



PRÉSENTATION DE PMVR TM (PATENTED MAPPABLE VR)

4.5A MANUAL

Ce document, dont l'essentiel a été traduit de l'anglais par Caroline Jouneau-Sion, peut être consulté dans son intégralité en anglais à l'adresse suivante :

<http://www.duckware.com/pmvr/>

1. Qu'est-ce que PMVR ?

PMVR est un programme (applet) Java TM qui fait défiler des images panoramiques dans une page web.

PMVR est une alternative au QTVR d'Apple qui exige l'installation préalable d'un important plug-in. QTVR convient bien si les utilisateurs ont déjà installé le plug-in. Dans le cas contraire, il est pénible d'attendre que le plug-in soit installé. Beaucoup d'utilisateurs n'attendront pas et passeront leur chemin... sans avoir visité votre site !

L'avantage de PMVR est d'être un très petit programme java (33K, il se charge en quelques secondes) qui fonctionne avec tous les navigateurs capables d'exécuter les programmes Java 1.1, quel que soit le système d'exploitation. Néanmoins, il est conseillé d'utiliser des navigateurs récents, pour des questions de sécurité et de rapidité d'exécution.

[Aucune expérience de programmation n'est nécessaire pour utiliser PMVR.](#) Tout ce dont vous avez besoin est de savoir comment éditer un document HTML. Lisez bien le guide de démarrage rapide pour comprendre ce dont il s'agit.

L'applet optionnel 'FloorPlan' permet de joindre un plan correspondant à votre vue panoramique, indiquant la direction dans laquelle on regarde et permettant une meilleure localisation.

Vous pouvez visiter <http://www.idyll-by-the-sea.com/> pour voir un site web qui utilise pmvr.class et floorplan.class (les programmes java). Les pages [photos](#) décrivent la façon dont les panoramiques ont été créés.

2. Les avantages, en résumé

Des panoramas associés au support plan/carte .

Le moyen le plus facile d'ajouter des panoramas à des pages web ou à des CD-- [démarrage rapide](#) [§4] Pas besoin d'installer des plug-in au préalable !

Complètement contrôlable à l'aide du langage Native or JavaScript.

Possibilité de zoomer dans l'image en utilisant les touches 'a'/'z' ou la barre de zoom.

L'applet Java est une plateforme commune acceptée par tous les navigateurs (à condition que la fonction java soit activée !)

Un applet Java très petit (33k) qui se télécharge automatiquement en quelques secondes.

Des zones sensibles flexibles en forme, en taille... x-axis section, cercle, rectangle, and polygone -- [link-spec](#) [§6]

Les zones sensibles peuvent lancer de nouvelles pages web ou appeler vos "Scripting fonctions" pour des actions programmées ! -- [url-target](#) [§6]

3. Exemple PMVR

Un exemple de PMVR en action. Pour faire défiler le panorama à droite ou à gauche, cliquez à l'intérieur de l'image et faites glisser du côté choisi. Cliquer-glisser sur la barre en bas du panorama (elle apparaît quand votre curseur entre dans le panorama) pour zoomer en avant ou en arrière (ou presser 'a' et 'z'). Remarquez que le plan vous indique dans quelle direction vous regardez et reste synchronisé avec le panorama, ce qui donne une idée précise de la direction.

Voir aussi [Idyll-By-The-Sea](#) .



Le code HTML qui a créé cette image PMVR et le plan ressemble à cela :

```
<applet code="pmvr.class" width=500 height=252>
  <param name="image" value="images/upperporch360.jpg">
  <param name="view" value="360">
  <param name="center" value="2222">
  <param name="pixdeg" value="0=14,570=96,1680=264,2304=14">
</applet>
<appletname="FloorPlan"code="FloorPlan.class"width=205 height=252>
  <param name="image" value="images/secondfloor.gif">
  <param name="x" value="100">
  <param name="y" value="492">
</applet>
```

Remarquez comment le programme pmvr fait tourner [l'image panoramique](#) et comment le programme floorplan fait tourner [l'image du plan](#). Les deux programmes communiquent entre eux pour garder les vues synchronisées. (l'utilisation du paramètre 'pixdeg' du programme pmvr signale que vous voulez que les deux programmes communiquent.)

On peut utiliser le même plan pour plusieurs panoramas. La seule chose qui change d'un panorama à l'autre est le "x" et le "y" du point fixe sur le plan. Le programme floorplan adaptera automatiquement l'image du plan pour centrer la position fixe "x" et "y".

Pour aider à comprendre comment fonctionnent les formules d'<APPLET> ci-dessus, consultez [Java Applet Tag Reference](#). Pour d'autres exemples et idées, consultez [la sélection de sites qui utilisent PMVR](#) [§18]

4. Démarrage rapide

Avez-vous déjà une image panoramique à afficher ? Sinon, la section [créer des panoramas](#) [§5] ci-dessous vous aidera à démarrer. Vous pouvez également utiliser cette [image de démonstration](#) (en la sauvegardant sur votre disque dur).

Ensuite, dès que vous avez une image panoramique à afficher, placez-la dans un dossier avec un fichier html (que vous créez vous-même, ou en utilisant un éditeur HTML) et vous ajoutez les lignes de code HTML suivantes dans le corps du document :

```
<applet code="pmvr.class" width=500 height= 300 >
  <param name="image" value=" votrepan.jpg ">
  <param name="view" value="360">
</applet>
```

Vérifiez que vous avez bien changé le 'height' (hauteur) indiquée ci-dessus en bleu pour l'adapter à la hauteur de votre panorama (ce n'est pas absolument nécessaire, mais c'est recommandé! Vous pouvez également adapter la hauteur de votre panoramique...) et que vous avez remplacé 'votrepan.jpg' indiqué ci-dessus en bleu par le nom sous lequel vous avez sauvegardé votre panoramique (indispensable). Si votre panoramique n'est pas une image à 360 degrés, vous pouvez supprimer la ligne du paramètre 'view'.

Enfin, placez les fichiers `acore.class`, `pmvr.class`, and `FloorPlan.class` (fichiers du `pmvr.zip` [télécharger](#) [§11]) dans le même dossier et vous y êtes !

Visualisez votre fichier HTML dans votre navigateur et vous verrez votre panorama. Placez simplement vos fichiers dans un dossier, sur une disquette ou un CD. Vous pouvez aussi placer vos fichiers sur un serveur web pour que chacun puisse les voir.

Vous désirez ajouter un plan? Si vous n'avez pas d'image de plan ou de carte, utilisez ce [plan de démonstration](#), en le sauvegardant sur votre disque dur. Ensuite, dans le même document HTML, ajoutez les lignes de code HTML suivantes après l'applet pmvr :

```
<applet code="FloorPlan.class" name="FloorPlan" width= 205 height= 300 >
  <param name="image" value=" votreplan.gif ">
  <param name="x" value=" 100 ">
  <param name="y" value=" 492 ">
</applet>
```

Vérifiez simplement que vous avez remplacé la largeur (width, 205) et la hauteur (height, 300) de l'applet floorplan par les dimensions de votre plan et remplacez 'votreplan.gif' par le nom sous lequel vous avez enregistré votre plan. Ensuite changez les paramètres 'x' (100) et 'y' (492) qui reflètent la position sur le plan d'où votre vue panoramique a été prise. Visualisez cet emplacement à l'aide de votre navigateur jusqu'à ce que vous soyez satisfait de l'emplacement (il faut tâtonner un peu) mais seul le point en couleur apparaît pour l'instant. L'étape finale pour ajouter un plan est d'ajouter le paramètre 'pixdeg' dans le programme pmvr ci-dessus, qui fera apparaître les lignes de couleur sur le plan et le fera fonctionner. Par exemple, ajoutez les lignes suivantes dans votre programme pmvr (attention : l'applet pmvr, pas l'applet floorplan!!!) :

```
<param name="pixdeg" value="0=90, 2000 =90">
```

Dans lesquelles vous remplacez **2000** par le nombre de pixels correspondant à la largeur de votre panorama. Bien que ce ne soit pas le plan que vous utiliserez finalement, vous voyez au moins apparaître dans votre navigateur les lignes sur votre plan et vous le verrez fonctionner. Maintenant, utilisez le [tutoriel PMVR 'pixdeg'](#) pour apprendre comment utiliser un pixdeg pour votre panorama.

5. Créer des panoramas

Un des moyens les plus rapides pour créer des panoramas est d'utiliser les logiciels de raccord d'image (Image stitching). Ces logiciels fonctionnent en prenant des images individuelles et en les assemblant pour former un panorama. La section [web ressources](#) [§19] contient des liens pour obtenir de plus amples informations sur ces logiciels. En particulier, <http://www.panoguide.com/> fournit un guide qui compare beaucoup de logiciels d'assemblage d'images.

Note: Beaucoup d'appareils photo numériques sont vendus aujourd'hui avec un CD contenant des logiciels pour cet appareil, incluant un logiciel d'assemblage de photos. De même, beaucoup d'appareils photos ont une fonction d'aide à la prise de panoramiques. Alors avant d'aller chercher une troisième solution, regardez les richesses que vous possédez!

Il vaut mieux utiliser un appareil photo numérique pour prendre vos photos, pour éviter de prendre vos photos avec un appareil argentique et d'avoir à les scanner ensuite.

TIP: Créer un panoramique de bonne qualité / résolution et sauvegardez-le sur votre disque dur. Ensuite vous le redimensionnez à la bonne taille et le sauvegarderez en JPG pour votre site web.

Ainsi vous disposerez toujours d'un panoramique de bonne qualité si vous en avez besoin ensuite. Par exemple, inclure une image dans une brochure, extraire un détail...

6. Paramètres de l'applet PMVR

(Tous les paramètres ne sont pas traduits. Voir pour cela la page originale, en anglais <http://duckware.com/pmvr>)

Les tableaux suivants décrivent tous les paramètres qui peuvent être utilisés par les programmes pmvr et Floorplan. Ne soyez pas effrayés par le nombre de paramètres: quelques-uns seulement sont indispensables.

Les nombreux paramètres optionnels sont là pour permettre d'agrémenter les applets, si vous en avez besoin. Les paramètres sont définis dans les `<APPLET>` tag et chaque paramètre défini ressemble à la ligne de code ci-dessous :

```
<param name =name value =value >
```

ou les noms (name) et valeurs (value) possibles sont décrits dans les tableaux ci-dessous.

Cliquez sur le nom d'un paramètre d'applet dans le tableau suivant pour obtenir une description détaillée de ce paramètre.

Notez que plusieurs paramètres d'applet ont été ajoutés dans des versions récentes (spécifiées entre crochets). Vous pourrez utiliser tous les paramètres si vous utilisez la version la plus récente.

Pour l'applet pmvr, seul le paramètre d'image est indispensable. Si l'image que vous regardez est à 360 degrés assurez-vous que vous utilisez le paramètre 'view' avec une valeur de 360. Les paramètres pmvr de base sont :

Paramètres de base de l'applet Java pmvr.class		
nom	valeur	Description
image	URL	L'URL de l'image panoramique (JPG) à afficher (image de démo)
view	nbre entier	La largeur de l'image panoramique, en degrés (1-360).
auto	nbre entier	défilement automatique horizontal, un pixel à chaque nombre de millisecondes spécifié
center	nbre entier	Définit le centre de l'image (horizontal) à l'endroit de ce pixel.
pixdeg	pixdeg-spec	A utiliser avec l'applet Floorplan pour préciser l'angle de vue.
delay	Nbre entier	Délai en millisecondes avant que le défilement automatique ne reparte (par défaut: 30000) [2.3b]

Pour utiliser l'applet floorplan, vous devez utiliser le paramètre **pixdeg** dans l'applet pmvr-- cela signale au programme pmvr qu'il doit chercher et utiliser le programme floorplan. Dans l'applet Floorplan, l'image, les paramètres 'x' et 'y' sont indispensables. Les paramètres de bases de l'applet FloorPlan sont :

Paramètres de base de l'applet Java FloorPlan.class		
nom	valeur	Description
image	URL	L'URL vers l'image du plan ou de la carte(GIF/JPG/etc) à montrer (image de démo)
x	Entier	Le pixel de localisation sur la carte du X (horizontal) où l'angle de vue est centré.
y	Integer	Le pixel de localisation sur la carte du Y (vertical) où l'angle de vue est centré.

Dans la plupart des cas, vous aurez besoin seulement des paramètres décrits dans les tableaux des paramètres de base ci-dessus. Cependant, si vous devez agrémenter votre applet avec des zones sensibles, des couleurs supplémentaires ou des textes, et bien d'autres options encore, consultez les paramètres avancés qui suivent (ils ne sont pas tous traduits, pour l'intégralité des paramètres voir <http://duckware.com/pmvr> :

Paramètres avancés de l'applet Java pmvr.class		
nom	valeur	Description
links	file-ref	L'URL des hyperliens (zones sensibles), où chaque ligne est un link-spec [2.2a]
link#	link-spec	Crée une zone sensible hypertexte (une alternative au fichier-liens)
hover	yes/no	La zone sensible devient-elle visible quand l'utilisateur la survole? [yes/no] (default: no) [3.0a]
ll	ll-spec	Lier les zones sensibles de pmvr aux zones sensibles de Floorplan [3.0a]
floorplan	nom	le nom de l'applet floorplan (default: FloorPlan)
zoom	yes/no	la ligne contrôlant le zoom est-elle apparente? [yes/no] (default: yes) [2.4a]
showlinks	yes/no	Les zones sensibles sont-elles apparentes? [yes/no] (default: no) [3.0d]
background	hex-color	Couleur de l'arrière-plan de l'applet (par défaut gris: C0C0C0) [2.1c]

Advanced FloorPlan.class Java Applet Parameters		
Nom	valeur	Description
links	URL	L'URL d'un fichier qui contient les zones sensibles, où chaque ligne du fichier est un link-spec [2.2a]
link#	link-spec	Crée une zone sensible hypertexte (une alternative au fichier-links) [2.1e]
hover	yes/no	La zone sensible devient-elle visible quand l'utilisateur la survole? [yes/no] (default: no) [3.0a]
ll	ll-spec	Lier les zones sensibles de FloorPlan aux zones sensibles de pmvr

		[3.0a]
spots	URL	L'URL vers un fichier overlay/spots ?, où chaque ligne est un spot-spec [3.0a]
showlinks	yes/no	Toutes les zones sensibles sont-elles surlignées ?[yes/no] (par défaut: no) [3.1a]
background	hex-color	Couleur de l'arrière plan de l'applet (par default gris: C0C0C0) [2.1c]

image -- <param name="image" value=" pano.jpg "> -- La valeur du paramètre 'image' est une URL vers une image (JPG/GIF/etc) sur le serveur web. Elle peut être une URL absolue (ex: <http://www.xyz.com/pano.jpg>) ou une URL relative (ex : [pano.jpg](#)) mais c'est presque toujours une URL relative (parce que cela permet de mettre le site web sur un CD sans perdre le lien vers l'image), liée à l'URL du document HTML. Remarquez que cette URL est sujette au [Java Applet Security Model \(modèle de sécurité des applet Java\)](#). Ce paramètre est obligatoire et doit être placé dans le code .

x,y -- <param name="x" value=" 100 "><param name="y" value=" 492 "> -- Les valeurs des paramètres x et y du plan précisent la localisation x et y d'un pixel, sur une carte, à l'endroit où le [point coloré](#) est dessiné: cette position x,y sur l'image de la carte est l'endroit d'où partent les lignes de l'angle de vue, signifiant "J'étais ici quand le panorama a été pris". Dans la plupart des cas, l'image du plan ou de la carte est plus grande que la partie visible de l'applet (parce que cette carte peut être partagée par plusieurs panoramas), donc l'applet va essayer de centrer la position x,y dans la partie visible de l'applet. Ces paramètres sont obligatoires et doivent être placés quand vous utilisez l'applet FloorPlan.

view -- <param name="view" value=" 360 "> -- La valeur du paramètre view est un nombre entier qui précise combien de degrés sont couverts par l'image panoramique (1-360). Quand elle est spécifiée à 360, PMVR permettra que l'image défile de façon continue. Sinon, le défilement de l'image s'arrête aux extrémités droite et gauche du panorama. Ce paramètre est optionnel et par défaut considère que le panorama ne fait pas 360 degrés.

auto -- <param name="auto" value=" 20 "> -- La valeur du paramètre auto est un entier qui précise combien de millisecondes le programme doit attendre avant de faire défiler automatiquement d'un pixel (horizontal) l'image panoramique. La valeur minimale est de 20 (millisecondes). Le signe de ce paramètre contrôle la direction du défilement: un signe positif entraîne un défilement vers la droite, un signe négatif fait défiler vers la gauche. Dans la version 2.5c, on peut également indiquer un coefficient multiplicateur qui précise combien de pixels doivent défiler à chaque intervalle (la valeur par défaut est 1). Il suffit de faire suivre la valeur d'un 'x' suivi du nombre de pixels désiré. Par exemple, "20x5" va faire défiler l'image de 5 pixels chaque 20 millisecondes. Sur une image qui n'est pas à 360, la direction de défilement s'inverse à chaque extrémité rencontrée. Placer ce paramètre à 0 arrêtera le défilement. Il faut noter que l'action de l'utilisateur dans le panorama arrêtera temporairement le défilement pour une durée donnée en millisecondes ([delay](#)) . Ce paramètre est optionnel et par défaut est de 0 (pas de défilement)

center -- <param name="center" value=" 1500 "> -- La valeur du paramètre center est un entier qui précise la localisation du pixel (horizontal)de l'image sur lequel le panorama doit être centré lorsque le panorama apparaît dans l'applet pour la première fois. Cela vous permet de contrôler quelle partie du panorama est vue en premier par l'utilisateur. Ce paramètre est surtout utilisé avec les panoramas horizontaux. Ce paramètre est optionnel et par défaut est au pixel 0.

links -- <param name="links" value=" firstfloor.links "> -- La valeur du paramètre links est une URL (absolue ou relative) vers un fichier, où chaque ligne dans le fichier est un [link-spec](#) (qui définit les zones sensibles) -- [example links file](#) . Si les mêmes zones sensibles sont utilisées encore et toujours (comme c'est le cas en général sur un plan), pointer vers un fichier

(qui contient les définitions des zones sensibles) est plus facile que de spécifier ces zones à chaque fois qu'on en a besoin avec le paramètre [link#](#) . Remarquez que cette URL est sujette au [Java Applet Security Model \(modèle de sécurité des applet Java\)](#) .. Ce paramètre est optionnel et par défaut aucun lien n'est spécifié.

link# -- `<param name="link0" value=" 300,100,50,kitchen.html,Kitchen ">` -- La valeur du paramètre `link#` est la définition d'une zone sensible [link-spec](#) (hot spot). Il suffit de continuer de numéroter autant de liens que vous voulez en définissant, en commençant à 0. Par exemple, `link0`, `link1`, `link2`, ..., `link10`, `link11`, etc. Ces paramètres sont optionnels.

hover -- `<param name="hover" value=" yes ">` -- La valeur du paramètre `hover` contrôle l'apparition ou non des lignes encadrant la zone sensible quand l'utilisateur la survole (avec la souris). Si la zone sensible est visible par l'utilisateur, il faut la définir avec soin. Ces paramètres sont optionnels et par défaut les contours des zones sensibles n'apparaissent pas ("no")

ll -- `<param name="ll" value=" 0=2,1=0 ">` -- Le paramètre `ll` (linked links, ou lien lié) lie les zones sensibles dans l'applet utilisé (`pmvr` or `FloorPlan`) aux zones sensibles de l'autre applet (`FloorPlan` or `pmvr`). Référez-vous à [ll-spec](#) pour une description détaillée de ces paramètres. Il est souvent utilisé avec le paramètre [hover](#). Ce paramètre est optionnel et par défaut il n'y a pas de lien lié.

spots -- `<param name="spots" value=" firstfloor.spots ">` -- La valeur du paramètre `spots` est une URL (absolue ou relative) vers un fichier, dans lequel chaque ligne est un [spot-spec](#) (ligne qui définit une image et une zone à recouvrir avec cette image). Remarquez que cette URL est sujette au [Java Applet Security Model \(modèle de sécurité des applet Java\)](#). Ce paramètre est optionnel et par défaut aucune zone n'est spécifiée.

pixdeg -- `<param name="pixdeg" value=" 0=14,570=96,1680=264,2304=14 ">` -- Veuillez vous référer au [pixdeg-spec](#) pour une description détaillée et un tutoriel qui permet de comprendre comment régler les valeurs de ce paramètre. Ce paramètre est nécessaire seulement si vous utilisez l'applet optionnel `floorplan` en parallèle avec `PMVR`.

floorplan -- `<param name="floorplan" value=" FloorPlan ">` -- La valeur du paramètre `floorplan` est le nom que vous avez attribué dans le code HTML à l'applet `FloorPlan`. Dans la plupart des cas, vous nommerez l'applet `Floorplan` "`FloorPlan`" comme dans `name="FloorPlan"` , et dans ce cas ce paramètre ne doit pas être spécifié. N'utilisez ce paramètre seulement si vous devez donner à `FloorPlan` un autre nom que "`FloorPlan`". Ce paramètre est optionnel et par défaut équivaut à "`FloorPlan`". Ce paramètre n'est utilisé que si vous avez besoin de plus d'un applet `FloorPlan` dans une même page web (en effet, chaque `FloorPlan` doit avoir son propre nom).

delay -- `<param name="delay" value=" 15000 ">` -- La valeur du paramètre `delay` précise le délai d'attente avant la reprise du défilement automatique (si cela est configuré par le paramètre [auto](#)). En effet, si une personne agit sur le panorama, les défilements [auto \(horizontal\)](#) et [vauto](#) (vertical) vont s'arrêter momentanément pour permettre à l'utilisateur d'agir sur l'image. Lorsqu'il n'y a pas eu d'activité pendant le nombre de millisecondes que vous avez spécifié par ce paramètre, le défilement reprend. Ce paramètre est optionnel et par défaut est défini à 30000 (30 secondes).

showlinks -- `<param name="showlinks" value=" yes ">` -- La valeur du paramètre `showlinks` contrôle si les zones sensibles sont surlignées ou non.

The primary use of this parameter is through Scripting control to provide a GUI to the user to turn highlights on/off at any time - [Example](#) . This parameter is optional and defaults to "no" (highlight off).

background -- `<param name="background" value=" C0C0C0 ">` -- La valeur du paramètre `background` contrôle la couleur [hex-color](#) d'arrière-plan de l'applet. Vous voyez l'arrière-plan

de l'applet lorsque l'image se charge ou si l'image est plus petite que la surface de l'applet. Ce paramètre est optionnel et par défaut correspond "C0C0C0" (gris clair).

file-ref : est un "nom de fichier(:section)" , dans lequel le nom de fichier est lié au document html, suivi éventuellement par une ":section" pour préciser la section du fichier à utiliser (plutôt que le fichier entier). Une aide à l'utilisation de ces sections de fichiers est disponible dans le [Native Scripting Manual](#).

link-spec : il s'agit de " num-list ,url-target ,description" ("numéro de liste, cible URL, description, soit la définition d'une zone sensible), dans lequel ' num-list ' précise le type de forme (rectangle, cercle, etc -- voir num-list plus bas), ' url-target ' est l'URL à lancer quand l'utilisateur clique sur la zone sensible, et 'description' est la description qui est affichée dans une étiquette lorsque l'utilisateur passe sa souris sur la zone sensible. Pour exemple :

Type de Région	exemple de link-spec (voir la num-list plus bas pour + d'info)
x-axis section	100,200,salon.html in _blank,Great Room
cercle	300,100,50,cuisine.html,Kitchen
rectangle	400,100,500,200,http://www.duckware.com,Duckware
polygone	100,0,200,200,0,200,pergola.html,Pergola

Un link-spec est utilisé aussi bien par les paramètres [link#](#) or [links](#) de l'applet pour définir une zone sensible cliquable dans l'image.

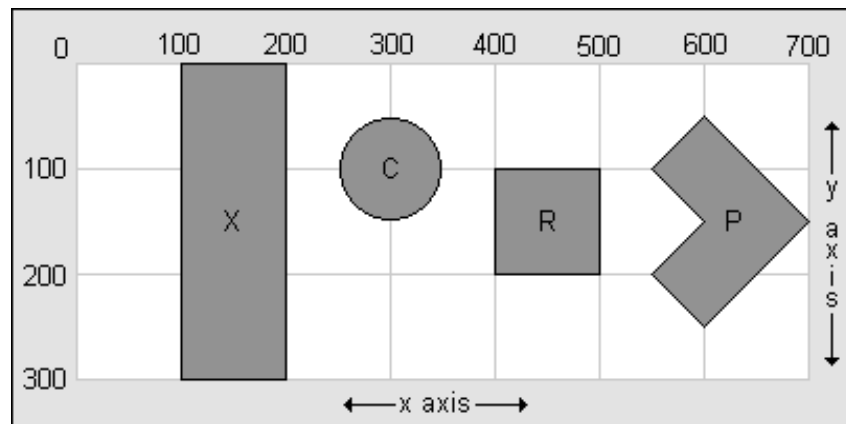
url-target : Une URL (absolue ou relative) suivie d'un éventuel " in target ", où la target (cible) est un cadre, une fenêtre (votre propre nom, ou _blank, _parent, _self, or _top). Ou "javascript:" suivi par un code JavaScript. Notez cependant que ce code ne doit pas contenir de virgule (qui sont déjà utilisées par le [link-spec](#)). Une URL relative est liée à l'URL du document HTML. Parfois vous désirez simplement créer une zone sensible qui annote l'image. Dans ce cas, indiquez un "-" comme objectif d'URL (un commentaire -- ne fait rien). Ci-dessous, quelques exemples d'objectifs d'URL.

Type	url-target Syntax	Exemple
relative	nom de fichier	kitchen.html
absolute	URL	http://www.xyz.com
target	URL in name	http://www.duckware.com in _top
javascript	javascript:code	javascript:alert('Clicked!')
variable	set:{applet.}parameter=value	set:pmvr.view=180
script	set:{applet.}script= file-ref	set:script=script.txt:gotoKitchen
comment	-	-

num-list : La taille de la liste de nombres varie en fonction du type de zone que vous créez. Pour une simple section d'axe-x (tous les pixels y, voir zone X ci-dessous), on utilise deux nombres, comme "gauche,droite". Pour un cercle, utilisez trois nombres, comme "x-du-centre,y-du-centre,rayon". Pour un rectangle, utilisez quatre nombres, comme "gauche,haut,droite,bas". Pour un polygone, utilisez plusieurs couples x,y, comme dans "x1,y1,x2,y2,.....,xn,yn". En résumé :

Region Type	num-list specification	exemples num-list	area below
x-axis section	gauche,droite	100,200	X

cercle	x-du-centre,y-du-centre,rayon	300,100,50	C
rectangle	gauche,haut,droite,bas	400,100,500,200	R
polygone	x1,y1,x2,y2,....,xn,yn	600,50,700,150,600,250, 550,200,600,150,550,100	P



pixdeg-spec : Une liste de 'pixel=degré' -- qui identifie la direction (en degré) dans laquelle vous regardez à plusieurs endroits de pixels. **TIP: Utilisez le tutoriel pixdeg interactif** pour apprendre comment utiliser le paramètre d'applet 'pixdeg'. À savoir, vous avez besoin de fournir à PMVR une liste de "quand vous regardez le pixel 'pixel' dans le panorama, dessinez une ligne en direction de 'degré sur la carte". Au minimum, le premier et dernier pixel/degré doivent être spécifiés. Il en faudra davantage si le panoramique n'est pas un panoramique parfait. Par exemple, dans [l'image du porche](#) utilisée au-dessus, (et le 'pixdeg' paramètre au-dessus), le pixel zero est pointé à 14°(en relation avec l'image du plan), ce qui fait "0=14". Et parce que l'image est un panoramique à 360 degrés, le dernier pixel est pointé dans la même direction, ce qui donne "2304=14". Cela donne un pixdeg-spec complet de "0=14,2304=14". Les autres 'pixel,degré' aident à corriger les imperfections de l'image.

spot-spec : Une "x,y,image" qui définit une image et l'emplacement d'une zone à recouvrir avec cette image. L'image est affichée centrée sur la zone "x,y". Cela permet aux zones d'être ajoutées de façon dynamique à l'image du plan. Par exemple, "100,100,spot1.gif" ordonne de recouvrir la zone "100,100" sur le plan. par l'image "spot1.gif" centrée à cet endroit.

ll-spec : Une liste, séparée par des virgules, de 'spot1=spot2' -- cela signifie 'quand l'utilisateur bouge sa souris sur le spot 1 dans l'applet utilisé, le spot 2 de l'autre applet est surligné.' Par exemple, "0=2,2=1" signifie "si l'utilisateur survole la zone sensible 0 dans l'applet utilisé, cela fait apparaître la zone sensible 2 dans l'autre applet" et si l'utilisateur survole la zone sensible 2 dans l'applet utilisé, cela fait apparaître la zone sensible 1 dans l'autre applet"

hex-color : Une couleur définie en 'RRGGBB' (red/green/blue = rouge/vert/bleu) en format hexadécimal. Par exemple, 'FF0000' est rouge pur, '00FF00' est vert pur, '0000FF' est bleu pur, 'FFFFFF' est blanc, '000000' est noir, 'CCCCCC' est gris clair etc... . Reportez-vous à <http://www.utexas.edu/learn/html/colors.html> pour avoir une liste de couleurs avec les codes hexadécimaux correspondants.

TIPS : Appuyez sur **d**(pour "debug") dans le panoramique ci-dessus (après avoir cliqué dans le panorama pour lui donner le 'focus') pour obtenir les valeurs des correspondances pixel/degré des extrémités gauches et droites *de ce que vous voyez dans la fenêtre de l'applet* . Très utile pour vous aider à créer le *pixdeg-spec* . De la même façon, appuyez sur les touches des flèches gauche/droite pour bouger l'image pixel par pixel (pressez la flèche vers le bas en même temps pour bouger de dix pixels à la fois). Appuyez sur **c** (comme copyright) pour afficher les informations de copyright. Appuyez sur la touche **v** (pour version) pour afficher les informations concernant la version. Appuyez sur la touche **x** (x marque la zone) pour afficher la localisation x,y du pixel de la souris.