

Carta Genius

Guide utilisateur

Version 5.2.0

20 juin 2006

Amaury Bouchard
Association Pandocréon

Statut de ce document

Ce document décrit le logiciel Carta Genius et explique pas-à-pas l'ensemble des fonctionnalités qu'il propose.

Auteur

Amaury Bouchard (amaury@amaury.net)

Merci à...

Corinne pour sa patience infinie, Vicnet et Nono pour avoir activement testé Carta Genius.

Copyright (c) 2004-2005, Amaury Bouchard

La permission est donnée de copier, distribuer et/ou modifier ce document selon les termes de la Licence de Documentation Libre GNU, dans sa version 1.2 ou ultérieure publiée par la Free Software Foundation ; sans section invariante, sans texte de couverture, ni texte de dos de couverture. Plus d'information, et une copie de cette licence sont incluses dans la section intitulée « GNU Free Documentation License ».

Table des matières

1. Préface.....	6
2. Premiers pas.....	7
2.1 Installation.....	7
2.2 Utilisation.....	7
2.3 Déclarations de base.....	9
2.4 Méta-données.....	10
2.5 Types de données.....	11
3. Objets graphiques simples.....	12
3.1 Rectangle.....	12
3.2 Rectangle arrondi.....	13
3.3 Cercle.....	14
3.4 Ligne.....	14
3.5 Grille.....	15
3.6 Hexagone.....	16
3.7 Rotation des objets.....	17
3.8 Pointillés.....	17
4. Objets graphiques complexes.....	19
4.1 Courbe de Bézier.....	19
4.2 Polygone.....	20
5. Zones de texte.....	22
5.1 Utilisation de base.....	22
5.2 Alignement du texte.....	23
5.3 Interlignage.....	24
5.4 Cadres.....	24
5.5 Rotation.....	26
5.6 Adaptation de la taille du texte.....	27
5.7 Caractères spéciaux.....	28
5.8 Polices de caractères intégrées.....	29
5.9 Polices externes.....	29
5.10 Soulignage, surlignage, barrage.....	30
5.11 Rayures horizontales.....	31
5.12 Police par défaut.....	32
5.13 Espacement entre les caractères.....	33
5.14 Echelle horizontale.....	34
6. Texte formaté.....	36
6.1 Définition de texte enrichi.....	36
6.2 Zones d'affichage.....	37
7. Insertion d'images.....	39
7.1 Insertion directe.....	39
7.2 Adaptation.....	40
7.3 Utilisation multiple.....	40
7.4 Types d'images et transparence implicite.....	42
7.5 Transparence explicite.....	42
7.6 Réutilisation de masques.....	44
8. Options concernant le papier.....	45
8.1 Positionnement des cartes.....	45
8.2 Format de papier.....	45
8.3 Marges et traits de coupe.....	45
8.4 Fond-perdu.....	46

8.5	Format et espacement des cartes.....	47
8.6	Utilisation multiple de la même carte.....	48
9.	Gabarits.....	49
9.1	Déclaration de gabarit simple.....	49
9.2	Gabarits multiples.....	50
9.3	Sous-gabarits.....	51
9.4	Dos des cartes.....	52
9.5	Appels en ligne des gabarits.....	53
10.	Options graphiques.....	55
10.1	Transparence des objets graphiques.....	55
10.2	Mélange des couleurs.....	56
11.	Evaluation d'expressions.....	58
11.1	Expressions dans les attributs.....	58
11.2	Types de constantes et de variables.....	59
11.3	Variables dans les expressions.....	60
11.4	Déclaration de variables.....	61
11.5	Déclaration initiale des variables.....	63
11.6	Insertion d'expressions dans du texte.....	64
11.7	Variables de couleur.....	65
11.8	Fonctions.....	65
11.9	Variables de variables.....	67
12.	Instructions de branchement et expressions booléennes.....	69
12.1	Instructions conditionnelles.....	69
12.2	Fonctions (et opérateurs) de comparaison.....	70
12.3	Fonctions (et opérateurs) logiques.....	71
12.4	Boucles.....	72
13.	Messages d'information.....	74
14.	Fichiers multiples.....	75
14.1	Fonctionnement.....	75
14.2	Répertoires d'inclusion.....	76
14.3	Application aux images.....	77
14.4	Application aux polices de caractères.....	77
15.	Extensions.....	78
15.1	Fabrication d'étuis de cartes.....	78
Annexe A.	Compilation de Carta Genius.....	82
A.1	PDFLib.....	82
A.2	Compilation dynamique.....	82
A.3	Compilation statique.....	82
A.4	Compilation sous Windows.....	83
Annexe B.	Installation de Carta Genius.....	84
B.1	Sous Linux.....	84
B.2	Sous Windows - compilation.....	84
B.3	Sous Windows – installeur.....	84
Annexe C.	Interface graphique Carta-GUI.....	85
Annexe D.	Options du programme à l'exécution.....	86
Annexe E.	Formats de papier supportés.....	87
E.1	Formats de papiers standards (A).....	87
E.2	Formats de papiers standards agrandis (RA).....	87
E.3	Formats de papiers standards agrandis supplémentaires (SRA).....	87
E.4	Formats de papiers standards additionnels (B).....	87
E.5	Formats de papiers non standards.....	88

E.6 Formats d'enveloppes standards habituels (C).....	88
E.7 Formats d'enveloppes standards supplémentaires (D).....	88
E.8 Formats d'enveloppes standards supplémentaires (E).....	88
E.9 Formats d'enveloppe non standard.....	89
E.10 Formats ID (cartes d'identité, passeports, cartes de crédit, ...)	89
Annexe F. Polices de caractères intégrées.....	90
Annexe G. Variables et fonctions de l'évaluateur d'expressions.....	91
G.1 Variables contextuelles.....	91
G.2 Variables de formats de page.....	91
G.3 Variables de couleurs.....	92
G.4 Variables additionnelles.....	92
G.5 Fonctions de transtypage.....	93
G.6 Fonctions de comparaison.....	93
G.7 Fonctions d'opérations booléennes.....	93
G.8 Fonctions de calcul.....	94
G.9 Fonctions sur les éléments.....	95
G.10 Fonctions sur les chaînes.....	95
G.11 Fonctions de test.....	95
G.12 Fonctions de date et heure.....	95
G.13 Fonctions d'aléa.....	97
Annexe H. Convertir des polices TrueType en polices Postscript Type 1.....	98
H.1 Conversion locale.....	98
H.2 Conversion en ligne.....	98
Annexe I. Générer des images bitmap.....	99
I.1 En local.....	99
I.2 En ligne.....	99
Annexe J. Table iso-8859-1.....	100
Annexe K. Versions de Carta Genius.....	103
Annexe L. Versions de ce document.....	107
Annexe M. Licence de ce document : GNU Free Documentation license.....	108
K.1 Preamble.....	108
K.2 Applicability and definitions.....	108
K.3 Verbatim copying.....	109
K.4 Copying in quantity.....	109
K.5 Modifications.....	110
K.6 Combining documents.....	111
K.7 Collections of documents.....	112
K.8 Aggregation with independent work.....	112
K.9 Translation.....	112
K.10 Termination.....	112
K.11 Future revisions of this license.....	112
Annexe N. Licence de Carta Genius : GNU General Public license.....	114
L.1 Preamble.....	114
L.2 Terms and conditions for copying, distribution and modification.....	115

1. Préface

Carta Genius est un logiciel dont la destination première est de générer des fichiers PDF de qualité professionnelle, contenant des cartes de jeu, avec un certain nombre de traitements automatiques qui en faciliteront l'utilisation par un imprimeur professionnel ; les cartes sont placées sur les pages en fonction de la dimension des pages, le dos des cartes sont disposés en fonction du côté par lequel les pages sont tournées, des traits de coupe sont placés au bon endroit, un fond-perdu peut être employé pour un meilleur résultat.

Incidemment, le logiciel offre des possibilités de mise en page d'éléments graphiques qui en font un outil de choix pour tout type de document.

Carta Genius a été développé pour satisfaire les besoins de l'association Pandocréon au cours de la réalisation de plusieurs jeux de société. En plus de servir pour générer des fichiers PDF contenant les cartes de ces jeux, le logiciel a aussi été utilisé pour faire les boîtes de jeu ainsi que divers productions graphiques telles que papier à lettre, cartes de visite, prospectus, etc. La plupart de ces réalisations sont disponibles au téléchargement sur le site www.pandocreon.com.

Ce logiciel utilise PDFLib, une bibliothèque de création de fichiers PDF. Elle est commerciale, mais une version (PDFLib Lite) est disponible gratuitement et sous licence libre lorsqu'elle est utilisée au sein d'un logiciel qui est lui-même sous licence libre, ce qui est le cas de Carta Genius. Plus d'informations sur la PDFLib et sur la licence de PDFLib Lite sur le site www.pdflib.com.

J'espère qu'en lisant ce manuel vous prendrez goût à l'utilisation de Carta Genius. Pour toute question, suggestion, remontée de bug ou pour signaler l'utilisation que vous faites du logiciel, n'hésitez pas à m'écrire à l'adresse amaury@amaury.net.

Vous pouvez retrouver tous les exemples présent dans cette documentation, dans l'arborescence de documentation des archives du logiciel. Vous pouvez ainsi les tester directement sans avoir à les retaper.

Bonne lecture.

Amaury Bouchard

2. Premiers pas

Même si Carta Genius peut être utilisé pour réaliser tout type de documents, il se destine initialement à la création de jeux de cartes. Plusieurs concepts du logiciel découlent de cela. Nous allons donc commencer par faire un premier document contenant quelques cartes.

2.1 Installation

Le programme est disponible sous forme de sources à compiler, sous forme binaire (directement exécutable sous Linux ou Windows), avec un programme d'installation graphique (sous Windows) ou encore utilisable en ligne sur le Web à l'adresse :

<http://www.pandocreon.com/logiciels/carta-genius>

Référez-vous à l'Annexe A. pour connaître la marche à suivre pour compiler vous-même le programme. Dans un premier temps, n'hésitez pas à utiliser la version en ligne du logiciel ; ses fonctions ne sont pas réduites, seul les temps de chargement vous limiteront lorsque vous voudrez créer de vrais documents complets.

2.2 Utilisation

Consultez l'Annexe B. pour connaître les options utilisables sur la ligne de commande lors de l'exécution du programme. Mais basiquement, l'idée est de fournir des ordres en entrée, et le logiciel crée en sortie un fichier PDF.

Carta Genius s'utilise en rédigeant un document XML, c'est-à-dire un document texte avec des balises permettant d'organiser hiérarchiquement l'information. Ce fichier va contenir tous les ordres qui aboutiront au fichier PDF. Les autres données nécessaires (images, polices de caractères) sont séparées mais le fichier XML y fait référence.

Voyons un premier fichier XML, dont l'effet sera de générer deux cartes, une contenant un cercle bleu, l'autre un carré vert.

Ligne 1 : Déclaration usuelle au début d'un fichier XML.

Ligne 2 : Toutes les informations d'un fichier XML Carta Genius sont contenues dans un élément `carta-genius`.

Ligne 3 : Il est possible de faire un certain nombre de déclarations au début du fichier, mais elles ne sont pas utilisées ici. Un élément `deck` est utilisé pour contenir les définitions des cartes.

Ligne 4 : L'élément `paper` sert à définir le papier utilisé. C'est ici du papier au format A4 (21cm par 29,7cm), avec une marge de 2 centimètres sur le tour.

Ligne 5 : L'élément `cardsize` permet de définir le format des cartes. Nous avons choisi du A7 (7,4cm par 10,5cm). Comme pour le papier, il est possible d'utiliser des formats standardisés, ou bien de spécifier une largeur et une hauteur.

Ligne 6 : Début de la définition de la première carte.

Ligne 7 : On définit ici un cercle, dont le centre est placé aux coordonnées 3cm par 5cm, avec un rayon de 2cm, et rempli de bleu.

Ligne 8 : Fin de la déclaration de la première carte.

Ligne 9 : Déclaration de la deuxième carte.

Lignes 10 et 11 : On crée un rectangle, dont le coin inférieur-gauche est aux coordonnées 2cm par 3cm, qui fait 3 centimètre de large et de haut, et rempli de vert.

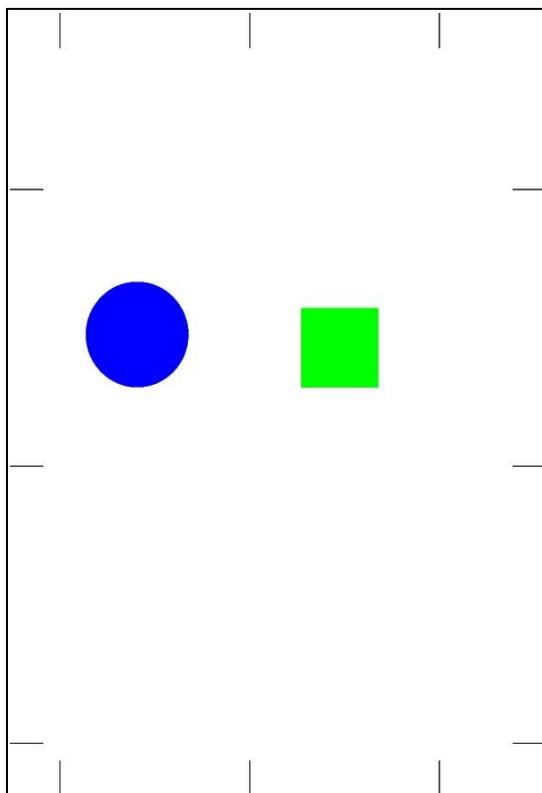
Ligne 12 : Fin de la déclaration de la deuxième carte.

Ligne 13 : Fin de la déclaration du jeu de cartes.

Ligne 14 : Fin du fichier

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a7"/>
06     <card>
07       <circle x="3cm" y="5cm" radius="20mm" fill-color="#0000ff"/>
08     </card>
09     <card>
10       <box x="2cm" y="3cm" width="3cm" height="3cm"
11         fill-color="#00ff00"/>
12     </card>
13   </deck>
14 </pandocreon:carta-genius>
```

Si vous exécutez Carta Genius en lui fournissant ce fichier, vous devriez récupérer un document PDF de la forme :



(Une bordure a été ici ajoutée sur le tour de l'image)

On voit les deux cartes l'une à côté de l'autre, et les traits de coupe qui sont placés dans la marge. L'utilité des traits de coupe est de faciliter la découpe des cartes par un imprimeur professionnel (en utilisant un massicot électrique) ; ils restent dans les marges, et donc à l'extérieur des cartes elles-mêmes, afin de ne pas apparaître après la découpe.

On peut aussi voir qu'avec cette taille de marge, et ces formats de papier et de carte, on peut mettre 4 cartes sur une page. Le logiciel place automatiquement les cartes sur les pages.

2.3 Déclarations de base

Les fichiers XML Carta Genius sont construits de la manière suivante :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <info author="Amaury Bouchard" title="Exemple" />
04   <pdf version="1.4"/>
05   <unit base="mm"/>
06   <back side="height" reverse="no"/>
07   <fonts>
08     ...
09   </font>
10   <images>
11     ...
12   </images>
13   <templates>
14     ...
15   </templates>
16   <includes>
17     ...
18   </includes>
19   <deck>
20     <paper .../>
21     <cardsize .../>
22     <card>
23       ...
24     </card>
25   </deck>
26   <deck>
27     ...
28   </deck>
29 </pandocreon:carta-genius>
```

Lignes 1 et 2 : Vous connaissez maintenant.

Ligne 3 : Vous pouvez renseigner plusieurs informations, qui seront intégrées dans le document PDF généré. Ces informations (auteur, titre, sujet, mots-clés, copyright, version, langue et notes) sont ensuite visible dans les méta-données PDF, accessibles dans Acrobat Reader par le menu Fichier->Propriétés.

Ligne 4 : Il est possible de spécifier la version du format PDF qui doit être utilisée pour générer le fichier. Cette information est optionnelle, si vous ne la donnez pas la valeur par défaut est utilisée (PDF version 1.4), qui est habituellement la bonne. Les valeurs possibles sont '1.3', '1.4' ou '1.5'. N'utilisez pas la version 1.3 si votre document contient de la transparence.

Ligne 5 : Cette déclaration, optionnelle elle aussi, permet de dire au programme quelle est l'unité de mesure par défaut de votre document. Cela sera utilisé si certaines valeurs de vos objets graphiques sont données sans leur unité (par exemple "25" sera interprété comme "25mm" si l'unité par défaut est le millimètre). Les valeurs possibles sont 'pt' (le point), 'mm' (millimètre), 'cm' (centimètre) ou 'in' (« inch », pouce en français). La valeur par défaut est le millimètre.

Ligne 6 : Cette déclaration optionnelle contient deux arguments, qui sont indépendants l'un de l'autre (et chacun est optionnel). L'argument `side` sert à spécifier le côté de retournement du papier ; s'il vaut "height" le dos des cartes sera placé sur les pages de manière à ce que les feuilles soient retournées suivant l'axe vertical ; s'il vaut "width" ce sera suivant l'axe horizontal. Bien évidemment, cette propriété n'est utilisée que si un verso est défini pour les cartes. L'argument `reverse` permet de demander à inverser l'ordre des pages ; la dernière page apparaît alors en premier dans le PDF, et la première en dernier.

Lignes 7 à 8 : Déclaration optionnelle des polices de caractère à utiliser. Un certain nombre de fontes sont incluses par défaut dans la norme PDF, mais il est possible d'en inclure de nouvelles. Plus d'information sur les fontes au Chapitre 5.9.

Lignes 9 à 10 : Déclaration optionnelle des images qui seront utilisées dans le PDF. En déclarant une image, il est possible de la réutiliser autant de fois que désiré sans alourdir inutilement le fichier. Plus d'information sur la manipulation des images au Chapitre 7.

Lignes 11 à 12 : Déclaration optionnelle des gabarits. Si des éléments graphiques se répètent entre plusieurs cartes, il suffit de définir un gabarit renfermant ces éléments, puis y faire référence lors de la création des cartes. Une même carte peut incorporer plusieurs gabarits, et les gabarits peuvent se référencer entre eux. Plus d'information sur les gabarits au Chapitre 9.

Lignes 13 à 14 : Déclaration optionnelle d'inclusion de fichiers. Il est possible d'avoir plusieurs fichiers XML et de vouloir les utiliser en commun pour créer un document PDF, ou encore de vouloir fragmenter un fichier pour le rendre plus facile à maintenir et mettre à jour. Plus d'information sur l'inclusion de fichiers au Chapitre 14.

Lignes 15 à 20 : Déclaration d'un deck. Pour être traité, un fichier doit contenir au moins un deck avec au moins une carte.

Lignes 21 à 22 : Il est possible de définir autant de decks que vous le souhaitez. Cela permet de définir des pages ou des cartes de tailles différentes au sein d'un même document PDF.

2.4 Méta-données

Nous venons de voir dans l'exemple précédent qu'il était possible d'insérer des informations relatives au document généré, qui sont intégrées au fichier PDF créé par Carta-Genius. Ces informations sont indiquées grâce à des attributs de l'élément `info` du fichier XML. Cet élément peut contenir les attributs suivants :

- `author` : Le ou les auteurs du document. Si vous souhaitez donner les adresses email, utilisez la convention des courriers électroniques (exemple : Amaury Bouchard <amauryamaury.net>).
- `title` : Le titre du document.
- `subject` : Le sujet auquel le document se rapporte.
- `keywords` : Liste de mots-clés décrivant le document, permettant de faciliter la recherche.
- `copyright` : Informations concernant le copyright et/ou la licence s'appliquant au document.
- `version` : Numéro de version du fichier, éventuellement en précisant la date de versionnement.
- `langue` : La langue du document.
- `note` : Toute autre information.

Notez que la présence de l'élément `info` est optionnelle ; tous ses attributs sont optionnels eux aussi (en particulier, ne renseignez pas le champ `subject` si le sujet est propre au document, et est donc contenu dans le titre).

2.5 Types de données

Dans les documents XML de Carta Genius, les paramètres peuvent être de plusieurs types :

scalaire : Un simple nombre à virgule (ex: 23.56, 123, 0.004). La notation est de type américaine, les virgules sont représentées par des points.

valeur : Distance, c'est-à-dire un scalaire avec une unité (ex: 12cm, 5.56in, 198mm)

couleur : Les couleurs sont spécifiées de la même manière que dans les pages HTML. Elles commencent par un caractère dièse, suivi de 6 caractères. Chaque groupe de 2 caractères représente une couleur (rouge, puis vert, puis bleu), dont la valeur est exprimée en hexadécimal. Pour faire du blanc, il faut que toutes les couleurs aient leurs valeurs au maximum, ce qui donne "#ffffff" ('ff' égale 255). Pour faire du noir, c'est l'inverse, tout à zéro : "#000000". Un rouge vif sera "#ff0000", alors qu'un rouge foncé sera "#800000" ('80' en hexadécimal donne 128 en décimal, soit environ la moitié de la valeur maximale). Pour faire du jaune, il vous faudra mélanger du rouge et du vert, et enlever le bleu : "#ffff00".

spécial : Peut prendre une valeur parmi une liste déterminée, selon le paramètre en question.

3. Objets graphiques simples

3.1 Rectangle

Nous avons déjà vu, dans le premier exemple, comment créer un simple rectangle coloré. Allons maintenant un peu plus loin.

Voici la liste complète des arguments que peut prendre l'élément `box` :

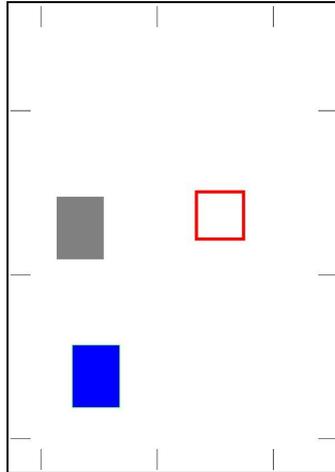
`x` : abscisse du coin inférieur-gauche du rectangle [valeur]
`y` : ordonnée du coin inférieur-gauche du rectangle [valeur]
`width` : largeur du rectangle [valeur]
`height` : hauteur du rectangle [valeur]
`line-width` : épaisseur de la ligne de bordure [valeur] (optionnel)
`line-color` : couleur de la ligne [couleur] (optionnel)
`fill-color` : couleur de fond [couleur] (optionnel)
`rotation` : angle de rotation [scalaire] (optionnel, voir Chapitre 3.7)
`opacity` : opacité/transparence de l'objet [scalaire] (optionnel, voir Chapitre 10.1)
`blendmode` : mode de mélange des couleurs [spécial] (optionnel, voir Chapitre 10.2)

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a7"/>
06     <card>
07       <box x="1cm" y="1cm" width="3cm" height="4cm"
08         fill-color="#808080"/>
09     </card>
10     <card>
11       <box x="25mm" y="2.3cm" width="3cm" height="3cm"
12         line-width="2mm" line-color="#ff0000"/>
13     </card>
14     <card>
15       <box x="20mm" y="2cm" width="3cm" height="4cm"
16         line-color="#00ff00" fill-color="#0000ff"/>
17     </card>
18   </deck>
19 </pandocreon:carta-genius>
```

On voit ici un exemple simple, avec trois rectangles. Le premier est rempli de gris, le deuxième n'est pas rempli, mais possède une ligne de bordure rouge épaisse de 2 millimètres, et le troisième est rempli de bleu avec une ligne verte dont l'épaisseur n'est pas spécifiée (utilisation d'une valeur

par défaut). La couleur de la ligne de bordure et la couleur de remplissage peuvent être utilisées séparément ou conjointement, mais au moins l'une des deux doit être présente.

Et voici le résultat de l'exemple ci-dessus :

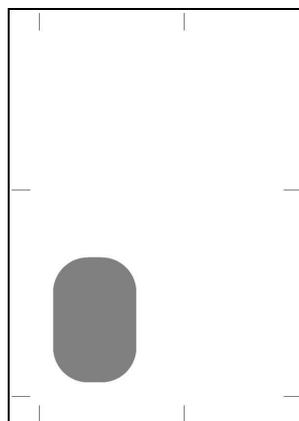


3.2 Rectangle arrondi

Un rectangle arrondi se gère de la même manière qu'un rectangle normal, sauf qu'un paramètre supplémentaire fait son apparition, `radius`, qui indique le rayon des coins arrondis.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <box x="1cm" y="1cm" width="6cm" height="9cm"
08         radius="2.5cm" fill-color="#808080"/>
09     </card>
10   </deck>
11 </pandocreon:carta-genius>
```

Et voici le résultat :



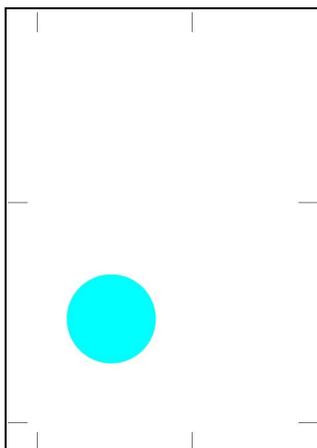
3.3 Cercle

Liste complète des options de l'élément `circle` :

`x` : abscisse du centre du cercle [valeur]
`y` : ordonnée du centre du cercle [valeur]
`radius` : rayon du cercle [valeur]
`line-width` : épaisseur de la ligne de bordure [valeur] (optionnel)
`line-color` : couleur de la ligne [couleur] (optionnel)
`fill-color` : couleur de fond [couleur] (optionnel)
`opacity` : opacité/transparence de l'objet [scalaire] (optionnel, voir Chapitre 10.1)
`blendmode` : mode de mélange des couleurs [spécial] (optionnel, voir Chapitre 10.2)

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <circle x="5cm" y="7cm" radius="3cm" fill-color="#00ffff"/>
08     </card>
09   </deck>
10 </pandocreon:carta-genius>
```

Le résultat :



3.4 Ligne

Options pour les lignes :

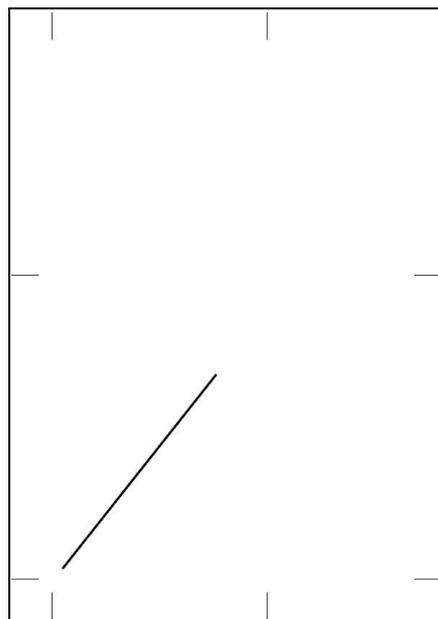
`x1` : abscisse du début de la ligne [valeur]
`y1` : ordonnée du début de la ligne [valeur]
`x2` : abscisse de la fin de la ligne [valeur]
`y2` : ordonnée de la fin de la ligne [valeur]
`width` : épaisseur de la ligne [valeur] (optionnel)
`color` : couleur de la ligne [couleur]
`opacity` : opacité/transparence de l'objet [scalaire] (optionnel, voir Chapitre 10.1)
`blendmode` : mode de mélange des couleurs [spécial] (optionnel, voir Chapitre 10.2)

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <line x1="5mm" y1="5mm" x2="8cm" y2="10cm" width="1mm"
08         color="#000000"/>
09     </card>
10   </deck>
11 </pandocreon:carta-genius>

```

Résultat :



3.5 Grille

Un type supplémentaire d'objet graphique permet de dessiner des grilles, c'est-à-dire des quadrillages. Pour cela, il faut spécifier les coordonnées du coin inférieur-gauche de la grille, le nombre de cases horizontalement et verticalement, et enfin la largeur et la hauteur des cases.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" landscape="yes" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <grid x="5mm" y="1cm" cell-width="2cm" cell-height="1cm"
08         nbr-width="3" nbr-height="6" line-width="0.4mm"
09         line-color="#800000"/>
10     </card>
11     <card>
12       <grid x="5mm" y="2cm" cell-width="5cm" cell-height="2cm"
13         nbr-width="2" nbr-height="4" fill-color="#8080ff"

```

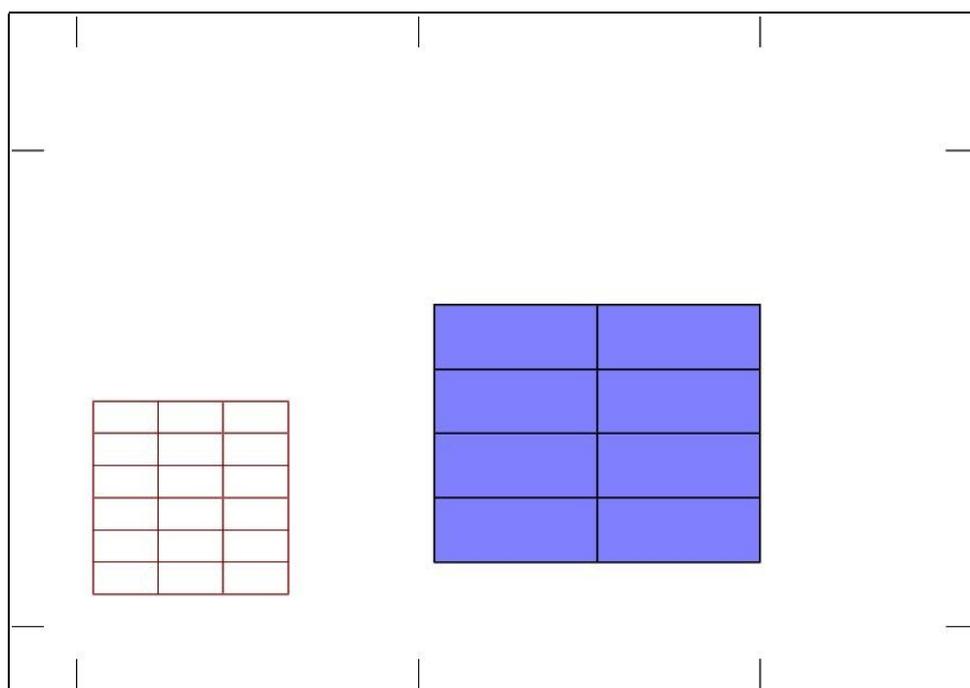
```

14     line-width="0.5mm" line-color="#000000"/>
15     </card>
16     </deck>
17 </pandocreon:carta-genius>

```

Lignes 7 à 9 : Définition d'une première grille, en utilisant le mot-clé `grid`. Les coordonnées sont définies avec les attributs `x` et `y`, les dimensions des cellules avec `cell-width` et `cell-height`, et le nombre de cellules avec `nbr-width` et `nbr-height`. On crée ici une grille sans couleur de fond, mais avec des lignes dont l'épaisseur et la couleur sont spécifiées.

Lignes 12 à 14 : Définition d'une seconde grille. Une couleur de remplissage est utilisée.



3.6 Hexagone

Les hexagones sont des formes géométriques bien souvent utilisées dans les jeux de société. Elles sont constituées de 6 côtés d'égale longueur, et l'angle entre chacun de ces côtés est de 120 degrés. Lors de la création d'un hexagone, vous devez spécifier le coin inférieur gauche du rectangle « imaginaire » qui le contiendra, ainsi que sa largeur. Vous n'avez pas à spécifier la hauteur de l'hexagone, car celle-ci est calculée : la hauteur est égale à deux fois la largeur multipliée par la tangente de 30 degrés, soit environ la hauteur multipliée par 1,15470053837.

Voici un exemple comportant plusieurs hexagones :

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" landscape="yes" margin="0cm"/>
05     <cardsize type="a5"/>
06     <card>
07       <hexagon x="3cm" y="3cm" width="9cm" line-color="#808080"

```

```

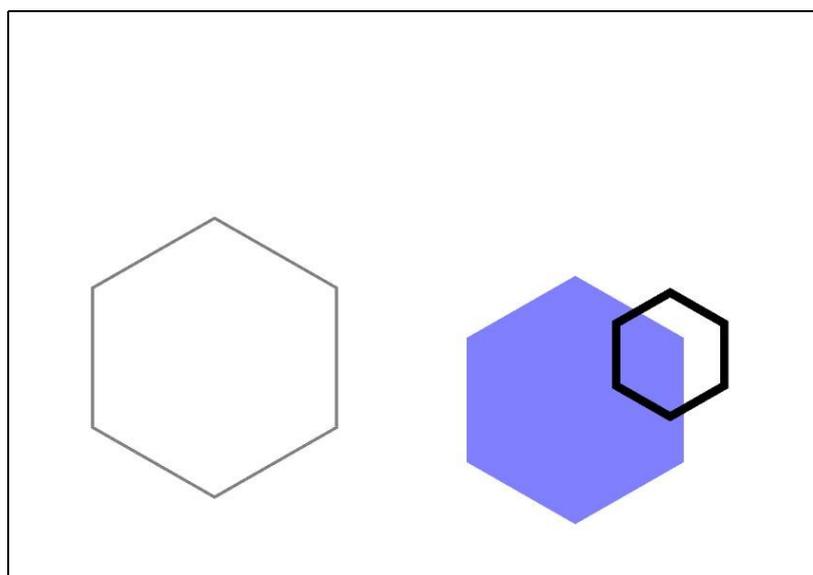
08     line-width="1mm"/>
09   </card>
10   <card>
11     <hexagon x="2cm" y="2cm" width="8cm" fill-color="#8080ff"/>
12     <hexagon x="7.5cm" y="6cm" width="4cm" line-color="#000000"
13       line-width="3mm"/>
14   </card>
15 </deck>
16 </pandocreon:carta-genius>

```

Lignes 7 et 8 : Le mot-clé hexagon permet de définir un hexagone. Ici seule la ligne de contour est demandée.

Ligne 11 : Un autre hexagone, sans ligne de bordure, mais rempli de bleu ciel.

Lignes 12 et 13 : Un dernier hexagone, avec une ligne et une couleur de remplissage.



3.7 Rotation des objets

La plupart des objets graphiques possèdent un attribut nommé "rotation". Il permet d'affecter un angle en degrés à l'objet. Le point de référence (coin inférieur gauche de l'objet) ne bouge pas. La rotation s'effectue dans le sens trigonométrique (sens inverse des aiguilles d'une montre). L'angle peut être négatif.

3.8 Pointillés

Tous les objets graphiques pouvant afficher une ligne peuvent être utilisés avec les paramètres dash1 et dash2. Ceux-ci permettent de faire en sorte que la ligne ne soit pas pleine mais pointillée. L'attribut dash1 contient alors la longueur des traits pleins, et dash2 la longueur des traits vides.

Voici un exemple de ligne pointillée :

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>

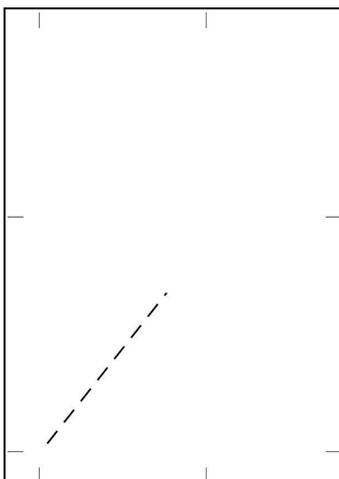
```

```

03 <deck>
04   <paper type="a4" margin="2cm"/>
05   <cardsize type="a6"/>
06   <card>
07     <line x1="5mm" y1="5mm" x2="8cm" y2="10cm" width="1mm"
08       color="#000000" dash1="1cm" dash2="7mm"/>
09   </card>
10 </deck>
11 </pandocreon:carta-genius>

```

Ligne 8 : On déclare que la ligne est pointillée, avec des traits pleins de 1 centimètre de long, et des parties vides de 7 millimètres de long.



4. Objets graphiques complexes

4.1 Courbe de Bézier

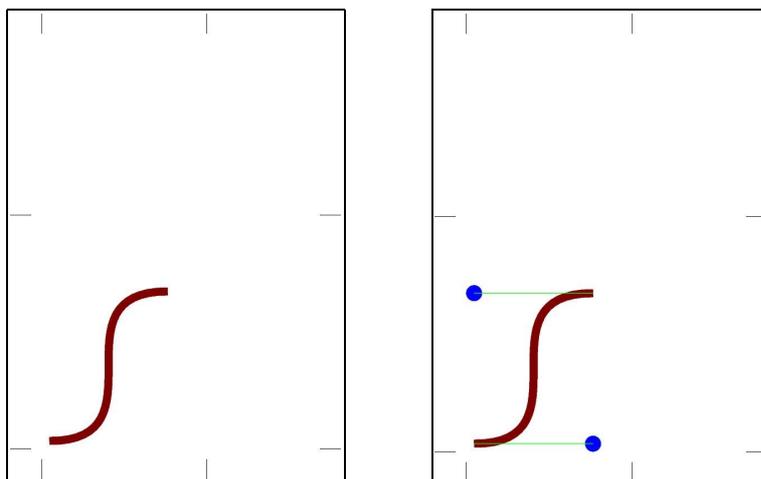
Les courbes de Bézier sont des lignes qui ne sont pas définies par 2 points, mais par 4 points : le début de la ligne, la fin de la ligne, et deux points de contrôle qui donnent la tangente à la courbe au niveau des deux premiers points. Voyons un exemple :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <bezier x1="5mm" y1="5mm" x2="8cm" y2="10cm"
08         cp-x1="8cm" cp-y1="5mm" cp-x2="5mm" cp-y2="10cm"
09         width="5mm" color="#800000"/>
10     </card>
11     <card>
12       <bezier x1="5mm" y1="5mm" x2="8cm" y2="10cm"
13         cp-x1="8cm" cp-y1="5mm" cp-x2="5mm" cp-y2="10cm"
14         width="5mm" color="#800000"/>
15       <circle x="8cm" y="5mm" radius="5mm"
16         fill-color="#0000ff"/>
17       <circle x="5mm" y="10cm" radius="5mm"
18         fill-color="#0000ff"/>
19       <line x1="5mm" y1="5mm" x2="8cm" y2="5mm"
20         color="#00ff00"/>
21       <line x1="8cm" y1="10cm" x2="5mm" y2="10cm"
22         color="#00ff00"/>
23     </card>
24   </deck>
25 </pandocreon:carta-genius>
```

Lignes 7 à 9 : Création de la courbe de Bézier sur la première carte.

Lignes 12 à 22 : Création sur la deuxième carte de la même courbe, avec en plus des cercles pour symboliser l'emplacement des points de contrôle, et des lignes droites pour montrer les tangentes.

Et voici le résultat de ces deux cartes :



Options complètes pour les courbes de Bézier :

x1 : abscisse du début de la courbe [valeur]
 y1 : ordonnée du début de la courbe [valeur]
 x2 : abscisse de la fin de la courbe [valeur]
 y2 : ordonnée de la fin de la courbe [valeur]
 cp-x1 : abscisse du premier point de contrôle [valeur]
 cp-y1 : ordonnée du premier point de contrôle [valeur]
 cp-x2 : abscisse du second point de contrôle [valeur]
 cp-y2 : ordonnée du second point de contrôle [valeur]
 width : épaisseur de la ligne [valeur] (optionnel)
 color : couleur de la ligne [couleur]
 opacity : opacité/transparence de l'objet [scalaire] (optionnel, voir Chapitre 10.1)
 blendmode : mode de mélange des couleurs [spécial] (optionnel, voir Chapitre 10.2)

4.2 Polygone

Les polygones sont des formes composées de plusieurs lignes et/ou courbes de Bézier. Voyons un exemple avec un polygone constitué de 3 lignes et d'une courbe :

```

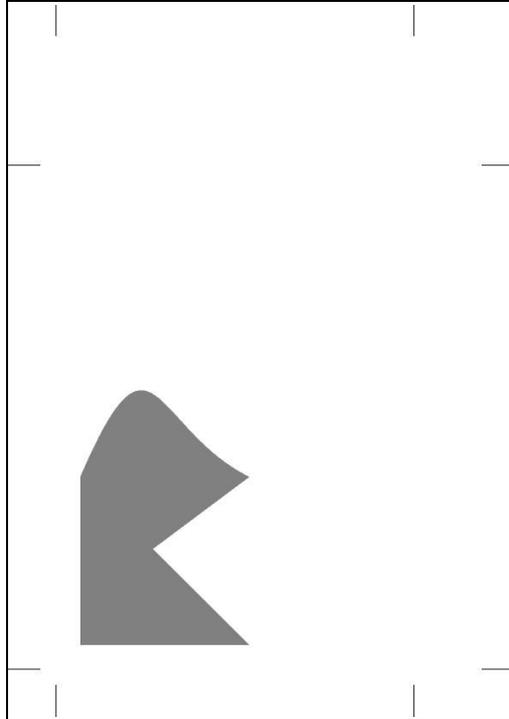
01  <?xml version="1.0"?>
02  <pandocreon:carta-genius>
03    <deck>
04      <paper type="a4" margin="2cm"/>
05      <cardsize type="a5"/>
06      <card>
07        <polygon x="1cm" y="1cm" fill-color="#808080">
08          <line x="8cm" y="1cm"/>
09          <line x="4cm" y="5cm"/>
10          <line x="8cm" y="8cm"/>
11          <bezier x="1cm" y="8cm" cp-x1="4cm" cp-y1="10cm"
12            cp-x2="4cm" cp-y2="15cm"/>
13        </polygon>
14      </card>
15    </deck>
16  </pandocreon:carta-genius>
  
```

Ligne 7 : Lors de la création du polygone, il faut donner les coordonnées du point qui va servir à

commencer la forme. Le polygone étant automatiquement « refermé », ce point constituera aussi la fin de la forme.

Lignes 8 à 10 : On place 3 lignes. Pour chacune, on se contente en fait de donner les coordonnées du prochain point, qui doit être relié au dernier point par une ligne droite.

Lignes 11 et 12 : On crée une courbe de Bézier, en donnant le prochain point à relier, puis les coordonnées des deux points de contrôle.



Options complètes pour les polygones :

`x` : abscisse du début du polygone [valeur]

`y` : ordonnée du début du polygone [valeur]

`line-width` : épaisseur de la ligne de bordure [valeur] (optionnel)

`line-color` : couleur de la ligne [couleur] (optionnel)

`fill-color` : couleur de fond [couleur] (optionnel)

`rotation` : angle de rotation [scalaire] (optionnel, voir Chapitre 3.7)

`opacity` : opacité/transparence de l'objet [scalaire] (optionnel, voir Chapitre 10.1)

`blendmode` : mode de mélange des couleurs [spécial] (optionnel, voir Chapitre 10.2)

Options pour les lignes de polygones :

`x` : abscisse du prochain point à relier [valeur]

`y` : ordonnée du prochain point à relier [valeur]

Options pour les courbes de Bézier de polygones :

`x` : abscisse du prochain point à relier [valeur]

`y` : ordonnée du prochain point à relier [valeur]

`cp-x1` : abscisse du premier point de contrôle [valeur]

`cp-y1` : ordonnée du premier point de contrôle [valeur]

`cp-x2` : abscisse du second point de contrôle [valeur]

`cp-y2` : ordonnée du second point de contrôle [valeur]

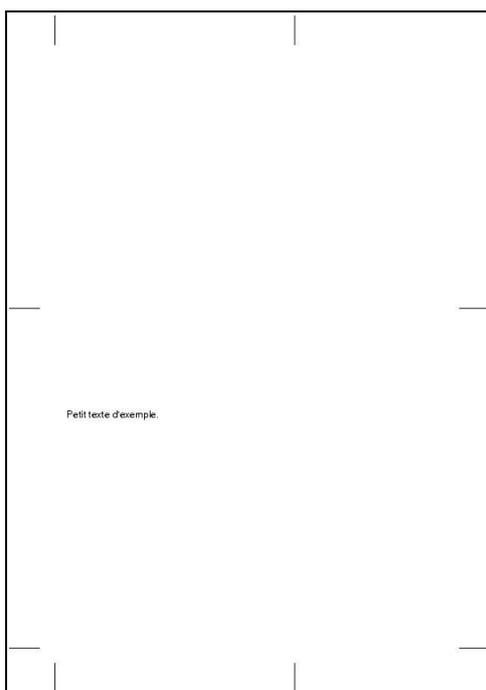
5. Zones de texte

5.1 Utilisation de base

Les zones de texte ressemblent de près aux rectangles vus précédemment. Il faut en spécifier l'emplacement et la taille de la même manière. Mais de nouvelles propriétés permettent de choisir la police et la taille des caractères, et l'élément reçoit des informations textuelles.

Voici un exemple simple :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="8cm" height="10cm"
08         font="Helvetica" size="12pt" color="#000000">
09         Petit texte d'exemple.
10       </text>
11     </card>
12   </deck>
13 </pandocreon:carta-genius>
```



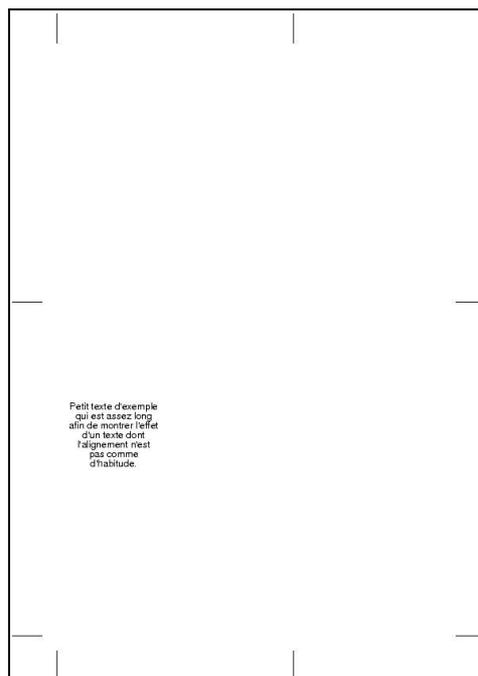
Nous voyons donc les propriétés de base pour l'élément `text` :

- `x` : abscisse du coin inférieur-gauche de la zone de texte [valeur]
- `y` : ordonnée du coin inférieur-gauche de la zone [valeur]
- `width` : largeur de la zone [valeur]
- `height` : hauteur de la zone [valeur]
- `font` : nom de la police de caractère à utiliser [spécial]
- `size` : taille des caractères [valeur]
- `color` : couleur du texte [couleur]

5.2 Alignement du texte

Il existe une propriété optionnelle `align`, qui peut prendre comme valeur 'left' (gauche), 'right' (droite), 'center' (centré) ou 'justify' (justifié), qui va décider de l'alignement du texte. Par défaut, le texte est aligné à gauche.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="4cm" height="10cm"
08         font="Helvetica" size="12pt" color="#000000"
09         align="center">
10         Petit texte d'exemple qui est assez long afin de montrer
11         l'effet d'un texte dont l'alignement n'est pas comme
12         d'habitude.
13       </text>
14     </card>
15   </deck>
16 </pandocreon:carta-genius>
```



5.3 Interlignage

Par défaut, les zone de texte affiche les lignes avec un interlignage simple. On considère alors que la distance entre la base de chaque ligne est égale à la taille choisie pour les caractères. Il est possible d'utiliser le paramètre `space` pour utiliser une autre valeur. Ce paramètre attend un scalaire ; un interlignage simple sera équivalent à 1, un interlignage double vaudra 2.

Reprenons le dernier exemple, avec un interlignage plus grand.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="4cm" height="10cm"
08         font="Helvetica" size="12pt" color="#000000"
09         align="center" space="2.5">
10         Petit texte d'exemple qui est assez long afin de montrer
11         l'effet d'un texte dont l'alignement n'est pas comme
12         d'habitude.
13       </text>
14     </card>
15   </deck>
16 </pandocreon:carta-genius>
```



La différence saute aux yeux et se passe de commentaires.

5.4 Cadres

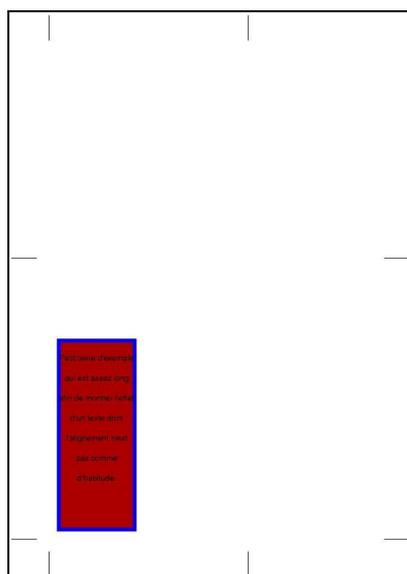
Vous aurez remarqué qu'il n'est pas forcément évident de situer, dans une page, l'étendue d'une zone de texte, ce qui a un impact direct sur l'emplacement réel de l'écriture. Pour remédier à cela, il est possible d'utiliser les propriétés suivantes, qui ont exactement le même effet qu'avec les rectangles :

fill-color : couleur de fond [couleur] (optionnel)
line-width : épaisseur de la ligne de bordure [valeur] (optionnel)
line-color : couleur de la ligne de bordure [couleur] (optionnel)

Modifions légèrement l'exemple précédent :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="4cm" height="10cm"
08         font="Helvetica" size="12pt" color="#000000"
09         align="center" space="2.5" fill-color="#aa0000"
10         line-width="2mm" line-color="#0000ff">
11         Petit texte d'exemple qui est assez long afin de montrer
12         l'effet d'un texte dont l'alignement n'est pas comme
13         d'habitude.
14       </text>
15     </card>
16   </deck>
17 </pandocreon:carta-genius>
```

Voyons le résultat :



Comme pour les rectangles habituels, le paramètre `radius` peut être employé pour obtenir des coins ronds à la place de coins carrés.

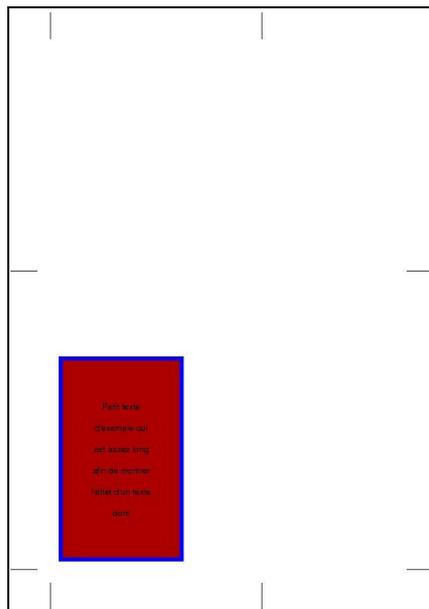
En plus de cela, il est possible de créer un tel cadre, tout en spécifiant une distance entre le texte et le bord du cadre. Il faut pour cela utiliser un paramètre supplémentaire, `border`, dont le contenu doit être une valeur.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
```

```

05     <cardsize type="a6"/>
06     <card>
07         <text x="5mm" y="5mm" width="6cm" height="10cm"
08             font="Helvetica" size="12pt" color="#000000"
09             align="center" space="2.5" fill-color="#aa0000"
10             line-width="2mm" line-color="#0000ff" border="15mm">
11             Petit texte d'exemple qui est assez long afin de montrer
12             l'effet d'un texte dont l'alignement n'est pas comme
13             d'habitude.
14         </text>
15     </card>
16 </deck>
17 </pandocreon:carta-genius>

```



Là encore, le résultat s'explique de lui-même.

5.5 Rotation

Comme expliqué au Chapitre 3.7, les zone de texte font partie des éléments graphiques sur lesquels peuvent être appliquées des rotations.

```

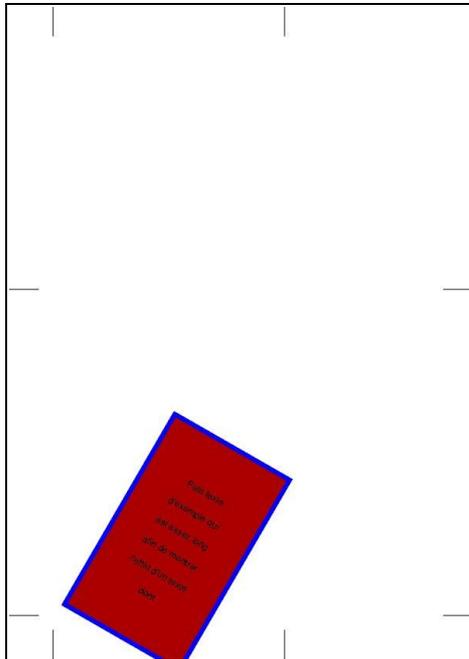
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03     <deck>
04         <paper type="a4" margin="2cm"/>
05         <cardsize type="a6"/>
06         <card>
07             <text x="5mm" y="5mm" width="6cm" height="10cm"
08                 font="Helvetica" size="12pt" color="#000000"
09                 align="center" space="2.5" fill-color="#aa0000"
10                 line-width="2mm" line-color="#0000ff" border="15mm"
11                 rotation="-30">
12                 Petit texte d'exemple qui est assez long afin de montrer

```

```

13         l'effet d'un texte dont l'alignement n'est pas comme
14         d'habitude.
15     </text>
16 </card>
17 </deck>
18 </pandocreon:carta-genius>

```



Comme vous pouvez le voir, la rotation fait ici sortir la zone de texte de la zone de la carte, mais aussi du papier. Cela peut parfois être pratique (c'est pour cela aussi qu'il est possible d'affecter des positions négatives aux objets), mais faites attentions lorsque vous avez plusieurs cartes qui risquent de se marcher dessus les unes les autres.

5.6 Adaptation de la taille du texte

Parfois, vous pouvez mettre du texte dans une zone, en donnant une taille de caractères ou d'interlignage trop grande ; résultat, le texte est tronqué et la fin n'apparaît pas. Normal, elle sortirait de la zone définie. Ceci est le cas de tous les exemples précédents, depuis que l'on a augmenté l'interlignage.

Grâce au paramètre `adapt`, il est possible de demander à ce que le texte soit automatiquement réduit jusqu'à entrer en entier dans la zone de texte. Il suffit de mettre ce paramètre à 'yes'.

```

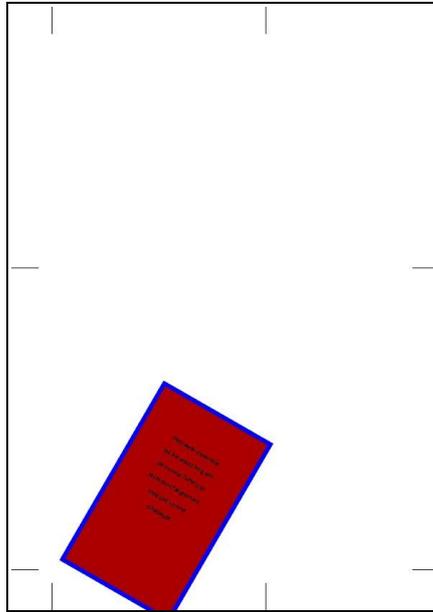
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03     <deck>
04         <paper type="a4" margin="2cm"/>
05         <cardsize type="a6"/>
06         <card>
07             <text x="5mm" y="5mm" width="6cm" height="10cm"
08                 font="Helvetica" size="12pt" color="#000000"
09                 align="center" space="2.5" fill-color="#aa0000"
10                 line-width="2mm" line-color="#0000ff" border="15mm"

```

```

11     rotation="-30" adapt="yes">
12     Petit texte d'exemple qui est assez long afin de montrer
13     l'effet d'un texte dont l'alignement n'est pas comme
14     d'habitude.
15     </text>
16 </card>
17 </deck>
18 </pandocreon:carta-genius>

```



5.7 Caractères spéciaux

Il n'est pas toujours évident de pouvoir intégrer à un fichier XML les blocs de texte que l'on souhaite. En effet, les retours à la ligne ne sont pas interprétés comme tels mais juste vus comme des espaces. Pour ajouter un retour à la ligne dans un texte, il faut donc ajouter une balise `
`. Pour ajouter un saut de paragraphe (deux sauts de lignes consécutifs), il faut utiliser une balise `<p />`.

N'oubliez pas que plusieurs caractères ne peuvent s'écrire tels quels dans un fichier XML. Par exemple, le caractère inférieur (`<`) marquant le début d'une balise, il n'est pas possible de l'utiliser directement. Voici les conversions :

```

< :    &lt;
> :    &gt;
& :    &amp;

```

Enfin, si vous voulez ajouter des caractères qui font partie du jeu de caractère latin1 (aussi appelé iso-8859-1) mais qui ne sont pas présent sur votre clavier, vous pouvez les ajouter sous la forme « `&XXX;` », où XXX est la valeur décimale du caractère dans la table latin1. Reportez-vous à l'Annexe J pour connaître les valeurs de cette table.

5.8 Polices de caractères intégrées

Jusqu'ici, tous les exemples de zone de texte ont été fait avec la police de caractères Helvetica. Cette police fait partie des 11 polices intégrées dans la norme PDF, et peut donc être employée librement, sans devoir inclure de fichiers externes.

La liste des polices de caractères intégrées est la suivante :

Courier, Courier-Bold, Courier-Oblique, Courier-BoldOblique, Helvetica, Helvetica-Bold, Helvetica-Oblique, Helvetica-BoldOblique, Times-Roman, Times-Bold, Times-Italic, Times-BoldItalic, Symbol, ZapfDingbats.

Faites attention à bien respecter la casse (majuscules/minuscules) lorsque vous utilisez ces polices. Pour avoir plus de renseignement sur ces fontes, reportez-vous à l'Annexe F.

5.9 Polices externes

Les polices intégrées sont très pratiques, mais on se sent vite limité et le besoin d'utiliser d'autres fontes se fait pressant. Pas de problème, Carta Genius est capable de gérer les polices Postscript Type 1 (si vous avez des fontes TrueType, reportez-vous à l'Annexe H pour savoir comment les transformer en Postscript Type 1).

Nous avons vu au Chapitre 2.3 que des déclarations générales pouvaient être faites au début du fichier XML (en dehors des « decks »). Parmi ces déclarations, il y a l'élément `fonts`, qui peut contenir autant de balises `font` que vous avez de polices à utiliser.

L'élément `font` a besoin de trois paramètres :

`id` : le nom Postscript de la fonte (attention, la casse majuscules/minuscules doit être respectée, et ce nom peut être très différent du nom qui apparaît sous Windows ; ouvrez le fichier d'outline pour connaître ce nom)

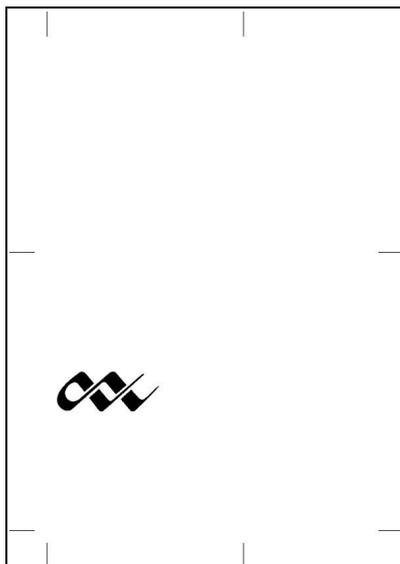
`outline` : le chemin vers le fichier d'« outline » de la police (fichier PFA ou PFB)

`metrics` : le chemin vers le fichier de « metrics » de la police (fichier AFM)

Voici un exemple :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <fonts>
04     <font id="PandocreonSymbol"
05       outline="pandocreon-symbol-0.4.0.pfa"
06       metrics="pandocreon-symbol-0.4.0.afm"/>
07   </font>
08   <deck>
09     <paper type="a4" margin="2cm"/>
10     <cardsize type="a6"/>
11     <card>
12       <text x="5mm" y="5mm" width="6cm" height="10cm"
13         font="PandocreonSymbol" size="120pt" color="#000000">
14         P
15       </text>
16     </card>
17   </deck>
18 </pandocreon:carta-genius>
```

Et le résultat :



Pour que cet exemple fonctionne, il faut bien évidemment que vous possédiez la police de caractères PandocreonSymbol (disponible sur www.pandocreon.com).

5.10 Soulignage, surlignage, barrage

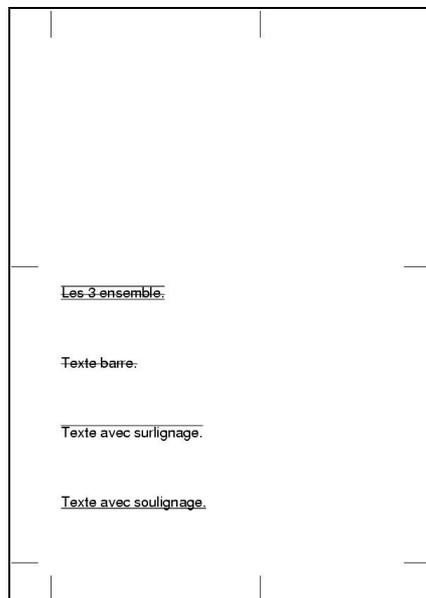
Quelques soit la police de caractère employée, il est possible de demander au logiciel de placer une ligne sous les caractères pour souligner le texte, et/ou une ligne au-dessus du texte, et/ou un ligne à travers le texte. Il faut utiliser les paramètres optionnels `underline`, `overline` et `strikeout` du tag `text` et leur donner la valeur "YES".

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="8cm" height="3cm"
08         font="Helvetica" size="20pt" color="#000000"
09         underline="yes">
10         Texte avec soulignage.
11       </text>
12       <text x="5mm" y="40mm" width="8cm" height="3cm"
13         font="Helvetica" size="20pt" color="#000000"
14         overline="yes">
15         Texte avec surlignage.
16       </text>
17       <text x="5mm" y="75mm" width="8cm" height="3cm"
18         font="Helvetica" size="20pt" color="#000000"
19         strikeout="yes">
20         Texte barre.
21       </text>
22       <text x="5mm" y="110mm" width="8cm" height="3cm"
23         font="Helvetica" size="20pt" color="#000000">
```

```

24     underline="yes" overline="yes" strikeout="yes">
25     Les 3 ensemble.
26     </text>
27 </card>
28 </deck>
29 </pandocreon:carta-genius>

```



5.11 Rayures horizontales

L'option de soulignage ajoute une ligne sous les caractères du texte. Mais pour inscrire des lignes qui recouvrent toute la largeur de la zone de texte, il faut utiliser les paramètres `ladder-color` (couleur des lignes), avec optionnellement `ladder-bottom` (booléen, mettre à "yes" pour que les lignes partent du bas de l'empatement des caractères) et `ladder-rise` (valeur, distance de déplacement vertical de la ligne).

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <text x="5mm" y="5mm" width="8cm" height="3cm"
08         font="Helvetica" size="20pt" color="#000000"
09         line-color="#800000"
10         ladder-color="#000080">
11         Texte avec des lignes.
12       </text>
13       <text x="5mm" y="40mm" width="8cm" height="3cm"
14         font="Helvetica" size="20pt" color="#000000"
15         line-color="#008000"
16         ladder-color="#000000" ladder-bottom="yes">
17         Un deuxieme texte avec des lignes

```

```

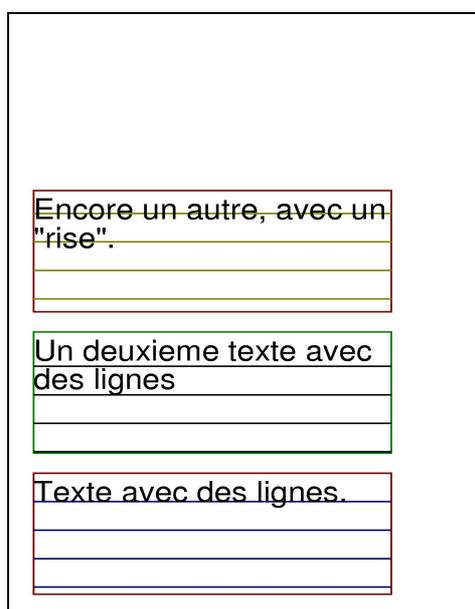
18     </text>
19     <text x="5mm" y="75mm" width="8cm" height="3cm"
20         font="Helvetica" size="20pt" color="#000000"
21         line-color="#800000"
22         ladder-color="#808000" ladder-rise="4pt">
23         Encore un autre, avec un "rise".
24     </text>
25 </card>
26 </deck>
27 </pandocreon:carta-genius>

```

Ligne 10 : Définition de la couleur des rayures horizontales.

Ligne 16 : Autres rayures, cette fois-ci en bas de l'empatement des caractères.

Ligne 22 : Cette fois on déplace explicitement les lignes vers le haut (la valeur peut être négative pour déplacer les lignes vers le bas).



5.12 Police par défaut

La plupart du temps, il vous faudra déclarer pour chaque zone de texte le nom de la police de caractères à utiliser. Néanmoins, vous pourriez avoir envie que toutes les zones pour lesquelles vous ne donnez pas le nom de la fonte à employer se retrouvent automatiquement avec une police par défaut (par exemple celle que vous utilisez le plus souvent).

Cela se fait en ajoutant à l'élément `fonts` (qui servait jusqu'ici à déclarer les polices externes) un argument `default`. Celui-ci peut contenir aussi bien le nom d'une police externe que celui d'une police intégrée.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <fonts default="Helvetica">
04     <font id="PandocreonSymbol"
05       outline="pandocreon-symbol-0.4.0.pfa"
06       metrics="../../../resources/pandocreon-symbol-0.4.0.afm"/>

```

```

07     </fonts>
08     <deck>
09         <paper type="a4" margin="2cm"/>
10         <cardsize type="a6"/>
11         <card>
12             <text x="5mm" y="5mm" width="8cm" height="10cm"
13                 size="12pt" color="#000000">
14                 Petit texte d'exemple.
15             </text>
16             <text x="5mm" y="5mm" width="6cm" height="10cm"
17                 font="PandocreonSymbol" size="120pt" color="#000000">
18                 P
19             </text>
20         </card>
21     </deck>
22 </pandocreon:carta-genius>

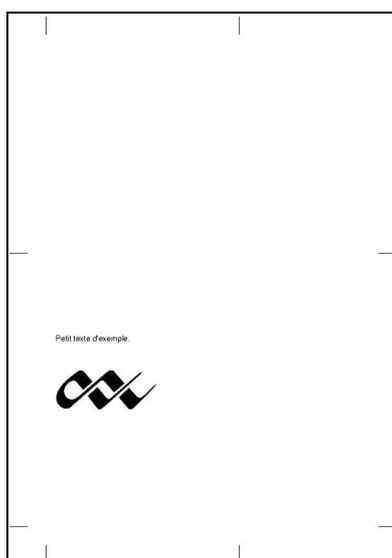
```

Ligne 3 : On déclare que la police Helvetica sera celle par défaut.

Lignes 4 à 6 : On ajoute une police externe.

Lignes 12 à 15 : Une première zone de texte, qui utilise la police par défaut.

Lignes 16 à 19 : Une seconde zone de texte, qui utilise une police déclarée préalablement.



5.13 Espacement entre les caractères

Il est possible de modifier l'espacement entre les caractères, en utilisant l'argument `char-space` qui attend une valeur (c'est-à-dire une distance). Par défaut, cette distance est nulle.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03     <deck>
04         <paper type="a4" margin="2cm"/>
05         <cardsize type="a6"/>
06         <card>
07             <text x="5mm" y="5mm" width="8cm" height="10cm"

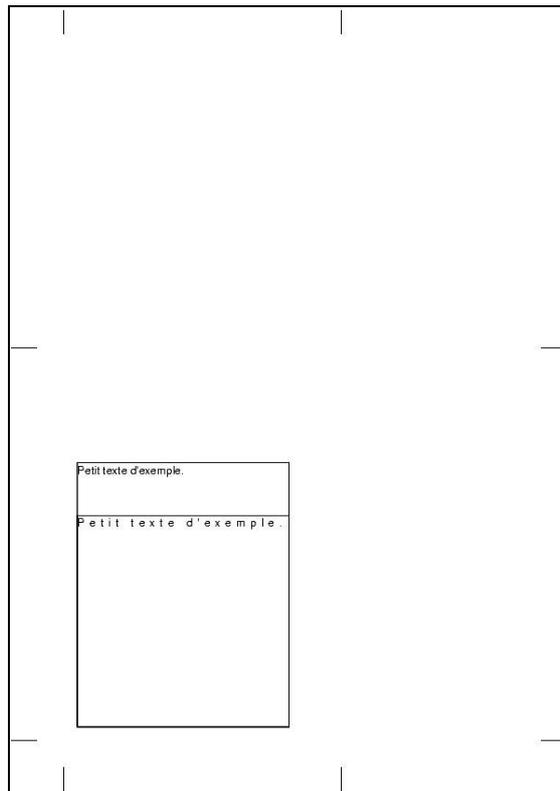
```

```

08     font="Helvetica" size="12pt" color="#000000"
09     line-color="#000000">
10     Petit texte d'exemple.
11     </text>
12     <text x="5mm" y="5mm" width="8cm" height="8cm"
13         font="Helvetica" size="12pt" color="#000000"
14         line-color="#000000" char-space="5pt">
15         Petit texte d'exemple.
16     </text>
17 </card>
18 </deck>
19 </pandocreon:carta-genius>

```

Ligne 14 : On déclare un espacement de 15 points entre les caractères.



5.14 Echelle horizontale

Il est aussi possible de modifier l'échelle horizontale d'un texte, grâce à l'argument `h-scale`. Par défaut, cette échelle vaut 100. Il est possible de lui affecter n'importe quelle valeur positive. En-dessous de 100, le texte apparaîtra plus étroit, au-dessus il sera plus large.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>

```

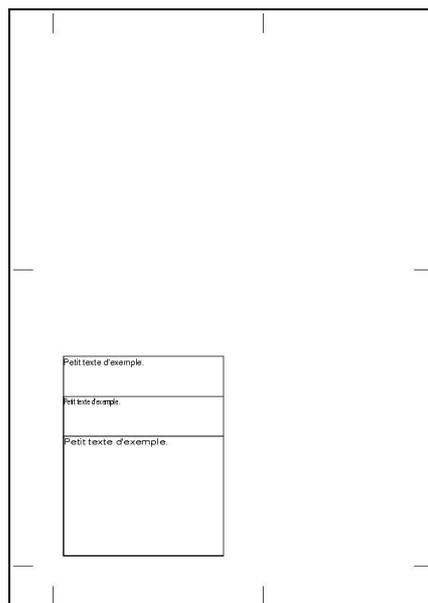
```

05 <cardsize type="a6"/>
06 <card>
07   <text x="5mm" y="5mm" width="8cm" height="10cm"
08     font="Helvetica" size="12pt" color="#000000"
09     line-color="#000000">
10     Petit texte d'exemple.
11   </text>
12   <text x="5mm" y="5mm" width="8cm" height="8cm"
13     font="Helvetica" size="12pt" color="#000000"
14     line-color="#000000" h-scale="70">
15     Petit texte d'exemple.
16   </text>
17   <text x="5mm" y="5mm" width="8cm" height="6cm"
18     font="Helvetica" size="12pt" color="#000000"
19     line-color="#000000" h-scale="130">
20     Petit texte d'exemple.
21   </text>
22 </card>
23 </deck>
24 </pandocreon:carta-genius>

```

Ligne 14 : On définit une échelle horizontale de 70%.

Ligne 19 : On définit une échelle horizontale de 130%.



6. Texte formaté

Les zones de texte vues dans le chapitre précédent permettent d'insérer de manière aisée du texte à un endroit donné. Mais leurs limites sont atteintes lorsque l'on cherche à placer des caractères ayant des caractéristiques (police, taille, couleur, ...) différentes, ou bien si on veut qu'un texte se déroule automatiquement sur plusieurs zones d'affichage sans avoir à le découper en fonction des zones en question. Deux nouveaux éléments permettent, à partir de la version 3.0.0 de Carta Genius, de réaliser cela ; le premier sert à définir les textes, et le second à placer les zones d'affichage.

6.1 Définition de texte enrichi

L'élément `text-define` sert à définir les textes enrichis, en leur affectant un identifiant (paramètre `id`). Cet élément doit contenir un ou plusieurs éléments `block`, qui sert à définir les caractéristiques des caractères regroupés par bloc de texte. Attention, un texte qui a été défini ne va pas être affiché pour autant ; cela se fera par l'utilisation de zones d'affichages (voir le sous-chapitre suivant). L'élément `block` prend les paramètres suivants (similaires aux paramètres du même nom des zones de texte classiques) :

- `font` : nom de la police à utiliser
- `size` : taille des caractères [valeur]
- `color` : couleur des caractères [couleur] (optionnel ; noir par défaut)
- `space` : interlignage [scalaire] (optionnel)
- `underline` : soulignage [booléen] (optionnel)
- `overline` : surlignage [booléen] (optionnel)
- `strikeout` : barrage [booléen] (optionnel)
- `char-space` : espacement entre les caractères [valeur] (optionnel)
- `h-scale` : échelle horizontale [scalaire] (optionnel)

Voyons un petit exemple de définition de texte :

```
01 <text-define id="txt1">
02   <block font="Helvetica-Bold" size="12pt" color="#800000">
03     Texte du premier<br/>bloc.
04   </block>
05   <block font="Courier-Oblique" size="10pt" color="#000000">
06     &nbsp;  Texte du second bloc,<p/>en deux parties.
07   </block>
08   <block font="Helvetica" size="12pt" color="#0000ff">
09     Et un dernier bloc pour finir...
10   </block>
05 </text-define>
```

Le texte défini ici, avec l'identifiant "txt1", comporte trois blocs, chacun ayant une fonte, une taille, et/ou une couleur différente.

Comme vous pouvez le remarquer, les blocs peuvent contenir les mêmes caractères spéciaux que les textes habituels (, ...), ainsi que les instructions de retour à la ligne (<p/>,
).

6.2 Zones d'affichage

L'exemple précédent va maintenant être intégré dans un ensemble de cartes. Le texte défini va être affiché sur 3 zones différentes. Dans cet exemple, chaque zone est sur une carte séparée, mais rien n'oblige cela.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize width="8.5cm" height="12cm"/>
06     <card>
07       <text-define id="txt1">
08         <block font="Helvetica-Bold" size="24pt" color="#800000">
09           Texte du premier<br/>bloc.
10         </block>
11         <block font="Courier-Oblique" size="20pt" color="#000000">
12           &nbsp; Texte du second bloc,<p/>en deux parties.
13         </block>
14         <block font="Helvetica" size="24pt" color="#0000ff">
15           Et un dernier bloc pour finir...
16         </block>
17       </text-define>
18       <text-area id="txt1" x="5mm" y="5mm"
19         width="5cm" height="5cm" font="Helvetica" size="12pt"
20         color="#000000" line-color="#000080" border="3mm"
21         fill-color="#ffaana" radius="5mm"/>
22     </card>
23     <card>
24       <text-area id="txt1" x="5mm" y="5mm"
25         width="5cm" height="5cm" font="Helvetica" size="12pt"
26         color="#000000" line-color="#000080" rotation="10"/>
27     </card>
28     <card>
29       <text-area id="txt1" x="5mm" y="5mm"
30         width="5cm" height="5cm" font="Helvetica" size="12pt"
31         color="#000000" line-color="#000080" rotation="-5"
32         dash1="4mm" dash2="3mm"/>
33     </card>
34   </deck>
35 </pandocreon:carta-genius>
```

Lignes 6 à 22 : Définition de la première carte.

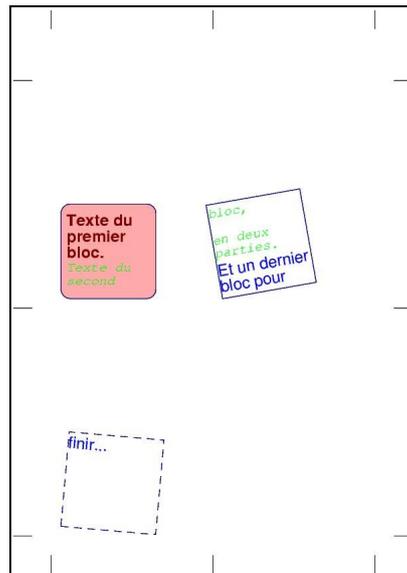
Lignes 7 à 17 : Définition du texte.

Lignes 18 à 21 : Première zone d'affichage du texte.

Lignes 23 à 27 : Deuxième carte, contenant la deuxième zone d'affichage.

Lignes 28 à 33 : Troisième carte, avec la troisième zone d'affichage. Notez que la ligne de cadre est en pointillés.

Voyons le résultat de ce fichier XML :



Les zones d'affichage possèdent les caractéristiques suivantes :

- `x` : abscisse du coin inférieur-gauche de la zone de texte [valeur]
- `y` : ordonnée du coin inférieur-gauche de la zone [valeur]
- `width` : largeur de la zone [valeur]
- `height` : hauteur de la zone [valeur]
- `radius` : rayon de courbure des coins [valeur] (optionnel)
- `rotation` : angle de rotation [scalaire] (optionnel)
- `opacity` : opacité [scalaire] (optionnel)
- `blendmode` : méthode de mélange des couleurs [spécial] (optionnel)
- `fill-color` : couleur de fond de la zone [couleur] (optionnel)
- `line-color` : couleur de bordure [couleur] (optionnel)
- `line-width` : épaisseur de la bordure [couleur] (optionnel)
- `border` : taille de marge dans la zone d'affichage [valeur] (optionnel)

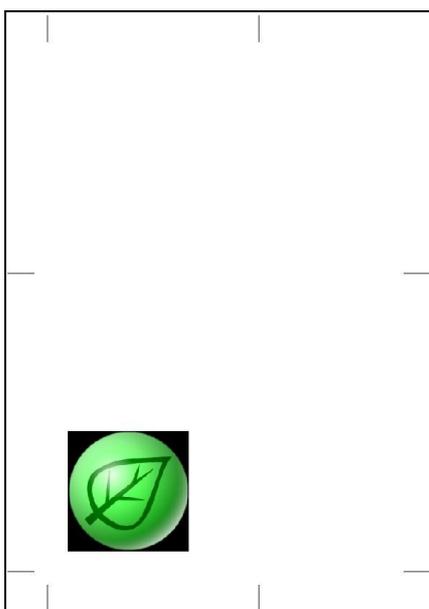
7. Insertion d'images

Nous avons vu les primitives graphiques de base, ainsi que les zones de texte. L'utilisation de ces objets vous assure d'un très bonne qualité graphique (dessin vectoriel), ainsi qu'un poids de fichier minimal. Néanmoins, vous aurez sûrement besoin d'intégrer à vos documents des images bitmap (fichiers GIF, JPEG ou PNG). Nous allons voir comment faire, et les différentes options qui s'offrent à vous.

7.1 Insertion directe

Nous allons faire un premier document contenant une image bitmap. La procédure est simple : on indique le fichier à insérer, son emplacement et sa taille.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <image file="nature.png"
08         x="1cm" y="1cm" width="6cm" height="6cm"/>
09     </card>
10   </deck>
11 </pandocreon:carta-genius>
```

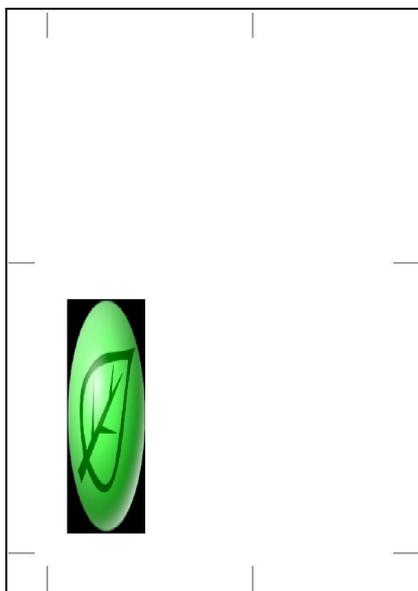


7.2 Adaptation

Par défaut, les dimensions (largeur et hauteur) définissent une zone d'affichage dans laquelle l'image va être placée de manière à être la plus grande possible à l'intérieur de cette zone tout en respectant ses dimensions propres.

Si vous voulez qu'une image soit déformée pour qu'elle s'adapte à l'intégralité de la zone, il faut utiliser le paramètre `entire`, en lui donnant la valeur `yes` (ou `YES`).

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a6"/>
06     <card>
07       <image file="nature.png" entire="yes"
08         x="1cm" y="1cm" width="4cm" height="12cm"/>
09     </card>
10   </deck>
11 </pandocreon:carta-genius>
```



7.3 Utilisation multiple

Nous avons intégré une image dans un PDF. Bien. Mais si on utilise la même image plusieurs fois, son contenu binaire va être chargé à autant de reprises, rendant le fichier PDF inutilement lourd. Il y a donc un mécanisme qui permet de déclarer les images au préalable, puis de les utiliser quand bon vous semble ; cela n'aura quasiment plus aucun impact sur le poids du fichier généré. En plus, lors de cette déclaration, il faudra donner la taille d'affichage par défaut de chaque image, ce qui permet ensuite de ne pas avoir à la donner, sauf si on souhaite l'afficher avec une taille différente.

Voici un exemple, où la même image est utilisée deux fois. La première fois, elle utilise la taille définie au moment de la déclaration, la seconde fois elle utilise une autre taille tout en appliquant une rotation sur l'image.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <images>
04     <image id="nature" file="nature.png"
05       width="5cm" height="5cm"/>
06   </images>
07   <deck>
08     <paper type="a4" margin="2cm"/>
09     <cardsize type="a7"/>
10     <card>
11       <image id="nature" x="5mm" y="5mm"/>
12     </card>
13     <card>
14       <image id="nature" x="1.5cm" y="1.5cm"
15         width="2.5cm" height="2.5cm" rotation="-25"/>
16     </card>
17   </deck>
18 </pandocreon:carta-genius>

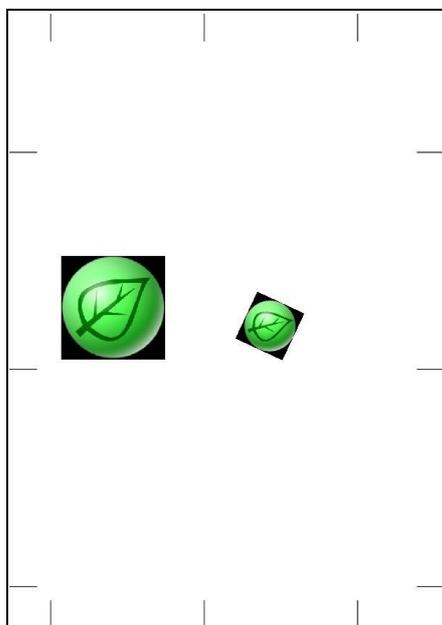
```

Lignes 3 à 6 : Déclaration des images. On peut déclarer autant d'images que nécessaire.

Lignes 4 et 5 : Déclaration d'un image. Le champ `id` contient le nom auquel feront référence par la suite les ordres de placement des images dans les cartes. Le champ `file` contient le chemin vers le fichier de l'image. Les champs `width` et `height` sont la largeur et la hauteur par défaut de l'image.

Ligne 11 : Utilisation de l'image définie. Le champ `id` indique l'image à laquelle on fait référence. Les champs `x` et `y` donnent l'emplacement de l'image. Aucune autre information n'est spécifié, donc l'image va être affichée en utilisant la largeur et la hauteur par défaut.

Lignes 14 et 15 : Seconde utilisation de l'image. En plus de fournir la position de l'image, les dimensions sont redéfinies. En plus, une rotation de 25 degrés dans le sens des aiguilles d'une montre est appliquée.

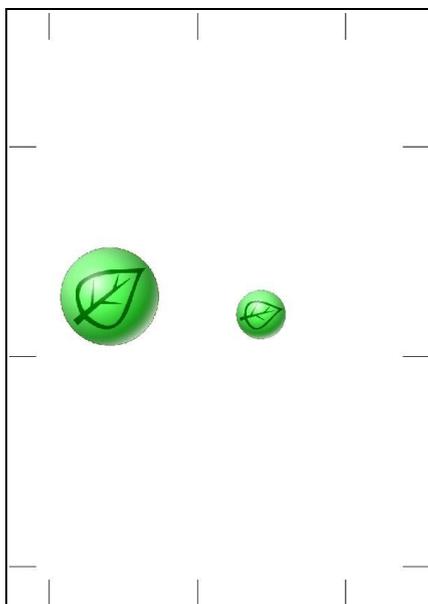


7.4 Types d'images et transparence implicite

Carta Genius supporte les images au format GIF, JPEG et PNG. Les images GIF et PNG peuvent comporter des informations de transparence, qui sont traitées de manière binaire : si une couleur est défini comme étant transparente (GIF) ou dont l'opacité est inférieure à 50% (PNG), les pixels concernés sont complètement transparents.

Reprenons l'exemple précédent en utilisant une image GIF transparente à la place du PNG sans transparence qui était employé :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <images>
04     <image id="nature" file="nature.gif"
05       width="5cm" height="5cm"/>
06   </images>
07   <deck>
08     <paper type="a4" margin="2cm"/>
09     <cardsize type="a7"/>
10     <card>
11       <image id="nature" x="5mm" y="5mm"/>
12     </card>
13     <card>
14       <image id="nature" x="1.5cm" y="1.5cm"
15         width="2.5cm" height="2.5cm" rotation="-25"/>
16     </card>
17   </deck>
18 </pandocreon:carta-genius>
```



7.5 Transparence explicite

Nous venons de voir les images qui contiennent une information sur leur transparence. Cela n'est possible qu'avec les images GIF et PNG. Mais il est aussi possible d'appliquer un « masque » sur

une image, pour indiquer les zones qui doivent s'afficher et celles qui seront transparentes. Cette fonctionnalité peut être utilisée aussi bien avec les GIF et les PNG qu'avec les JPEG.

Les images de masque doivent être au format PNG, en noir et blanc. Les parties noires seront affichées, les parties blanches correspondent aux zones transparentes.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <images>
04     <image id="nature" file="nature.png"
05       mask="masque_dogmes.png"
06       width="5cm" height="5cm"/>
07   </images>
08   <deck>
09     <paper type="a4" margin="2cm"/>
10     <cardsize type="a7"/>
11     <card>
12       <image file="masque_dogmes.png"
13         x="5mm" y="5mm" width="5cm" height="5cm"/>
14     </card>
15     <card>
16       <image id="nature" x="5mm" y="5mm"/>
17     </card>
18     <card>
19       <image file="chaos.png"
20         mask="masque_dogmes.png"
21         x="5mm" y="5mm" width="5cm" height="5cm"/>
22     </card>
23   </deck>
24 </pandocreon:carta-genius>
```

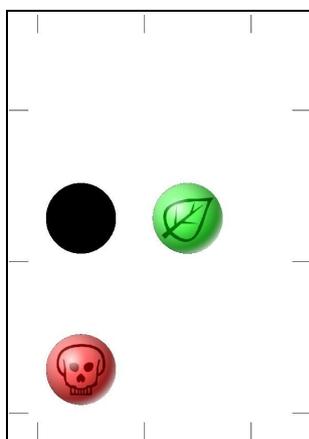
Lignes 3 à 8 : Déclaration des images.

Lignes 4 à 6 : Définition d'une image. Par rapport aux exemples précédent, l'image (qui n'a pas de transparence) a un attribut mask, dans lequel on donne le chemin vers l'image de masque pour définir explicitement la transparence.

Lignes 12 et 13 : On affiche directement l'image de masque, pour bien voir qu'il s'agit d'une image bitmap noire et blanche.

Ligne 16 : On affiche l'image définie préalablement. Elle apparaîtra avec la transparence définie par le masque.

Lignes 19 à 21 : On affiche une autre image, non définie auparavant, en lui appliquant un masque à elle aussi.



7.6 Réutilisation de masques

Nous venons de voir, dans le cadre de la transparence explicite, comment déclarer un masque de transparence à utiliser avec une image. Mais si vous avez plusieurs images différentes qui utilisent le même fichier de masque, la méthode précédente consomme une quantité de mémoire inutile : les masques sont alors chargés à chaque nouvelle déclaration.

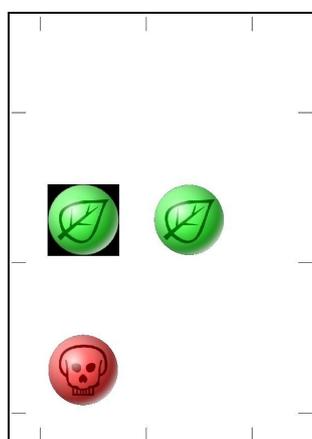
La solution est donc de déclarer le masque une seule fois puis, lorsque les images avec transparence explicite sont déclarées, le définir comme masque.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <images>
04     <mask id="noirblanc" file="masque_dogmes.png"/>
05     <image id="nature" file="nature.png"
06       mask-id="noirblanc" width="5cm" height="5cm"/>
07   </images>
08   <deck>
09     <paper type="a4" margin="2cm"/>
10     <cardsize type="a7"/>
11     <card>
12       <image file="nature.png"
13         x="5mm" y="5mm" width="5cm" height="5cm"/>
14     </card>
15     <card>
16       <image id="nature" x="5mm" y="5mm"/>
17     </card>
18     <card>
19       <image file="chaos.png"
20         mask-id="noirblanc" x="5mm" y="5mm"
21         width="5cm" height="5cm"/>
22     </card>
23   </deck>
24 </pandocreon:carta-genius>
```

Ligne 4 : Au même niveau que la déclaration des images réutilisables, on utilise l'élément `mask` pour définir un masque de transparence. Cet élément contient deux attributs, `id` qui est le nom par lequel on fera référence au masque par la suite, et `file` qui est le chemin vers le fichier.

Ligne 6 : Lors de la déclaration de cette image, on utilise l'attribut `mask-id` pour indiquer le masque préalablement déclaré à utiliser.

Ligne 20 : On utilise à nouveau le masque, ici pour une image déclarée à la volée.



8. Options concernant le papier

8.1 Positionnement des cartes

Les cartes sont placées automatiquement sur la page, en fonction de plusieurs paramètres qui sont détaillés dans ce chapitre (format du papier, format des cartes, marge du papier, espacement entre les cartes). Vous pouvez être certain que le positionnement est optimal en fonction de ces paramètres ; le seul ajustement qui pourrait vous incomber serait de changer l'orientation du papier (mode portrait / mode paysage) car dans certain cas ce changement permet de mettre plus de cartes sur une même page.

8.2 Format de papier

Avec tous les exemples des pages précédentes, vous connaissez maintenant la balise `paper` présente dans tous les « decks ». Cette balise sert à définir le format de la page. Si elle n'est pas présente, Carta Genius utilise comme papier par défaut le format A4 (210x297 millimètres). Voici tous les arguments que cette balise peut recevoir :

`type` : Le nom d'un format de papier standardisé [spécial] (voir l'Annexe E pour avoir la liste).

`landscape` : Inverse la largeur et la hauteur du papier si vaut YES [spécial]. Valeurs possibles : yes, no.

`width` : Largeur du papier [valeur].

`height` : Hauteur du papier [valeur].

`margin` : Taille de la marge tout autour du papier [valeur].

`margin-width` : Taille de la marge en largeur (bord droit et bord gauche) [valeur].

`margin-height` : Taille de la marge en hauteur (bord supérieur et bord inférieur) [valeur].

Les paramètres `width` et `height` sont à utiliser si le paramètre `type` n'est pas employé (si le papier ne respecte pas un format standard). Les paramètres `margin-width` et `margin-height` sont à utiliser si le paramètre `margin` n'est pas employé (si la marge ne doit pas être la même sur tous les bords du papier).

8.3 Marges et traits de coupe

Vous savez grâce au paragraphe précédent comment modifier la marge de vos documents. Mais à quoi cette marge peut bien servir ? En premier lieu, les imprimantes de bureau ne peuvent pas – la plupart du temps – imprimer jusqu'au bord du papier ; il faut donc définir des marges sinon les cartes ne seront pas imprimées en entier.

Mais le plus important (et c'est une des raisons qui ont conduit initialement à la création du programme Carta Genius) est que les imprimeurs professionnels, qui peuvent imprimer puis

découper plusieurs pages rapidement et facilement, utilisent des massicots électriques qui ont besoin de guides pour se caler. Ces guides sont des petits traits sur le bord du papier, des repères qui indiquent les bords des cartes. Mais si ces traits étaient à l'intérieur de la zone imprimée, « dans » les cartes, vous risqueriez de les voir à l'intérieur des cartes après l'impression et le découpage. C'est pour cette raison que les traits de coupe sont placés dans les marges ; leur positionnement est fait automatiquement par le programme.

8.4 Fond-perdu

Vous pouvez avoir besoin d'une couleur de fond pour vos cartes, qui soit unique sur tout le document. Mais, de la même manière que les traits de coupe ne doivent pas se retrouver à l'intérieur des parties découpées, cette couleur de fond doit déborder et empiéter sur la marge. Cela s'appelle un fond-perdu. L'idée est que la découpe morde dans la matière. Dans le cas contraire, vous risqueriez de voir apparaître des liserets blancs sur le bord de vos cartes.

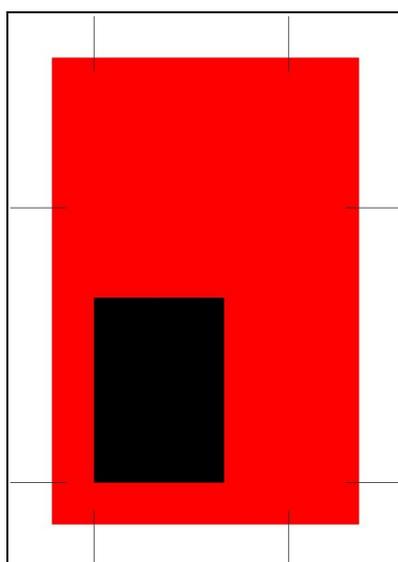
Pour définir un fond-perdu, il faut ajouter à l'intérieur de votre « deck » une balise `hidden-ditch`, qui peut accueillir deux paramètres :

`odd` : couleur pour le fond-perdu du recto des pages [couleur]

`even` : couleur pour le fond-perdu du verso des pages [couleur]

Voici un exemple tout simple pour vous montrer l'effet d'un fond-perdu.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="4.5cm"/>
05     <cardsize type="a6"/>
06     <hidden-ditch odd="#ff0000"/>
07     <card>
08       <box x="0" y="0" width="7cm" height="10cm"
09         fill-color="#000000"/>
10     </card>
11   </deck>
12 </pandocreon:carta-genius>
```

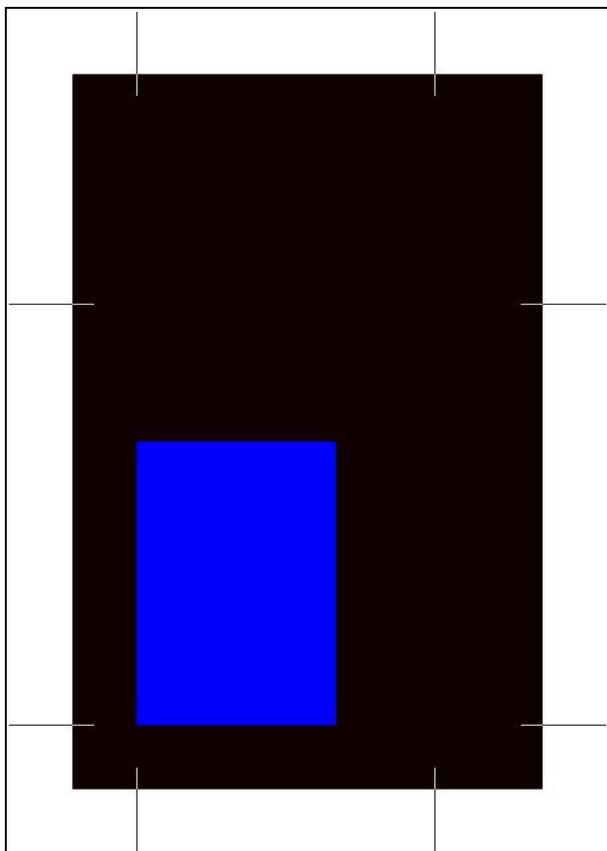


Comme vous pouvez le voir, le fond-perdu (ici de couleur rouge) déborde sur un tiers de la

marge, empiétant même sur les traits de coupe (qui font les trois quarts de la marge).

Par contre, lorsqu'un fond-perdu est très sombre, il empêche de voir les traits de coupe qui sont noirs. Cela est dommageable car ça gêne la découpe automatique correcte des cartes. Le logiciel s'adapte donc et change la couleur des traits de coupe qui sont à l'intérieur du fond-perdu.

```
<?xml version="1.0"?>
<pandocreon:carta-genius>
  <deck>
    <paper type="a4" margin="4.5cm"/>
    <cardsize type="a6"/>
    <hidden-ditch odd="#100000"/>
    <card>
      <box x="0" y="0" width="7cm" height="10cm"
          fill-color="#0000ff"/>
    </card>
  </deck>
</pandocreon:carta-genius>
```



8.5 Format et espacement des cartes

Comme le papier, le format des cartes possèdent un certain nombre de réglages possibles. Pour cela, vous connaissez déjà la balise `cardsize`, qui peut prendre en paramètre :

`type` : le nom d'un format de papier standard (voir la liste à l'Annexe E)

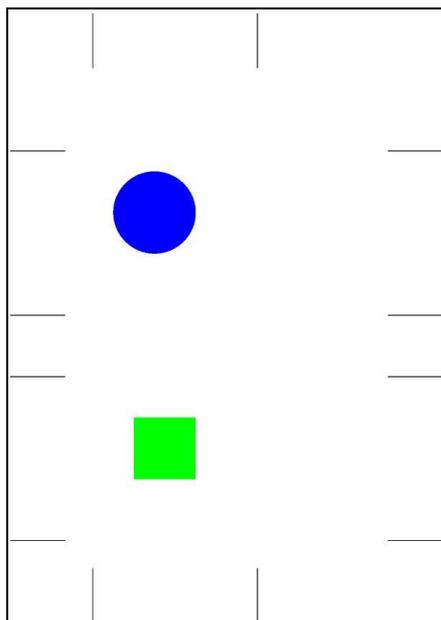
`landscape` : inverse la largeur et la hauteur des cartes si mis à YES (valeurs : yes, no)

`width` : largeur des cartes [valeur]

height : hauteur des cartes [valeur]
space : espace horizontal et vertical entre les cartes [valeur]
space-width : espace horizontal entre les cartes [valeur]
space-height : espace vertical entre les cartes [valeur]

L'espacement entre les cartes fait que les cartes ne sont plus jointes ; il y a donc 2 traits de coupe entre chaque carte. Voici un exemple avec des cartes de dimension carrée et espacées :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="4cm"/>
05     <cardsize width="8cm" height="8cm" space="3cm"/>
06     <card>
07       <circle x="3cm" y="5cm" radius="20mm" fill-color="#0000ff"/>
08     </card>
09     <card>
10       <box x="2cm" y="3cm" width="3cm" height="3cm"
11         fill-color="#00ff00"/>
12     </card>
13   </deck>
14 </pandocreon:carta-genius>
```



8.6 Utilisation multiple de la même carte

Parfois l'on souhaite afficher plusieurs fois la même carte à la suite. Plutôt que de redéfinir cette carte en insérant plusieurs fois l'élément `<card>`, il est possible d'indiquer tout simplement le nombre de fois où la carte doit s'afficher en utilisant l'attribut `"count"`.

```
<card count="3" template="recto" back="verso"/>
```

9. Gabarits

Bien souvent, les cartes possèdent des éléments similaires les unes-les autres. Le fond des cartes, ou encore leur dos, voire même d'autres éléments graphiques, peuvent se répéter. Plutôt que de les recopier de carte en carte, il suffit d'utiliser le système de gabarits (« templates » en anglais) pour n'avoir qu'à déclarer une seule fois un ou plusieurs éléments qui sont ensuite réutilisables à l'infini.

9.1 Déclaration de gabarit simple

Voyons un exemple avec un gabarit simple, contenant deux carrés l'un sur l'autre. Ce gabarit est ensuite utilisé par deux cartes comme fond.

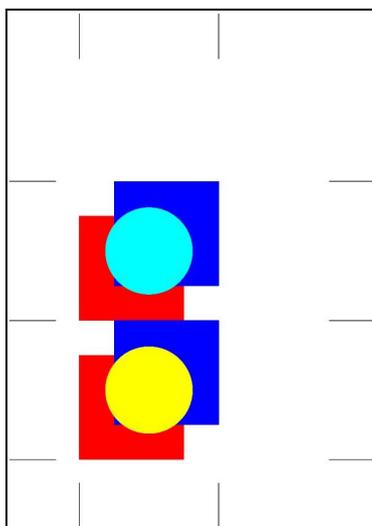
```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="fond">
05       <box x="0mm" y="0mm" width="6cm" height="6cm"
06         fill-color="#ff0000"/>
07       <box x="2cm" y="2cm" width="6cm" height="6cm"
08         fill-color="#0000ff"/>
09     </template>
10   </templates>
11   <deck>
12     <paper type="a4" margin="4cm"/>
13     <cardsize width="8cm" height="8cm"/>
14     <card template="fond">
15       <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"/>
16     </card>
17     <card template="fond">
18       <circle x="4cm" y="4cm" radius="25mm" fill-color="#ffff00"/>
19     </card>
20   </deck>
21 </pandocreon:carta-genius>
```

Lignes 3 à 9 : Les gabarits sont déclarés à l'intérieur d'un élément `templates`.

Lignes 4 à 8 : Un gabarit est déclaré en utilisant une balise `template`. Un nom est affecté à chaque gabarit grâce à l'argument `id`.

Ligne 14 : Une carte est définie, utilisant le gabarit « fond » (l'argument `template` permet de nommer le gabarit à utiliser).

Ligne 17 : Une deuxième carte est définie, utilisant le même gabarit.



9.2 Gabarits multiples

Une carte peut utiliser plusieurs gabarits. Il suffit pour cela de nommer les différents gabarits dans l'ordre où ils doivent être ajoutés dans la carte.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="rouge">
05       <box x="0mm" y="0mm" width="6cm" height="6cm"
06         fill-color="#ff0000"/>
07     </template>
08     <template id="vert">
09       <box x="1cm" y="1cm" width="6cm" height="6cm"
10         fill-color="#00ff00"/>
11     </template>
12     <template id="bleu">
13       <box x="2cm" y="2cm" width="6cm" height="6cm"
14         fill-color="#0000ff"/>
15     </template>
16   </templates>
17   <deck>
18     <paper type="a4" margin="2cm"/>
19     <cardsize width="8cm" height="8cm"/>
20     <card template="rouge,vert">
21       <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"/>
22     </card>
23     <card template="vert,bleu">
24       <circle x="4cm" y="4cm" radius="25mm" fill-color="#ffff00"/>
25     </card>
26     <card template="rouge,bleu,vert">
27       <circle x="4cm" y="4cm" radius="25mm" fill-color="#ff00ff"/>
28     </card>
29   </deck>
30 </pandocreon:carta-genius>

```

Lignes 4 à 7 : Définition du premier gabarit.

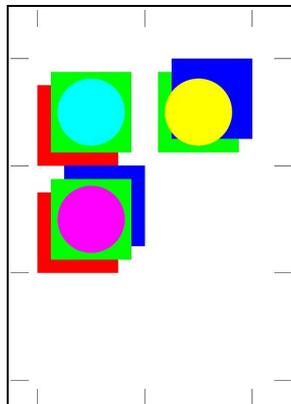
Lignes 8 à 11 : Définition du deuxième gabarit.

Lignes 12 à 15 : Définition du troisième gabarit.

Lignes 20 à 22 : Définition de la première carte, qui utilise 2 gabarits.

Lignes 23 à 25 : Définition de la deuxième carte, qui utilise 2 gabarits.

Lignes 26 à 28 : Définition de la troisième carte, qui utilise les 3 gabarits.



9.3 Sous-gabarits

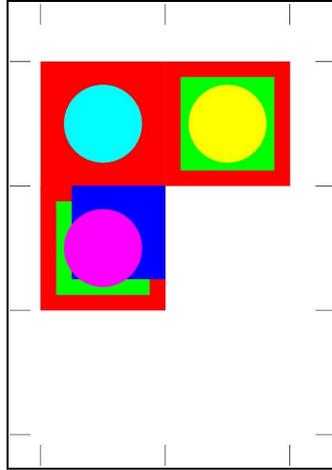
Il est possible de définir un gabarit qui fait lui-même référence à un ou plusieurs autres gabarits. Les éléments graphiques de ces sous-gabarits sont interprétés avant que le nouveau gabarit soit rendu. Il suffit d'ajouter un argument `template` aux déclarations de gabarits.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="fond">
05       <box x="0mm" y="0mm" width="8cm" height="8cm"
06         fill-color="#ff0000"/>
07     </template>
08     <template id="vert" template="fond">
09       <box x="1cm" y="1cm" width="6cm" height="6cm"
10         fill-color="#00ff00"/>
11     </template>
12     <template id="bleu" template="fond,vert">
13       <box x="2cm" y="2cm" width="6cm" height="6cm"
14         fill-color="#0000ff"/>
15     </template>
16   </templates>
17   <deck>
18     <paper type="a4" margin="2cm"/>
19     <cardsize width="8cm" height="8cm"/>
20     <card template="fond">
21       <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"/>
22     </card>
23     <card template="vert">
24       <circle x="4cm" y="4cm" radius="25mm" fill-color="#ffff00"/>
```

```

25     </card>
26     <card template="bleu">
27         <circle x="4cm" y="4cm" radius="25mm" fill-color="#ff00ff"/>
28     </card>
29 </deck>
30 </pandocreon:carta-genius>

```



9.4 Dos des cartes

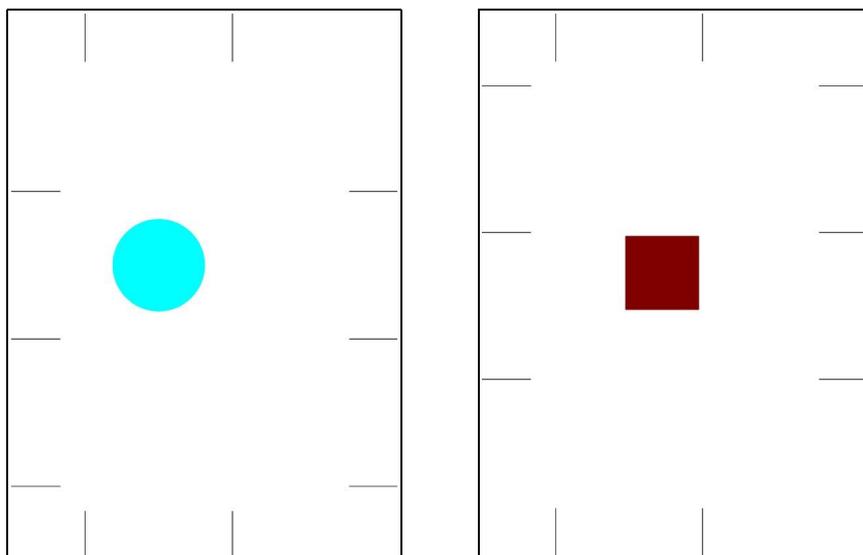
Nous avons vu, depuis le début de cette documentation, uniquement comment composer le recto des cartes. Pour ajouter un verso, il suffit de désigner pour chaque carte le gabarit à utiliser pour le composer, en ajoutant un argument `back` aux définitions des cartes.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03     <back side="width"/>
04     <templates>
05         <template id="fond">
06             <box x="2mm" y="2mm" width="4cm" height="4cm"
07                 fill-color="#800000"/>
08         </template>
09     </templates>
10 <deck>
11     <paper type="a4" margin="4cm"/>
12     <cardsize width="8cm" height="8cm"/>
13     <card back="fond">
14         <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"/>
15     </card>
16 </deck>
17 </pandocreon:carta-genius>

```

Lors de la génération, grâce à l'instruction de la ligne 3, Carta Genius crée deux fois plus de pages que d'habitude. Une page sur deux est constitué des versos.



Comme vous pouvez le voir, le template est utilisé sur la deuxième page, à l'emplacement qui correspond à la carte.

9.5 Appels en ligne des gabarits

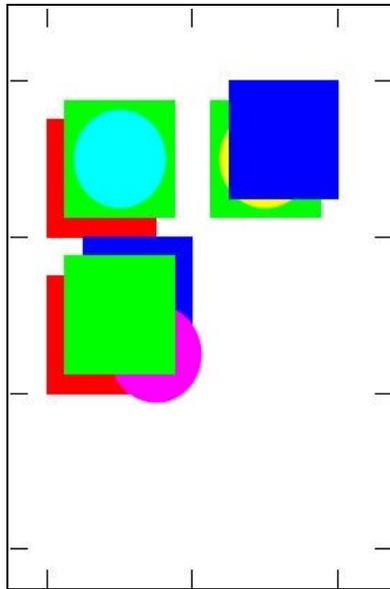
Pour une utilisation plus souple des gabarits, il est possible de les appeler explicitement dans les cartes. Pour cela, il faut employer la balise `<template>`, dont l'attribut `id` sert à spécifier le nom du ou des gabarits à inclure.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="rouge">
05       <box x="0mm" y="0mm" width="6cm" height="6cm"
06         fill-color="#ff0000"/>
07     </template>
08     <template id="vert">
09       <box x="1cm" y="1cm" width="6cm" height="6cm"
10         fill-color="#00ff00"/>
11     </template>
12     <template id="bleu">
13       <box x="2cm" y="2cm" width="6cm" height="6cm"
14         fill-color="#0000ff"/>
15     </template>
16   </templates>
17   <deck>
18     <paper type="a4" margin="2cm"/>
19     <cardsize width="8cm" height="8cm"/>
20     <card template="rouge,vert">
21       <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"/>
22     </card>
23     <card template="vert">
24       <circle x="4cm" y="4cm" radius="25mm" fill-color="#ffff00"/>
25     </card>
26   </deck>
27 </pandocreon:carta-genius>

```

```
26     </card>
27     <card>
28         <template id="rouge,bleu"/>
29         <circle x="4cm" y="4cm" radius="25mm" fill-color="#ff00ff"/>
30         <template id="vert"/>
31     </card>
32 </deck>
33 </pandocreon:carta-genius>
```



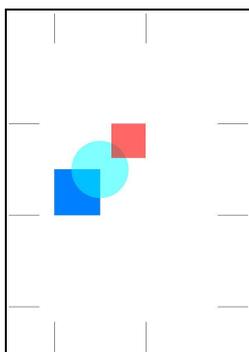
10. Options graphiques

Les objets graphiques (cercles, rectangles, rectangles arrondis, lignes, textes, mais pas les images) possèdent tous deux attributs supplémentaires, `opacity` et `blendmode`, dont nous n'avons pas encore parlé. Ils affectent la manière dont les objets s'insèrent dans le document PDF par rapport aux autres objets graphiques qu'ils recouvrent ou qui les recouvrent. Attention, ces paramètres ne sont pas utilisables avec des fichiers à la norme PDF 1.3 ; il vous faudra donc utiliser le logiciel Adobe Acrobat Reader version 5 ou supérieure (ou tout autre lecteur PDF supportant le PDF version 1.4 ou supérieure) et vous renseigner éventuellement sur les capacités des machines employées par l'imprimeur auquel vous comptez faire appel.

10.1 Transparence des objets graphiques

L'attribut `opacity` permet de modifier la transparence d'un objet. Par défaut, les objets n'ont pas de transparence, ce qui veut dire qu'ils sont opaques. L'opacité est un nombre flottant allant de 0 (transparence complète) à 1 (objet opaque). Voici un exemple :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="4cm"/>
05     <cardsize width="8cm" height="8cm"/>
06     <card>
07       <box x="0cm" y="0cm" width="4cm" height="4cm"
08         fill-color="#0080ff"/>
09       <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"
10         opacity="0.5"/>
11       <box x="5cm" y="5cm" width="3cm" height="3cm"
12         fill-color="#ff0000" opacity="0.6"/>
13     </card>
14   </deck>
15 </pandocreon:carta-genius>
```

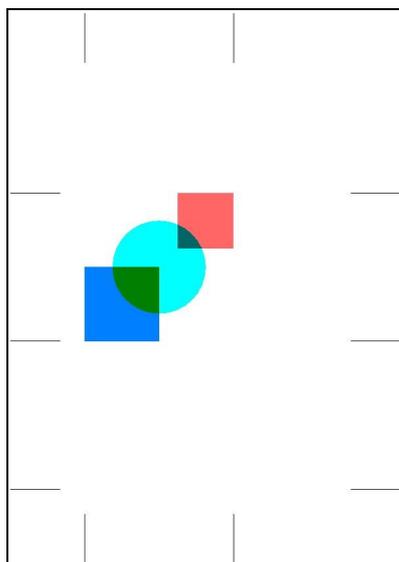


10.2 Mélange des couleurs

Le paramètre `blendmode` permet de choisir la méthode de mélange des couleurs. Cela permet de faire des effets similaires à celui de la transparence, mais aussi d'aboutir à des résultats impossibles avec le simple réglage de l'opacité. De plus, ces deux réglages peuvent être utilisés conjointement.

Les valeurs que peut prendre le paramètre `blendmode` sont : `Multiply`, `Screen`, `Overlay`, `Darken`, `Lighten`, `ColorDodge`, `ColorBurn`, `HardLight`, `Difference`, `Exclusion`, `Hue`, `Saturation`, `Color`, `Luminosity` (en respectant la casse majuscules/minuscules).

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03 <deck>
04 <paper type="a4" margin="4cm"/>
05 <cardsize width="8cm" height="8cm"/>
06 <card>
07 <box x="0cm" y="0cm" width="4cm" height="4cm"
08 <fill-color="#0080ff"/>
09 <circle x="4cm" y="4cm" radius="25mm" fill-color="#00ffff"
10 <blendmode="Exclusion"/>
11 <box x="5cm" y="5cm" width="3cm" height="3cm"
12 <fill-color="#ff0000" opacity="0.6" blendmode="Multiply"/>
13 </card>
14 </deck>
15 </pandocreon:carta-genius>
```



Multiply : Multiplie les valeurs de la couleur de fond et de la couleur source. La couleur résultant est toujours au moins aussi foncée que l'une ou l'autre des deux couleurs initiales. Multiplier une couleur avec du noir produit du noir. Multiplier une couleur avec du blanc laisse la couleur inchangée. Dessiner plusieurs objets se recouvrant successivement avec une couleur autre que noir ou blanc produit des couleurs de plus en plus sombres.

Screen : Multiplie les couleurs complémentaires de la couleur de fond et de la couleur source. La couleur résultant est toujours au moins aussi claire que l'une ou l'autre des deux couleurs initiales. 'Screener' avec du noir laisse la couleur inchangée. 'Screener' avec du blanc produit du blanc. L'effet est similaire à la projection simultanée de plusieurs transparents sur un même écran.

Overlay : Multiplie ou 'screene' les couleurs, en fonction de la couleur de fond. La couleur source recouvre la couleur de fond tout en préservant les reflets et les ombres de la couleur de fond. La couleur de fond n'est pas remplacée mais elle est mixée avec la couleur source en reflétant ses reflets et ses ombres.

SoftLight : Assombrit ou éclaircit les couleurs, en fonction de la couleur source. L'effet est similaire à l'application d'un spot lumineux diffus sur la couleur de fond. Si la couleur source est plus claire que sombre, la couleur de fond est éclaircie, comme si elles étaient combinées. Si la couleur source est plus sombre que claire, la couleur de fond est rendue plus sombre, comme si elles étaient 'cuites'. Effectué avec du noir ou du blanc produit des zones distinctement sombres ou claires, mais ne donne pas du noir ou du blanc pur.

HardLight : Multiplie ou 'screen' les couleurs, en fonction de la couleur source. L'effet est similaire à l'application d'un spot lumineux franc sur la couleur de fond. Si la couleur source est plus claire que sombre, la couleur de fond est éclaircie, comme si elles étaient 'screenées'. C'est pratique pour ajouter des reflets à une image. Si la couleur source est plus sombre que claire, la couleur de fond est assombrie, comme si elles étaient multipliées. C'est pratique pour ajouter des ombres à une image. Effectué avec du noir ou du blanc pur produit du noir ou du blanc pur.

ColorDodge : Eclaircie la couleur de fond pour refléter la couleur source. Effectué avec du noir ne produit aucun changement.

ColorBurn : Assombrie la couleur de fond pour refléter la couleur source. Effectué avec du blanc ne produit aucun changement.

Darken : Sélectionne le plus sombre de la couleur de fond et de la couleur source. La couleur de fond est remplacée par la couleur source là où la couleur source est plus sombre, autrement l'image reste inchangée.

Lighten : Sélectionne le plus clair de la couleur de fond et de la couleur source. La couleur de fond est remplacée par la couleur source là où la couleur source est plus claire, autrement l'image reste inchangée.

Difference : Soustrait la couleur source de la couleur de fond, ou la couleur de fond de la couleur source, en fonction de laquelle est la plus claire. Effectué avec du blanc inverse la couleur de fond, avec du noir cela ne produit aucun changement.

Exclusion : Produit un effet similaire, mais avec moins de contraste, que le mode 'Difference'. Effectué avec du blanc inverse la couleur de fond, du noir ne produit aucun changement.

Hue : Crée une couleur avec la luminosité et la saturation de la couleur de fond, et la teinte de la couleur source.

Saturation : Crée une couleur avec la luminosité et la teinte de la couleur de fond, et la saturation de la couleur source. Effectué sur du gris pur (pas de saturation) ne produit aucun changement.

Color : Crée une couleur avec la luminosité de la couleur de fond, et la teinte et la saturation de la couleur source. Cela préserve les niveaux de gris de la couleur de fond et est pratique pour coloriser les images monochromes et pour teinter les images couleurs.

Luminosity : Crée une couleur avec la teinte et la saturation de la couleur de fond, et la luminosité de la couleur source. Cela produit un effet inverse du mode 'Color'.

11. Evaluation d'expressions

Grâce à ce tout ce que nous avons passé en revue jusqu'ici dans cette documentation, vous êtes capable de créer de superbes (n'ayons pas peur des mots) cartes de jeu, et tout type de documents en fait. Mais vous aurez remarqué que le plus sûr moyen de réussir vos fichiers est encore de préparer minutieusement vos cartes sur du papier millimétré avant de créer vos fichiers XML. C'est le principe même de fonctionnement du logiciel, et cela fonctionne assez bien.

Pourtant, pour faciliter les choses, Carta Genius intègre un évaluateur d'expressions, qui vous permet de rendre les choses un peu plus souples à l'emploi. Par exemple, plutôt que de définir une zone de texte de 4 centimètres de large, n'est-il pas mieux de dire qu'elle fait la moitié de la largeur de la carte ? Ou encore, pourquoi ne pas utiliser une variable pour faire varier la composition des cartes en fonction de la page ?

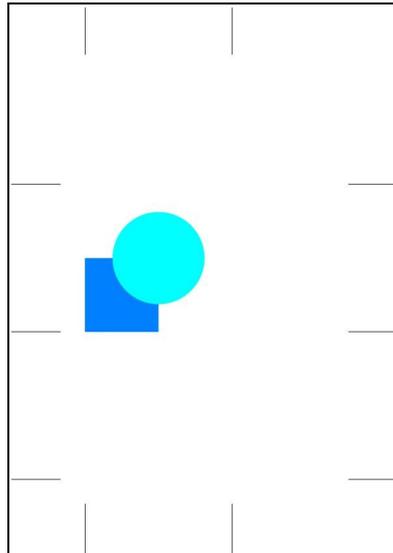
11.1 Expressions dans les attributs

Nous allons, comme à notre habitude, commencer avec un exemple simple, qui va illustrer la base de l'évaluateur d'expressions. La première chose à savoir est qu'il est désormais possible de mettre des formules mathématiques dans les attributs des objets graphiques.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="4cm"/>
05     <cardsize width="8cm" height="8cm"/>
06     <card>
07       <box x="0cm" y="0cm" width="2cm + 2cm" height="8cm - 40mm"
08         fill-color="#0080ff"/>
09       <circle x="(8cm + 16cm / 2) / 4" y="2cm * 2" radius="25mm"
10         fill-color="#00ffff"/>
11     </card>
12   </deck>
13 </pandocreon:carta-genius>
```

Comme vous pouvez le voir aux lignes 7 et 9, on utilise des formules mathématiques simples pour aboutir au résultat auquel on pourrait s'attendre. L'intérêt est nul dans cet exemple, mais il illustre bien que l'interpréteur comprend les constructions mathématiques standard et qu'il gère la priorité de certaines opérations par rapport aux autres (la multiplication, la division et le modulo sont prioritaires par rapport à l'addition et la soustraction).

Les opérateurs mathématiques utilisables sont l'addition (+), la soustraction (-), la multiplication (*), la division (/), et le modulo (%) qui est le reste de la division entière. Le parenthésage peut être utilisé sans retenue.



11.2 Types de constantes et de variables

Les types de données utilisables par l'évaluateur d'expressions sont au nombre de quatre :

- les scalaires, c'est à dire les nombres entiers ou flottants (12, 15.3, 2534.9924)
- les valeurs, distances constituées d'un scalaire suivant d'une unité (12in, 29.7cm, 450mm)
- les chaînes de caractères, obligatoirement encadrées de guillemets ("aaa", "titre")
- les éléments, représentant une surface en 2 dimensions, sont constitués de deux valeurs
- les couleurs, représentées par 3 nombres (rouge, vert, bleu) sur 2 caractères hexadécimal
- les booléens, qui ne peuvent prendre que 2 valeurs (« vrai » ou « faux »)

Et il existe un type supplémentaire un peu spécial, les variables, dont le nom commence par un caractère dollar (\$). Une variable peut contenir n'importe quelle donnée un peu donc prendre n'importe lequel des quatre types vus ci-dessus.

Ces différents types ont un impact sur la manière dont sont évaluées les opérations. Si l'on reprend la ligne 9 de l'exemple précédent, on peut lire :

```
y="2cm * 2"
```

Cette expressions est évaluée de gauche à droite, et le type de son résultat est une valeur, car on multiplie une valeur par autre chose (ici un scalaire). Si nous avons utilisé à la place :

```
y="2 * 2cm"
```

Cela nous aurait généré une erreur, car le résultat de ce calcul n'est plus une valeur mais un scalaire. Le type du résultat est le même que celui de l'opérande de gauche. Si vous avez besoin de modifier le type d'un calcul, il faut le transtyper (voir plus bas).

Les calculs sont un peu plus compliqués pour les couleurs. Il est possible de faire des opérations directement sur des couleurs :

```
line-color="#FF0000 + #0000FF"
```

La ligne sera alors de couleur violette (rouge plus bleu). Mais dans le cas d'opération entre une

couleur et un scalaire, il faut savoir que la représentation interne de chaque composante (rouge vert et bleu) est codée comme un nombre flottant entre 0 (zéro) et 1. Un bleu foncé vaudra par exemple 0.5 (la moitié de la valeur maximale). Par exemple, lors d'une addition entre une couleur et un scalaire, le scalaire est ajouté à chacune des composantes de la couleur :

```
fill-color="#800000 + 0.5"
```

Cela produira du rouge clair (rouge au maximum, vert et bleu à 50%).

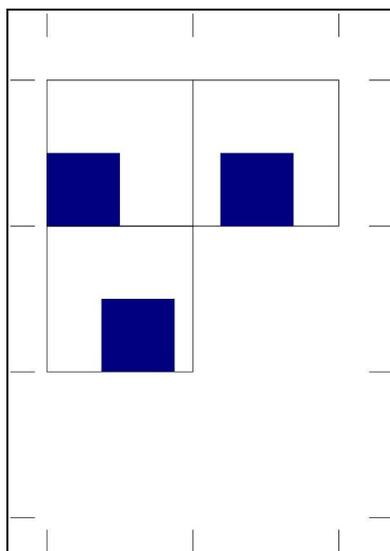
11.3 Variables dans les expressions

C'est bien beau de pouvoir mettre des expressions mathématiques dans les attributs des objets graphiques, mais encore faut-il que cela puisse servir à quelque chose d'intéressant. Pour ce faire, on peut utiliser des variables qui sont définies automatiquement par le programme. Par exemple, voici un fichier où les cartes contiennent un carré. Ce carré n'est pas à la même position pour toutes les cartes, uniquement grâce à l'utilisation de la variable `$CARD_INDEX`, qui contient le numéro de la carte dans le « deck ».

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="carre">
05       <box x="1.5cm * ($CARD_INDEX - 1)" y="0cm"
06         width="4cm" height="4cm" fill-color="#000080"/>
07     </template>
08   </templates>
09   <deck>
10     <paper type="a4" margin="2cm"/>
11     <cardsize width="8cm" height="8cm"/>
12     <card template="carre">
13       <box x="0cm" y="0cm" width="8cm" height="8cm"
14         line-color="#000000"/>
15     </card>
16     <card template="carre">
17       <box x="0cm" y="0cm" width="8cm" height="8cm"
18         line-color="#000000"/>
19     </card>
20     <card template="carre">
21       <box x="0cm" y="0cm" width="8cm" height="8cm"
22         line-color="#000000"/>
23     </card>
24   </deck>
25 </pandocreon:carta-genius>
```

Lignes 4 à 7 : Un gabarit est déclaré. Il contient un rectangle bleu dont le calcul de l'abscisse fait intervenir la variable `$CARD_INDEX`.

Lignes 12, 16 et 20 : Création des cartes, qui font référence au gabarit précédemment créé.



La liste des variables définies par le système est disponible à l'Annexe G.

11.4 Déclaration de variables

Comme nous venons de le voir, il existe plusieurs variables pré-définies. Mais il est aussi possible de définir vos propres variables, puis de les utiliser dans toutes les expressions où vous en aurez besoin. Une variable n'est pas une « constante », il est donc possible de redéfinir une variable déjà existante pour en modifier la valeur, que ce soit une variable que vous ayez créé ou qu'elle soit créée par le programme.

Pour définir ou redéfinir une variable, il suffit d'ajouter un élément `var` dans une carte ou dans un gabarit. A partir de là, la variable existe et est utilisable. Attention à trois choses :

- A la déclaration, le nom de la variable est donné sans caractère '\$', mais il ne faut pas l'oublier à l'utilisation.
- Tant qu'elle n'est pas redéfinie, la valeur d'une variable reste fixe. Elle n'est pas réévaluée.
- Si vous définissez une variable dans un gabarit, elle sera redéfinie à chaque utilisation du gabarit. Cela peut être l'effet recherché, mais attention si la valeur de votre variable est calculée dynamiquement.

```

01 <?xml version="1.0"?>
02   <pandocreon:carta-genius>
03     <templates>
04       <template id="fond">
05         <box x="0cm" y="0cm" width="8cm" height="8cm"
06           line-color="#000000"/>
07       </template>
08     </templates>
09     <deck>
10       <paper type="a4" margin="2cm"/>
11       <cardsize width="8cm" height="8cm"/>
12       <card template="fond">
13         <var name="abscisse">1.5cm * ($CARD_INDEX - 1)</var>
14         <box x="$abscisse" y="0cm" width="4cm" height="4cm"
15           fill-color="#000080"/>
16       </card>
17     <card template="fond">

```

```

18     <box x="$abscisse" y="0cm" width="4cm" height="4cm"
19         fill-color="#000080"/>
20     </card>
21     <card template="fond">
22         <var name="abscisse">1.5cm * ($CARD_INDEX - 1)</var>
23         <box x="$abscisse" y="0cm" width="4cm" height="4cm"
24             fill-color="#000080"/>
25     </card>
26 </deck>
27 </pandocreon:carta-genius>

```

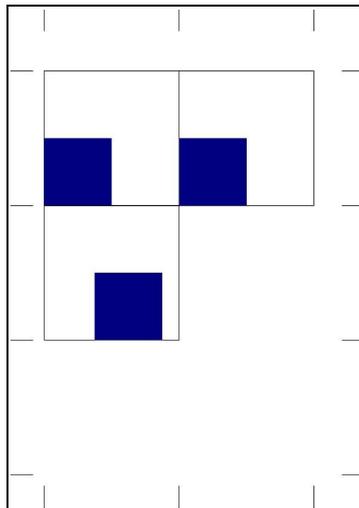
Ligne 13 : Une variable, nommée `abscisse` est créée. Sa valeur est fixée (ici le calcul vaut 0cm).

Ligne 14 : Création d'un rectangle dont le paramètre `x` vaut la variable `abscisse`.

Ligne 18 : Création d'un autre rectangle dont le paramètre `x` vaut la variable `abscisse`. La variable n'ayant pas été redéfinie, l'abscisse de ce rectangle est identique à celle du premier.

Ligne 22 : Redéfinition de la variable `abscisse`. Elle vaut maintenant 3cm.

Ligne 23 : Création d'un troisième rectangle, dont le paramètre `x` vaut la variable `abscisse`. La variable ayant été redéfinie, l'abscisse de ce rectangle est différente de celle des deux premiers.



Comment faire pour que la variable soit correctement redéfinie pour chaque carte ? Tout simplement en utilisant un gabarit dont le seul contenu sera la définition de la variable.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="init">
05       <var name="abscisse">1.5cm * ($CARD_INDEX - 1)</var>
06     </template>
07     <template id="fond">
08       <box x="0cm" y="0cm" width="8cm" height="8cm"
09         line-color="#000000"/>
10     </template>
11   </templates>
12 </deck>

```

```

13 <paper type="a4" margin="2cm"/>
14 <cardsize width="8cm" height="8cm"/>
15 <card template="init,fond">
16   <box x="$abscisse" y="0cm" width="4cm" height="4cm"
17     fill-color="#000080"/>
18 </card>
19 <card template="init,fond">
20   <box x="$abscisse" y="0cm" width="4cm" height="4cm"
21     fill-color="#000080"/>
22 </card>
23 <card template="init,fond">
24   <box x="$abscisse" y="0cm" width="4cm" height="4cm"
25     fill-color="#000080"/>
26 </card>
27 </deck>
28 </pandocreon:carta-genius>

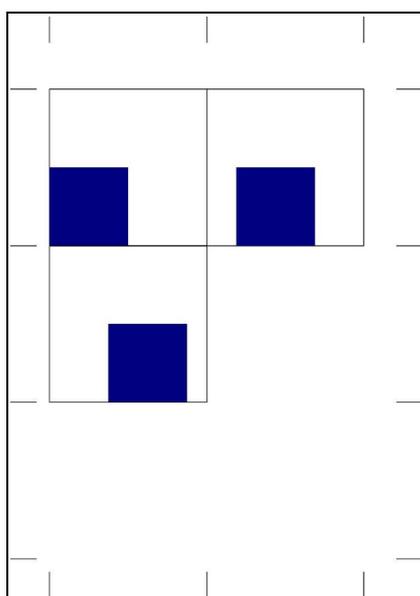
```

Lignes 4 à 6 : Déclaration d'un gabarit qui servira à initialiser la variable.

Ligne 5 : Définition de la variable.

Lignes 15, 19 et 23 : Les cartes font référence au gabarit d'initialisation.

Lignes 16, 20 et 24 : Les rectangles sont créés, avec la variable telle qu'elle a été initialisée.



11.5 Déclaration initiale des variables

Il est possible de déclarer des variables avant même l'appel de la première carte. Cela permet de définir proprement les variables en dehors des cartes ou des gabarits. Pour cela, il existe une section `<variables>` au même niveau que les déclarations de gabarits (`<templates>`), d'images (`<images>`) ou les inclusions de fichiers (`<includes>`). Cette section peut contenir autant d'éléments `<var>` que vous désirez.

11.6 Insertion d'expressions dans du texte

Les expressions, avec ou sans utilisation de variables, permettent donc de faire varier les valeurs des arguments des objets graphiques. Mais il est aussi possible de les utiliser pour ajouter des données dans les zones de texte. Pour cela, il faut utiliser l'élément `value`, qui peut contenir n'importe quelle expression (qui sera transformée en chaîne de caractères).

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize width="16cm" height="8cm"/>
06     <card>
07       <var name="tralala">"pouet pouet"</var>
08       <var name="numero">172.99</var>
09       <text font="Helvetica" size="20pt" color="#000000"
10         x="1cm" y="1cm" width="10cm" height="6cm">
11         Une chaîne : <value>$tralala</value><br/>
12         Un nombre : <value>$numero * 0.5</value><br/>
13         Index : <value precision="0">${CARD_INDEX}</value>
14       </text>
15     </card>
16   </deck>
17 </pandocreon:carta-genius>
```

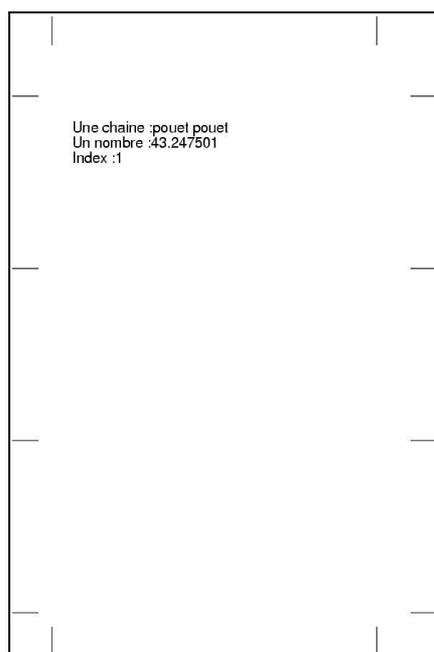
Ligne 7 : Définition d'une variable contenant une chaîne de caractères. Notez les guillemets de chaque côté de la chaîne.

Ligne 8 : Définition d'une variable contenant un scalaire.

Ligne 11 : Insertion de la chaîne.

Ligne 12 : Insertion du résultat d'un calcul effectué sur une variable.

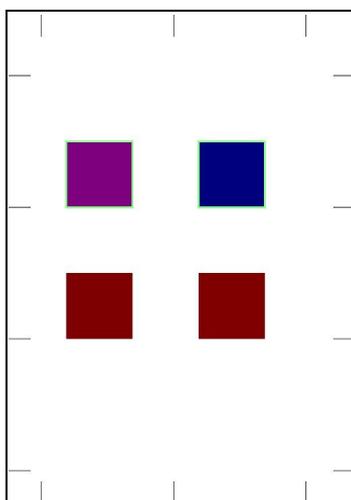
Ligne 13 : Insertion du numéro de la carte. Notez le paramètre `precision` de la balise `value`, qui détermine le nombre de chiffres à afficher après la virgule des scalaires. Si ce paramètre n'est pas renseigné, il est mis au maximum (6).



11.7 Variables de couleur

Voici un petit exemple illustrant l'utilisation des variables pour les couleurs, ainsi que les calculs qui peuvent être effectués avec ces variables.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize width="8cm" height="8cm"/>
06     <card>
07       <var name="couleur">$DARK_BLUE + $DARK_RED</var>
08       <box x="1.5cm" y="0cm" width="4cm" height="4cm"
09         fill-color="$couleur" line-color="#80ff80"
10         line-width="1mm"/>
11     </card>
12     <card>
13       <box x="1.5cm" y="0cm" width="4cm" height="4cm"
14         fill-color="#0000ff / 2" line-color="#008000 + 0.5"
15         line-width="1mm"/>
16     </card>
17     <card>
18       <box x="1.5cm" y="0cm" width="4cm" height="4cm"
19         fill-color="#ffffff - #80ffff"/>
20     </card>
21     <card>
22       <box x="1.5cm" y="0cm" width="4cm" height="4cm"
23         fill-color="#800000"/>
24     </card>
25   </deck>
26 </pandocreon:carta-genius>
```



11.8 Fonctions

En plus des variables et des opérations mathématiques usuelles, l'interpréteur contient un certain nombre de fonctions utilisables dans les expressions. La liste complète de ces fonctions est disponible à l'Annexe G.

Prenons comme exemple une carte dans laquelle on veut insérer :

- un rectangle sur la moitié de la hauteur de la carte
- un cercle qui soit centré et dont le rayon soit égale à la marge de la page
- un texte et son nombre de caractères

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize width="16cm" height="8cm"/>
06     <card>
07       <var name="texte">"ceci est un test"</var>
08       <box x="0mm" y="0mm" width="width($CARD) "
09         height="height($CARD) / 2" fill-color="#0000ff"/>
10       <circle x="width($CARD) / 2" y="height($CARD) / 2"
11         radius="width($MARGIN) " fill-color="#ff0000"/>
12       <text font="Helvetica" size="20pt" color="#000000"
13         x="1cm" y="1cm" width="10cm" height="6cm">
14         <value>$texte</value><p/>
15         <value precision="0">strlen($texte)</value> caracteres
16       </text>
17     </card>
18   </deck>
19 </pandocreon:carta-genius>
```

Ligne 7 : On définit la chaîne de caractère qui sera insérée.

Ligne 8 : Définition du rectangle. Sa largeur sera égale à la largeur de la carte.

Ligne 9 : La hauteur du rectangle sera égale à la moitié de la hauteur de la carte.

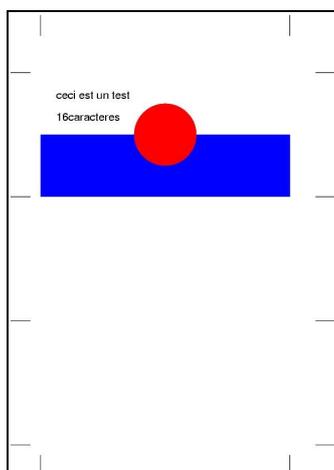
Ligne 10 : Définition du cercle. Ses coordonnées correspondent au centre de la carte (moitié de la largeur en abscisse, moitié de la hauteur en ordonnée).

Ligne 11 : Le rayon du cercle vaut la largeur de la marge. La marge étant la même en largeur et en hauteur (car elle est défini grâce à l'attribut margin ligne 4, et non pas en utilisant les attributs margin-width et margin-height), on aurait pu écrire tout aussi bien

```
radius="height($MARGIN) "
```

Ligne 14 : On insère le texte.

Ligne 15 : On ajoute la longueur du texte.



11.9 Variables de variables

Il existe une variable de transtypage, nommée `var()`, qui permet d'utiliser une chaîne de caractère (éventuellement contenue dans une variable) pour récupérer la valeur d'une expression ou d'une variable (à partir de son nom). Cela permet d'utiliser différentes variables suivant, par exemple, la carte en cours de traitement. Voici un exemple illustrant cela.

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="init">
05       <var name="couleur1">#800000</var>
06       <var name="couleur2">#008000</var>
07     </template>
08   </templates>
09   <deck>
10     <paper type="a4" margin="2cm"/>
11     <cardsize width="16cm" height="8cm"/>
12     <card template="init">
13       <var name="fond" precision="0">"couleur" + $CARD_INDEX</var>
14       <box x="0mm" y="0mm" width="width($CARD) "
15         height="height($CARD) / 2"
16         fill-color="var($fond)"/>
17     </card>
18     <card template="init">
19       <var name="fond" precision="0">"couleur" + $CARD_INDEX</var>
20       <box x="0mm" y="0mm" width="width($CARD) "
21         height="height($CARD) / 2"
22         fill-color="var($fond)"/>
23     </card>
24   </deck>
25 </pandocreon:carta-genius>
```

Lignes 5 et 6 : Création de deux variables contenant deux couleurs différentes.

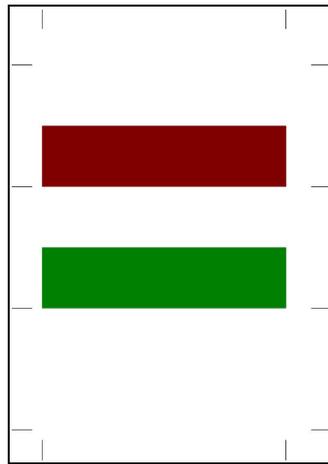
Ligne 12 : Première carte.

Ligne 13 : Création de la variable `$fond`, résultat de la concaténation d'une chaîne de caractère avec un scalaire. Le paramètre `precision` est utilisé afin que la variable `$CARD_INDEX` soit prise sous la forme "1" et non pas "1.0000000". La variable `$fond` contient donc la chaîne "couleur1".

Ligne 16 : Un rectangle est créé, et rempli de la couleur de la variable `$couleur1`. La fonction `var()` interprète le contenu de la variable `$fond`, et retourne le contenu de la variable dont le nom y est placé.

Ligne 19 : La variable `$fond` est redéfinie. Elle vaut maintenant "couleur2".

Ligne 22 : Création d'un second rectangle. Sa couleur de fond sera celle contenue dans la variable `$couleur2`.



12. Instructions de branchement et expressions booléennes

L'évaluation d'expressions est assez pratique, associée à la déclaration de variables, pour rendre le code de description des cartes plus lisible. Mais, associée aux expressions booléennes, elle prend toute son ampleur avec les instructions de branchement ; ces instructions sont de deux types : les instructions conditionnelles et les boucles.

12.1 Instructions conditionnelles

Les conditions sont utilisables avec la balise `<if>`, qui permet de tester une expression booléenne. Si cette expression est positive (ou plutôt si elle est « vraie »), le code conditionnel est exécuté.

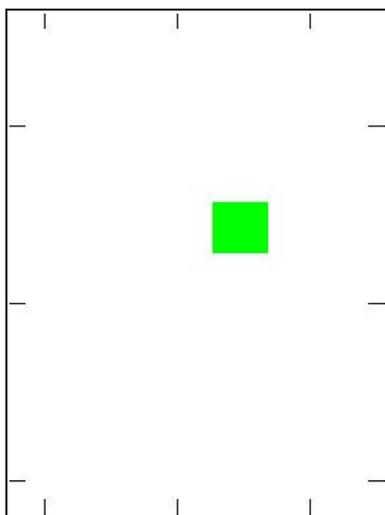
```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <deck>
04     <paper type="a4" margin="2cm"/>
05     <cardsize type="a7"/>
06     <card>
07       <if test="$FALSE">
08         <circle x="3cm" y="5cm" radius="20mm"
09           fill-color="#0000ff"/>
10       </if>
11     </card>
12     <card>
13       <if test="$TRUE">
14         <box x="2cm" y="3cm" width="3cm" height="3cm"
15           fill-color="#00ff00"/>
16       </if>
17     </card>
18   </deck>
19 </pandocreon:carta-genius>
```

Ligne 7 : Début d'une condition. Le test est simple, et sa valeur sera fausse, car égale à la variable booléenne `$FALSE`.

Ligne 10 : Fin de la première condition.

Ligne 13 : Une deuxième condition, dans une seconde carte. Le test est ici tout aussi simple, et le code conditionnel sera exécuté.

Si on regarde le résultat, on peut voir que le cercle de la première carte ne s'affiche pas, alors que le carré de la seconde carte s'affiche bien. Les tests conditionnels ont été interprétés de la manière attendue.



12.2 Fonctions (et opérateurs) de comparaison

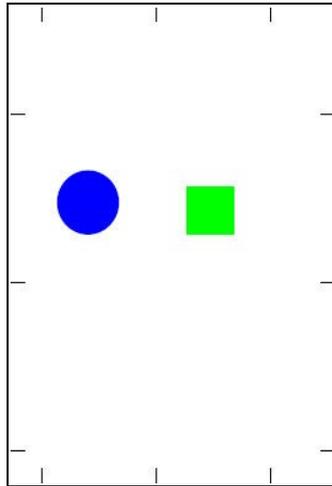
Plusieurs fonctions permettent de rendre les instructions conditionnelles plus intéressantes. Elles servent, par exemple, à tester la valeur d'une variable.

Voici un premier exemple, dans lequel on définit un gabarit qui teste le numéro de carte courante, et décide en fonction de cela de la forme à afficher :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="gabarit">
05       <if test="$CARD_INDEX = 1">
06         <circle x="3cm" y="5cm" radius="20mm"
07           fill-color="#0000ff"/>
08       </if>
09       <if test="eq($CARD_INDEX, 2)">
10         <box x="2cm" y="3cm" width="3cm" height="3cm"
11           fill-color="#00ff00"/>
12       </if>
13     </template>
14   </templates>
15 <deck>
16   <paper type="a4" margin="2cm"/>
17   <cardsize type="a7"/>
18   <card template="gabarit"/>
19   <card template="gabarit"/>
20 </deck>
21 </pandocreon:carta-genius>
```

Ligne 5 : On déclare une condition dans laquelle on teste l'égalité entre le numéro de carte courante et la valeur scalaire « 1 ». Ce code ne s'exécutera donc que lorsque le gabarit sera appelé par la première carte.

Ligne 9 : On définit une autre condition, qui teste elle aussi l'égalité du numéro de carte courante. La fonction booléenne `eq()` est identique à l'utilisation de l'opérateur « = ».



Les fonctions de comparaison sont au nombre de 3 :

- `eq()`, qui teste l'égalité entre deux valeurs (opérateur « = »).
- `lt()`, qui teste si une valeur est inférieure à une autre (opérateur « < »).
- `gt()`, qui teste si une valeur est supérieure à une autre (opérateur « > »).

12.3 Fonctions (et opérateurs) logiques

Il est possible de composer des expressions booléennes complexes en utilisant les opérateurs logiques suivants :

- `and()`, qui est vrai si les deux expressions sont vraies (opérateur « & »).
- `or()`, qui est vrai si l'une des deux expressions est vraie (opérateur « | »).
- `not()`, qui retourne une valeur booléenne inverse à celle donnée en paramètre (pas d'opérateur).

Voici un exemple :

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="gabarit">
05       <if test="($CARD_INDEX = 2 | $CARD_INDEX = 3) &
06         $position > 3cm">
07         <circle x="$position" y="5cm" radius="20mm"
08           fill-color="#0000ff"/>
09       </if>
10       <var name="fill">#00ff00</var>
11       <if test="$CARD_INDEX = 2 | $CARD_INDEX = 4">
12         <var name="fill">#ff0000</var>
13       </if>
14       <box x="2cm" y="3cm" width="3cm" height="3cm"
15         fill-color="$fill"/>
16     </template>
17   </templates>
18   <deck>
19     <paper type="a4" margin="2cm"/>
20     <cardsize type="a7"/>
```

```

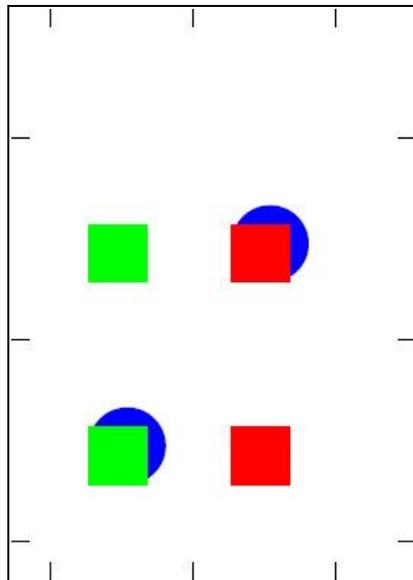
21     <card>
22         <var name="position">2cm</var>
23         <template id="gabarit"/>
24     </card>
25     <card>
26         <var name="position">4cm</var>
27         <template id="gabarit"/>
28     </card>
29     <card template="gabarit"/>
30     <card template="gabarit"/>
31 </deck>
32 </pandocreon:carta-genius>

```

Lignes 5 et 6 : Le code de ce test sera exécuté s'il s'agit de la 2ème ou de la 3ème carte, et que la variable \$position vaut plus de 3 centimètres.

Lignes 7 et 8 : On ajoute dans ce cas-là un cercle bleu.

Lignes 11 à 13 : S'il s'agit de la 2ème ou de la 4ème carte, la couleur du carré suivant sera rouge.



12.4 Boucles

Les boucles permettent de relancer plusieurs fois un même ensemble d'instructions, en donnant une condition qui va décider si la boucle doit continuer à se faire ou non. Elles sont définies avec la balise `<while>`. Attention à ne pas créer une boucle infinie, ce qui peut arriver si votre condition de bouclage est vraie en toutes circonstances.

Voici un exemple un peu complexe, qui intègre un peu toutes les notions vues précédemment.

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03     <deck>
04         <paper type="a4" landscape="yes" margin="2cm"/>
05         <cardsize type="a7"/>

```

```

06     <card>
07         <circle x="3cm" y="5cm" radius="20mm" fill-color="#0000ff"/>
08         <var name="offset">5.0</var>
09         <while test="$offset < 25.0">
10             <if test="($offset % 2) > 0">
11                 <var name="dedans">#aaaaaa</var>
12                 <var name="dehors">#cccccc</var>
13             </if>
14             <if test="($offset - 1) % 2) = 0">
15                 <var name="dedans">#cccccc</var>
16                 <var name="dehors">#aaaaaa</var>
17             </if>
18             <var name="temp">string(($offset / 2) + 10) + "cm"</var>
19             <var name="y">2cm * cos($offset * 20)</var>
20             <circle x="var($temp)" y="8cm + $y" radius="2.2cm"
21                 fill-color="$dedans" line-color="$dehors"/>
22             <var name="offset">$offset + 1</var>
23         </while>
24     </card>
25 </deck>
26 </pandocreon:carta-genius>

```

Ligne 8 : On définit la variable \$offset.

Ligne 9 : On crée une boucle ; celle-ci va s'effectuer tant que la variable \$offset sera inférieure à une certaine valeur.

Lignes 10 et 14 : On teste la valeur de \$offset modulo 2. Cela permet d'effectuer des traitements une fois sur deux. Ici nous utilisons cela pour définir des couleurs qui vont changer à chaque tour de boucle.

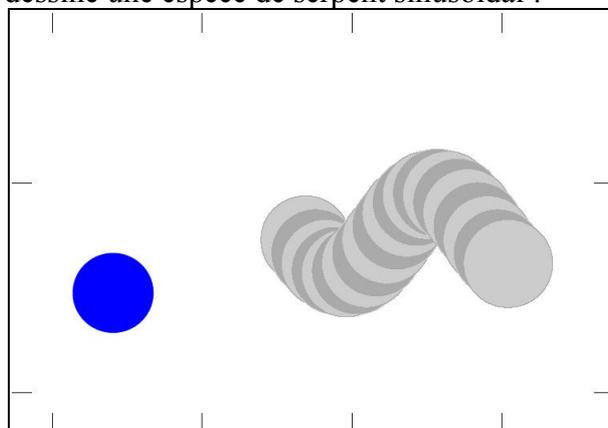
Ligne 18 : On crée une variable \$temp, qui va contenir une chaîne de caractères correspondant à une longueur en centimètres.

Ligne 19 : On crée une variable \$y, en utilisant la fonction mathématique cosinus, afin d'avoir une valeur cyclique.

Lignes 20 et 21 : On affiche un cercle, dont les coordonnées et les couleurs sont le résultat des expressions précédentes. On peut remarquer que l'abscisse utilise la fonction var(), qui permet d'interpréter une chaîne de caractères ; ici nous interprétons la variable \$temp, qui contient une chaîne décrivant une longueur.

Ligne 22 : On incrémente la variable \$offset.

Et voici le résultat, qui dessine une espèce de serpent sinusoïdal :



13. Messages d'information

Lorsque l'on travaille sur un document un peu délicat, avec de nombreuses fonctions et variables, il est souvent intéressant de connaître la valeur de certaines variables, sans pour autant avoir besoin de définir une zone de texte où afficher ces variables. Dans ces cas-là, il est possible d'utiliser la balise `<log>`, qui permet d'afficher une variable (ou le résultat d'une expression) sur la sortie d'erreur du programme.

L'utilisation est très simple :

```
<log>$variable</log>
<log>"une chaîne" + "une autre"</log>
<log>$var1 + $var2</log>
<log precision="2">sin($variable)</log>
<log precision="0">$CARD_INDEX</log>
```

L'attribut `precision` sert à définir le nombre de chiffres après la virgule qui s'afficheront.

14. Fichiers multiples

14.1 Fonctionnement

Nous avons toujours travaillé, jusqu'ici, avec un seul et unique fichier XML monolithique, qui contenait toutes les déclarations nécessaires à la constitution du document PDF. La plupart du temps vous pourrez procéder de la sorte, mais il vous sera parfois nécessaire de découper votre document en plusieurs fichiers XML, pour les rendre plus facilement maintenant et réutilisables.

Il est donc possible d'inclure un ou plusieurs fichiers XML dans un autre. Les éléments qui sont pris, dans les XML inclus, sont les déclarations des gabarits, des polices de caractères, des images et des « decks ». Le résultat est identique à ce que donnerait un unique fichier contenant toutes les déclarations.

Concrètement, il faut ajouter une balise `includes` au même niveau que `fonts`, `images` et `templates`. Cette balise va contenir autant de balise `include` qu'il y a de fichiers à inclure, avec un paramètre `file` contenant le nom du fichier à inclure. Un fichier inclus peut à son tour inclure d'autres fichiers. Attention alors à ne pas faire de boucle !

```
01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="init">
05       <var name="taille">width($CARD) * 2 / 3</var>
06     </template>
07   </templates>
08   <includes>
09     <include file="33-inclusion_inclus.xml"/>
10   </includes>
11   <deck>
12     <paper type="a4" margin="2cm"/>
13     <cardsize width="16cm" height="8cm"/>
14     <card template="init,fond">
15       <circle x="width($CARD) / 2" y="height($CARD) / 2"
16         radius="width($MARGIN)" fill-color="#ff0000"/>
17     </card>
18   </deck>
19 </pandocreon:carta-genius>
```

Lignes 4 à 6 : Un template d'initialisation de variable.

Ligne 9 : On demande à inclure un autre fichier XML.

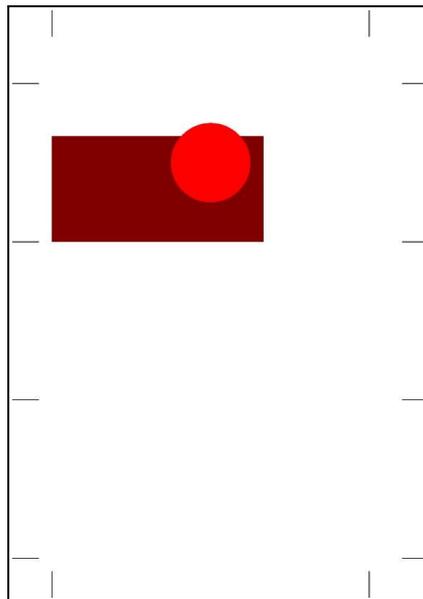
Et voici le fichier inclus :

```

01 <?xml version="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="fond">
05       <box x="0mm" y="0mm" width="$taille" height="$taille / 2"
06         fill-color="#800000"/>
07     </template>
08   </templates>
09 </pandocreon:carta-genius>

```

Ce fichier se content de définir un template, qui utilise la variable définie dans le fichier principal. Cela ne pose aucun problème. Voici le résultat :



14.2 Répertoires d'inclusion

Par défaut, les chemins vers les fichiers sont donnés en entier, et peuvent être relatifs ou absolus. Mais si vous souhaitez placer tous vos fichiers à inclure dans un même répertoire, il peut être fastidieux de mettre le chemin complet vers ce répertoire pour chaque fichier ; surtout si le chemin vers le répertoire peut être amené à changer.

Il est donc possible, au même endroit que la déclaration des fichiers à inclure, d'ajouter des répertoire pouvant servir de base aux chemins vers les fichiers :

```

01 <includes>
02   <path dir="/home/" />
03   <include file="toto.xml" />
04   <path dir="/usr/local/" />
04   <include file="titi/tutu.xml" />
05 </includes>

```

Ligne 2 : On ajoute le répertoire /home/ à la liste des chemins d'inclusion.

Ligne 3 : On veut inclure le fichier toto.xml. Le programme va donc chercher "./toto.xml", puis "/home/toto.xml".

Ligne 4 : On ajoute le répertoire `/usr/local/` à la liste des chemins d'inclusion.

Ligne 5 : On veut inclure le fichier `titi/tutu.xml`. Le programme va donc chercher `./titi/tutu.xml`, puis `/home/titi/tutu.xml`, puis `/usr/local/titi/tutu.xml`.

Le programme arrête de chercher dès qu'il trouve un fichier correspondant. Il faut remarquer que les chemins déclarés doivent obligatoirement se terminer par un slash (/), ou un backslash (\) sous Windows, pour garantir que la concaténation des chemins se fera correctement.

14.3 Application aux images

De la même manière que pour les fichiers à inclure, il est possible de spécifier des répertoires de recherche des images. La syntaxe est strictement la même que celle vue dans le paragraphe précédent. Si nous prenons un exemple :

```
01 <images>
02   <path dir="subdir"/>
03   <image id="fond" file="fond.jpg" width="15cm" height="20cm"/>
04   <path dir="../autre_images"/>
05   <image id="icone" file="icone.gif" width="3cm" height="4cm"/>
06 </images>
```

Ligne 2 : On ajoute le répertoire `subdir` à la liste des chemins de recherche d'image.

Ligne 3 : On importe une image. Le programme va chercher le fichier `fond.jpg` dans le répertoire courant, et sinon dans le répertoire `subdir`.

Ligne 4 : On ajoute le répertoire `../autre_images` à la liste des chemins.

Ligne 5 : On importe une deuxième image. Le programme va la chercher dans le répertoire courant, puis dans le répertoire `subdir`, puis dans `../autre_images`.

14.4 Application aux polices de caractères

De la même manière que ce que nous venons de voir pour les fichiers à inclure et pour les images, il est possible d'ajouter un ou plusieurs éléments `path` dans les éléments `fonts`, pour définir un chemin de recherche des fichiers de polices de caractères.

15. Extensions

Vous connaissez maintenant tout ce qu'il faut savoir pour faire des cartes de jeu et tout type de documents, en utilisant le programme Carta Genius. Nous allons maintenant passer en revue quelques fichiers qui sont fournis dans le répertoire 'resources' (présent dans l'archive de Carta Genius, version sources ou version compilée), et qui peuvent être inclus dans vos fichiers XML pour réaliser facilement certains documents pour vous faciliter les choses.

15.1 Fabrication d'étuis de cartes

Une des choses dont on peut avoir rapidement besoin, après avoir fait des cartes de jeu, c'est un étui qui puisse contenir ces cartes. Pour cela, vous pouvez inclure le fichier `inc_cardcase.xml`, qui comporte un gabarit nommé « cardcase » qui fait tout le boulot.

```
01 <?xml verison="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="init">
05       <var name="case_width">5.4cm</var>
06       <var name="case_height">9cm</var>
07       <var name="case_depth">4cm</var>
08       <var name="case_flap_big">2cm</var>
09       <var name="case_flap_small">1.3cm</var>
10       <var name="case_gap_big">8mm</var>
11       <var name="case_gap_small">4mm</var>
12     </template>
13   </templates>
14   <includes>
15     <include file="inc_cardcase.xml"/>
16   </includes>
17   <deck>
18     <paper type="a4" landscape="yes" margin="0mm"/>
19     <cardsize type="a4" landscape="yes"/>
20     <hidden-ditch odd="#800000"/>
21     <card template="init,cardcase"/>
22   </deck>
23 </pandocreon:carta-genius>
```

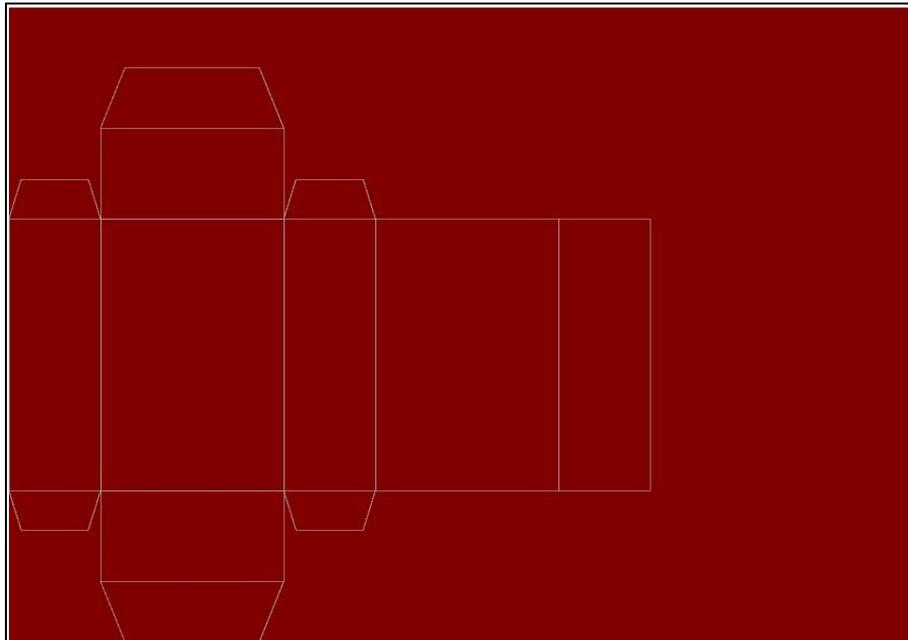
Lignes 4 à 12 : Définition d'un gabarit qui contient certaines variables utilisées par `cardcase`.

Ligne 5 : La variable `case_width` contient la largeur de l'étui.

Ligne 6 : `case_height` contient la hauteur de l'étui.

Ligne 7 : `case_depth` contient la profondeur de l'étui.

- Ligne 8 : `case_flap_big` contient la taille des rabats supérieur et inférieur.
- Ligne 9 : `case_flap_small` contient la taille des rabats latéraux.
- Ligne 10 : `case_gap_big` contient la taille du décalage des rabats supérieur et inférieur.
- Ligne 11 : `case_gap_small` contient la taille du décalage des rabats latéraux.
- Ligne 18 : On définit un papier de taille A4 mode paysage, sans marge.
- Ligne 19 : On définit une taille de carte égale à celle du papier.
- Ligne 20 : On place un fond-perdu rouge foncé.
- Ligne 21 : Une seule carte, qui appelle le gabarit d'initialisation des variables, puis le gabarit qui crée l'étui.



Et voilà le résultat. De fines lignes grises sont placées dans la carte, dessinant un étui de cartes. N'oubliez pas de spécifier une taille de page et de carte suffisamment grandes pour pouvoir afficher l'intégralité de l'étui. Le gabarit `cardcase` doit être appelé en dernier, afin que les lignes grises recouvrent vos autres éléments graphiques.

Maintenant, pour placer des objets graphiques facilement par rapport à la forme de boîte ainsi créée, il serait bien d'avoir à disposition les coordonnées et la taille des différentes zones qui composent le dessin. Pour cela, ces zones vont être nommées (A, B, C, D et E pour les quatre rectangles figurant l'avant, l'arrière et les côtés de la boîte, F et G pour les rectangles figurants les côtés supérieur et inférieur).

Pour avoir accès aux variables qui fournissent ces informations, il faut passer par un gabarit nommé `cardcase_init`. Attention surtout à utiliser les variables mais à ne pas les modifier, sinon le dessin de la boîte serait modifié.

```

01 <?xml verison="1.0"?>
02 <pandocreon:carta-genius>
03   <templates>
04     <template id="init">
05       <var name="case_width">6cm</var>
06       <var name="case_height">9cm</var>
07       <var name="case_depth">3cm</var>

```

```

08     <var name="case_flap_big">2cm</var>
09     <var name="case_flap_small">1.3cm</var>
10     <var name="case_gap_big">8mm</var>
11     <var name="case_gap_small">4mm</var>
12     </template>
13     <template id="etui" template="cardcase_init">
14         <box x="$CARD_CASE_A_X + 5mm" y="$CARD_CASE_A_Y + 5mm"
15             width="$CARD_CASE_A_W - 1cm" height="$CARD_CASE_A_H - 1cm"
16             fill-color="#0000ff"/>
17         <box x="$CARD_CASE_B_X + 5mm" y="$CARD_CASE_B_Y + 5mm"
18             width="$CARD_CASE_B_W - 1cm" height="$CARD_CASE_B_H - 1cm"
19             fill-color="#00ff00"/>
20         <box x="$CARD_CASE_C_X + 5mm" y="$CARD_CASE_C_Y + 5mm"
21             width="$CARD_CASE_C_W - 1cm" height="$CARD_CASE_C_H - 1cm"
22             fill-color="#0000ff"/>
23         <box x="$CARD_CASE_D_X + 5mm" y="$CARD_CASE_D_Y + 5mm"
24             width="$CARD_CASE_D_W - 1cm" height="$CARD_CASE_D_H - 1cm"
25             fill-color="#00ff00"/>
26         <box x="$CARD_CASE_E_X + 5mm" y="$CARD_CASE_E_Y + 5mm"
27             width="$CARD_CASE_E_W - 1cm" height="$CARD_CASE_E_H - 1cm"
28             fill-color="#0000ff"/>
29         <box x="$CARD_CASE_F_X + 5mm" y="$CARD_CASE_F_Y + 5mm"
30             width="$CARD_CASE_F_W - 1cm" height="$CARD_CASE_F_H - 1cm"
31             fill-color="#ffff00"/>
32         <box x="$CARD_CASE_G_X + 5mm" y="$CARD_CASE_G_Y + 5mm"
33             width="$CARD_CASE_G_W - 1cm" height="$CARD_CASE_G_H - 1cm"
34             fill-color="#ffff00"/>
35     </template>
36 </templates>
37 <includes>
38     <include file="inc_cardcase.xml"/>
39 </includes>
40 <deck>
41     <paper type="a4" landscape="yes" margin="0mm"/>
42     <cardsize type="a4" landscape="yes"/>
43     <hidden-ditch odd="#800000"/>
44     <card template="init,etui,cardcase"/>
45 </deck>
46 </pandocreon:carta-genius>

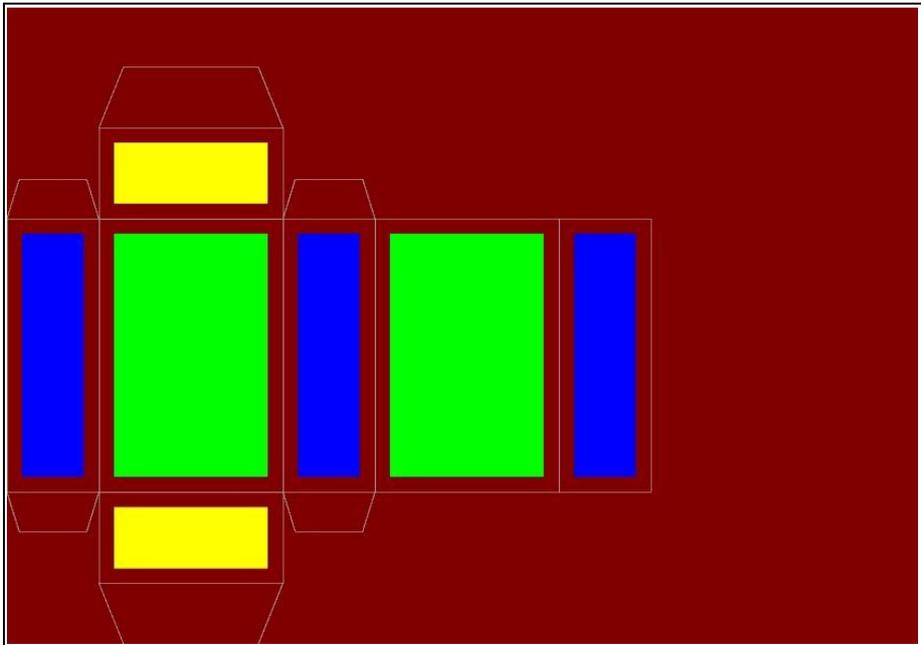
```

Lignes 4 à 12 : On définit les variables nécessaires à la constitution de l'étui.

Lignes 13 à 35 : On crée un autre gabarit, qui appelle le gabarit `cardcase_init`. Il récupère ainsi plusieurs variables. Par exemple, les variables `$CARD_CASE_A_X` et `$CARD_CASE_A_Y` donnent les coordonnées du rectangle A, alors que `$CARD_CASE_A_W` et `$CARD_CASE_A_H` donnent respectivement sa largeur et sa hauteur. Plusieurs rectangles de couleur sont placés dans les zones A, B, C, D, E, F et G de l'étui.

Ligne 38 : Inclusion du fichier XML qui va constituer l'étui.

Ligne 44 : Une seule et unique carte, qui appelle le gabarit d'initialisation des variables, puis le gabarit qui remplit la carte, puis enfin le gabarit qui crée l'étui.



Annexe A. Compilation de Carta Genius

A.1 PDFLib

Vous devez avoir la PDFLib d'installée. Cette bibliothèque de fonctions est gratuite lorsqu'elle est utilisée avec un logiciel libre comme Carta-Genius. Vous pouvez récupérer la version gratuite de la PDFLib ici : <http://www.pdflib.com/products/download-source.html>

1. Téléchargez et décompressez le fichier.
2. Configurez PDFLib : `./configure --prefix=/usr`
3. Compilez la bibliothèque : `make`
4. Installez-la : `make install`

Vous devez être root pour exécuter la quatrième étape. Si vous ne pouvez pas l'installer en tant qu'utilisateur root, lisez le paragraphe "Compilation statique" ci-dessous.

A.2 Compilation dynamique

Une fois que la PDFLib est compilée et installée à un endroit connu du système (comme /usr/lib), vous pouvez faire :

1. Allez dans le répertoire de Carta-Genius et tapez `./configure` afin de vérifier que la configuration du système est correcte.
2. Tapez `make` pour compiler l'application.

Le fichier exécutable appelé `carta-genius` se trouve dans le répertoire `bin`.

Vous pouvez effacer le binaire du programme et tous les fichiers objets du répertoire source en tapant `make clean`.

A.3 Compilation statique

Si vous avez compilé la PDFLib mais que vous ne pouvez pas l'installer en étant root, il est possible de compiler Carta-Genius lié statiquement à la PDFLib.

1. Compilez la PDFLib. Vous trouverez la version statique de la bibliothèque dans le sous-répertoire `libs/pdflib/.libs/`. Le fichier se nomme `libpdf.a`. Vous trouverez aussi le fichier d'en-tête `pdflib.h` dans le sous-répertoire `libs/pdflib/`.

2. Copiez `libpdf.a` dans le répertoire `lib` de Carta-Genius. Copiez `pdflib.h` dans le répertoire `include` de Carta-Genius.

3. Dans le répertoire de Carta-Genius, tapez `make static`.

Le fichier exécutable appelé `carta-genius` se trouve dans le répertoire `bin`.

Vous pouvez effacer le binaire du programme et tous les fichiers objets du répertoire source en tapant `make clean`.

A.4 Compilation sous Windows

Pour compiler et installer Carta-Genius sous Windows, vous devez utiliser l'environnement Cygwin, disponible sur <http://www.cygwin.com>.

Sous l'environnement Cygwin, vous pouvez compiler et installer la PDFLib comme sous Linux (voir le paragraphe "PDFLib").

Pour compiler Carta-Genius, tapez `make cygwin` dans le répertoire de Carta-Genius.

Annexe B. Installation de Carta Genius

B.1 Sous Linux

Une fois que vous avez la version binaire du programme (que vous l'avez compilé ou que vous l'avez téléchargé), vous pouvez soit l'exécuter depuis l'endroit où est placé le fichier, soit le copier vers un emplacement qui le rendra disponible pour l'ensemble du système, tel `/usr/local/bin`.

B.2 Sous Windows - compilation

Si vous avez compilé le programme vous-même (sous Cygwin), vous n'avez qu'à copier le binaire dans le répertoire `c:\windows` ou `c:\winnt` pour le rendre disponible à l'ensemble du système.

B.3 Sous Windows – installeur

Si vous avez téléchargé l'installeur pour Windows, exécutez-le tout simplement. Il placera les programmes automatiquement au bon endroit. Il vous fournira de plus l'interface graphique Carta-GUI, qui vous permettra d'éditer vos fichiers XML et de lancer simplement la génération des PDF sans avoir à passer par la ligne de commandes.

Annexe C. Interface graphique Carta-GUI

L'interface graphique Carta-GUI est fournie de base lors de l'installation Windows. Elle n'est actuellement pas fonctionnelle sur les autres environnements (Linux et MacOS X).

Cette interface est très simple, la signification des menus est évidente.

Le menu Fichier/Nouveau vous ouvre une fenêtre dans laquelle vous pouvez entrer toutes les informations générales concernant votre nouveau document.

Le menu Fichier/Ouvrir vous permet d'ouvrir un fichier XML existant.

Le menu Générer vous permet de lancer la création du fichier PDF à partir de votre document XML. Si la génération s'est déroulée sans problème, le fichier PDF s'ouvrira automatiquement. Si par contre une erreur a été rencontrée, un fichier texte s'ouvrira, contenant les logs d'erreur. Nous vous conseillons d'installer le logiciel Textpad (<http://www.textpad.com>) et d'y associer les fichiers ayant une extension ".log".

Les « éléments basiques » vous ouvrent des fenêtres de saisie qui vous permettent d'insérer simplement dans votre document XML les données nécessaires pour créer les éléments graphiques qui vont composer vos PDF.

Attention à deux choses :

- Lorsque vous créez un nouveau document XML, à partir du menu Fichier/Nouveau, le document que vous obtenez n'inclut pas de balise `<card>`. C'est à vous de les insérer à la main, avant d'ajouter dedans les éléments graphiques.
- Quand vous ajoutez un élément graphique en utilisant l'interface graphique, le code XML correspondant est ajouté à l'endroit où est placé le curseur dans la zone de texte. Veillez à bien la placer à l'endroit où vous souhaitez ajouter les éléments.

Annexe D. Options du programme à l'exécution

Vous trouverez un fichier binaire exécutable nommé 'carta-genius' dans le repertoire 'bin' de Carta Genius.

Pour l'utiliser (après installation, voir Annexe B), tapez :
`carta-genius -i file.xml -o result.pdf`

`file.xml` doit être un fichier XML au format Carta-Genius.
`result.pdf` sera le fichier PDF résultant.

Toutes les informations de configuration du fichier XML (taille du papier et de la margin, côté de retournement, ...) peuvent être remplacées par des options sur la ligne de commande.

Pour indiquer le type de papier (et donc la taille du papier) :

`-p TYPE-PAPIER` exemple: A4

Pour indiquer la taille du papier, si format inhabituel :

`-w LARGEUR-PAPIER` exemple: 450mm

`-h HAUTEUR-PAPIER` exemple: 32cm

Pour utiliser le papier en mode paysage (inversion de la hauteur et de la largeur du papier) :

`-l MODE-PAYSAGE` valeurs: yes, YES, no, NO

Pour indiquer la taille de la marge (identique pour la largeur et la hauteur) :

`-m TAILLE-MARGE` exemple: 10mm

Pour indiquer des tailles précises de marge :

`-x LARGEUR-MARGE` exemple: 1.2cm

`-y HAUTEUR-MARGE` exemple: 8mm

Pour indiquer le côté de retournement des pages :

`-b COTE-RETOURNEMENT` valeurs: width, WIDTH, height, HEIGHT

Pour indiquer la version PDF :

`-v VERSION-PDF` valeurs: 1.3, 1.4, 1.5

Pour inverser l'ordre des pages (dernière page en premier, première page en dernier) :

`-r INVERSION` valeurs: yes, YES, no, NO

Pour affecter des mots de passe PDF (le mot de passe utilisateur est nécessaire pour ouvrir le fichier, le mot de passe administrateur sert pour modifier les permissions et changer les mots de passe) :

`-a MDP-ADMINISTRATEUR`

`-u MDP-UTILISATEUR`

Pour indiquer le ou les répertoires où le logiciel doit chercher les fichiers à inclure :

`-d CHEMINS_INCLUSION` exemple: "/home/"

Annexe E. Formats de papier supportés

Vous trouverez ci-dessous les différents formats utilisables pour spécifier le type de papier ou de cartes à utiliser (paramètre type des éléments paper et cardsize), avec les tailles correspondantes. Ces noms peuvent être utilisés en majuscule ou en minuscule.

E.1 Formats de papiers standards (A)

4A0	1682 x 2378 mm
2A0	1189 x 1682 mm
A0	841 x 1189 mm
A1	594 x 841 mm
A2	420 x 594 mm
A3	297 x 420 mm
A4	210 x 297 mm
A5	148 x 210 mm
A6	105 x 148 mm
A7	74 x 105 mm
A8	52 x 74 mm
A9	37 x 52 mm
A10	26 x 37 mm

E.2 Formats de papiers standards agrandis (RA)

RA0	860 x 1120 mm
RA1	610 x 860 mm
RA2	430 x 610 mm
RA3	305 x 430 mm
RA4	215 x 305 mm

E.3 Formats de papiers standards agrandis supplémentaires (SRA)

SRA0	900 x 1280 mm
SRA1	640 x 900 mm
SRA2	450 x 640 mm
SRA3	320 x 450 mm
SRA4	225 x 320 mm

E.4 Formats de papiers standards aditionnels (B)

B0	1000 x 1414 mm
B1	707 x 1000 mm
B2	500 x 707 mm
B3	353 x 500 mm
B4	250 x 353 mm
B5	176 x 250 mm
B6	125 x 176 mm
B7	88 x 125 mm
B8	62 x 88 mm
B9	44 x 62 mm
B10	31 x 44 mm

E.5 Formats de papiers non standards

LETTER	215,9 x 279,4 mm	8,5 x 11 pouces
LEGAL	215,9 x 355,6 mm	8,5 x 14 pouces
SEMI-LETTER		5,5 x 8,5 pouces
EXECUTIVE	184,15 x 266,7 mm	7,25 x 10,5 pouces
TABLOID	279,4 x 431,8 mm	
LEDGER	431,8 x 279,4 mm	

E.6 Formats d'enveloppes standards habituels (C)

C0	917 x 1297 mm
C1	648 x 917 mm
C2	458 x 648 mm
C3	324 x 458 mm
C4	229 x 324 mm
C5	162 x 229 mm
C6	114 x 162 mm
C7	81 x 114 mm
C8	57 x 81 mm
C9	40 x 57 mm
C10	28 x 40 mm

E.7 Formats d'enveloppes standards supplémentaires (D)

D0	779 x 1090 mm
D1	545 x 779 mm
D2	385 x 545 mm
D3	272 x 385 mm
D4	192 x 272 mm
D5	136 x 192 mm
D6	96 x 136 mm
D7	68 x 96 mm
D8	48 x 68 mm
D9	34 x 48 mm
D10	24 x 34 mm

E.8 Formats d'enveloppes standards supplémentaires (E)

E0	1120 x 1600 mm
E1	800 x 1120 mm
E2	560 x 800 mm
E3	400 x 560 mm
E4	280 x 400 mm
E5	200 x 280 mm
E6	140 x 200 mm
E7	100 x 140 mm
E8	70 x 100 mm
E9	50 x 70 mm
E10	35 x 50 mm
E-A3	312 x 441 mm
E-A3/2	156 x 441 mm

E-A4	220 x 312 mm
E-A4/2	115 x 312 mm
E-A5	156 x 220 mm
E-A5/2	110 x 220 mm
E-A6	110 x 156 mm
E-A7	78 x 110 mm
E-B4	262 x 371 mm
E-B4/2	131 x 371 mm
E-B5	185 x 262 mm
E-B6	131 x 185 mm
E-B7	92 x 131 mm
E-B8	65 x 92 mm

E.9 Formats d'enveloppe non standard

DL	110 x 220 mm	
COM10	104,8 x 241,3 mm	4 1/8 x 9 1/2 pouces
MONARCH	98,4 x 190,5 mm	3 7/8 x 7 1/2 pouces

E.10 Formats ID (cartes d'identité, passeports, cartes de crédit, ...)

ID1	53,98 x 85,60 mm	3,370 x 2,125 pouces
ID2 (=A7)	74 x 105 mm	
ID3 (=B7)	88 x 125 mm	
BUSINESS-CARD	51 x 90 mm	3,5 x 2 pouces

Annexe F. Polices de caractères intégrées

Les fontes intégrées au format PDF sont réparties en 5 groupes, certaines avec des déclinaisons italique et/ou gras. Ces groupes sont :

Courier

Caractères à largeur fixe. Correspond sous Windows à la police « Courier New ».

Courier-Bold : en gras

Courier-Oblique : en italique

Courier-BoldOblique : en gras et italique

Helvetica

Caractères sans empattement. Correspond sous Windows à la police « Arial ».

Helvetica-Bold : en gras

Helvetica-Oblique : en italique

Helvetica-BoldOblique : en gras et italique

Times-Roman

Caractères avec empattement. Correspond sous Windows à la police « Times New Roman ».

Times-Bold : en gras

Times-Italic : en italique

Times-BoldItalic : en gras et italique

Symbol

Caractères grecs et latins, souvent utilisés pour les formules mathématiques.

ZapfDingbats

Caractères symboliques, utilisables pour des illustrations.

Annexe G. Variables et fonctions de l'évaluateur d'expressions

G.1 Variables contextuelles

\$PAGE, \$PAGE_DIM, \$PAGE_DIMENSION

type : élément

Contient les dimensions de la page courante.

\$CARD, \$CARD_DIM, \$CARD_DIMENSION

type : élément

Contient les dimensions de la carte courante.

\$MARGIN, \$MARGIN_DIM, \$MARGIN_DIMENSION

type : élément

Contient les dimensions des marges.

\$SPACE, \$SPACE_DIM, \$SPACE_DIMENSION

type : élément

Contient les dimensions des espaces entre les cartes.

\$CARD_INDEX

type : scalaire

Numéro de la carte courante dans le deck.

\$CARD_DECK, \$CARDS_IN_DECK, \$NBR_CARDS_IN_DECK

type : scalaire

Nombre de cartes dans le deck

\$DECK_INDEX

type : scalaire

Numéro du deck courant.

\$DECK_TOTAL

type : scalaire

Nombre total de decks.

G.2 Variables de formats de page

Toutes ces variables sont de type élément.

\$4A0, \$2A0, \$A0, \$A1, \$A2, \$A3, \$A4, \$A5, \$A6, \$A7, \$A8, \$A9, \$A10, \$RA0, \$RA1, \$RA2, \$RA3, \$RA4, \$SRA0, \$SRA1, \$SRA2, \$SRA3, \$SRA4, \$B0, \$B1, \$B2, \$B3, \$B4, \$B5, \$B6, \$B7, \$B8, \$B9, \$B10, \$C0, \$C1, \$C2, \$C3, \$C4, \$C5, \$C6, \$C7, \$C8, \$C9, \$C10, \$D0, \$D1, \$D2, \$D3, \$D4, \$D5, \$D6, \$D7, \$D8, \$D9, \$D10, \$LETTER, \$LEGAL, \$LEDGER, \$SEMI_LETTER, \$EXECUTIVE, \$TABLOID, \$DL, \$COM10, \$MONARCH, \$E0, \$E1, \$E2, \$E3, \$E4, \$E5, \$E6, \$E7, \$E8, \$E9, \$E10, \$E_A3, \$E_A3_2, \$E_A4, \$E_A4_2, \$E_A5, \$E_A5_2, \$E_A6, \$E_A7, \$E_B4, \$E_B4_2, \$E_B5, \$E_B6, \$E_B7, \$E_B8, \$ID1, \$ID2, \$ID3, \$BUSINESS_CARD

G.3 Variables de couleurs

Toutes ces variables sont de type couleur.

\$BLACK

Noir. #000000

\$WHITE

Blanc. #ffffff

\$GREY

Gris. #808080

\$RED

Rouge. #ff0000

\$GREEN

Vert. #00ff00

\$BLUE

Bleu. #0000ff

\$YELLOW

Jaune. #ffff00

\$TURQUOISE

Bleu turquoise. #00ffff

\$PURPLE

Violet. #ff00ff

\$DARK_RED

Rouge foncé. #800000

\$DARK_GREEN

Vert foncé. #008000

\$DARK_BLUE

Bleu foncé. #000080

\$DARK_YELLOW

Jaune foncé. #808000

\$DARK_TURQUOISE

Bleu turquoise foncé. #008080

\$DARK_PURPLE

Violet foncé. #800080

G.4 Variables additionnelles

\$PI

type : scalaire
3.14159265358979323846

\$TRUE
type : booléen
Valeur « vraie ».

\$FALSE
type: booléen
Valeur « fausse ».

G.5 Fonctions de transtypage

`var (expr)`

Retourne une variable dont le nom est passé en paramètre. N'importe quelle variable ou expression peut être passée en paramètre tant qu'il s'agit d'une chaîne de caractère, et qu'elle contient le nom d'une variable existante.

`string (expr)`
Convertit en chaîne de caractères une variable ou une expression.

`scalar (expr)`
Convertit en scalaire une variable ou une expression.

`value (expr)`
Convertit en valeur (scalaire avec une unité) une variable ou une expression.

`element (expr)`
Convertit en élément (couple de 2 valeurs) une variable ou une expression.

`bool (expr)`
Convertit en booléen une variable ou une expression.

G.6 Fonctions de comparaison

`eq (expr, expr)`

Retourne \$TRUE si les deux expressions ou variables passées en paramètre sont égales. Les deux expressions doivent être du même type.

`lt (expr)`

Retourne \$TRUE si la première expression est inférieure à la seconde. Les deux expressions doivent être du même type.

`gt (expr)`

Retourne \$TRUE si la première expression est supérieure à la seconde. Les deux expressions doivent être du même type.

G.7 Fonctions d'opérations booléennes

`not (bool)`

Retourne la valeur booléenne inverse à celle passée en paramètre.

`and (bool, bool)`

Retourne \$TRUE si les deux paramètres valent \$TRUE.

`or (bool, bool)`

Retourne \$TRUE si l'un des deux paramètres vaut \$TRUE.

G.8 Fonctions de calcul

`plus (expr, expr)`

Additionne une expression à une autre. Equivalent à '+'.
Si le premier paramètre est une chaîne, retourne cette chaîne à laquelle est concaténée le second paramètre, transtypé en chaîne. Si le premier paramètre est un scalaire ou une valeur, le second paramètre doit être indifféremment un scalaire ou une valeur. Le type renvoyé est le même que celui du premier paramètre. Si le premier paramètre est un élément, le second paramètre ne doit pas être une chaîne. Le type renvoyé est un élément.

`minus (expr, expr)`

Soustrait une expression à une autre. Equivalent à '-'.
Les types renvoyés sont les mêmes que pour la fonction `plus ()`.

`mult (expr, expr)`

Multiplie une expression par une autre. Equivalent à '*'.
Aucun des deux paramètres ne peut être une chaîne. Si le premier paramètre est un scalaire ou une valeur, le second paramètre doit être indifféremment un scalaire ou une valeur. Le type renvoyé est le même que celui du premier paramètre.

`div (expr, expr)`

Divise une expression par une autre. Equivalent à '/'.
Les types renvoyés sont les mêmes que pour la fonction `mult ()`.

`mod (expr, expr)`

Calcule de reste de la division entière d'une expression par une autre. Equivalent à '%'.
Les types renvoyés sont les mêmes que pour la fonction `mult ()`.

`sqrt (expr)`

Calcule la racine carrée d'une expression.
Le paramètre ne doit pas être une chaîne ni une couleur, ni être négatif. Le type retourné est le même que celui reçu en paramètre.

`pow (expr, expr)`

Elève une expression par une puissance donnée en paramètre.
Les paramètre ne doivent pas être des chaînes ni des couleurs.

`sin (scalaire)`

Calcule le sinus d'un scalaire (angle).
Retourne un scalaire.

`cos (scalaire)`

Calcule le cosinus d'un scalaire (angle).
Retourne un scalaire.

`tan (scalaire)`

Calcule la tangente d'un scalaire (angle).

Retourne un scalaire.

`asin(scalaire)`

Calcule l'arc-sinus d'un scalaire (angle).
Retourne un scalaire.

`acos(scalaire)`

Calcule l'arc-cosinus d'un scalaire (angle).
Retourne un scalaire.

`atan(scalaire)`

Calcule l'arc-tangente d'un scalaire (angle).
Retourne un scalaire.

G.9 Fonctions sur les éléments

`width(element)`

Retourne la largeur d'un élément.
Le type renvoyé est une valeur.

`height(element)`

Retourne la hauteur d'un élément.
Le type renvoyé est une valeur.

`landscape(element)`

Inverse la hauteur et la largeur d'un élément.
Le type renvoyé est un élément.

G.10 Fonctions sur les chaînes

`strlen(chaine)`

Retourne le nombre de caractères d'une chaîne.
Le type retourné est un scalaire.

`strwidth(chaine, valeur, chaine)`

Retourne la longueur d'une chaîne en fonction de la police utilisée et de la taille de caractère.
Premier paramètre : le nom de la police à employer.
Deuxième paramètre : la taille de caractère à utiliser.
Troisième paramètre : le texte lui-même.
Le type retourné est une valeur.

G.11 Fonctions de test

`min(expr, expr)`

Retourne une variable égale à la plus faible expression passée en paramètre.

`max(expr, expr)`

Retourne une variable égale à la plus grande expression passée en paramètre.

G.12 Fonctions de date et heure

`time(scalaire)`

Retourne un timestamp Unix (nombre de secondes écoulées depuis le 1er janvier 1970).

Le paramètre est un décalage horaire (nombre d'heures positif ou négatif). Mettre zéro pour obtenir le temps GMT.

```
strftime(scalaire, chaine)
```

Formate une chaîne de caractères correspondant au timestamp Unix passé en premier paramètre. Le deuxième paramètre est une chaîne de caractères qui définit la chaîne produite ; elle peut être constituée de :

- %a : Le jour de la semaine abrégé.
- %A : Le jour de la semaine.
- %b : Le nom du mois abrégé.
- %B : Le nom du mois.
- %C : Le siècle (année / 100) sur 2 chiffres.
- %d : Le jour du mois (01 à 31).
- %D : Equivalent à "%m/%d/%y".
- %e : Comme %d, sauf que le zéro du début est remplacé par un espace.
- %F : Equivalent à "%Y-%m-%d" (le format de date ISO 8601).
- %G : Le numéro de semaine dans l'année (comme %V) sauf si on déborde sur une autre année, auquel cas le numéro de l'année est utilisé.
- %g : Comme %G, mais sur 2 chiffres.
- %h : Equivalent à %b.
- %H : L'heure (00 à 23).
- %I : L'heure (01 à 12).
- %j : Le jour de l'année (001 à 366).
- %k : Comme %H, sauf que le zéro du début est remplacé par un espace.
- %l : Comme %I, sauf que le zéro du début est remplacé par un espace.
- %m : Le mois (01 à 12).
- %M : La minute (00 à 59).
- %n : Un retour à la ligne.
- %p : `AM` ou `PM`.
- %P : `am` ou `pm`.
- %r : L'heure dans la notation AM/PM. Equivalent à "%I:%M:%S %p".
- %R : L'heure complète (%H:%M).
- %s : Le timestamp Unix.
- %S : Les secondes (00 à 61).
- %t : Une tabulation.
- %T : L'heure complète avec les secondes (%H:%M:%S).
- %u : Le jour de la semaine (1 à 7).
- %U : Le numéro de la semaine (00 à 53), avec le premier Dimanche comme premier jour de la semaine 01.
- %V : Le numéro de la semaine ISO 8601:1988 (01 à 53), avec la semaine 01 étant la première semaine avec au moins 4 jours dans l'année courante, et Lundi comme premier jour de la semaine.
- %w : Le jour de la semaine (0 à 6).
- %W : Le numéro de la semaine (00 à 53), avec le premier Lundi comme premier jour de la semaine 01.
- %y : L'année sur 2 chiffres (00 à 99).
- %Y : L'année sur 4 chiffres.
- %z : Le décalage horaire par rapport au temps universel.
- %% : Un caractère '%'.

G.13 Fonctions d'aléa

`srand(scalaire)`

Initialise le générateur de nombres aléatoires.

Le paramètre est utilisé comme base pour l'initialisation. Conseil : vous pouvez utiliser le résultat de la fonction `time()` pour obtenir un résultat toujours différent.

Retourne un nombre aléatoire compris entre 0 et 1.

`rand(scalaire)`

Retourne un nombre aléatoire compris entre 0 et le nombre donné en paramètre.

Annexe H. Convertir des polices TrueType en polices Postscript Type 1

H.1 Conversion locale

1. Récupérez et installez le programme `ttf2pt1` (<http://ttf2pt1.sourceforge.net>).
2. Tapez `'ttf2pt1 -e font.ttf'`, où `font.ttf` est le nom de votre police TrueType.
3. Vous récupérez un fichier `font.pfa` et un fichier `font.afm`.
4. Ouvrez le fichier `font.afm`, et supprimez la ligne commençant par 'Character'.
5. C'est fini.

H.2 Conversion en ligne

Allez sur la page <http://www.pandocreon.com/logiciels/carta-genius/online/> et vous trouverez une interface Web vous permettant de convertir des polices TrueType en polices Postscript Type 1. Envoyez simplement votre fichier TTF, et vous pourrez télécharger directement un fichier ZIP contenant les deux fichiers correspondant (fichier PFA et fichier AFM).

Annexe I. Générer des images bitmap

I.1 En local

Dans le répertoire 'bin' de Carta Genius, il y a un script shell appelé "pdf_to_bitmap". Il sert à convertir un fichier PDF en plusieurs images bitmap (une image par page).

Exemples d'utilisation :

- Vous avez besoin d'une image JPEG de vos cartes de jeu ? Créez un document Carta Genius dont la marge est explicitement mise à 0 (zéro), avec la taille du papier égale à celle des cartes. Créez le fichier PDF. Vous devriez avoir un fichier PDF avec une page par carte. Maintenant, convertissez le PDF pour avoir autant de fichiers JPEG que de cartes dans le PDF.

- Vous voulez insérer un carte générée avec Carta Genius dans un autre document Carta Genius ? Exportez la carte en JPEG (comme expliqué ci-dessus), et utilisez-la dans votre nouveau document.

Pour utiliser pdf_to_bitmap, vous devez avoir, installé sur votre ordinateur, le logiciel ghostscript (téléchargeable sur <http://www.ghostscript.com>) version GNU ou APFL. Si votre document PDF contient de la transparence, vous devrez utiliser la version 8.14 ou supérieure.

Comment utiliser pdf_to_bitmap

1. Vous devez décider la résolution en "points par pouces" des images qui seront créées. Cela déterminera la taille de l'image en pixels. Par exemple, si vos pages PDF font 2 pouces x 2 pouces, et que vous spécifiez une résolution de 300 points par pouces (aussi appelé "dpi", pour "dot per inch" en anglais), vous obtiendrez une image de 600x600 pixels.
2. Il y a plusieurs options :
 - Le type d'image de sortie. Par défaut il s'agit du format JPEG, mais vous pouvez choisir le BMP (bmp16m), le PBM (pbm), le PNG (png16m) ou le TIFF (tiff24nc).
 - Vous pouvez choisir la première et la dernière page à traiter dans le PDF.
 - Il est possible de demander un anti-aliasing sur les textes et/ou les graphiques pour obtenir un meilleur résultat.
3. Tapez "./pdf_to_bitmap -h" pour avoir les options exactes de la ligne de commande.

I.2 En ligne

Allez sur la page <http://www.pandocreon.com/logiciels/carta-genius/online/> et vous trouverez une interface Web vous permettant de convertir vos fichiers PDF en images bitmap. Envoyez simplement votre fichier PDF (en modifiant au besoin les options de rendu), et vous pourrez télécharger directement un fichier ZIP contenant une image pour chaque page du PDF.

Annexe J. Table iso-8859-1

	 	no-break space
¡	¡	inverted exclamation mark
¢	¢	cent sign
£	£	pound sterling sign
¤	¤	general currency sign
¥	¥	yen sign
¦	¦	broken (vertical) bar
§	§	section sign
¨	¨	umlaut (dieresis)
©	©	copyright sign
^a	ª	ordinal indicator, feminine
«	«	angle quotation mark, left
¬	¬	not sign
	­	soft hyphen
®	®	registered sign
¯	¯	macron
°	°	degree sign
±	±	plus-or-minus sign
²	²	superscript two
³	³	superscript three
´	´	acute accent
µ	µ	micro sign
¶	¶	pilcrow (paragraph sign)
·	·	middle dot
¸	¸	cedilla
¹	¹	superscript one
º	º	ordinal indicator, masculine
»	»	angle quotation mark, right
¼	¼	fraction one-quarter
½	½	fraction one-half
¾	¾	fraction three-quarters
¿	¿	inverted question mark
À	À	capital A, grave accent
Á	Á	capital A, acute accent
Â	Â	capital A, circumflex accent
Ã	Ã	capital A, tilde

Ä	Ä	capital A, dieresis or umlaut mark
Å	Å	capital A, ring
Æ	Æ	capital AE diphthong (ligature)
Ç	Ç	capital C, cedilla
È	È	capital E, grave accent
É	É	capital E, acute accent
Ê	Ê	capital E, circumflex accent
Ë	Ë	capital E, dieresis or umlaut mark
Ì	Ì	capital I, grave accent
Í	Í	capital I, acute accent
Î	Î	capital I, circumflex accent
Ï	Ï	capital I, dieresis or umlaut mark
Ð	Ð	capital Eth, Icelandic
Ñ	Ñ	capital N, tilde
Ò	Ò	capital O, grave accent
Ó	Ó	capital O, acute accent
Ô	Ô	capital O, circumflex accent
Õ	Õ	capital O, tilde
Ö	Ö	capital O, dieresis or umlaut mark
×	×	multiply sign
Ø	Ø	capital O, slash
Ù	Ù	capital U, grave accent
Ú	Ú	capital U, acute accent
Û	Û	capital U, circumflex accent
Ü	Ü	capital U, dieresis or umlaut mark
Ý	Ý	capital Y, acute accent
Þ	Þ	capital THORN, Icelandic
ß	ß	small sharp s, German (sz ligature)
à	à	small a, grave accent
á	á	small a, acute accent
â	â	small a, circumflex accent
ã	ã	small a, tilde
ä	ä	small a, dieresis or umlaut mark
å	å	small a, ring
æ	æ	small ae diphthong (ligature)
ç	ç	small c, cedilla
è	è	small e, grave accent
é	é	small e, acute accent
ê	ê	small e, circumflex accent

ë	ë	small e, dieresis or umlaut mark
ì	ì	small i, grave accent
í	í	small i, acute accent
î	î	small i, circumflex accent
ï	ï	small i, dieresis or umlaut mark
ð	ð	small eth, Icelandic
ñ	ñ	small n, tilde
ò	ò	small o, grave accent
ó	ó	small o, acute accent
ô	ô	small o, circumflex accent
õ	õ	small o, tilde
ö	ö	small o, dieresis or umlaut mark
÷	÷	divide sign
ø	ø	small o, slash
ù	ù	small u, grave accent
ú	ú	small u, acute accent
û	û	small u, circumflex accent
ü	ü	small u, dieresis or umlaut mark
ý	ý	small y, acute accent
þ	þ	small thorn, Icelandic
ÿ	ÿ	small y, dieresis or umlaut mark

Annexe K. Versions de Carta Genius

v 5.2.0 - 20 juin 2006

- Ajout de la section de déclaration des variables.
- Ajout du « card count ».
- Nouvelle fonction : strftime().
- Débuggage concernant le décalage horaire.
- Débuggage de la déclaration de masques d'images.
- Débuggage de la priorité des opérateurs dans l'évaluateur d'expressions.

v 5.0.0 - 16 octobre 2005

- Ajout de la balise <template>.
- Ajout de la balise <log>.
- Ajout de la balise <if>.
- Ajout de la balise <while>.
- Nouveau type de données ajouté : booléen.
- Nouvelles fonctions de comparaison : eq(), lt() et gt().
- Nouvelles fonctions logiques : not(), and() et or().

v 4.0.0 - 30 septembre 2005

- Ajout d'une véritable interface graphique basée sur le XUL.
- Support des fichiers de log.
- Support de l'exécution silencieuse.

v 3.12.1 - 13 décembre 2004

- Débuggage de la largeur des rayures horizontales sur les textes.
- Débuggage sur la recherche des répertoires de fontes.

v 3.12.0 - 07 décembre 2004

- Ajout du support des rayures horizontale sur les textes.
- Débuggage des textes avec boîte avec rotation.

v 3.10.0 - 15 octobre 2004

- Ajout d'une interface graphique (CartaGUI) pour Windows et Linux.
- Programme d'installation sous Windows.

v 3.8.0 - 24 août 2004

- Correction d'un bug affectant les dos de carte multi-gabarits.
- Ajout de nouvelles fonctions : min(), max(), time(), srand() et rand().

v 3.6.0 - 01 juillet 2004

- Correction d'un bug affectant les fond-perdus sur les versos.
- Ajout de nouvelles fonctions : sqrt() et pow().

v 3.4.0 - 21 juin 2004

- Support des grilles.
- Support des hexagones.
- Support de la déclaration de chemins de recherche des images.
- Support de la déclaration de chemins de recherche des polices de caractères.

v 3.2.0 - 08 juin 2004

Support de masques de transparence d'images réutilisables.
Support de méta-données PDF (auteur, titre, ...).
Support de lignes pointillées dans tous les objets graphiques.
Ajout du numéro de ligne dans les messages d'erreur.

v 3.0.0 - 16 mai 2004

Support d'objets texte traversant plusieurs zones (même sur différentes cartes).
Conversions plus précises des valeurs.
Ajout des fonctions asin(), acos() et atan().
Support du paramètre radius dans les 'box' et les 'text'.
Modification du 'round-box' comme synonyme de 'box' (pour compatibilité).
Ajouts dans l'Aide-Mémoire.

v 2.4.0 - 07 mai 2004

Support de la déclaration des chemins d'inclusion dans le XML et la ligne de commande.
Noms de variables alternatifs (plus explicites, moins concis).
Correction de bogue dans l'évaluateur d'expressions.
Ajout des variables et des fonctions dans l'Aide-Mémoire.

v 2.2.1 - 29 avril 2004

Ajout d'un Aide-Mémoire en français et en anglais.

v 2.2.0 - 21 avril 2004

Support des couleurs dans l'évaluateur d'expressions et de variables.
Support de la déformation des images pour couvrir les zones définies.
Ajout des courbes de Bézier.
Ajout des polygones constitués de lignes et/ou de courbes.
Ajout de lignes sur/sous/à travers les textes.
Support des variables de variables.
Ajout de l'accès aux variables internes lors de la création d'étuis de cartes.
Modification de la couleur des traits de coupe dans les fond-perdus sombres.
Support de l'espacement entre les caractères.
Support de l'échelle horizontale du texte.

v 2.0.0 - 14 avril 2004

Correction d'un bogue dans le parseur XML (fuite de mémoire).
Support de l'écriture (valeur de caractères latin1) dans les textes.
Evalueur d'expressions et de variables.
Support des bordures dans les boîtes de texte.
Support de l'inclusion de fichiers XML.
Guide utilisateur complet en français et anglais.
Fichier XML externe pour la création d'étuis de cartes.

v 1.14.0 - 24 mars 2004

Support des déclarations directes des images.
Support de plusieurs méthodes de mélange des couleurs.

v 1.12.0 - 23 mars 2004

Correction d'un bug lorsque les cartes sont plus grandes que la taille du papier.
Correction d'un bug concernant les documents vides.

- v 1.11.0 - 19 mars 2004
 - Support de l'espacement entre les cartes.
 - Ajout d'un script de conversion de PDF en images bitmap.
- v 1.10.2 - 03 mars 2004
 - Support d'un plus grand nombre de types de papiers.
 - Support d'un plus grand nombre d'options sur la ligne de commande.
- v 1.10.1 - 29 février 2004
 - Modification des tailles des traits de coupe et du fond-perdu.
- v 1.10.0 - 10 décembre 2003
 - Support de l'opacité des objets graphiques.
- v 1.8.0 - 07 décembre 2003
 - Support de
, <p/> et dans les textes.
 - Correction d'un bug dans l'adaptation de la taille des textes.
- v 1.6.0 - 05 décembre 2003
 - Support des fonds-perdus.
- v 1.4.0 - 17 octobre 2003
 - Support de l'inversion de l'ordre des pages.
- V 1.2.0 - 29 septembre 2003
 - Support de marges différentes entre la largeur et la hauteur des pages.
 - Support de la définition de l'interlignage des textes.
- v 1.0.0 - 27 août 2003
 - Support des gabarits multiples.
- v 0.9.0 - 21 août 2003
 - Support du numéro de version PDF explicite.
 - Support de la protection PDF par mot de passe.
 - Script de configuration.
 - Courte documentation et exemple en français et anglais.
- v 0.7.0 - 20 août 2003
 - Support de la transparence explicite des images par masque.
- v 0.5.0 - 19 août 2003
 - Support du verso des cartes.
 - Support des gabarits de gabarits.
- v 0.3.0 - 18 août 2003
 - Support des gabarits.
- v 0.1.0 - 17 août 2003
 - Support des cercles, rectangles, rectangles arrondis, lignes, textes et images.

v 0.0.1 - 16 août 2003

Import de fichiers XML, création de PDF vides avec des traits de coupe.

Projet démarré le 16 août 2003

Annexe L. Versions de ce document

5.2.0 – 20 juin 2006

Ajout des fonctionnalités de Carta Genius 5.2.0.
Passage au format OpenDocument.

5.0.0 – 16 octobre 2005

Ajout des fonctionnalités de Carta Genius 5.0.0.

4.0.0 – 30 septembre 2005

Ajout des fonctionnalités de Carta Genius 4.0.0 et de la nouvelle interface Carta-GUI.

3.12.1 – 13 décembre 2004

Synchronisation avec Carta Genius 3.12.1.

3.12.0 – 07 décembre 2004

Ajout des fonctionnalités de Carta Genius 3.12.0.
Corrections de fautes.

3.10.0 – 15 octobre 2004

Ajout de l'interface graphique CartaGUI et de l'installeur Windows.

3.8.0 – 24 août 2004

Ajout des fonctionnalités de Carta Genius 3.8.0.

3.6.0 – 01 juillet 2004

Ajout des fonctionnalités de Carta Genius 3.6.0.
Correction de fautes relevées par Vicnet.

3.4.0 – 21 juin 2004

Ajout des fonctionnalités de Carta Genius 3.4.0.

3.2.0 – 08 juin 2004

Ajout des fonctionnalités de Carta Genius 3.2.0.

3.0.0 – 16 mai 2004

Ajout des nouvelles fonctionnalités de Carta Genius 3.0.0.

2.4.0 – 07 mai 2004

Ajout des nouvelles fonctionnalités de Carta Genius 2.4.0.
Correction d'erreurs minimes.

2.2.0 – 21 avril 2004

Ajout des nouvelles fonctionnalités de Carta Genius 2.2.0.
Corrections de fautes relevées par Vicnet.

2.0.0 – 14 avril 2004

Documentation complète correspondant à la version 2.0.0 de Carta Genius.

Document commencé le 23 mars 2004.

Annexe M. Licence de ce document : GNU Free Documentation license

Version 1.2, November 2002

Copyright (c) 2000, 2001, 2002 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this licence document, but changing it is not allowed.

K.1 Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

K.2 Applicability and definitions

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-

Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

K.3 Verbatim copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

K.4 Copying in quantity

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover

Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

K.5 Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- **A.** Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- **B.** List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- **C.** State on the Title page the name of the publisher of the Modified Version, as the publisher.
- **D.** Preserve all the copyright notices of the Document.
- **E.** Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- **F.** Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- **G.** Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- **H.** Include an unaltered copy of this License.
- **I.** Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

- **J.** Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- **K.** For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- **L.** Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- **M.** Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- **N.** Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- **O.** Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

K.6 Combining documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections

Entitled "Endorsements."

K.7 Collections of documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

K.8 Aggregation with independent work

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

K.9 Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

K.10 Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

K.11 Future revisions of this license

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See

<http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

Annexe N. Licence de Carta Genius : GNU General Public license

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

L.1 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public

License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

L.2 Terms and conditions for copying, distribution and modification

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- **a)** The modified work must itself be a software library.
- **b)** You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- **c)** You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- **d)** If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this

License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- **a)** Accompany the work with the complete corresponding machine-readable source code for

the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

- **b)** Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
- **c)** Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- **d)** If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- **e)** Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- **a)** Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- **b)** Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so,

and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Index

2A0	87	
4A0	87	
A0	87	
A1	87	
A10	87	
A2	87	
A3	87	
A4	87	
A5	87	
A6	87	
A7	87	
A8	87	
A9	87	
acos	95	
adapt	27	
Adaptation de la taille du texte		27
align	23	
and	93	
and()	71	
asin	95	
atan	95	
author	10	
B0	87	
B1	87	
B10	87	
B2	87	
B3	87	
B4	87	
B5	87	
B6	87	
B7	87	
B8	87	
B9	87	
back	52	
bitmap		39
blendmode		12, 14, 20 sv, 38, 56
block	36	
bool	93	
booléens		59
border	25, 38	
box	12 sv	
BUSINESS-CARD		89
C0	88	
C1	88	
C10	88	
C2	88	
C3	88	
C4	88	
C5	88	
C6	88	
C7	88	
C8	88	
C9	88	
Cadres		24
cardcase		78
cardcase_init		79
cardsize		47
Carta-GUI		85
case_depth		78
case_flap_big		79
case_flap_small		79
case_gap_big		79
case_gap_small		79
case_height		78
case_width		78
cell-height		16
cell-width		16
center	23	
Cercle	14	
chaînes de caractères		59
char-space		33, 36
circle	14	
color	14, 20, 23, 36	
Color	56	
ColorBurn		56
ColorDodge		56
COM10		89
Compilation		82
Compilation dynamique		82
Compilation sous Windows		83
Compilation statique		82
copyright		10
cos	94	
couleurs		59
count	48	
Courier		29, 90
Courier-Bold		29, 90
Courier-BoldOblique		29, 90
Courier-Oblique		29, 90
cp-x1	20 sv	
cp-x2	20 sv	
cp-y1	20 sv	
cp-y2	20 sv	
D0	88	
D1	88	
D10	88	
D2	88	
D3	88	

D4	88		fond-perdu	46
D5	88		font	23, 29, 36
D6	88		Free Documentation license	108
D7	88		gabarits	49
D8	88		General Public license	114
D9	88		ghostscript	99
Darken		56	GIF	39
Déclarations		9	GNU	108, 114
default		32	grid	16
Difference		56	gt	93
dir	76		gt()	71
div	94		h-scale	34, 36
DL	89		HardLight	56
E-A3	88		height	12, 23, 38, 45, 48, 95
E-A3/2		88	Helvetica	29, 90
E-A4	89		Helvetica-Bold	29, 90
E-A4/2		89	Helvetica-BoldOblique	29, 90
E-A5	89		Helvetica-Oblique	29, 90
E-A5/2		89	hidden-ditch	46
E-A6	89		Hue	56
E-A7	89		id	29, 36, 44, 49
E-B4	89		ID1	89
E-B4/2		89	ID2	89
E-B5	89		ID3	89
E-B6	89		image	39
E-B7	89		images	41
E-B8	89		include	75
E0	88		includes	75
E1	88		info	10
E10	88		Installation	7, 84
E2	88		installeur	84
E3	88		Interlignage	24
E4	88		iso-8859-1	28, 100
E5	88		JPEG	39
E6	88		justify	23
E7	88		keywords	10
E8	88		ladder-bottom	31
E9	88		ladder-color	31
element		93	ladder-rise	31
éléments		59	landscape	45, 47, 95
entire	40		langue	10
eq	93		latin1	28
eq()	71		LEDGER	88
espacement		47	left	23
even	46		LEGAL	88
Exclusion		56	LETTER	88
EXECUTIVE		88	Lighten	56
expressions		58	Ligne	14
file	44		line	15
fill-color		12, 14, 21, 25, 38	line-color	12, 14, 21, 25
fonctions		65	line-color	38

line-width	12, 14, 21, 25, 38	right	23
lt	93	rotation	12, 21, 38
lt()	71	Rotation	17, 26
Luminosity	56	Saturation	56
margin	45	scalaires	59
margin-height	45	scalar	93
margin-width	45	Screen	56
mask	43 sv	SEMI-LETTER	88
mask-id	44	sin	94
max	95	size	23, 36
metrics	29	space	24, 36, 48
min	95	space-height	48
minus	94	space-width	48
mod	94	sqrt	94
MONARCH	89	SRA0	87
mult	94	SRA1	87
Multiply	56	SRA2	87
name	61	SRA3	87
nbr-height	16	SRA4	87
nbr-width	16	srand	97
not	93	strftime	96
not()	71	strikeout	30, 36
note	10	string	93
odd	46	strlen	95
opacité	55	strwidth	95
opacity	12, 14, 20 sv, 38, 55	subject	10
Options	86	Symbol	29, 90
or	94	TABLOID	88
or()	71	tan	94
outline	29	template	49
Overlay	56	text-define	36
overline	30, 36	texte	22
path	76	time	95
paysage	45	Times-Bold	29, 90
pdf_to_bitmap	99	Times-BoldItalic	29, 90
plus	94	Times-Italic	29, 90
PNG	39	Times-Roman	29, 90
portrait	45	title	10
Postscript Type 1	98	transparence	42
pow	94	TrueType	98
precision	64, 74	ttf2pt1	98
Préface	6	type	45, 47
RA0	87	Types de données	11
RA1	87	underline	30, 36
RA2	87	valeurs	59
RA3	87	value	64, 93
RA4	87	var	61, 67, 93
radius	13 sv, 25, 38	variables	59
rand	97	Variables contextuelles	91
Rectangle	12	Variables de formats de page	91
Rectangle arrondi	13	version	10

Versions	103, 107	\$C3	91
width	12, 14, 20, 23, 38, 45, 47, 95	\$C4	91
x	12, 14, 16, 21, 23, 38	\$C5	91
x1	14, 20	\$C6	91
x2	14, 20	\$C7	91
y	12, 14, 16, 21, 23, 38	\$C8	91
y1	14, 20	\$C9	91
y2	14, 20	\$CARD	91
ZapfDingbats	29, 90	\$CARD_DECK	91
&	28	\$CARD_DIM	91
>	28	\$CARD_DIMENSION	91
<	28	\$CARD_INDEX	91
&XXX;	28	\$CARD_CASE_A_H	80
 	28	\$CARD_CASE_A_W	80
<if>	69	\$CARD_CASE_A_X	80
<log>	74	\$CARD_CASE_A_Y	80
<p/>	28	\$CARDS_IN_DECK	91
<template>	53	\$COM10	91
<variables>	63	\$D0	91
<while>	72	\$D1	91
\$2A0	91	\$D10	91
\$4A0	91	\$D2	91
\$A0	91	\$D3	91
\$A1	91	\$D4	91
\$A10	91	\$D5	91
\$A2	91	\$D6	91
\$A3	91	\$D7	91
\$A4	91	\$D8	91
\$A5	91	\$D9	91
\$A6	91	\$DARK_BLUE	92
\$A7	91	\$DARK_GREEN	92
\$A8	91	\$DARK_PURPLE	92
\$A9	91	\$DARK_RED	92
\$B0	91	\$DARK_TURQUOISE	92
\$B1	91	\$DARK_YELLOW	92
\$B10	91	\$DECK_INDEX	91
\$B2	91	\$DECK_TOTAL	91
\$B3	91	\$DL	91
\$B4	91	\$E_A3	91
\$B5	91	\$E_A3_2	91
\$B6	91	\$E_A4	91
\$B7	91	\$E_A4_2	91
\$B8	91	\$E_A5	91
\$B9	91	\$E_A5_2	91
\$BLACK	92	\$E_A6	91
\$BLUE	92	\$E_A7	91
\$BUSINESS_CARD	91	\$E_B4	91
\$C0	91	\$E_B4_2	91
\$C1	91	\$E_B5	91
\$C10	91	\$E_B6	91
\$C2	91	\$E_B7	91

\$E_B8		91		\$NBR_CARDS_IN_DECK		91
\$E0	91			\$PAGE	91	
\$E1	91			\$PAGE_DIM	91	
\$E10	91			\$PAGE_DIMENSION		91
\$E2	91			\$PI	92	
\$E3	91			\$PURPLE		92
\$E4	91			\$RA0	91	
\$E5	91			\$RA1	91	
\$E6	91			\$RA2	91	
\$E7	91			\$RA3	91	
\$E8	91			\$RA4	91	
\$E9	91			\$RED	92	
\$EXECUTIVE			91	\$SEMI_LETTER		91
\$FALSE		93		\$SPACE	91	
\$GREEN		92		\$SPACE_DIM		91
\$GREY		92		\$SPACE_DIMENSION		91
\$ID1	91			\$SRA0	91	
\$ID2	91			\$SRA1	91	
\$ID3	91			\$SRA2	91	
\$LEDGER		91		\$SRA3	91	
\$LEGAL		91		\$SRA4	91	
\$LETTER		91		\$TABLOID	91	
\$MARGIN		91		\$TRUE	93	
\$MARGIN_DIM			91	\$TURQUOISE		92
\$MARGIN_DIMENSION			91	\$WHITE	92	
\$MONARCH		91		\$YELLOW	92	