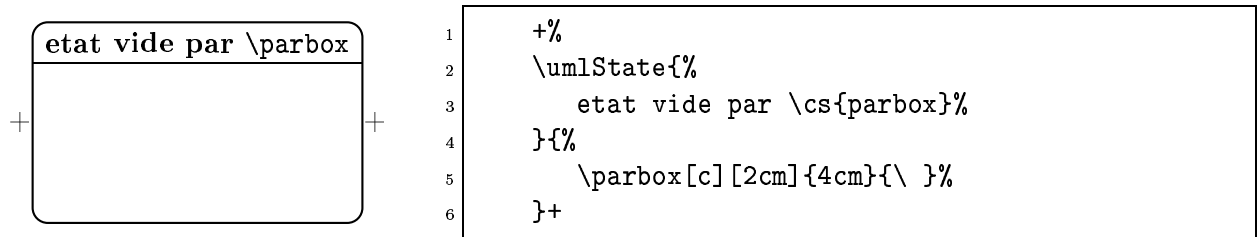
```
1 +%
2 \umlState{%
3   Utilis. de \cs{strut}\rule[-2ex]{1mm}{6ex}%
4 }{%
5   \fbox{\rule{0cm}{2cm}\rule{4cm}{0cm}}%
6 }+
```



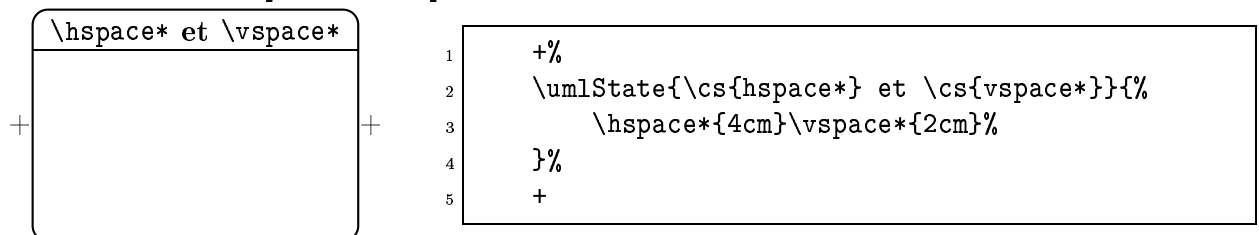
Une commande `\umlEmptyBox` est proposée pour créer une boîte vide (mais est-elle bien utiles??)



On peut spécifier une boîte "paragraphe" de largeur 4 cm et optionnellement de hauteur 2 cm dans le corps. Il faut mettre un espace forcé pour que le paragraphe ne soit pas vide.



On peut utiliser `\hspace*` et `\vspace*`, c'est la solution la plus précise.



On pourrait encore utiliser le saut de ligne paramétré `\\[2cm]` (moins précis, mais pratique dans les tableaux).

## 2.4 `\umlCase` et `\umlCasePut`

### Syntaxe

```
\umlCase[keyOptions]{body}
```

Permet un bloc de texte dans un oval. Cependant la boîte  $\text{\LaTeX}$  résultante est forcément rectangulaire (comme toute boîte  $\text{\LaTeX}$  : il n'est plus possible alors de créer simplement un ovalnode par la suite, d'où la quasi-inutilité de cette commande.

```
\umlCasePut[keyOptions]{coord}{nodeName}{body}
```

Permet un bloc de texte dans un ovalnode. Personnellement je n'ai utilisé que cette commande pour créer (et surtout connecté) des use-case).

### Paramètres

**coord** : coordonnées du centre de la boîte à positionner.

**nodeName** : nom du node (au sens  $\text{\PSTricks}$ ). Celui-ci sera utilisé pour les connexions ultérieures.

**body** : texte éventuellement sur plusieurs lignes séparées par des `\\` (utilisation en interne de la commande `\umlStack`).

**Options** : pas d'option spécifique pour l'instant

**Bug**

**À faire**

**Exemples 1**

Voir fichier séparé

## 2.5 \umlNote

**Syntaxe**

```
\umlNote[keyOptions]{noteBody}
```

Permet de dessiner une note dans un ardre au coin corné.

**Paramètres**

**noteBody** : corps de la note. Il peut contenir plusieurs lignes séparées par `\\`.

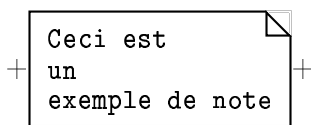
**Options**

**umlNoteCoin (dimension: 1.5ex)** : (NON IMPLÉMENTÉ) coin du petit carré replié.

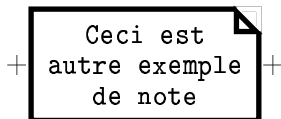
**Bug**

**À faire**

**Exemples 1**



```
1 +%
2 \umlNote{Ceci est \\ un \\ exemple de note}%
3 +
```



```
1 +%
2 \umlNote[linewidth=2pt,umlAlign=c]{%
3     Ceci est \\ autre exemple \\ de note%
4 }%
5 +
```

## 3 Les commandes auxiliaires de pst-uml

Ces commandes ne permettent pas de créer des boites au sens UML, mais simplifient la vie lors du dessin des différents type de diagrammes.

### 3.1 \umlStack

**Syntaxe**

```
\umlStack[keyOptions]{stackBody}
```

permet de mettre du texte sur plusieurs lignes dans un boite sans avoir d'espace supplémentaire en utilisant un environnement `tabular`. Par rapport à la commande `LATEX \shortstack`, l'espacement entre ligne n'est pas nul et est régulier.

**Paramètres (un seul)**

**stackBody** : corps de la boite. Il peut contenir plusieurs lignes séparées par `\\`.

## Options

`umlStackSep` (**dim** : 0) : distance de séparation entre le contenu de la boîte et le bord (pour une frame éventuelle).

Cette option utilise d'une part l'option `tabcolsep` de `tabular` pour régler l'espacement sur les bords verticaux, et d'autre part des `strut` pour rajouter des espacements horizontaux avant la première ligne et après la dernière ligne.

`umlStackWidth` (**dim** : 0) : largeur de la boîte. La valeur 0 indique une largeur automatique en fonction du contenu.

`umlStackLinesStretch` (**réel** : 0.85) : écart relatif de l'espace interligne par rapport à un tableau normal. La valeur par défaut de 0.85 permet de tasser suffisamment tout en permettant une marge pour absorber les différences de profondeur ou de hauteur des différentes lettres (x, g, j, h, t, ...)

`umlAlign` (**l, c, r** : c) : alignement horizontal du texte à l'intérieur de la boîte.

`umlPos` (**t, c, b** : c) : alignement vertical de la boîte par rapport à la ligne de base.

## Bug

### À faire

#### Exemples 1 : comparaison de `\umlStack`, `\shortstack` et `tabular`

Dans les exemples suivants, les `\umlStack` utilisées sont entourées d'une `\fbox` ayant une `\fboxsep` nulle de façon à bien visualiser les limites réelles de la boîte produite.

```

1 +%
2 \setlength{\fboxsep}{0pt}\fbox{%
3   \umlStack{Ceci est \\\ un \\\ exemple \\\
4     de stack}%
5 }+%

```

```

1 +%
2 \setlength{\fboxsep}{0pt}\fbox{%
3   \shortstack{Ceci est \\\ un \\\ exemple \\\
4     de stack}%
5 }+%

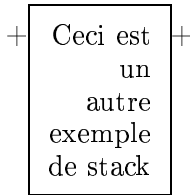
```

```

1 +%
2 \setlength{\fboxsep}{0pt}\fbox{%
3   \begin{tabular}{c}
4     Ceci est \\\ un \\\ exemple \\\
5     de stack
6   \end{tabular}%
7 }+%
8

```

#### Exemples 2 : utilisation des options

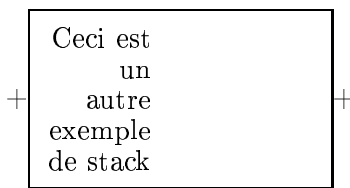


```

1 +\setlength{\fboxsep}{0pt}\fbox{%
2 \umlStack[
3     umlStackSep=1.5ex,
4     umlAlign=r,
5     umlPos=t,
6     umlStackLinesStretch=.9,
7     ]%
8     {Ceci est \\ un \\ autre \\
9     exemple \\ de stack}%
10 }+

```

### Exemples 3 : bug avec alignement à droite et largeur imposée



```

1 +\setlength{\fboxsep}{0pt}\fbox{%
2 \umlStack[
3     umlStackWidth=4,
4     umlStackSep=1.5ex,
5     umlAlign=r,
6     ]%
7     {Ceci est \\ un \\ autre \\
8     exemple \\ de stack}%
9 }+

```

## 4 Positionnement des objets

Le placement des différents objets peut se faire selon quatre méthodes :

- soit en utilisant les positions absolues dans un environnement `psgraphics` à l'aide de la commande `\rput` pour placer les objets et `\rnode` pour leur affecter un nom de noeud (en vue d'une future référence pour les connexions) ;
- soit à l'aide des commandes de haut niveau fournies par `PSTricks` à savoir l'environnement `psmatrix` et la commande `\pstree` ;
- soit à l'aide de commandes de placement relatif fournies par `pst-uml` (Merci à Denis GIROU) ;
- soit tout simplement dans le texte comme dans certains exemples montrés précédemment (en mode LR au sens  $\text{\LaTeX}$ ).

### 4.1 Utilisation de `psgraphics` et `\rput`

Il consiste à plasser les boîtes (précédemment créés ou non) dans un environnement `pspicture` à l'aide des commandes `\rput`. La macro `\psgrid` (compte tenu des options prédéfinies par le package `pst-uml`) permet de visualiser une grille de 1cm de coté. Cette grille pourra être commentée une fois la mise au point de la figure terminée. Dans cet exemple, les objets sont créés au moment du placement car ils sont simples. On remarque la création (par `\psdot`) d'un noeud ponctuel "pnode1" invisible sur le graphique, mais qui pourra être utilisé comme point de connexion intermédiaire (le petit cercle n'est là que pour sa visualisation).

```

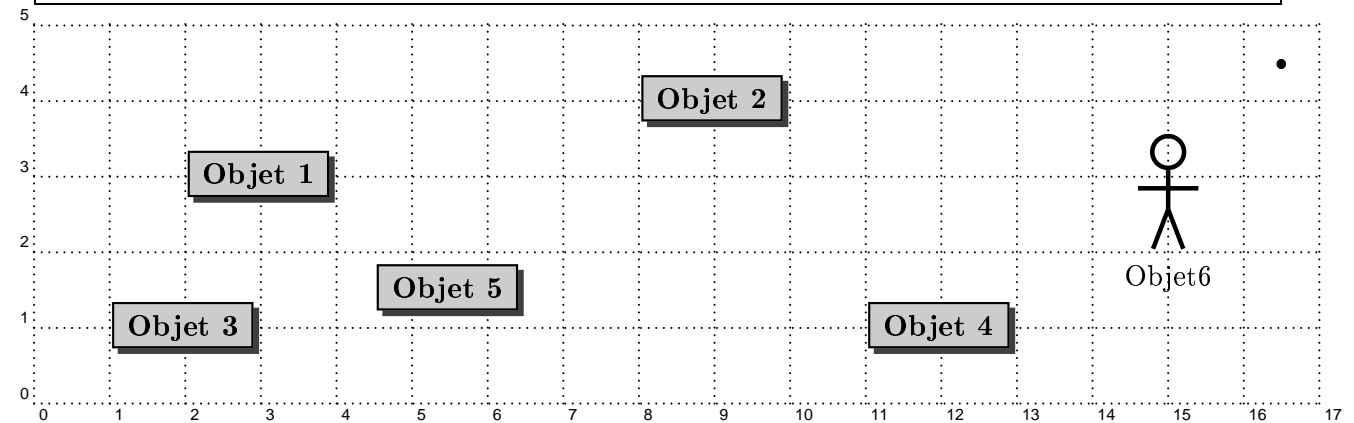
1 % positionnement des classes
2 \begin{pspicture}(17,5)\psgrid
3   \rput(3,3){\rnode{Objet1}{\umlClass{Objet 1}{}}}
4   \pnode(16.5,4.5){pnode1}\psdot(pnode1)% pour visu
5   \rput(9,4){\rnode{Objet2}{\umlClass{Objet 2}{}}}
6   \rput(2,1){\rnode{Objet3}{\umlClass{Objet 3}{}}}

```

```

7 \rput(12,1){\rnode{Objet4}{\umlClass{Objet 4}{}}}
8 \rput(5.5,1.5){\rnode{Objet5}{\umlClass{Objet 5}{}}}
9 %
10 \rput(15,2.5){\rnode{Objet6}{\umlActor{Objet6}}}
11 \end{pspicture}
12 %

```



## 4.2 Utilisation de psmatrix

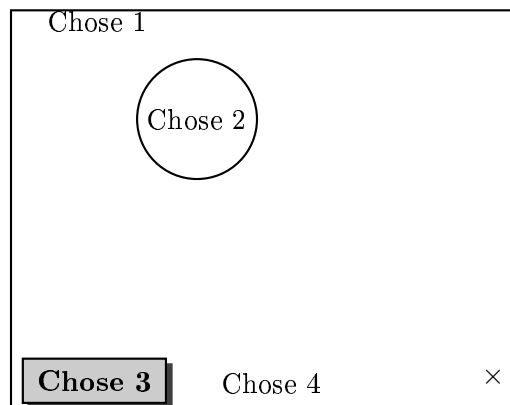
Les boites précédemment définies peuvent être positionnées à l'aide des macros de haut niveau de PSTricks telles que les environnements `psmatrix` ou `psTree`.

Ces macro sont plus simples à utiliser, mais sont moins souple au niveau du placement. Voici un exemple de placement utilisant l'environnement `psmatrix`. On peut remarquer l'utilisation d'une longueur négative pour la séparation des colonnes, ainsi que le changement possible de la longueur d'une seule colonne. De même, un changement d'espacement pour une ligne particulière peut se faire par `\\[2cm]` ou même `\\[-1cm]`.

```

1 \psshadowbox[framesep=0]{
2   \begin{psmatrix}[rowsep=0.3,colsep=-0.5,mnode=r]
3     % Dessin de classes predefinies
4     [name=Chose1] Chose 1 & & [colsep=2,name=P1] \\
5     & [name=Chose2] \pscirclebox{Chose 2} \\[-2cm]
6     [name=Chose3] \umlClass{Chose 3}{} & & [name=Chose4] Chose 4
7   \end{psmatrix}
8   % Visualisation d'un moeud ponctuel invisible par un X :
9   \ncput{pnode1}{\times}
10 }

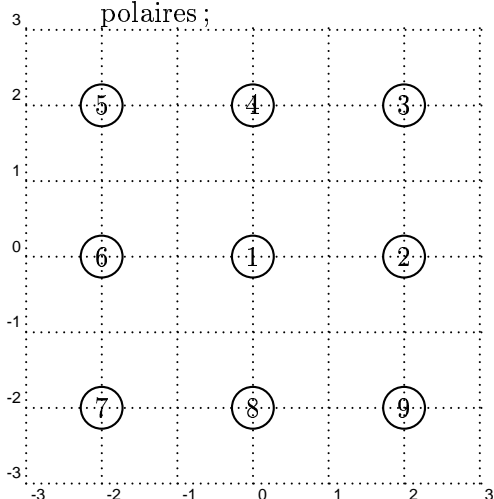
```



## 5 Utilisation des commandes de placement relatif

Les lettres E, N, W et S sont utilisés à la place de Right, Left, ... pour des raisons de cohérence avec les commandes de connexion du style `\ncNE` vues plus loin. Les commandes de gestion du placement relatif sont les suivantes :

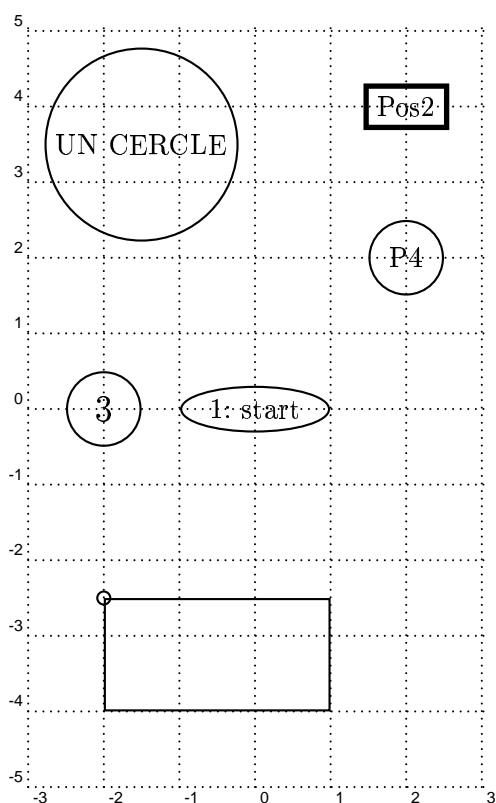
- affectation en absolu du point courant par rapport à l'origine (fixée à (0,0) pour l'instant) : `\ResetXY`, `\SetX`, `\SetY`, `\SetXY` en cours);
- accès au point courant par `\X` et `\Y`;
- les commandes de base : `\incrX`, `\incrY` (sont utilisées par les quatre suivantes);
- les déplacements relatifs `\moveE`, `\moveN`, `\moveW`, `\moveS`;
- positionnement d'un objet au point courant `\rputXY`.
- A FAIRE `\SetXY` pour l'affectation absolue du point courant à partir d'une coordonnée d'un nom de node mémorisé. Je n'ai pas trouvé la commande interne à `PSTricks` permettant de convertir les coordonnées à partir d'un nom d'un node en coordonnées X ou Y pour le point courant : (help);
- A FAIRE `\setOri` pour changer la position de l'origine. Cela affecte donc les commandes de positionnement absolues telles que `\resetXY`. Le but est d'affecter l'origine à un noeud précédemment créé par un `\pnode` ou autre `\node`;
- Note : vérifier si l'option `origin={coor}` de `PSTricks` ne conviendrait pas!
- A FAIRE `\move` (vers une direction incrémentale arbitraire pouvant utiliser les coordonnées polaires;



```

1 \begin{pspicture}(-3,-3)(3,3)\psgrid
2 \ResetXY % X and Y are rest at startup
3 \rputXY{\pscirclebox{1}}
4 \moveE{2} \rputXY{\pscirclebox{2}}
5 \moveN{2} \rputXY{\pscirclebox{3}}
6 \moveW{2} \rputXY{\pscirclebox{4}}
7 \moveW{2} \rputXY{\pscirclebox{5}}
8 \moveS{2} \rputXY{\pscirclebox{6}}
9 \moveS{2} \rputXY{\pscirclebox{7}}
10 \moveE{2} \rputXY{\pscirclebox{8}}
11 \moveE{2} \rputXY{\pscirclebox{9}}
12 \end{pspicture}

```



```

1 \begin{pspicture}(-3,-5)(3,5)\psgrid
2 %
3 \ResetXY % Don't forget !
4 %
5 \rputXY{\psovalbox{1: start}}
6 %
7 % mixing absolute and relative coords
8 \SetX{2}\SetY{2}\moveN{2}
9 \rputXY{\psframebox[linewidth=2pt]{Pos2}}
10 %
11 % use of negative coord, and coord with
12 % explicit units (10mm)
13 \moveW{4}\moveS{4}\moveE{-10mm}\moveE{1}
14 \rputXY{\Large 3}
15 % use of current coord X and Y as params
16 % for not using "\rputXY"
17 % (or for calcul : to be tested) :
18 \pscircle(\X,\Y){0.5}
19 %
20 % memorisation par un node
21 \moveE{4}\moveN{2}
22 \rputXY{\pnode{P4}}
23 \rputXY{P4}
24 \rputXY{\pscircle{0.5}}
25 %
26 % mixing absolute (SetX) and relative (moveN)
27 \SetX{-1.5}\moveN{1.5}
28 \rputXY{\pscirclebox{UN CERCLE}}
29 %
30 % use of current coord X and Y in calcul
31 % (doesn't work yet) :
32 \SetX{-2}\SetY{-2.5}
33 % \pssetlength allows \X+2 instead of \X+2cm
34 \newlength{\tmpX}\setlength{\tmpX}{\X}
35 \newlength{\tmpY}\setlength{\tmpY}{\Y}
36 \psaddtolength{\tmpX}{5}
37 \psaddtolength{\tmpY}{1}
38 \rputXY{\pscircle{0.1}} % at curent point
39 \rputXY{%
40     \psframe(0,0)(\tmpX,\tmpY)%
41 }
42 %
43 % acces direct to a memorised point
44 % NOT YET DONE
45 % \SetXY{P4}
46 % \rputXY{\pscirclebox{UN CERCLE}}
47 %
48 \end{pspicture}

```

## 6 Les connecteurs orthogonaux

En plus des divers connecteurs proposés par PStricks tels que `\ncline`, `\ncbar`, `\ncdiag`, `\ncdiagg`, `\ncangle`, `\ncangles` et `\nccurve`; `pst-uml` propose un certain nombre de connecteurs basés sur les précédents (par `\newpsobject`).

Le but est de simplifier le tracé des liens en se restreignant aux directions horizontales et verticales. Le principe est d'indiquer dans le nom même de la commande le nombre de segments à tracer et leur direction.

Par exemple, la première lettre (E dans `\ncEVW`) indique que le segment part vers l'Est, tourne verticalement (V : vers le haut ou vers le bas) puis tourne vers l'Ouest W pour se connecter.

- E, W, N, S pour Est, West, North, Sud,
- H, V pour Horizontal et Vertical,
- D pour diagonal,
- X pour indifférent.

Les commandes suivantes sont proposées :

- un seul segment : `\ncE \ncW \ncN \ncS`
- deux segments : `\ncEN \ncES \ncWN \ncWS \ncNE \ncNW \ncSE \ncSW`
- trois segments en U : `\ncEVW \ncWVE \ncSHN \ncNHS`
- trois segments en Z : `\ncEVE \ncWVW \ncNHN \ncSHS`
- trois segments en diagonale : `\ncEDE \ncWDW \ncNDN \ncSDS`
- quatre segments (voir trois) : `\ncSXE \ncSXW \ncEXS \ncEXN \ncWXS \ncWXN \ncNXE \ncNXW`

### Bug

La position par défaut des labels (utilisée par des `\naput...`) peut être affectée : dans ce cas imposer le positionnement explicitement par :

```
\ncE{nodeA}{nodeB}\naput[npos=0.5]{myLabel}
```

De même, dans les commandes à trois segments, quatre segments sont réellement dessinés : en cas de problème de positionnement des labels, il peut être utile d'imposer `armB=0` (en attendant un raffinement de ces commandes orthogonales).

### Exemples 1

A FAIRE (voir listing de dessin de classe ci-après)

## 7 Essais d'icônes "flèches" sur des courbes

Après des tentatives infructueuses (pour l'instant) de définir de nouvelle vraie flèche au sens PStricks (sans passer par `postscript` : en n'utilisant que des commande `TEX`), je propose une commande `\ncputicon` basée sur la commande `\ncput`. Par défaut, l'icône est positionnée en début de connecteur (cf. Bug ci-dessous)

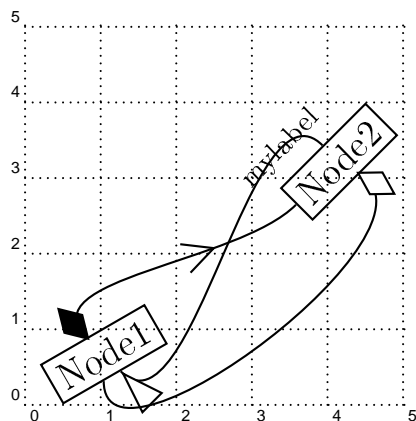
### Bug

#### À faire

Je souhaiterais que par défaut l'icône soit positionnée en fin de connecteur en mettant par exemple `\psset{npos=5}`. Cela ne marche pas pour tous les connecteurs (les `\ncline` et les `\nccurve`). C'est pour cette raison que la position par défaut est en début de connecteur (peut-être existe-t-il une variable PStricks qui indique le nombre total de segments présents dans le dernier connecteur utilisé??).



## Exemples 1



```

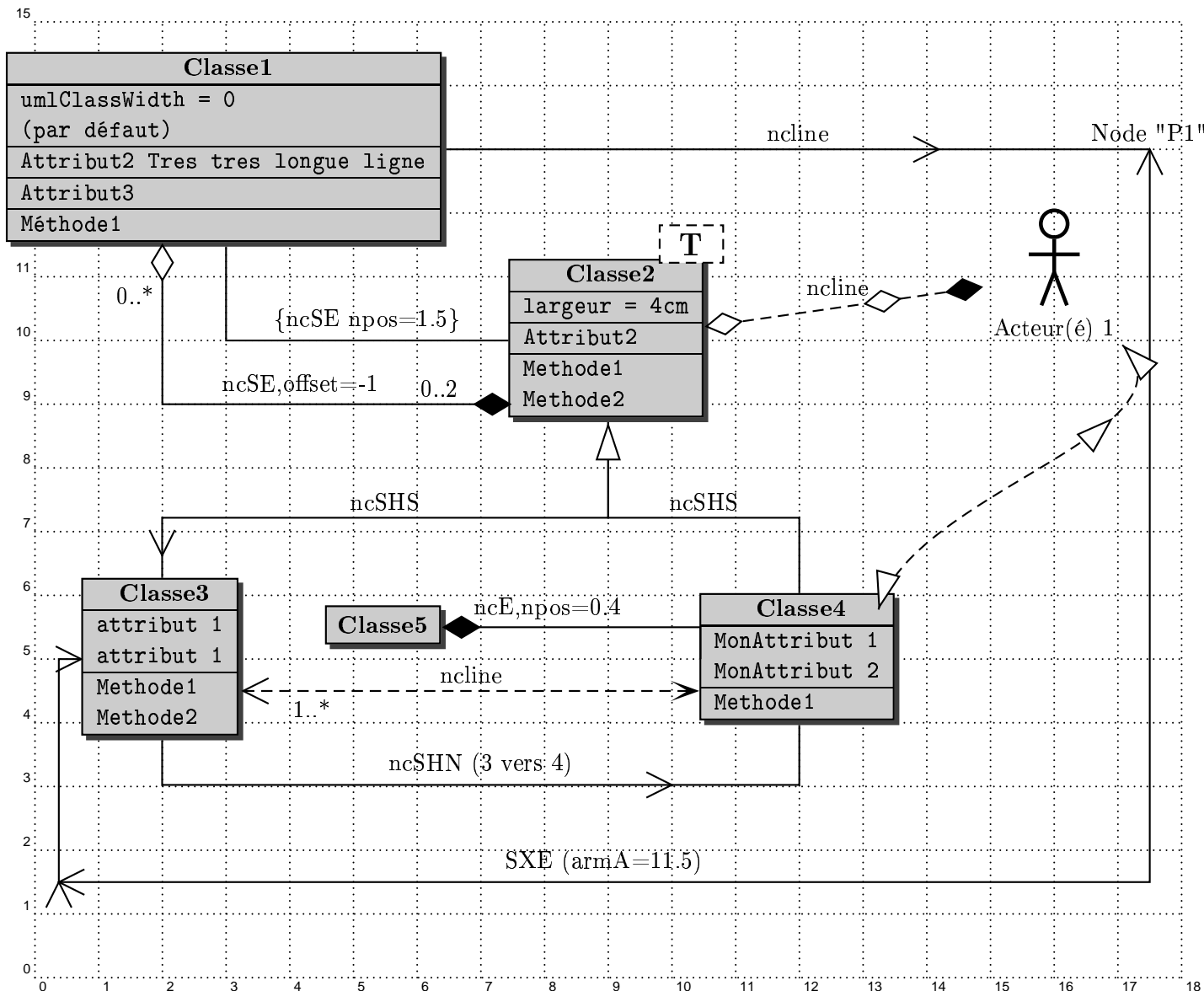
1 \begin{pspicture}(0,0)(5,5)\psgrid
2   \rput[b1]{30}(0.5,0){\rnode{Node1}}{%
3     \psframebox{\Large Node1}}
4   \rput[tr]{45}(4.5,4){\rnode{Node2}}{%
5     \psframebox{\Large Node2}}
6
7   \ncurve[angleA=-45,angleB=135]{Node1}{Node2}
8   \ncput[nrot=:U,npos=0.8]{mylabel}
9   \ncputicon{umlHerit}
10
11  \ncurve[angleA=-30,angleB=-90]{Node2}{Node1}
12  \ncputicon{umlAgreg}
13
14  \ncurve[angleA=135,angleB=-135]{Node1}{Node2}
15  \ncputicon{umlCompos}
16  \ncputicon[nrot=:U,npos=0.7]{umlV}
17
18 \end{pspicture}

```

## 8 Principales commandes PSTricks utilisées (A FAIRE)



# A Exemple de diagramme de classe



```

1 % \documentclass[11pt,a4paper,twoside]{article}
2 % \usepackage[T1]{fontenc}
3 % \usepackage[applemac]{inputenc}
4 % % \usepackage[latin1]{inputenc}
5 % \usepackage{pst-uml}
6 % \begin{document}
7
8 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9 % Placement des objet}
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 \newcommand{\drawClassi}{%
13   \umlClass{Classe1}{%
14     umlClassWidth = 0 \\  

15     (par défaut) \\\hline

```

```

16     Attribut2 Tres tres longue ligne \\ \hline
17     Attribut3 \\ \hline %
18     Methode1%
19 }}
20
21 \newcommand{\drawClassii}{%
22     \umlClass[umlClassWidth=4cm,umlParameter={\ T\ }]{Classe2}{%
23         largeur = 4cm \\ \hline
24         Attribut2 \\ \hline %
25         Methode1\\
26         Methode2%
27 }}
28
29 \newcommand{\drawClassiii}{%
30     \umlClass[umlClassWidth=2.7]{Classe3}{%
31         attribut 1 \\
32         attribut 1\\ \hline
33         Methode1 \\
34         Methode2%
35 }}
36
37 \newcommand{\drawClassiv}{%
38     \umlClass{Classe4}{%
39         MonAttribut 1 \\
40         MonAttribut 2 \\ \hline
41         Methode1%
42 }}
43
44 % Classe5 : largeur automatique et titre seul
45 \newcommand{\drawClassv}{%
46     \umlClass[umlClassWidth=0]{Classe5}{}}
47
48 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
49 % Placement des objets
50 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
51
52 \begin{pspicture}(18,15)\psgrid
53     \rput(3,13){\rnode{Class1}{\drawClassi}}
54     \pnode(17.5,13){pnode1}
55     \rput(9,10){\rnode{Class2}{\drawClassii}}
56     \rput(2,5){\rnode{Class3}{\drawClassiii}}
57     \rput(12,5){\rnode{Class4}{\drawClassiv}}
58     \rput(5.5,5.5){\rnode{Class5}{\drawClassv}}
59     %
60     \rput(16,11){\rnode{Actor1}{\umlActor{Acteur(é) 1}}}
61 \end{pspicture}
62 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
63 % Dessin des liens et labels
64 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
65 % La grande boucle en deux étapes :

```

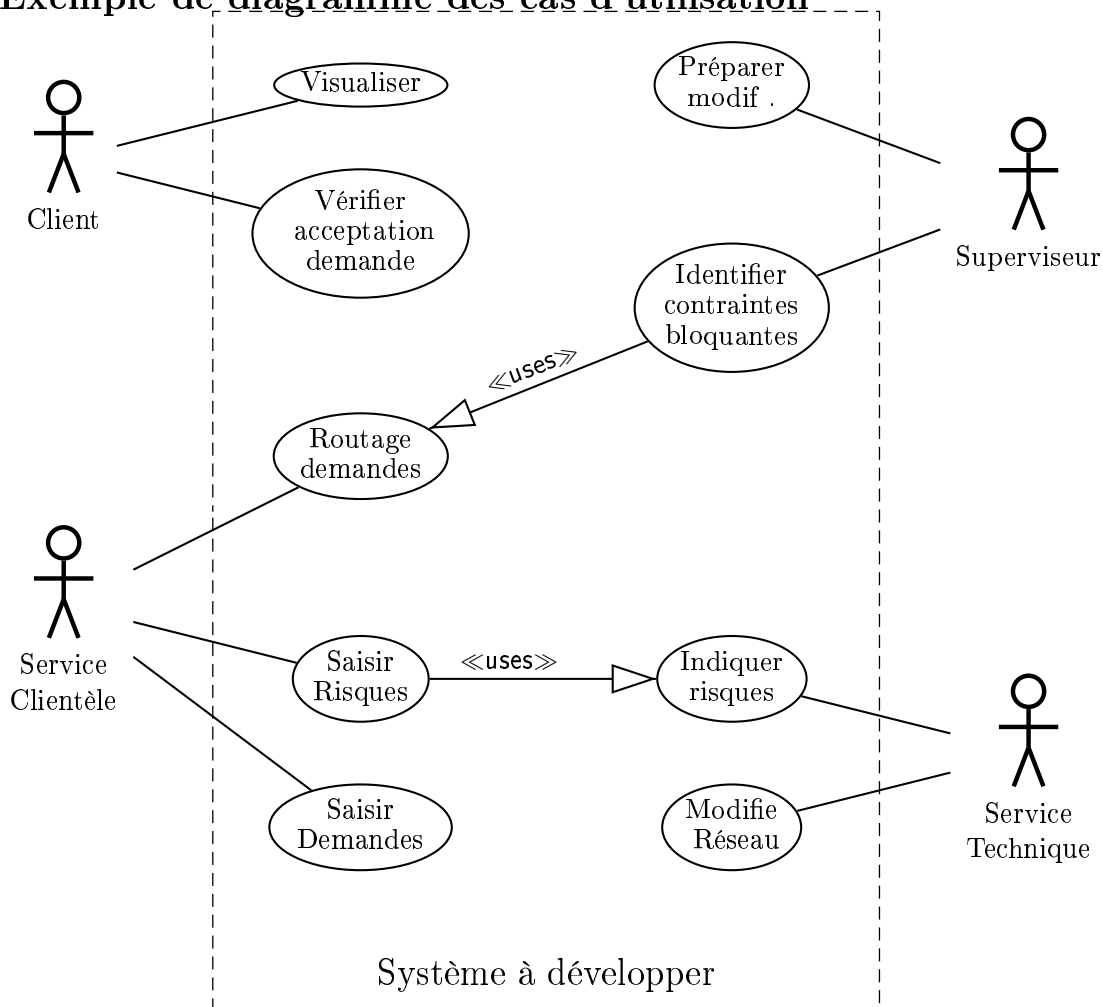
```

66 \ncpline{Class1}{pnode1}
67 \ncputicon [npos=0.7,nrot=:U] {umlV}
68 \naput{ncpline}\naput [npos=1,ref=r] {Node "P1"}
69 \ncSXE [armA=11.5]{pnode1}{Class3}
70 \nbput{SXE (armA=11.5)}
71 \ncputicon{umlV}% debut
72 \ncputicon [npos=1.9999,nrot=:U] {umlV}
73 \ncputicon [npos=2,nrot=:U] {umlV}
74 \ncputicon [npos=5,nrot=:U] {umlV}% fin ERREUR si nrot=4 ok pour 5!!!!
75 %
76 \ncSE{Class1}{Class2}
77 \naput [npos=1.5]{\{ncSE npos=1.5\}}
78 \ncSE [offset=-1]{Class1}{Class2}
79 \ncputicon{umlAgreg} % debut
80 \ncputicon [npos=2,nrot=:U] {umlCompos}% fin
81 \nbput [npos=0.3]{0..*}
82 \naput [npos=1.8]{0..2}
83 \naput [npos=1.4]{ncSE,offset=-1}
84 %
85 \ncSHS [armA=1.5]{Class2}{Class4}\naput{ncSHS}
86 \ncSHS [armA=1.5]{Class2}{Class3}\nbput{ncSHS}
87 \ncputicon{umlHerit}% héritage au debut
88 \ncputicon [npos=3,nrot=:U] {umlV}% V en fin
89 %
90 \ncSHN [arm=.7]{Class3}{Class4}
91 \naput{ncSHN (3 vers 4)}
92 \ncputicon [npos=1.8,nrot=:U] {umlV}% fleche au milieu vers destination !
93 %
94 % \ncE [npos=0.4]{Class5}{Class4}\naput{ncE,npos=0.4}
95 \ncE{Class5}{Class4}\naput [npos=0.4]{ncE,npos=0.4}
96 \ncputicon{umlCompos}
97 %
98 % Essai de définition d'un style personnalisé
99 \newsstyle{umlDependance}{%
100     linestyle=dashed,
101     arrows=->,
102     arrowsscale=3,
103     arrowinset=0.6
104 }
105 \ncpline [style=umlDependance,offset=-0.5]{Class3}{Class4}
106 \naput{ncpline}
107 \ncputicon{umlV}% fleche au debut
108 \nbput [npos=0.15]{1..*}
109 %
110 % % % On peut coller n'importe quoi par rapport à un node :
111 % % % Essai pour mettre un template sur une classe : prévoir
112 % % % une option du style [umlTemplate=myString]
113 % % \nput*[labelsep=-0.8,offset=1.4]%
114 % % {0}{Class2}{\psframebox%
115 % % [fillstyle=solid,fillcolor=white,linestyle=dashed]}%

```

```
116 % %      {\LARGE\textbf{\ T\ }}
117 % % %
118 % Lien de Class2 et Class4 vers l'acteur :
119 \ncline[linestyle=dashed]{Class2}{Actor1}
120 \naput{ncline}
121 \ncputicon{umlAgreg}
122 \ncputicon[npos=0.7,nrot=:U]{umlAgreg}
123 \ncputicon[npos=1,nrot=:U]{umlCompos} % en fin
124 %
125 \ncurve[linestyle=dashed, angleA=75,offsetA=-1,angleB=-45]{Class4}{Actor1}
126 \ncputicon{umlHerit} % debut
127 \ncputicon[npos=0.7,nrot=:U]{umlHerit}
128 \ncputicon[npos=1,nrot=:U]{umlHerit}% en fin
129
130 % \end{document}
```

## B Exemple de diagramme des cas d'utilisation



```

1 % \documentclass[11pt,a4paper,twoside]{article}
2 % \usepackage[T1]{fontenc}
3 % \usepackage[applemac]{inputenc}
4 % % \usepackage[latin1]{inputenc}
5 % \usepackage{pst-uml}
6 % \begin{document}
7
8 \begin{center}
9 % \scalebox{0.5}{%} Fonctionne egalement
10 \resizebox{0.9\linewidth}{!}{%
11 \begin{pspicture}(0,0.5)(15,13.5)\psgrid
12 \psset{framesep=0}
13 %
14 \psframe[linewidth=0.5pt,linestyle=dashed](3,14)(12,0.5)
15 \rput(7.5,1){\Large Système à développer}
16 %
17 \rput(1,12){\rnode{acCL}{\umlActor{Client}}}
18 \rput(1,6){\rnode{acSC}{\umlActor{Service\ Clientèle}}}
19 \rput(14,4){\rnode{acST}{\umlActor{Service\ Technique}}}
20 \rput(14,11.5){\rnode{acSU}{\umlActor{Superviseur}}}
21 %
  
```

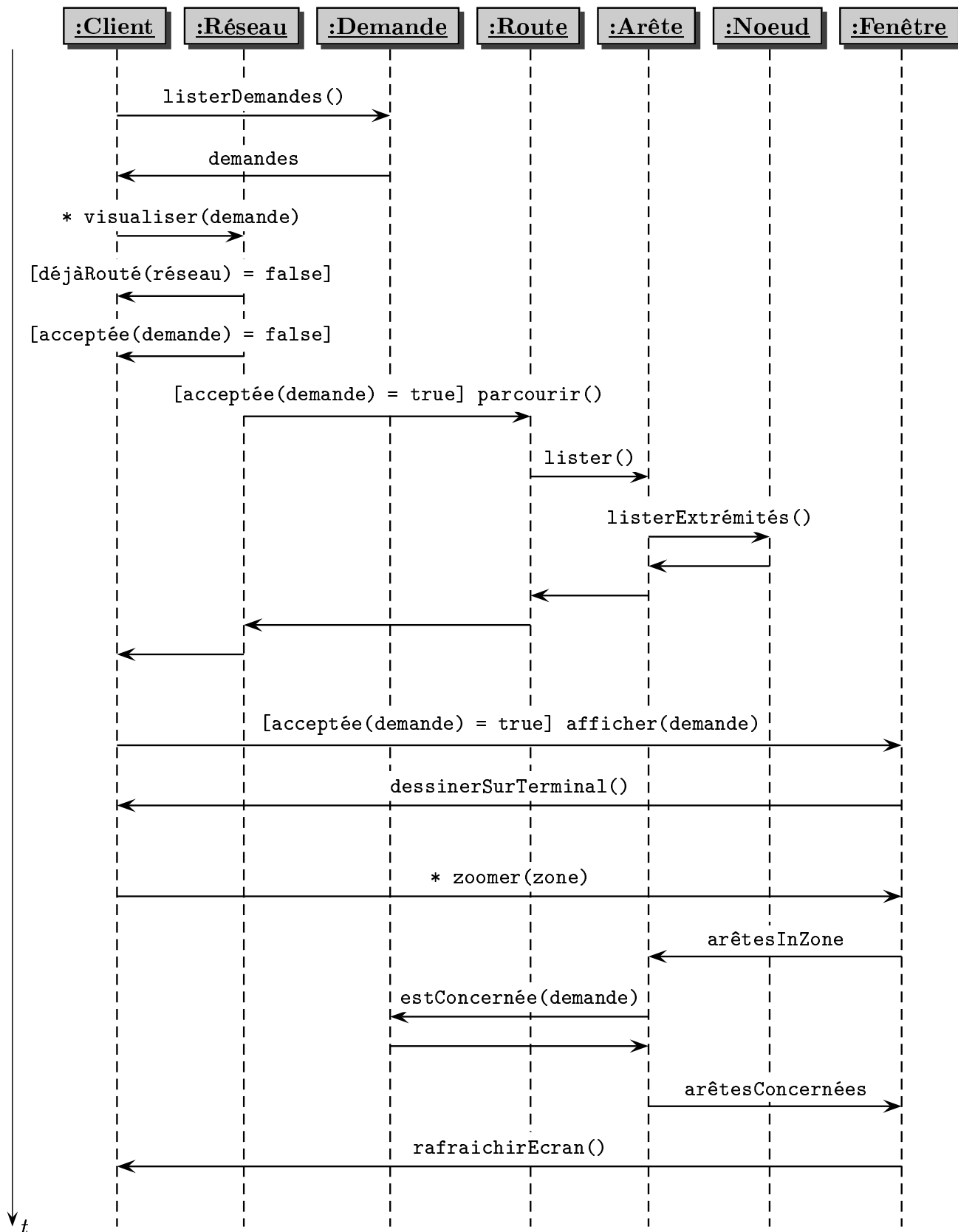
```

22 %
23 % \umlPutCase{5,13}{VISU}{\[[0mm]Visualiser\[[0mm]]}
24 \umlPutCase{5,13}{VISU}{Visualiser}
25 \umlPutCase{5,5}{SR}{Saisir\ Risques}
26 \umlPutCase{5,3}{SD}{Saisir\ Demandes}
27 \umlPutCase{5,8}{RD}{Routage\ demandes}
28 \umlPutCase{10,10}{ICB}{%
29     Identifier\
30     contraintes\
31     bloquantes}
32 \umlPutCase{10,3}{MR}{Modifie\ Réseau }
33 \umlPutCase{5,11}{VAD}{Vérifier\ acceptation\ demande }
34 \umlPutCase{10,5}{IR}{Indiquer\ risques }
35 \umlPutCase{10,13}{PM}{Préparer\ modif .}
36 %
37 \ncline{acCL}{VAD}
38 \ncline{acCL}{VISU}
39 \ncline{acSC}{SR}
40 \ncline{acSC}{SD}
41 \ncline{acSC}{RD}
42 \ncline{acSU}{ICB}
43 \ncline{acSU}{PM}
44 \ncline{acST}{IR}
45 \ncline{acST}{MR}
46 %
47 \ncline{RD}{ICB}\naput[nrot=:U]{\umlStereoType{uses}}
48 \ncputicon{umlHerit}
49 \ncline{IR}{SR}\nbput[nrot=:D,npos=0.65]{\umlStereoType{uses}}
50 \ncputicon{umlHerit}
51 \end{pspicture}
52 }%end resizeORscalebox
53 \end{center}
54
55
56 % \end{document}

```



### C Exemple de diagramme de séquences



```

1 % \documentclass[11pt,a4paper,twoside]{article}
2 % \usepackage[T1]{fontenc}
3 % \usepackage[applemac]{inputenc}
4 % % \usepackage[latin1]{inputenc}
5 % \usepackage{pst-uml}
6 % \begin{document}
7
8 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9 % Placement des objet}
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 \begin{center}
13 \resizebox{\linewidth}{!}{%
14 \begin{psmatrix}[colsep=0.2,rowsep=0.5]
15 %
16 % la ligne 1 contient le nom des objets
17 [name=client]\umlClass{\underline{:Client}}{}
18 & [name=reseau]\umlClass{\underline{:R\'eseau}}{}
19 & [name=demande]\umlClass{\underline{:Demande}}{}
20 & [name=route]\umlClass{\underline{:Route}}{}
21 & [name=arete]\umlClass{\underline{:Ar\^ete}}{}
22 & [name=noeud]\umlClass{\underline{:Noeud}}{}
23 & [name=fenetre]\umlClass{\underline{:Fen\^etre}}{}
24 \\ [+0.5cm] %1
25 % ATTENTION les lignes vides telle que :
26 % & & & & & \
27 % sont inaccessible par (3,2)
28 %
29 {} & {} & {} & {} & {} & {} & {} & {} \
30 {} & {} & {} & {} & {} & {} & {} & {} \
31 {} & {} & {} & {} & {} & {} & {} & {} \
32 {} & {} & {} & {} & {} & {} & {} & {} \ % 5
33 %
34 {} & {} & {} & {} & {} & {} & {} & {} \
35 {} & {} & {} & {} & {} & {} & {} & {} \
36 {} & {} & {} & {} & {} & {} & {} & {} \
37 {} & {} & {} & {} & {} & {} & {} & {} \ [-0.5cm]
38 {} & {} & {} & {} & {} & {} & {} & {} \ [-0.5cm] % 10
39 %
40 {} & {} & {} & {} & {} & {} & {} & {} \ [-0.5cm]
41 {} & {} & {} & {} & {} & {} & {} & {} \ [-0.5cm]
42 {} & {} & {} & {} & {} & {} & {} & {} \ [+0.5cm]
43 {} & {} & {} & {} & {} & {} & {} & {} \
44 {} & {} & {} & {} & {} & {} & {} & {} \ [+0.5cm] % 15
45 %
46 {} & {} & {} & {} & {} & {} & {} & {} \
47 {} & {} & {} & {} & {} & {} & {} & {} \
48 {} & {} & {} & {} & {} & {} & {} & {} \ [-0.5cm]
49 {} & {} & {} & {} & {} & {} & {} & {} \

```

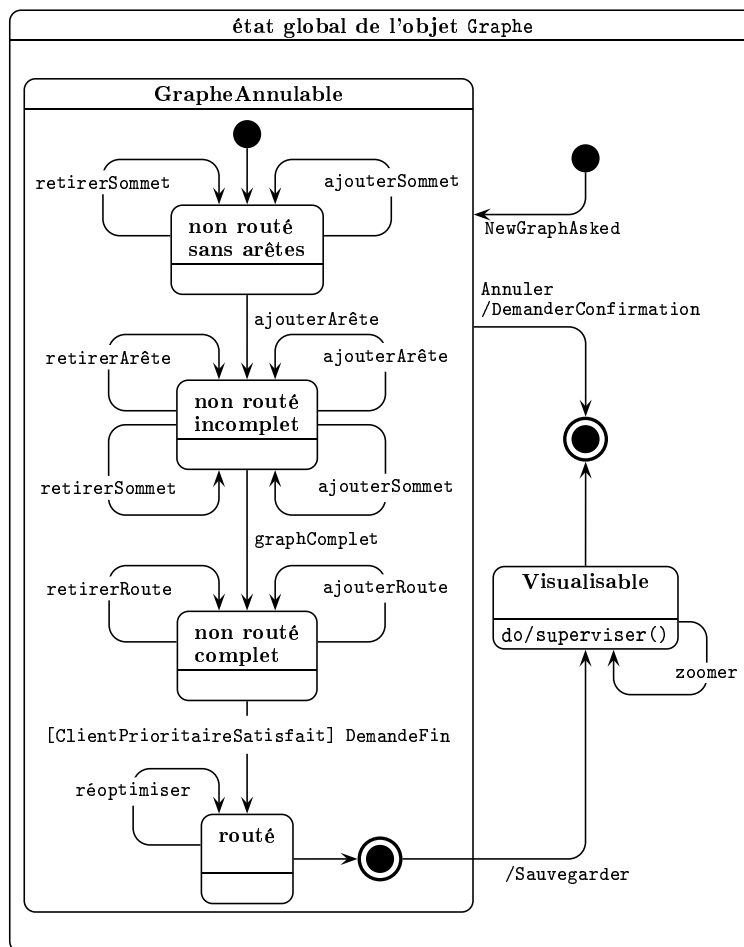
```

50 {} & {} & {} & {} & {} & {} & {} \\ % 20
51 %
52 {} & {} & {} & {} & {} & {} & {} \\[0cm] % 21 ([0cm] nécessaire : bug ?)
53 %
54 % Les noms pour les fins d'objets (invariant si nouvelles lignes)
55 [name=clientEnd]{}
56 & [name=reseauEnd]{}
57 & [name=demandeEnd]{}
58 & [name=routeEnd]{}
59 & [name=areteEnd]{}
60 & [name=noeudEnd]{}
61 & [name=fenetreEnd]{}
62 & \\[-0.5cm] % Saut de ligne sans vertic pour corrigé problème
63 %
64 % Le trait d'axe pour l'échelle des temps :
65 \ncline[linewidth=0.5pt,linestyle=solid,offset=-1.7,nodesep=0.0]%
66 {->}{client}{clientEnd}
67 \naput[npos=1]{\emph{t}}
68 %
69 % Les pointillés verticaux
70 \ncline[linestyle=dashed]{client}{clientEnd}
71 \ncline[linestyle=dashed]{reseau}{reseauEnd}
72 \ncline[linestyle=dashed]{demande}{demandeEnd}
73 \ncline[linestyle=dashed]{route}{routeEnd}
74 \ncline[linestyle=dashed]{arete}{areteEnd}
75 \ncline[linestyle=dashed]{noeud}{noeudEnd}
76 \ncline[linestyle=dashed]{fenetre}{fenetreEnd}
77 %
78 % Les connexions horizontales ave leur commentaires associés
79 \small\ttfamily% Fonctionne
80 \psset{labelsep=1.5mm}
81 \ncline{->}{2,1}{2,3}\naput*{listerDemandes()}
82 \ncline{->}{3,3}{3,1}\nbput*{demandes}
83 \ncline{->}{4,1}{4,2}\naput*{* visualiser(demande)}
84 \ncline{->}{5,2}{5,1}\nbput*{[déjàRouté(réseau) = false]}
85 \ncline{->}{6,2}{6,1}\nbput*{[acceptée(demande) = false]}
86 \ncline{->}{7,2}{7,4}\naput*{[acceptée(demande) = true] parcourir()}
87 \ncline{->}{8,4}{8,5}\naput*{lister()}
88 \ncline{->}{9,5}{9,6}\naput*{listerExtrémités()}
89 \ncline{->}{10,6}{10,5}
90 \ncline{->}{11,5}{11,4}
91 \ncline{->}{12,4}{12,2}
92 \ncline{->}{13,2}{13,1}
93 \ncline{->}{14,1}{14,7}\naput*{[acceptée(demande) = true] %
94 afficher(demande)}
95 \ncline{->}{15,7}{15,1}\nbput*{dessinerSurTerminal()}
96 \ncline{->}{16,1}{16,7}\naput*{* zoomer(zone)}
97 \ncline{->}{17,7}{17,5}\nbput*{arêtesInZone}
98 \ncline{->}{18,5}{18,3}\nbput*{estConcernée(demande)}
99 \ncline{->}{19,3}{19,5}

```

```
100     \ncline{->}{20,5}{20,7}\naput*{arêtesConcernées}
101     \ncline{->}{21,7}{21,1}\nbput*{rafraichirEcran()}
102     % \ncEVW[armA=2]{->}{4,3}{10,3} % Est Vertical West
103     %
104     \end{psmatrix}
105 }%end resizeORscalebox
106 \end{center}
107
108
109 % \end{document}
```

## D Exemple de diagramme d'états



```

1 % \documentclass[11pt,a4paper,twoside]{article}
2 % \usepackage[T1]{fontenc}
3 % \usepackage[applemac]{inputenc}
4 % % \usepackage[latin1]{inputenc}
5 % \usepackage{pst-uml}
6 % \begin{document}
7
8 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9 % définition des objets
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 \newcommand{\StateGlobal}{%
13   \umlState{état global de l'objet \texttt{Graphe}}{\umlEmptyBox{13cm}{16cm}}%
14 }
15 \newcommand{\StateNRSA}{%
16   \umlState{non routé \ \ sans arêtes}{\space}%
17 }
18 \newcommand{\StateNRI}{%
19   \umlState{non routé \ \ incomplet}{\space}%
20 }

```

```

21 \newcommand{\StateNRC}{%
22   \umlState{non routé \ \ complet}{\space}%
23 }
24 \newcommand{\StateROU}{%
25   \umlState{routé \ \ \mbox{}}{\space}%
26 }
27 \newcommand{\StateVisu}{%
28   \umlState{Visualisable \ \ \mbox{}}{do/superviser()}%
29 }
30 \newcommand{\StateAnu}{%
31   \umlState{GrapheAnnulable}{%
32     \hspace*{2.25cm}
33     \rmfamily% car normalement un corps d'état est en ttfamily
34     \begin{psmatrix}[colsep=1,rowsep=1.5,mnode=r]
35                                     \ \ [-1.4cm]
36       [name=StateInAnu] \umlStateIn   \ \ [-0.5cm]
37       [name=StateNRSA]  \StateNRSA    \ \ [0cm]
38       [name=StateNRI]  \StateNRI     \ \ [1cm]
39       [name=StateNRC]  \StateNRC     \ \ [0.5cm]
40       [name=StateROU]  \StateROU
41       & \umlPutStateOut{0,0}{StateOutAnu} \ \ [-1.5cm]
42       {} % boite vide NECESSAIRE sur la dernière ligne si vide !
43     \end{psmatrix}%
44     \hspace*{1.5cm}
45     %
46     % Connection propre à chaque node
47     %
48     {\ttfamily\small
49     %
50     % \input{75}{StateInAnu}{EntréeAnnulable}
51     % \input{-75}{StateOutAnu}{SortieAnnulable}
52     %
53     \ncEXS[offsetA=0.25,offsetB=0.5]{StateNRSA}{StateNRSA}%
54     \ncput*[npos=1.7]{ajouterSommet}
55     \ncWXS[offsetA=-0.25,offsetB=-0.5]{StateNRSA}{StateNRSA}%
56     \ncput*[npos=1.7]{retirerSommet}
57     %
58     \ncEXS[offsetA=0.25,offsetB=0.5]{StateNRI}{StateNRI}%
59     \ncput*[npos=1.7]{ajouterArête}
60     \ncWXS[offsetA=-0.25,offsetB=-0.5]{StateNRI}{StateNRI}%
61     \ncput*[npos=1.7]{retirerArête}
62     \ncEXN[offsetA=-0.0,offsetB=-0.5]{StateNRI}{StateNRI}%
63     \ncput*[npos=1.7]{ajouterSommet}
64     \ncWXN[offsetA=0.0,offsetB=0.5]{StateNRI}{StateNRI}%
65     \ncput*[npos=1.7]{retirerSommet}
66     %
67     \ncEXS[offsetA=0.25,offsetB=0.5]{StateNRC}{StateNRC}%
68     \ncput*[npos=1.7]{ajouterRoute}
69     \ncWXS[offsetA=-0.25,offsetB=-0.5]{StateNRC}{StateNRC}%
70     \ncput*[npos=1.7]{retirerRoute}

```

```

71 %
72 \ncWXS[offsetA=-0.25,offsetB=-0.5]{StateROU}{StateROU}%
73 \ncput*[npos=1.7]{réoptimiser}
74 %
75 % Connections INTERNODE interne au macro-état
76 %
77 \ncline{->}{StateInAnu}{StateNRSA}%
78 \naput [npos=0.3]{}%
79 %
80 \ncline{->}{StateNRSA}{StateNRI}%
81 \naput [npos=0.3]{ajouterArête}%
82 %
83 \ncline{->}{StateNRI}{StateNRC}%
84 % \naput [npos=0.3]{graphComplet}%
85 \naput{graphComplet}%
86 %
87 \ncline{->}{StateNRC}{StateROU}%
88 \ncput*[npos=0.3]{\umlStack{[ClientPrioritaireSatisfait] DemandeFin}}
89 %
90 \ncline{->}{StateROU}{StateOutAnu}%
91 %
92 }%
93 }%
94 }
95
96 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
97 % Placement des objets
98 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
99 \begin{center}
100 % \scalebox{0.5}{%} Fonctionne egalement
101 % \resizebox{0.9\linewidth}{!}{%}
102 \resizebox{!}{13cm}{%
103 \begin{pspicture}(-5,-9.5)(9.5,8)%\psgrid
104 %
105 \psset{%
106 lineararc=0.3,%
107 % arm=1.2,%
108 armA=1.2,%
109 armB=0.8,%
110 arrows=->,%
111 arrowsscale=2,%
112 ncurv=2,% instead of 0.67
113 }%
114 \rput(2.4,-0.75){\rnode{StateGlobal}{\StateGlobal}}%
115 %
116 \umlPutStateIn{6,5}{StateIn}%
117 %
118 \rput(0,-1){\rnode{StateAnu}{\StateAnu}}%
119 %
120 \rput(6,-3){\rnode{StateVisu}{\StateVisu}}%

```

```

121 %
122 \rput(6,0){\rnode{StateOut}{\umlStateOut}}%
123 %
124 % LES CONNECTIONS INTERNODE
125 %
126 {\ttfamily\small
127 %
128 \ncEXN[offsetA=-0.25,offsetB=-0.5,armA=0.5]{StateVisu}{StateVisu}%
129 \ncput*[npos=1.7]{zoomer}
130 %
131 \ncSW[offsetB=-5]{->}{StateIn}{StateAnu}%
132 \naput[npos=1.3]{NewGraphAsked}
133 %
134 \ncEN{->}{StateOutAnu}{StateVisu}%
135 \nbput[npos=0.9]{/Sauvegarder}
136 \ncline{->}{StateVisu}{StateOut}%
137 %
138 \ncES[offsetA=3]{->}{StateAnu}{StateOut}%
139 % \naput[npos=0.6]{Annuler}
140 % on ajoute un espace en début de chaque ligne " Annuler"..
141 \naput[npos=0.99]{\umlStack[umlAlign=1]%
142 {\ Annuler\\\ /DemanderConfirmation}}
143 }
144 \end{pspicture}%
145 }%end resizeORscalebox
146 \end{center}
147
148
149 % \end{document}

```



## E Listing du package pst-uml.sty

```

1  %%
2  %% This is file 'pst-uml.sty',
3  %%
4  %% IMPORTANT NOTICE:
5  %%
6  %% Package 'pst-uml.sty'
7  %%
8  %% Maurice Diamantini <diam@ensta.fr>
9  %%
10 %% May 07, 2003
11 %%
12 %% This program can be redistributed and/or modified under the terms
13 %% of the LaTeX Project Public License Distributed from CTAN archives
14 %% in directory macros/latex/base/lppl.txt.
15 %%
16 %% DESCRIPTION:
17 %% 'pst-uml' is a PSTricks package to draw UML diagrams
18 %%
19 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
20 % Fichier pst-uml.sty $Id: pst-uml.sty,v 1.30 1998/08/24 07:10:32 diam Exp $
21 %
22 % Ce fichier fournit des macro pstricks pour faciliter la création
23 % de diagrammes UML :
24 %
25 % Inspir\`e des exemples de Denis GIROU
26 %
27 % mise `a jour par diam@ensta.fr :
28 % 22/02/98 : création
29 % 15/03/98 : utilisation des cl\`e pour les option de \umlClass
30 % 10/11/04 : using pst-xkey and pstricks (hv)
31
32 \def\filename{pst-uml}
33 \def\fileBut{Quelques macros pstricks pour diagrammes UML}
34 \def\fileversion{0.81}
35 \def\filedate{2006/08/16}
36 \def\fileMailAutor{diam@ensta.fr}
37 %
38 \NeedsTeXFormat{LaTeX2e}
39 \ProvidesPackage{\filename}%
40   [\filedate\space v\fileversion\space-\space\fileMailAutor]
41 \typeout{\space\space\space But\space:\space\fileBut.}
42 %
43 % Chargement de pstricks puis de graphicx
44 % pstcol charge color dans le bon ordre, necessite un fichier color.cfg
45 \RequirePackage{pstricks} % Require a color.cfg file defined
46 % \RequirePackage{pst-grad}
47 % \RequirePackage{pst-plot}
48 % \RequirePackage{pst-coil}
49 % \RequirePackage{pst-text}
50 % \RequirePackage{pst-char}
51 \RequirePackage{pst-node}
52 % \RequirePackage{pst-3d}
53 % \RequirePackage{pst-eps}
54 % \RequirePackage{pst-fill}
55 \RequirePackage{pst-tree}
56 \RequirePackage{multido}
57
58 \RequirePackage{calc}
59 \RequirePackage{ifthen}
60

```

```

61 \RequirePackage{graphicx} % a charger APRES pstricks
62 \RequirePackage{pst-xkey}
63 \pst@addfams{pst-uml}
64 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
65 % Définition de quelques couleur par défaut utilisées pour UML
66 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
67 \definecolor{uml_greyLight} {rgb}{0.9,0.9,0.9}
68 \definecolor{uml_grey80} {rgb}{0.8,0.8,0.8}
69
70 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
71 % Quelques variable ou boite temporaires :
72 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
73
74 % compteur (i.e. entiers) pour les calculs temporaires
75 \newcounter{umlint@a}
76 \newcounter{umlint@b}
77 \newcounter{umlint@c}
78 \newcounter{umlint@d}
79 % longueurs pour les calculs temporaires
80 \newlength{\umldim@a}
81 \newlength{\umldim@b}
82 \newlength{\umldim@c}
83 \newlength{\umldim@d}
84 \newlength{\umldim@D}% Depth
85 \newlength{\umldim@H}% Height
86 \newlength{\umldim@W}% Width
87 % boite temporaire de memorisation des parametres
88 \newsavebox{\umlbox@a}
89 \newsavebox{\umlbox@b}
90 \newsavebox{\umlbox@c}
91 \newsavebox{\umlbox@d}
92 \newsavebox{\umlbox@out}
93 % boolean temporaires
94 \newboolean{umlbool@a}
95 \newboolean{umlbool@b}
96 \newboolean{umlbool@c}
97 \newboolean{umlbool@d}
98
99 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
100 % Affectation des valeurs par default standard de pstricks :
101 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
102
103 \psset{
104   % labelsep=0.5mm, % au lieu de +/- 2mm
105   labelsep=1mm, % au lieu de +/- 2mm
106   subgriddiv=1, % pour une grille de 1cm en pointilles
107   griddots=10,
108   gridlabels=6pt,
109   arrowscale=2,
110 }
111 \SpecialCoor
112
113 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
114 % Definition des parametres par défaut (nouvelles longueurs, ...)
115 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
116
117 % % For debug : ShowPosition
118 % %
119 % % usage :
120 % % \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
121 % %
122 % \newboolean{umlDebug}\setboolean{umlDebug}{false}
123

```

```

124 % Un entier pour Degoguer
125 % ATTENTION : si pb de portée de variable (si besoin compteur global) :
126 %         passer par \edef
127 %
128 \newcounter{umlDebug}
129 \define@key[pst-uml]{pst-uml}{umlDebug}{%
130     \setcounter{umlDebug}{#1}%
131     %
132     % \setcounter{umlint@a}{#1}
133     % \setcounter{umlDebug}{\the\umlint@a}%
134     %
135     % \pst@cntg=#1\relax%
136     % \edef\psk@Integer{\the\pst@cntg}%
137 }
138
139
140 % Alignement du texte dans la boite umlstack (l, c, r)
141 \newcommand{\umlstr@Align}{}
142 \define@key[psset]{pst-uml}{umlAlign}{\renewcommand{\umlstr@Align}{#1}}
143
144 % Positionnement vertical de la boite umlstack (t, b, c)
145 \newcommand{\umlstr@Pos}{}
146 \define@key[psset]{pst-uml}{umlPos}{\renewcommand{\umlstr@Pos}{#1}}
147
148 % Distance de séparation du texte à la boite umlstack
149 %   Influe sur colsep pour l'espace horizontal et sur des strut pour
150 %   l'espace vertical
151 \newlength\umldim@StackSep
152 \define@key[psset]{pst-uml}{umlStackSep}{\pssetlength\umldim@StackSep{#1}}
153
154 % cooéficient de modif de l'interligne de umlstack
155 \newcommand{\umlreal@StackLinesStretch}{}
156 \define@key[psset]{pst-uml}{umlStackLinesStretch}{%
157     \renewcommand{\umlreal@StackLinesStretch}{#1}}
158
159 % Largeur de la boite "umlStack"
160 \newlength\umldim@StackWidth
161 \define@key[psset]{pst-uml}{umlStackWidth}{\pssetlength\umldim@StackWidth{#1}}
162
163 % Largeur de la boite "classe"
164 \newlength\umldim@ClassWidth
165 \define@key[psset]{pst-uml}{umlClassWidth}{\pssetlength\umldim@ClassWidth{#1}}
166
167 % distance entre deux \hline dans la boite "classe"
168 \newlength\umldim@umlDoubleRuleSep
169 \define@key[psset]{pst-uml}{umlDoubleRuleSep}{\pssetlength\umldim@umlDoubleRuleSep{#1}}
170
171 % Chaîne "Template"
172 % umlParameter
173 \newcommand{\umlstr@umlParameter}{}
174 \define@key[psset]{pst-uml}{umlParameter}{\renewcommand{\umlstr@umlParameter}{#1}}
175
176
177 % % un parametre boolean
178 % \newif\ifuml@Shadow
179 % \define@key[psset]{pst-uml}{umlShadow}[true]{\@nameuse{uml@Shadow#1}}
180 % un parametre boolean
181 \newboolean{umlbool@Shadow}
182 \define@key[psset]{pst-uml}{umlShadow}[true]{\setboolean{umlbool@Shadow}{#1}}
183
184
185 % Epaisseur de ligne pour le dessin d'un acteur
186 \newlength\umldim@ActorLineWidth

```

```

187 \define@key[psset]{pst-uml}{umlActorLineWidth}{\pssetlength\umldim@ActorLineWidth{#1}}
188
189 % Nom d'un objet (umlState, ...) N'EST PAS UTILISE !
190 \newcommand{\umlname@umlTitle}{}
191 \define@key[psset]{pst-uml}{umlTitle}{\renewcommand{\umlname@umlTitle}{#1}}
192
193 % Affectation des valeurs par default sprecifique a pst-uml:
194 \psset{
195   umlClassWidth=0,
196   umlShadow=true,
197   umlActorLineWidth=0.6mm,
198   umlParameter=,
199   umlTitle=,
200   umlDoubleRuleSep=2mm,
201   umlStackLinesStretch=0.85,
202   umlStackSep=0,
203   umlStackWidth=0,
204   umlAlign=c,
205   umlPos=c,
206 }
207 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
208 % Quelques commandes utilitaires generales :
209 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
210
211 % pour passer plusieurs lignes (par \\) dans un parametre simple
212 %
213 % La difficulté était de gérer l'option "StackSep" pour le
214 % haut et le bas du texte.
215 % ATTENTION il faut s'assurer que cette commande fonctionne même
216 % si le texte passé en parametre commence ou finit par \hline !
217 %
218 %
219 \newcommand{\umlStack}[2][]{%
220   \psset{#1}%
221   \setlength{\tabcolsep}{\umldim@StackSep}%
222   \renewcommand{\arraystretch}{\umlreal@StackLinesStretch}%
223   \ifthenelse{\lengthtest{\umldim@StackWidth=Opt}}{%
224     % dimensionnement automatique
225     \begin{tabular}[\umlstr@Pos]{\umlstr@Align}%
226   }{% else
227     % dimensionnement fixe
228     \begin{tabular*}{\umldim@StackWidth}[\umlstr@Pos]{\umlstr@Align}%
229   }% endif
230   % \begin{tabular}[\umlstr@Pos]{\umlstr@Align}%
231     \setlength{\umldim@a}{1.5ex+\umldim@StackSep}%
232     \rule{0pt}{\umldim@a}%
233     \l[-2.5ex]%
234     #2%
235     \l[-2.5ex]%
236     \setlength{\umldim@a}{-\umldim@StackSep}%
237     \rule[-\umldim@StackSep]{0pt}{0pt}%
238   % \end{tabular}%
239   \ifthenelse{\lengthtest{\umldim@StackWidth=Opt}}{%
240     % dimensionnement automatique
241     \end{tabular}%
242   }{% else
243     % dimensionnement fixe
244     \end{tabular*}%
245   }% endif
246 }}
247
248 % FONCTIONNE MAIS LE PREMIER \rule gene si #2 commence par \hline
249 % \newcommand{\umlStack}[2][]{%

```

```

250 % \psset{#1}%
251 % \setlength{\tabcolsep}{\umldim@StackSep}%
252 % \renewcommand{\arraystretch}{\umlreal@StackLinesStretch}%
253 % \begin{tabular}{\umlstr@Align}%
254 % \setlength{\umldim@a}{1.5ex+\umldim@StackSep}%
255 % \rule{0pt}{\umldim@a}%
256 % #2%
257 % \setlength{\umldim@a}{-\umldim@StackSep}%
258 % \rule[-\umldim@StackSep]{0pt}{0pt}
259 % \end{tabular}%
260 % }}
261
262 % Pour creer une boite vide width, height
263 % Exemple \umlEmptyBox{7cm}{10cm}
264 \newcommand{\umlEmptyBox}[2]{%
265 \rule{#1}{0cm}%
266 \rule{0cm}{#2}%
267 }
268
269 % permet d'afficher <<MonStereoType>> sans serif.
270 \newcommand{\umlStereoType}[1]{%
271 {\footnotesize\ensuremath{\ll}}\textsf{#1}\{\footnotesize\ensuremath{\gg}}\}%
272 }
273
274 % permet d'afficher un triangle plein vers la droite.
275 % Pas d'option pour l'instant
276 \newcommand{\umlTriRight}[1][ ]{%
277 \psset{#1}%
278 \space
279 \begin{pspicture}(0,0)(1.1ex,1.1ex)%
280 \pstriangle[linewidth=0,fillstyle=solid,fillcolor=black,gangle=-90]%
281 (0.55ex,0.55ex)(1.1ex,1.1ex)%
282 \end{pspicture}%
283 \space
284 }}
285 % permet d'afficher un triangle plein vers la gauche.
286 % Pas d'option pour l'instant
287 \newcommand{\umlTriLeft}[1][ ]{%
288 \psset{#1}%
289 \space
290 \begin{pspicture}(0,0)(1.1ex,1.1ex)%
291 \pstriangle[linewidth=0,fillstyle=solid,fillcolor=black,gangle=90]%
292 (0.55ex,0.55ex)(1.1ex,1.1ex)%
293 \end{pspicture}%
294 \space
295 }}
296
297 % For drawing a grid of 1cm in pointed line
298 % but \showgrid could be already define by another package
299 % PLANTE SI \showgrid EST REDEFINIE DANS fvrtext.sty
300 % \providecommand{\showgrid}{%
301 % \psset{subgriddiv=1,griddots=10,gridlabels=6pt,}
302 % \psgrid
303 % }
304
305 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
306 % La macro principal pour dessiner une classe :
307 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
308
309 % Le corps de la commande est entre {{ }} pour assurer des changements
310 % LOCAUX des valeurs de parametres (tels que \psset{...})
311 \newcommand{\umlClass}[3][ ]{%
312 \psset{#1}% Reading keyOptions

```

```

313 \setlength{\doublerulesep}{\umldim@umlDoubleRuleSep}%
314 \def\next{#2}%
315 \ifx\next\empty%
316   % Title IS EMPTY (JAMAIS UTILISE ! : A VIRER ??)
317   \sbox{\umlbox@out}{\texttt{\umlStack[umlStackSep=1ex]{#3}}}%
318 \else%
319   % Title is NOT EMPTY
320   \def\next{#3}%
321   \ifx\next\empty%
322     % Body IS EMPTY
323     \sbox{\umlbox@out}{\textbf{\umlStack[umlStackSep=1ex]{#2}}}%
324   \else%
325     % Body is NOT EMPTY
326     \sbox{\umlbox@out}{\ttfamily%
327       \begin{tabular}{l}%
328         \hfil\normalfont\textbf{%
329           \setlength{\tabcolsep}{0pt}
330           \begin{tabular}{c}
331             #2
332           \end{tabular}
333         }%
334         \\ \hline%
335         #3%
336       \end{tabular}%
337     }%
338   \fi%
339 \fi%
340 % Output box is now define
341 %
342 % On dessine la boite finale :
343 \ifthenelse{\boolean{umlbool@Shadow}}{%
344   \psshadowbox[fillstyle=solid,framesep=0,fillcolor=uml_grey80]%
345     {\usebox{\umlbox@out}}%
346 }{%
347   {\setlength{\fboxsep}{0pt}\fbox{\usebox{\umlbox@out}}}%
348 }%
349 %
350 % On superpose éventuellement le template (rectangle pointillé)
351 %
352 \settowidth{\umldim@a}{\umlstr@umlParameter}%
353 \ifthenelse{\lengthtest{\umldim@a=0pt}}{%
354 }{%
355   \settoheight{\umldim@H}{\usebox{\umlbox@out}}% Height
356   \setlength{\umldim@a}{1.5ex}% dx
357   \setlength{\umldim@b}{\umldim@H - 0.3ex}% Height - dy
358   %
359   \rput[br](\umldim@a,\umldim@b){%
360     \psframebox[fillstyle=solid,fillcolor=white,linestyle=dashed]%
361       {\Large\textbf{\umlstr@umlParameter}}%
362   }%
363 }%
364 %
365 }}
366
367
368 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
369 % Memorisation d'une classe dans une boite
370 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
371
372 % Surcourage à la commande de dessin de classe :
373 % CECI FONCTIONNE MAL (la boite cree s'appel "#" )
374 % Je n'ai pas reussi en jonglant avec les \csname et autre \@nameuse
375 \newcommand{\umlSaveClass}[3] []{%

```

```

376 \newsavebox{\#2}%
377 \savebox{\#2}{%
378   \umlClass[#1]{#2}{#3}%
379 }%
380 }
381 % Pas de \umlUseClass car sera egalement utilisee pour les acteurs, ...
382 \newcommand{\umlUseBox}[1]{\usebox{\#1}}
383
384
385 %% Surcourage à la commande de dessin de classe :
386 %%
387 %% CECI FONCTIONNE PAS :
388 %%   \newsavebox{\csname#2\endcsname} PLANTE :
389 %% idem avec :
390 %%   \expandafter\newsavebox{\csname#2\endcsname}
391 %% ! LaTeX Error: Command \csnameClasse 1 bis\endcsname already defined.
392 %%
393 %% \newcommand{\umlSaveClass}[3] [] {%
394 %%   \typeout{*****avant newsavebox }
395 %%   \expandafter\newsavebox{\csname#2\endcsname}%
396 %%   \typeout{*****apres newsavebox }
397 %%   \savebox{\csname#2\endcsname}{%
398 %%     \typeout{*****apres savebox }
399 %%     \umlClass[#1]{#2}{#3}%
400 %%   }%
401 %% }
402 %% Pas de \umlUseClass car sera aussi utilisee pour les acteurs, ...
403 %% \newcommand{\umlUseBox}[1]{%
404 %%   \typeout{*****avant use box }
405 %%   \usebox{\csname#1\endcsname}%
406 %% }
407
408
409 %% Surcourage à la commande de dessin de classe :
410 %% CECICOMPILE MAIS FONCTIONNE PAS
411 %% affiche "Classe 1bisClasse 1bis" (nom double) sous la classe
412 %% \newcommand{\umlSaveClass}[3] [] {%
413 %%   \newsavebox{\@nameuse{#2}}%
414 %%   \savebox{\@nameuse{#2}}{%
415 %%     \umlClass[#1]{#2}{#3}%
416 %%   }%
417 %% }
418 %% Pas de \umlUseClass car sera aussi utilisee pour les acteurs, ...
419 %% \newcommand{\umlUseBox}[1]{%
420 %%   \usebox{\@nameuse{#1}}%
421 %% }
422
423 %% Surcourage à la commande de dessin de classe :
424 %% CECI FONCTIONNE PAS :
425 %% ! Illegal parameter number in definition of \@tempa.
426 %% Je n'ai pas réussi en jonglant avec les \csname et autre \@nameuse
427 %% \newcommand{\umlSaveClass}[3] [] {%
428 %%   \@namedef{boxname}{#2}
429 %%   \newsavebox{\expandafter\boxname}%
430 %%   \savebox{\expandafter\boxname}{%
431 %%     \umlClass[#1]{#2}{#3}%
432 %%   }%
433 %% }
434 %% Pas de \umlUseClass car sera aussi utilisee pour les acteurs, ...
435 %% \newcommand{\umlUseBox}[1]{%
436 %%   \@namedef{boxname}{#1}
437 %%   \usebox{\expandafter\boxname}%
438 %% }

```

```

439 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
440 % Memorisaton d'une classe dans une commande
441 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
442 %
443 % A faire (utiliser gdef par exemple ?)
444 % s'assurer que la commande cree est GLOGALE !!
445 %
446 %
447 % \newcommand{\umlDefClass}[3] []{%
448 %   \@ifundefined{\csname#2\endcsname}{%
449 %     % NOT DEFINE
450 %     \typeout{\^~Jcommande NON definie : "#2"^^J}%
451 %     \newcommand{\csname#2\endcsname}{%
452 %       \umlClass[#1]{#2}{#3
453 %     }
454 %   }{%
455 %     % IS DEFINE
456 %     \typeout{\^~Jcommande DEJA definie : "#2"^^J}%
457 %     \renewcommand{\csname#2\endcsname}{%
458 %       \umlClass[#1]{#2}{#3
459 %     }
460 %   }
461 % }
462 % }
463 % % Pas de \umlRunClass car sera aussi utilis6e pour les acteurs, ...
464 % \newcommand{\umlRun}[1]{\@username{#1}}
465 %
466 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
467 % La macro pour dessiner un acteur :
468 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
469 %
470 %
471 \newcommand{\umlActor}[2] []{{%
472   \psset{unit=0.8,arrows=-}%
473   \psset{#1}%
474   \begin{tabular}[t]{c}%
475     \begin{pspicture}(-0.5,0)(0.5,1.9)%
476       \psset{linewidth=\umldim@ActorLineWidth}%
477       \pscicle(0,1.6){0.3\psyunit}%      la tete
478       \psline(0cm,.65)(0cm,1.3)%      le tronc
479       \psline(-0.25,0cm)(0cm,0.65)%  jambe gauche
480       \psline(0.25,0cm)(0cm,0.65)%  jambe droite
481       \psline(-0.5,1)(0.5,1)%      les bras
482     \end{pspicture}\\
483     \begin{pspicture}(0,0)%
484       \setlength{\tabcolsep}{0pt}%
485       \begin{tabular}[t]{c}%
486         #2%
487       \end{tabular}%
488     \end{pspicture}\\
489   \end{tabular}%
490 }}
491 %
492 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
493 % La macro pour dessiner les useCase :
494 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
495 %
496 %
497 % \umlCase[keyOptions]{Title}{Body} : #2 = Title
498 % Option : idem que psovalbox
499 % Remarque : utiliser ovalnode et non pas rnode pour les placements
500 %
501 % BUG : \psovalbox creer une boite RECTANGULAIRE dans la quelle il y a

```



```

502 % un oval => difficile de séparer la CREATION du dessin et le PLACEMENT
503 % de celui-ci avec la création d'un node CONCLUSION : inutilisé !
504 % (Utiliser plutot la commande suivante \umlPutCase)
505 %
506 \newcommand{\umlCase}[2][ ]{{%
507   \psset{#1}%           Reading keyOptions
508   \sbox{\umlbox@out}{\umlStack{#2}}%
509   \psovalbox{\usebox{\umlbox@out}}
510 }}
511
512 % Pour placer directement dans un ovalnode
513 % UTILISATION :
514 % \umlPutCase[keyOptions{10,2}{C6}{Identifieur\contraintes bloquantes}%
515 %
516 \newcommand{\umlPutCase}[4][ ]{{%
517   \psset{#1}%           Reading keyOptions
518   \rput{#2}{\ovalnode[framesep=0]{#3}{\umlStack{#4}}}%
519 }}
520
521 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
522 % La macro pour dessiner les etats :
523 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
524
525 % \umlState[keyOptions]{Title}{Body} : #2 = Title ; #3 = Body
526 %
527 \newcommand{\umlState}[3][ ]{{%
528   \psset{#1}%           Reading keyOptions
529   % \typeout{***** DEBUT umlState : Titre=:#2:, Body=:#3:}%
530   \def\next{#2}%
531   \ifx\next\empty
532     % \typeout{**Title IS EMPTY : #2}
533     \sbox{\umlbox@out}{\texttt{\umlStack[umlStackSep=0.35em]{#3}}}%
534   \else
535     % \typeout{**Title is NOT EMPTY : #2}%
536     \def\next{#3}%
537     \ifx\next\empty
538       % \typeout{**Body IS EMPTY : #3}%
539       \sbox{\umlbox@out}{\textbf{\umlStack[umlStackSep=.35em]{#2}}}%
540     \else
541       % \typeout{**Body is NOT EMPTY : #3}%
542       \sbox{\umlbox@out}{\umlStack{%
543         \textbf{%
544           \umlStack[umlStackSep=.35em]{#2}%
545         }}\
546         \hline%
547         \texttt{%
548           \umlStack[umlStackSep=.35em,umlAlign=1]{#3}%
549         }}\%
550     }}%
551   \fi
552 \fi
553 \psframebox[framesep=0,cornersize=absolute,lineararc=0.2]%
554   {\usebox{\umlbox@out}}%
555 }}
556
557
558 \newcommand{\umlStateIn}[1][ ]{{%
559   \psset{#1}%           Reading keyOptions
560   \begin{pspicture}[](-0.25,-0.25)(0.25,0.25)
561     \pscircle[fillstyle=solid,fillcolor=black]{0.25}%
562   \end{pspicture}%
563 }}
564

```



```

628 % Commandes de placement relatif de type "LOGO"
629 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
630 %
631 % A partir d'une version TEX de Denis GIROU
632 % Inspired by an idea of Sebastian Rahtz <s.rahtz@elsevier.co.uk>
633 % (LGC, example 4-10-10)
634 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
635 %
636 % For debug : ShowPosition
637 %
638 % usage :
639 % \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
640 %
641 \newboolean{umlDebug}\setboolean{umlDebug}{false}
642 \newcommand{\ShowPosition}[1] []{%
643   \typeout{%
644     **** umlCpt@Direction=\the\value{umlCpt@Direction} - %
645     X=\the\X\space - Y=\the\Y^^J%
646     \space\space\space\space#1^^J%
647   }%
648 }%
649
650
651 % Direction (non utilisée pour l'instant) codé en nombre de quart de cercle
652 % (i x 90) => 0 right, 1 up, 2 left, 3 down
653 %
654 \newcounter{umlCpt@Direction}\setcounter{umlCpt@Direction}{-1}
655
656 % New length for recording current position :
657 \newlength{\umldim@PosX}\setlength\umldim@PosX{0pt}
658 \newlength{\umldim@PosY}\setlength\umldim@PosY{0pt}
659 % ALIAS for user ?
660 \newlength{\X}\setlength\X{\umldim@PosX}
661 \newlength{\Y}\setlength\Y{\umldim@PosY}
662
663 % Reset position
664 \newcommand{\ResetXY}{%
665   \pssetlength{\umldim@PosX}{0}%
666   \pssetlength{\umldim@PosY}{0}%
667   \pssetlength{\X}{0}%
668   \pssetlength{\Y}{0}%
669   \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
670 }
671 \ResetXY
672 % % For Getting X or Y position by the user.
673 % \newcommand{\X}{\the\umldim@PosX}
674 % \newcommand{\Y}{\the\umldim@PosY}
675
676 % Set position
677 \newcommand{\SetX}[1]{%
678   \pssetlength{\umldim@PosX}{#1}%
679   \pssetlength{\X}{#1}%
680   \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
681 }
682 \newcommand{\SetY}[1]{%
683   \pssetlength{\umldim@PosY}{#1}%
684   \pssetlength{\Y}{#1}%
685   \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
686 }
687 % IMPORTANT : essayer d'utiliser les nodes ou autre
688 % coordonnees speciales de pstricks pour \SetXY :
689 % Utilisation de \pst@getcoor et \pssetxlength
690 \newcommand{\SetXY}[1]{%

```

```

691 \pssetxlength{\umldim@PosX}{#1}%
692 \pssetylength{\umldim@PosY}{#1}%
693 \pssetxlength{\X}{#1}%
694 \pssetylength{\Y}{#1}%
695 \ifthenelse{\value{umlDebug} > 0}{\ShowPosition}{}%
696 }
697
698 \newcommand{\incrX}[1]{%
699 % record the direction for later (NOT DONE)
700 % \setcounter{umlCpt@Direction}{0} % if #1 is positive
701 % \setcounter{umlCpt@Direction}{3} % if #1 is negative
702 % increment X position
703 \psaddtolength{\umldim@PosX}{#1}
704 \psaddtolength{\X}{#1}
705 }
706 \newcommand{\incrY}[1]{%
707 % record the direction for later (NOT DONE)
708 % \setcounter{umlCpt@Direction}{1} % if #1 is positive
709 % \setcounter{umlCpt@Direction}{4} % if #1 is negative
710 % increment Y position
711 \psaddtolength{\umldim@PosY}{#1}
712 \psaddtolength{\Y}{#1}
713 }
714 % For moving Est, West, North or Sud
715 \newcommand{\moveE}[1]{\incrX{#1}}
716 \newcommand{\moveW}[1]{\incrX{-#1}}
717 \newcommand{\moveN}[1]{\incrY{#1}}
718 \newcommand{\moveS}[1]{\incrY{-#1}}
719
720
721 % % % \newcommand{\Right}[1]{%
722 % % % % record the direction for later
723 % % % \setcounter{umlCpt@Direction}{0}
724 % % % % increment X position
725 % % % \psaddtolength{\umldim@PosX}{#1}
726 % % % }
727 % % % \newcommand{\Up}[1]{%
728 % % % % record the direction for later
729 % % % \setcounter{umlCpt@Direction}{1}
730 % % % % increment Y position
731 % % % \psaddtolength{\umldim@PosY}{#1}
732 % % % }
733 % % %
734 % % % \newcommand{\Left}[1]{%
735 % % % % record the direction for later
736 % % % \setcounter{umlCpt@Direction}{2}
737 % % % % decrement X position
738 % % % \psaddtolength{\umldim@PosX}{-#1}
739 % % % }
740 % % %
741 % % % \newcommand{\Down}[1]{%
742 % % % % record the direction for later
743 % % % \setcounter{umlCpt@Direction}{3}
744 % % % % decrement Y position
745 % % % \psaddtolength{\umldim@PosY}{-#1}
746 % % % }
747
748 % % Put an object at current coordinates
749 % \newcommand{\rputXY}[2][ ]{%
750 % \psset{#1}%
751 % \rput{\umldim@PosX,\umldim@PosY}{#2}
752 % }
753 %

```

```

754 % Put an object at current coordinates
755 \newcommand{\rputXY}[2] [] {%
756   \psset{#1}%
757   \rput(\umldim@PosX,\umldim@PosY){#2}%
758 }
759
760
761
762 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
763 % Définition des "fleches" de UML (sous forme de labels)
764 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
765
766 \newcommand{\ncputicon}[2] [] {%
767   \psset{npos=0,nrot=:D}% default : au debut et vers la gauche
768   \psset{#1}%
769   \@ifundefined{umlicon@#2}%
770     {\typeout{^^JERREUR : FLECHE NON definie "#2"^^J}}%
771     {}%
772   \ncput{\@nameuse{umlicon@#2}}
773 }
774 % Definition des fleches de base (vers la droite)
775 % Utiliser nrot=:U pour une orientation vers la droite (Up)
776 % Utiliser nrot=:D pour une orientation vers la gauche (Down)
777 \newcommand{\umlicon@umlV}{%
778   \psline(-0.4,0.2)(0,0)(-0.4,-0.2)
779 }
780 \newcommand{\umlicon@umlHerit}{%
781   \pstriangle[gangle=-90,
782     linestyle=solid,
783     fillstyle=solid,
784     fillcolor=white,
785     ](-0.6,0)(0.4,0.6)% (centre)(Width,Length)
786 }
787 \newcommand{\umlicon@umlAgreg}{%
788   \psdiamond[linestyle=solid,
789     fillstyle=solid,
790     fillcolor=white,
791     ](-0.25,0)(-0.25,0.15)
792 }
793 \newcommand{\umlicon@umlCompos}{%
794   \psdiamond[linestyle=solid,
795     fillstyle=solid,
796     fillcolor=black,
797     ](-0.25,0)(-0.25,0.15)
798 }
799
800 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
801 % Nouvelles définition pour les interconnexions
802 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
803
804 % Ces connecteurs sont une surcouche aux différents connecteurs
805 % proposés par pstricks.
806 % Leur but est de simplifier le tracé des liens en se restreignant aux
807 % directions horizontales et verticales.
808 %
809 % Le principe est d'indiquer dans le nom même de la commande le nombre
810 % de segments à tracer et leur direction :
811 % - E, W, N, S pour Est, West, North, Sud
812 % - H, V pour Horizontal; Vertical
813 % - D pour diagonal
814 % - X pour indifférent
815
816 % Un seul segments

```

```

817 % Effet de bord : en fait, un deuxieme segment est dessin . Mais celui
818 % n'apparait g n ralement pas car il longe la fronti re de boite
819 \newpsobject{ncE}{ncangle}{angleA=0,angleB=180,armB=0,npos=0.5,nodesepB=-0.5pt}
820 % \newpsobject{ncE}{ncangle}{angleA=0,angleB=180,armB=0,npos=0.5}
821 \newpsobject{ncW}{ncangle}{angleA=180,angleB=0,armB=0,npos=0.5}
822 \newpsobject{ncN}{ncangle}{angleA=90,angleB=-90,armB=0,npos=0.5}
823 \newpsobject{ncS}{ncangle}{angleA=-90,angleB=90,armB=0,npos=0.5}
824
825 % Deux segments
826 \newpsobject{ncEN}{ncangle}{angleA=0,angleB=-90,armB=0}
827 \newpsobject{ncES}{ncangle}{angleA=0,angleB=90,armB=0}
828 \newpsobject{ncWN}{ncangle}{angleA=180,angleB=-90,armB=0}
829 \newpsobject{ncWS}{ncangle}{angleA=180,angleB=90,armB=0}
830 \newpsobject{ncNE}{ncangle}{angleA=90,angleB=180,armB=0}
831 \newpsobject{ncNW}{ncangle}{angleA=90,angleB=0,armB=0}
832 \newpsobject{ncSE}{ncangle}{angleA=-90,angleB=180,armB=0}
833 \newpsobject{ncSW}{ncangle}{angleA=-90,angleB=0,armB=0}
834
835 % Trois segments
836 % On peut utiliser armA ou armB pour imposer la longueur des extr mit s
837 %
838 % remplace \ncbar (connecteurs en forme de U)
839 \newpsobject{ncEVW}{ncangles}{angleA=0,angleB=0}
840 \newpsobject{ncWVE}{ncangles}{angleA=180,angleB=180}
841 \newpsobject{ncSHN}{ncangles}{angleA=-90,angleB=-90}
842 \newpsobject{ncNHS}{ncangles}{angleA=90,angleB=90}
843
844 % connecteurs en forme de Z (mais \a angles droits)
845 \newpsobject{ncEVE}{ncangles}{angleA=0,angleB=180}
846 \newpsobject{ncWVW}{ncangles}{angleA=180,angleB=0}
847 \newpsobject{ncNHN}{ncangles}{angleA=90,angleB=-90}
848 \newpsobject{ncSHS}{ncangles}{angleA=-90,angleB=90}
849
850
851 % connecteurs 3 segments dont segment median en diagonale (incomplet)
852 \newpsobject{ncEDE}{ncdiag}{angleA=0,angleB=180}
853 \newpsobject{ncWDW}{ncdiag}{angleA=180,angleB=0}
854 \newpsobject{ncNDN}{ncdiag}{angleA=90,angleB=-90}
855 \newpsobject{ncSDS}{ncdiag}{angleA=-90,angleB=90}
856
857
858 % quatre (voire trois) segments :
859 \newpsobject{ncSXE}{ncangles}{angleA=-90,angleB=180}
860 \newpsobject{ncSXW}{ncangles}{angleA=-90,angleB=0}
861 \newpsobject{ncEXS}{ncangles}{angleA=0,angleB=90}
862 \newpsobject{ncEXN}{ncangles}{angleA=0,angleB=-90}
863 \newpsobject{ncWXS}{ncangles}{angleA=180,angleB=90}
864 \newpsobject{ncWXN}{ncangles}{angleA=180,angleB=-90}
865 \newpsobject{ncNXE}{ncangles}{angleA=90,angleB=180}
866 \newpsobject{ncNXW}{ncangles}{angleA=90,angleB=0}
867
868
869
870
871 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
872 % FIN
873 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
874 \typeout{Package \filename.sty is loaded.}
875 \endinput

```

pour connaitre le numero de la derniere page