

ratpoison

ratpoison est un gestionnaire de fenêtres simple sans grosse dépendances. pas de gadgets graphiques, pas de décoration de fenêtre, et aucune dépendance à l'égard du mulot :) . Il est largement calquée sur GNU Screen qui a fait des merveilles sur le marché du terminal virtuel. Toutes les interactions avec le gestionnaire de fenêtres se font avec le clavier. ratpoison fonctionne avec un combo-clavier (Control-t) comme touche de modification, qui permet de ne pas paralyser Emacs ou d'autres logiciels de qualité. ratpoison a été écrit par Shawn Betts sabetts@gmail.com.

Sommaire

ratpoison.....	1
installation.....	2
depuis les sources.....	2
sur Debian GNU/Linux.....	2
Concept des Frames.....	2
Utilisation.....	3
Fenêtres.....	3
Manipuler les fenêtres.....	3
format "winfmt".....	5
Classes des fenêtres.....	5
fenêtres non-gérées.....	6
Rudeness.....	6
Groupes.....	7
Frames.....	7
Diviser les Frames.....	7
Taille des Frames.....	8
Naviguer entre les Frames.....	9
Sauvegarde des frames.....	9
Identifier les frames.....	10
Frames dédiées.....	10
Multi-Moniteurs.....	10
Raccourcis clavier.....	11
Table des raccourcis.....	11
Raccourcis par défaut.....	12
Hooks.....	13
la barre de messages.....	14
Utiliser un autre WM.....	15
Commandes diverses.....	15
Zone de saisie.....	17
Arguments en ligne de commande.....	17
ratpoisonrc.....	18
conclusion.....	18

installation

depuis les sources

au moment où j'écris ces lignes, la dernière version de ratpoison est la 1.4.8.

le processus d'installation est classique: on installe les dépendances:

```
# apt-get install automake makeinfo texinfo libx11-dev libxext-dev x11proto-core-dev libreadline5-dev x11proto-xext-dev libxtst-dev
```

on récupère l'archive, on décompresse, on configure, make, install. plus de détails dans les fichiers inclus (README, INSTALL...):

```
$ wget http://download.savannah.gnu.org/releases/ratpoison/ratpoison-1.4.8.tar.xz
$ tar -xJf ratpoison-1.4.8.tar.xz
$ cd ratpoison-1.4.8.tar.xz
$ ./configure
$ make
# make install
```

sur Debian GNU/Linux

ratpoison est présent dans les dépôts Debian Jessie en 1.4.8. pour l'installer:

```
# apt-get install ratpoison
```

Concept des Frames

ratpoison utilise le concept de **cadres** ou **frames** pour placer ou redimensionner les fenêtres: Au lieu de laisser les fenêtres se placer arbitrairement sur l'écran, celui-ci est divisé en cadres. les fenêtres se placent maximisées dans ces cadres. ratpoison se lance avec un cadre qui peut être divisé en plusieurs cadres plus petits. Chaque cadre peut être divisée en deux, horizontalement ou verticalement. Vous pouvez vous déplacer parmi eux, ou en créer de nouveaux. Pour plus d'informations, consultez la section **frames**.

Chaque cadre a une fenêtre ou plus qui lui est associée, qui est visible dans ce cadre. Si vous sélectionnez une fenêtre qui est associée à un cadre, le focus se déplace vers le cadre associé, plutôt que de déplacer la fenêtre dans le cadre actuel. Si vous sélectionnez une fenêtre qui n'est pas associée à un cadre, cette fenêtre sera ouverte dans le cadre actuel et redimensionnée pour s'adapter à ce cadre.

Si la fenêtre associée ne remplit pas complètement le cadre, les différentes commandes de gravité contrôlent la façon dont elle est placée.

Si aucune fenêtre n'était ouverte dans ce cadre avant la création du cadre, le bureau sera visible derrière elle.

Utilisation

Quand ratpoison démarre, vous devriez voir un serveur X vide. Pour ouvrir un terminal, taper "C-t c". Vous pouvez maintenant exécuter des commandes shell comme vous le feriez sur n'importe quel terminal. Remarquez le terminal maximisé plein écran. "C-t!" vous ouvrira une invite de commande simple vous permet d'économiser l'effort d'ouvrir un terminal.

Une fois que vous avez quelques programmes en cours sur X, vous aurez envie de naviguer entre les fenêtres. Pour voir quelles fenêtres sont gérées, tapez "C-t w". Chaque fenêtre a un numéro. Vous pouvez passer d'une fenêtre en appuyant sur "C-t" suivi par le numéro de la fenêtre. Cela suppose que le numéro de la fenêtre est un chiffre. Vous pouvez également passer à une fenêtre en tapant une partie de son nom. Pour ce faire, taper "C-t ".

ratpoison vous permet de faire défiler les fenêtres avec "C-t n" et "C-t p".

ceci est une brève introduction sur la façon d'utiliser ratpoison. Remarquez comment nous n'avons pas eu à faire glisser une fenêtre unique, ou cliquez sur un seul bouton d'agrandissement... *"It's modern computing at its best"*.

Fenêtres

Manipuler les fenêtres

liste des commandes acceptées pour la manipulation des fenêtres:

pour lancer une commande dans ratpoison: "C-t :." ("Control+t" puis ":",)

Command	Action & Keys
select nb	afficher la fenêtre "nb" où "nb" est le numéro de la fenêtre indiqué dans la barre des messages. vous pouvez aussi faire un "C-t nb". pour ne sélectionner aucune fenêtre (vider le frame actuel) command="select -" ou key="C-t -".
select window-name	afficher la fenêtre selon son nom. key="C-t '".
windows fmt	affiche les fenêtres gérées selon un format (fmt). le numéro devant chaque fenêtre vous permet de l'afficher. vous pouvez y accéder avec "C-t nb" (nb=numéro de la fenêtre). notez que cette option fonctionne de 0 à 9. pour atteindre une fenêtre avec nb>9, utiliser "C-t '" et entrer le numéro désiré après le prompt. le raccourcis par défaut: "C-t w". si vous utilisez cette commande depuis un terminal, "\$ ratpoison -c windows" au lieu de la barre de message, vous obtiendrez une liste des fenêtres dans stdout. cela vous permettra d'écrire quelques scripts. c'est là que le format (fmt) entre en jeu: utiliser "set winfmt" (voir plus bas) pour définir le format d'identification des fenêtres et ainsi modifier la sortie de vos scripts.
title title	renomme la fenêtre active. ce nom restera jusqu'à la fermeture de la fenêtre ou jusqu'à ce que vous le changiez. par défaut key="C-

	t A".
other	échanger la fenêtre actuelle avec la précédente. par défaut key="C-t C-t".
prev	afficher la fenêtre précédente. par défaut key="C-t p".
next	afficher la fenêtre suivante. par défaut key="C-t n", "C-t space", et "C-t enter".
kill	détruit la fenêtre actuelle. cette commande est utilisée si la fenêtre ne répond pas à la fermeture. par défaut key="C-t K".
info fmt	affiche les informations sur la fenêtre actuelle. un format peut être passé en option pour écraser le format par défaut. "info" accepte les même options de format que "windows". (voir le tableau à la suite)
gravity g	change la gravité de la fenêtre actuelle (son point de départ dans la frame). par défaut, la fenêtre est placée depuis le coin supérieur gauche du cadre, "g" permet d'aligner la fenêtre selon les 8 directions: "northwest", "north", "northeast", "east", "southeast", "south", "southwest" et "west". "center" centre la fenêtre dans le cadre. "g" peut être abrégé par une ou deux lettres (iex: "nw", "s", etc). sans argument "g", cette commande affiche la gravité de la fenêtre actuelle.
delete	efface la fenêtre actuelle. par défaut key="C-t k".
set infofmt fmt	définir le format utilisé par la commande "info". voir "winfmt" pour la liste des format acceptés.
set winname name	ratpoison peut nommer la fenêtres de 3 façons: avec son WMNAME, le res_name du WMCLASS, ou le res_class du WMCLASS (obtenus avec 'xprop'). le nom peut être le titre "title" comme dans la plupart des tasklists, le nom "name"=res_name, ou sa classe "class"=res_class. sans argument, cette commande affiche la valeur par défaut.
set wingravity g	définir la gravité par défaut pour les fenêtres normales. voir "gravity" sans argument, cette commande indique la valeur par défaut.
set winliststyle setting	la liste des fenêtres peut s'afficher en ligne "row" ou en colonnes "column".
number n target	attribuer un numéro "n" à une fenêtre. si une autre fenêtre possède ce numéro, les numéros sont échangés. le second argument, "target", est optionnel. il sert à définir la fenêtre dont on change le numéro. si omit, la fenêtre actuelle sera renumérotée.
set transgravity g	définir la gravité par défaut des fenêtres de transition. voir "gravity". sans argument, cette commande affiche la gravité par défaut.
set maxsizegravity g	définir la gravité par défaut des fenêtres à taille maximale fixe. voir "gravity". sans argument, cette commande affiche la gravité par défaut.
set border n	définir l'épaisseur de la bordure des fenêtres. sans argument, cette commande affiche la valeur par défaut.
set winfmt fmt	définir le format par défaut pour la commande "windows". par défaut '%n%s%t' (numéro-statut-titre). voir la tableau suivant pour la liste des formats acceptés. sans argument, cette commande affiche la valeur par défaut.

format "winfmt"

char	window format
'%a'	nom de l'application
'%c'	classe de l'application
'%f'	numéro du cadre dans lequel se trouve la fenêtre ou un espace si la fenêtre est hors-cadre
'%g'	paramètre de gravité de la fenêtre
'%h'	hauteur de la fenêtre
'%H'	échelle de redimensionnement vertical
'%i'	ID de la fenêtre
'%l'	historique d'accès de la fenêtre. numéro le plus haut=fenêtre la plus récente.
'%n'	numéro de la fenêtre
'%p'	ID du processus ('?' si _NET_WM_PID n'existe pas)
'%s'	statut de la fenêtre (fenêtre actuelle, dernière fenêtre, etc)
'%S'	écran de la fenêtre
'%t'	nom de la fenêtre
'%T'	fenêtre de transition ou non
'%M'	fenêtre à taille maximale fixe ou non
'%w'	largeur de la fenêtre
'%W'	échelle de redimensionnement horizontal
'%x'	écran xinerama de la fenêtre

vous pouvez ajouter un nombre devant le format afin de définir le nombre de caractères à afficher, ex: "%n%s%20t".

Classes des fenêtres

"Window classes" est une façon de regrouper les fenêtres. les fenêtres d'un même programme ont généralement la même propriété "classe" définie par 'xprop'. ratpoison utilise cette propriété pour naviguer entre le fenêtrés selon leur classe. bine pratique pour ne circuler que parmi les "XTerms" par exemple.

Command	Action
inext	afficher la fenêtre suivante ayant la même classe que la fenêtre actuelle.
iprev	afficher la fenêtre précédente ayant la même classe que la fenêtre actuelle.
iother	afficher la dernière fenêtre ayant la même classe que la fenêtre actuelle.
cnext	afficher la fenêtre suivante n'ayant pas la même classe que la fenêtre actuelle.
cprev	afficher la fenêtre précédente n'ayant pas la même classe que la fenêtre actuelle.
cother	afficher la dernière fenêtre n'ayant pas la même classe que la fenêtre actuelle.

fenêtres non-gérées

ratpoison peut ne pas gérer certaines fenêtres. ratpoison garde une liste de termes et toute nouvelle fenêtre correspondant à l'un de ces termes ne sera pas gérée par ratpoison.

The following are commands to manipulate this list:

Command	Action
clrunmanaged	vide la liste des fenêtres non-gérées.
unmanage text	ajoute "text" à la liste des fenêtres non-gérées. toute nouvelle fenêtre qui correspond ne sera pas gérée par ratpoison. sans argument, cette commande affiche la liste actuelle.

Rudeness



Some programs will attempt to steal the focus without the users permission. Not only is this a sign of a lame programmers attempt to fix a window manager problem in the wrong place, it's just plain rude. By default ratpoison will honour these rudeness requests, but it doesn't have to. Use the rudeness command to deal with such programs.

Command	Action
rudeness n	The rudeness command lets you decide what windows pop-up automatically and when. This is often useful for those deep hack sessions when you absolutely can't be disturbed. n is a number from 0 to 15. Each of the four bits determine which requests ratpoison grants. When called with no arguments, the current setting is returned.

There are two kinds of windows: normal windows (like an xterm) and transient windows (generally pop-up dialog boxes). When a client program wants to display a new window it makes a requests to ratpoison. ratpoison then decides whether to grant the request and display the window or ignore it. A client program can also request that one of its windows be raised. You can customize ratpoison to either honour these requests (the default operation) or ignore them.

- Bit 0 : Tells ratpoison to grant raise requests on transient windows
- Bit 1 : Tells ratpoison to grant raise requests on normal windows
- Bit 2 : Tells ratpoison to grant display requests on new transient windows
- Bit 3 : Tells ratpoison to grant display requests on new normal windows

For example, if you wanted only wanted to grant transient windows raise requests and display requests you would type `rudeness 5'. If a request is not granted ratpoison will tell you about the request with a message like `Raise request from window 1 (emacs)'.

Groupes

ratpoison peut grouper les fenêtres entre elles. cette option couplée à la configuration avancée des frames permet d'obtenir un équivalent aux "bureaux" ou "espaces de travail".

même si ratpoison ne gère pas les bureaux virtuels, il vous offre la possibilité d'écrire des scripts à cet effet. de tel scripts existent dans le dossier "contrib/" des sources: les "rpws". consultez ce dossier pour plus de détails sur le support des bureaux virtuels dans ratpoison.

les groupes sont plus flexibles que les bureaux virtuels. une fenêtre d'un groupe peut être visible dans d'autres groupes. si vous changez de groupe, rien ne change sauf la liste des fenêtres dans laquelle vous circulez. ratpoison vous permet de créer des groupes, de déplacer des fenêtres d'un groupe à l'autre, de fusionner des groupes (merge) ou de les détruire.

voici la liste des commandes applicables aux groupes.

Command	Action
gnew name	créer un nouveau groupe "name". name est optionel. le nouveau groupe devient le groupe actuel.
gnewbg name	idem "gnew" mais le groupe actuel reste le même.
groups	affiche une liste des groupes.
gmove group	intègre la fenêtre actuelle dans "group".
gnext	afficher le prochain groupe.
gother	afficher le dernier groupe visité.
gprev	afficher le groupe précédent.
grename	renomme le groupe actuel.
gselect group	sélectionne un groupe depuis son numéro ou son nom. si "group" est omit, ratpoison vous le demandera.
gmerge group	fusionne "group" avec le groupe actuel. toutes les fenêtres de "group" sont déplacées dans le groupe actuel. "group" n'est pas effacé (il peut accueillir de nouvelles fenêtres).
gdelete group	efface un groupe. "group" est optionel. si "group" est omit, ratpoison tentera d'effacer le groupe actuel. seul un groupe vide peut être effacé. pour vider un groupe, voir "gmerge".

Frames

les **frames** sont les cadres dans lesquels s'affichent les fenêtres.

Diviser les Frames

ou comment diviser les cadres dans ratpoison pour afficher plusieurs fenêtres à l'écran.

- pour diviser une frame horizontalement "C-t s"
 - pour diviser verticalement "C-t S"
- si vous avez plusieurs fenêtres ouvertes, vous constaterez que ratpoison occupe le nouveau cadre avec un autre fenêtre automatiquement. vous pouvez utiliser les touches de navigations pour circulez dans le cadre.
- notez que si vous sélectionnez une fenêtre avec son nom ou son numéro, le cadre de la fenêtre sélectionnée deviendra le cadre actuel.

- pour circuler entre les frames, "C-t tab".
- pour effacer une frame, "C-t R". ratpoison ajuste automatiquement la taille des autres frames pour occuper l'espace disponible. cependant, il est possible que ratpoison remplisse l'espace d'un manière inattendue.
- pour revenir à l'affichage en plein écran (éliminer les divisions), "C-t Q".

les commandes applicables aux **frames**

Command	Action
remove	détruit le cadre actuel, seulement si il y en existe d'autres.
only	détruit tous les cadres sauf le cadre actuel.
(v)split n*	divise le cadre en deux (vertical). la dernière fenêtre visité n'occupant pas un cadre spécifique vient occuper le nouveau cadre.
hsplit n*	divise le cadre en deux (horizontal). la dernière fenêtre visité n'occupant pas un cadre spécifique vient occuper le nouveau cadre.

*n peut être une fraction au format x/y ou un nombre de pixels.

- si "n" est une fraction, le cadre actuel sera divisé selon cette fraction, créant le nouveau cadre dans l'espace libéré. ex: "split 1/4" réduira le cadre actuel au 1/4 de sa taille originale et laissera les 3/4 restant pour le nouveau cadre.
- si "n" est un nombre de pixels, le cadre original sera agrandi de "n" pixels
- si "n" est un nombre négatif de pixels, le cadre original sera réduit de "n" pixels

Taille des Frames

ou comment redimensionner une **frame** avec la commande **resize**. le raccourcis par défaut: "C-t r".

la commande "resize" dépend de 2 arguments, qui définissent respectivement le redimensionnement horizontal et vertical du cadre. ex: pour agrandir le cadre actuel de 50px horizontalement et le réduire verticalement de 10 px, on utilisera la commande "resize 50 -10" pour un redimensionnement depuis les raccourcis clavier (confirmation de la taille avec [Enter]):

Keys	Action on Frames
C-t r C-p/Up	agrandit la hauteur du cadre.
C-t r C-n/Down	réduit la hauteur du cadre.
C-t r C-f/Left/Right	agrandit la largeur du cadre.
C-t r C-b/Left/Right	réduit la largeur du cadre.
return	valide la redimension.
C-t r C-g/Esc	abandonne la redimension, rétablit la dimension originale.

l'échelle de redimensionnement est définie par la commande "set resizeunit".

Command	Action
set resizeunit pixels	définir le nombre de pixels avec lequel un cadre sera réduit ou agrandi depuis la commande resize ou un raccourcis. sans argument, la commande affiche la valeur actuelle.
resize horiz vert	redimensionne la cadre de "horiz" pixels en largeur et

	"vert" pixels en hauteur. sans argument, ratpoison laisse l'utilisateur redimensionner le cadre avec les raccourcis "C-p", "C-n", "C-b", "C-f", et "s" pour réduire (voir "shrink") (See the shrink command). valider le redimensionnement avec [Enter].
shrink	si une fenêtre ne remplit pas entièrement le cadre (comme certaines fenêtres de terminal) cette commande va automatiquement adapter les cadres pour éliminer tout espace libre.

Naviguer entre les Frames

voici les commandes acceptées pour circuler entre les **frames**.

Command	Action
fselect n	sélectionne un cadre depuis son numéro "n". si "n" est spécifié, ratpoison tentera de sélectionner le cadre "n". si "n" est omit, ratpoison affiche un numéro dans le coin supérieur gauche de chaque cadre pour les identifier, et attends la sélection de l'utilisateur. il n'y a pour le moment aucun moyen de sélectionner un cadre avec "n">9 depuis un raccourcis, seule la commande le peut.
curframe	indique le cadre actuel.
focus	circule dans les cadres.
focusprev	circule dans les cadres précédents.
focusdown	affiche le cadre au-dessous le cadre actuel.
focuslast	échange le cadre actuels avec le dernier cadre sélectionné.
focusleft	sélectionne le cadre à gauche du cadre actuel.
focusright	sélectionne le cadre à droite du cadre actuel.
focusup	affiche le cadre au-dessus le cadre actuel.

Sauvegarde des frames

ratpoison propose deux commandes, **fdump** et **frestore**, qui vous permettent de sauvegarder un agencement des **frames**.

utilisation: vous avez divisé votre écran en plusieurs frames mais vous souhaitez temporairement revenir à un affichage fullscreen sans pour autant perdre votre agencement à l'écran. utilisez alors "fdump" pour sauvegarder vos frames, puis tapez "C-t Q" pour effacer vos divisions et sélectionnez le cadre sur lequel vous désirez travailler en plein écran. une fois fini, utilisez "frestore" pour rétablir votre agencement initial.

si une fenêtre occupe un cadre avant le "dump" et est fermée avant le "restore", l'espace de cette fenêtre restera vide.

lancer "fdump", copier-coller le "dump" à chaque fois est un peu contre-productif. visitez votre dossier sources doc/sample.ratpoisonrc pour une série de raccourcis accomplissant ce processus.

Command	Action
fdump screen-num	sauvegarde l'agencement actuel au format texte. sans argument, l'écran actuel est pris en compte. avec argument, le numéro d'écran passé en argument est pris en compte. voir la section multi-moniteurs.
frestore frames	rétablit l'agencement "frames". "frames" doit être le "texte" sauvegardé lors du dernier "fdump".
undo	annule la dernière commande.
redo	relance la dernière commande.

Identifier les frames

les frames sont indentifiées par des numéros commençant par 0, mais la commande "framesels" permet de renommer les frames avec des chiffres ou des lettres.

```
set framesels abcdefghijklmnopqrstuvwxyz
```

le code ci-dessus renomme les frames avec des lettres et les agence dans l'ordre alphabétique.

Command	Action
set framesels order	définir le noms et l'ordre des frames lors de leur création. sans argument, la commande affiche la valeur actuelle.

Frames dédiées

une frame dédiée "dedicated" est un cadre dans lequel aucune fenêtre ne s'affiche automatiquement: seul l'utilisateur peut lui assigner une fenêtre.

Command	Action
dedicate	rend le frame actuel dédié ou non.

Multi-Moniteurs

le système X window assigne un numéro à chaque écran connectés reconnus. pour naviguer entre ces écrans, utiliser les commandes "nextscreen" et "prevscreen", ou "sselect" pour aller directement à un écran spécifique. ratpoison indiquera quel frame détient le focus avec une fenêtre de dialogue.

la plupart des commandes de ratpoison ne s'appliquent qu'à l'écran actuel. si vous désirez circuler entre deux fenêtres sur deux écran avec la commande "other", vous verrez un message "No other window." car il n' y a pas d'autres fenêtres sur l'écran actuel. si vous désirez aller à l'écran suivant, vous pouvez sélectionner la fenêtre depuis son nom (avec "select" ou "C-t"), son numéro, ou utiliser "nextscreen", "prevscreen", et "sselect".

Command	Action
nextscreen	aller à l'écran suivant.
prevscreen	aller à l'écran précédent.
sselect n	aller à l'écran numéro "n". les écrans commencent à 0.
sdump	comme "fdump", mais sauvegarde l'écran courant.
sfdump	comme "fdump", mais sauvegarde tous les écrans.
sfrestore	rétablit l'agencement sauvegardé par "sfdump".

Raccourcis clavier

ratpoison se contrôle intégralement depuis le clavier: commandes ou raccourcis.

ratpoison utilise l'organisation des raccourcis Emacs: une combinaison de touches de modifications suivies de touches classiques pour exécuter une action. la syntaxe: une ou plusieurs touches de modifications séparées par "-" puis une touche classique.

liste des touches de modifications acceptées par ratpoison:

- S : Shift
- C : Control
- M : Meta
- A : Alt
- H : Hyper
- s : Super

ratpoison utilise la propriété keysym de X11. les touches alphanumériques sont celles que vous voyez sur votre clavier. les ponctuations ont des noms différents selon le serverX utilisé. pour trouver le nom d'une touche, voir la commande "describekey". pour savoir si une touche est assignée à une action, taper "C-t" puis la touche concernée: ratpoison affichera un message avec le nom de la touche (si elle n'est pas assignée).

Table des raccourcis

tous les raccourcis sont contenus dans une table ou "keymap". lorsque vous tapez le préfixe "C-t", vous appelez le keymap 'root'. par défaut, toutes les commandes sont accessibles depuis le keymap principal commençant par "C-t".

il existe aussi une table supérieure, "top keymap". tous les raccourcis de cette table sont accessibles simplement en pressant une touche. c'est là que le préfixe entre en jeu.

l'exemple suivant assigne "C-x b" à l'échange des fenêtres, comme le "C-x b" dans Emacs.

```
# création du keymap
newkmap ctrl-x
# assigne b à 'select' dans le nouveau keymap
definekey ctrl-x b select
# attache notre keymap au "top key map" via C-x.
definekey top C-x readkey ctrl-x
```

voici les commandes pour créer, éditer ou supprimer une table de raccourcis.

Command	Action on keymap
newkmap kmap	créer une table de raccourcis "kmap".
delkmap kmap	supprimer la table de raccourcis "kmap".
bind Key command	assigne "key" à "command" dans la table de raccourcis 'root' (accessible depuis "C-t"). ex: pour assigner "C-t R" au redémarrage de ratpoison, utiliser "bind R restart".

unbind key	désactive "key" de la table 'root'.
definekey kmap key command	fonctionne comme "bind" mais permet d'entrer un raccourcis dans n'importe quelle table (pas seulement 'root').
undefinekey kmap key	comme "unbind" mais pour la table "kmap".
readkey kmap	attend pour une touche et exécute la commande associée dans "kmap".
link key	appelle la commande liée à "key". ex: "link C-t" appelle la commande "other" et échange les fenêtres.
describekey keymap	une façon interactive de connaître une commande associée dans "keymap". cette commande attend une touche de l'utilisateur. une fois donnée, ratpoison affiche la commande associée à la touche tapée.
set topkmap kmap	définir "kmap" comme table de raccourcis de niveau supérieur. vous utiliserez cette commande pour changer de table en cours de session.

Raccourcis par défaut

les raccourcis par défaut sont rassemblés dans ce tableau. notez que toutes les commandes ne sont pas associées à des raccourcis.

Keys	Action
C-t C-t	afficher la dernière fenêtre.
C-t t	simule un Control+t dans la fenêtre active. pour ouvrir un onglet firefox par ex.
C-t 0-9	affiche la fenêtre portant le numéro sélectionné.
C-t -	ne sélectionne aucune fenêtre: essentiellement pour cacher toutes les fenêtres d'un frame.
C-t A / C-t C-A	renomme la fenêtre active.
C-t K / C-t C-K	détruit la fenêtre active et tue le processus associé.
C-t n / C-t C-n / C-t Return C-t C-Return / C- t Space / C-t C- Space	affiche la fenêtre suivante.
C-t p / C-t C-p	affiche la fenêtre précédente.
C-t ' / C-t C-'	affiche la fenêtre depuis son nom. notez que Tab fait défiler les noms des fenêtres.
C-t a / C-t C-a	affiche l'heure.
C-t c / C-t C-c	ouvre une fenêtre de terminal.
C-t :	exécute une commande ratpoison depuis la barre de messages.
C-t !	exécute une commande shell depuis la barre de messages.
C-t C-!	exécute une commande shell depuis un terminal.
C-t i / C-t C-i	affiche les informations de la fenêtre active.
C-t k / C-t C-k	ferme la fenêtre active.
C-t l / C-t C-l	redessine la fenêtre active: il arrive que certaines fenêtres ne se maximisent pas dans le cadre, au autre façon de dire qu'il reste quelques bugs dans ratpoison.
C-t m / C-t C-m	affiche le dernier message.
C-t v / C-t C-v	affiche la version de ratpoison.

C-t V / C-t C-V	affiche la license de ratpoison.
C-t w / C-t C-w	affiche la liste des fenêtres gérées. la fenêtre active est surlignée.
C-t s / C-t C-s	divise horizontalement la frame active.
C-t S / C-t C-S	divise verticalement la frame active.
C-t tab	circule entre les frames.
C-t M-tab	échange avec la dernière frame visitée.
C-t Q	élimine toutes les frames sauf la frame active.
C-t R	élimine la frame active, seulement si il en existe d'autres.
C-t r / C-t C-r	redimensionne la frame active.
C-t b / C-t C-b	renvoi le pointeur dans le coin inférieur droit.
C-t ?	affiche l'écran d'aide.
C-t f / C-t C-f	sélectionne une frame depuis son numéro.
C-t F	identifie la frame active.
C-t Down	déplace le focus sur la frame au-dessous.
C-t Left	déplace le focus sur la frame de gauche.
C-t Right	déplace le focus sur la frame de droite.
C-t Up	déplace le focus sur la frame au-dessus.
C-t C-Down	échange la fenêtre de la frame active avec la fenêtre de la frame au-dessous.
C-t C-Left	échange la fenêtre de la frame active avec la fenêtre de la frame de gauche.
C-t C-Right	échange la fenêtre de la frame active avec la fenêtre de la frame de droite.
C-t C-Up	échange la fenêtre de la frame active avec la fenêtre de la frame au-dessus.
C-t x / C-t C-x	choisir une frame et échanger la fenêtre de la frame active avec la fenêtre de la frame sélectionnée.

Hooks

un des buts de ratpoison est de permettre à l'utilisateur de créer des modifications en accord avec ses besoins ou sa façon de travailler. les "Hooks" permettent à l'utilisateur d'associer une série de commandes à un évènement.

chaque "hook" contient une série de commandes à exécuter lorsque qu'un évènement arrive dans ratpoison. par ex, si vous souhaitez renvoyer le pointeur à chaque exécution d'un raccourcis clavier, ajoutez cette ligne dans votre fichier .ratpoisonrc: "addhook key banish" (sans les ""). ça devrait suffir pour tenir le mulot hors de portée.

- Command: addhook hook command : ajoute "command" à "hook". lorsque "hook" arrive, "command" est exécuté.
voici la liste des "hooks" acceptés par ratpoison:

Hook	Description
key	exécuté lors d'un raccourcis de niveau supérieur (par défaut, le seul toplevel keymap est le préfixe: "C-t").
switchwin	exécuté lors d'un échange de fenêtres dans une frame.

switchframe	exécuté lors d'un échange de frame. exécuté aussi lors d'un changement d'écran.
switchgroup	exécuté lors d'un échange de group.
deletewindow	exécuté si une fenêtre est détruite.
newwindow	exécuté après l'ouverture d'une fenêtre.
titlechanged	exécuté si le titre de la fenêtre active est changé.
quit	exécuté lors de la fermeture de ratpoison.
restart	exécuté lors de redémarrage de ratpoison.

- Command: "remhook hook command" : efface "command" de "hook". voir "addhook" pour la liste des "hooks" acceptés.
- Command: "listhook hook" : liste des commandes exécutées lorsque "hook" arrive.

la barre de messages

ratpoison affiche les notifications, commandes, listes des fenêtres etc, dans une barre de messages située par défaut dans le coin supérieur droit de l'écran. cete barre s'efface en 5 secondes par défaut.

voici les commandes modifiant le comportement et l'aspect de la barre de messages:

Command	Action
msgwait n	définir le temps de masquage de la barre. sans argument, affiche la valeur actuelle.
lastmsg	affiche le dernier message.
echo text	affiche "text" dans la barre.
set inputwidth n	définir la taille de la fenêtre d'input (mode commande). sans argument, affiche la valeur actuelle.
set font font	définir la police utilisée dans la barre. "font" est au format "9x15bold" ou "Droid Sans Mono-11". sans argument, affiche la valeur actuelle.
set framefmt fmt	définir le format du message affiché lors de la commande "curframe". voir les formats acceptés par winfmt.
set fgcolor color	définir la couleur du texte des messages de ratpoison "gray80" ou "DeepSkyBlue4". sans argument, affiche la valeur actuelle.
set bgcolor color	définir la couleur du fond des messages de ratpoison. sans argument, affiche la valeur actuelle.
set fwcolor color	définir la couleur de la bordure de la fenêtre active. sans argument, affiche la valeur actuelle.
set bwcolor color	définir la couleur de la bordure des fenêtres inactives. sans argument, affiche la valeur actuelle.
set barpadding x y	définir la marge interne dans la barre de message. sans argument, affiche la valeur actuelle.
set bargravity g	définir l'alignement du texte dans la barre de messages. voir la commande gravity. sans argument, affiche la valeur actuelle.
set barborder n	Set the border width for the bar window. When called with no arguments, the current setting is returned.
set barinpadding n	définir si la barre s'affiche dans le coin de l'écran (1) ou la coin de la fenêtre (0). sans argument, affiche la valeur actuelle.

Utiliser un autre WM

il peut arriver qu'un programme ne puisse pas s'afficher correctement dans ratpoison. ratpoison fournit deux commandes pour pallier à ce problème: **tmpwm** et **newwm**.

ces commandes doivent être utilisées aussi rarement que possible. elles sont simplement là pour montrer à l'utilisateur que certains programmes sont si explicitement préconfigurés, qu'il est impossible de changer leur façon de gérer leurs interfaces.

- Command: tmpwm WM : donne le contrôle temporaire à un autre gestionnaire de fenêtres, puis revient à ratpoison lors de la fermeture du wm. "WM" est le nom de l'exécutable ou l'adresse du wm.
- Command: newwm window-manager : mauvaise commande. ça tue ratpoison et fait revenir ce vilain mulot! ne pas faire!

Commandes diverses

une petite liste de commandes acceptées par ratpoison et pas forcément détaillées précédemment.

Command	Keys & Action
abort	pratique. par défaut key="C-t g": abandonne la dernière combinaison de touches (comme C-g dans 'Emacs').
alias name command	permet de nommer une commande de ratpoison autrement. ex: création d'un alias pour firefox: "alias ffx exec firefox" un alias est traité comme une commande lancée depuis "C-t :", vous pouvez l'associer à une touche, et la lancer depuis "ratpoison -c".
banish	envoi le pointeur dans le coin inférieur droit de l'écran.
banishrel	envoi.le pointeur dans le coin inférieur droit de la fenêtre. si il n'y a pas de fenêtres, dans le coin de la frame.
chdir	change le répertoire courant de ratpoison.
colon command	lance une commande ratpoison.
compat	Install the now obsolete `def*' commands as aliases to the corresponding `set *' command. 
set padding left top right bottom	définir les marges aux bords d'écran. sans arguments, affiche les valeurs actuelles.
set waitcursor n	définir si le pointeur est "carré" en attendant une touche (0) ou pas(différent de 0). sans argument, affiche la valeur actuelle.
set historysize n	définir la taille de l'historique des commandes ratpoison. sans argument, affiche la valeur actuelle.
set historcompaction bool	définir si on efface les commandes identiques dans l'historique. sans argument, affiche la valeur actuelle.
set historexpanion bool	Set whether to expand ! using readline's libhistory in input. 
escape key	définir le préfixe ,la combinaison de touche qui accède au keymap 'root' ("C-t" par défaut). ex: 'escape C-b' défini le préfixe à <C-b>.
exchangedown	échange la frame active avec celle du dessous.
exchangeleft	échange la frame active avec celle de gauche.
exchangeright	échange la frame active avec celle de droite.

exchangeup	échange la frame active avec celle du dessus.
exec command	exécute une commande shell. par défaut key="C-t !".
execa command	exécute une commande shell sans enregistrer depuis quelle frame. 🛠️Fix Me!
execf frame command	Execute a shell command and choose which frame the client's first window will open in. The client must be netwm compliant for this to work. 🛠️Fix Me!
getenv env	affiche la variable d'environnement, "env".
getsel	affiche la dernière sélection X11.
help	affiche la liste des raccourcis claviers.
license	affiche la licence ratpoison. par défaut key="C-t V".
meta key	"key" est optionnel. si omit, la commande simule un "Control-t" sur la fenêtre active. si "key" est défini, "key" est envoyé à la fenêtre active.
prompt prompt	affiche une invite de commande dans la fenêtre de messages.
putsel text	définir "text" comme sélection X11.
quit	quitter ratpoison.
ratinfo	affiche les coordonnées relatives du pointeur sur l'écran.
ratrelinfo	affiche les coordonnées relatives du pointeur sur la fenêtre ou la frame active.
ratrelwarp x y	déplace le pointeur selon x et y depuis sa position initiale.
ratwarp x y	déplace le pointeur selon x et y sur l'écran.
ratclick button	simule un clic. "button" peut être 1, 2, ou 3. par défaut: "1".
rathold state button	clic ou relache le "bouton" selon son état initial.
redisplay	redessine la fenêtre active.
restart	redémarre ratpoison (relecture du ratpoisonrc).
set var value	définir la valeur d'une variable ratpoison voici la liste des variables modifiables: framesels, winliststyle, barpadding, bgcolor, fgcolor, winname, winfmt, waitcursor, inputwidth, barborder, border, padding, font, bargravity, maxsizegravity, transgravity, wingravity, maxundos, resizeunit, historysize, historycompaction, historyexpansion.
setenv env value	définir la variable d'environnement "env" à "value".
source file	lire un fichier contenant des commandes ratpoison.
startup_message state	masque le message d'accueil: à mettre dans votre ~/.ratpoisonrc.
swap destination-frame source-frame	échange une fenêtre d'une frame avec une autre.
time	affiche l'heure dans la barre de messages.
unalias name	efface "name" e la liste des alias.
unsetenv env	rétablir la variable d'environnement "env" à sa valeur par défaut.
verbexec command	exécute la commande shell "command" en mode verbeux.
version	affiche la version de ratpoison. par défaut key="C-t v".
warp state	par défaut, ratpoison sauvegarde la position du pointeur sur une fenêtre et la rétablit au retour sur cette fenêtre. cette commande permet de désactiver cette option "state" peut être "on" ou "off".

Zone de saisie

ratpoison vous invite souvent dans la zone de saisie ou **input**: pour lancer une commande, choisir une fenêtre, d'finir une valeur...

voici les raccourcis autorisés dans la zone de saisie:

Keys	Action
<C-g> / <escape>	abandonne la saisie.
<C-f> / <right arrow>	avance d'un caractère.
<C-b> / <left arrow>	recule d'un caractère.
<M-f>	avance d'un mot.
<M-b>	recule d'un mot.
<C-a> / <home>	retourne en début de ligne.
<C-e> / <end>	aller en fin de ligne.
<C-d> / <delete>	efface le caractère suivant.
<M-d>	efface le mot suivant.
<backspace>	efface le caractère précédent.
<M-backspace>	efface le mot précédent.
<C-k>	efface jusqu'à la fin de la ligne.
<C-u>	efface depuis le début de ligne.
<C-y>	colle la sélection X11.
<C-p> / <up arrow>	retour dans l'historique des commandes.
<C-n> / <down arrow>	avancer dans l'historique des commandes.
<return>	soumettre la commande.
<tab>	fait appel à l'auto-complétion si possible.
<S-iso-lefttab>	idem "tab" mais en arrière.

toutes les entrées saisies sont stockées dans le fichier `~/.ratpoison_history` contenant 100 lignes par défaut. ce fichier est lu au démarrage de ratpoison comme l'historique bash de votre terminal.

Arguments en ligne de commande

ratpoison accepte une série d'argument qui peuvent être passés directement en ligne de commande.

Options	Action
-h, --help	affiche l'écran d'aide
-v, --version	affiche la version
-d, --display	définir le \$DISPLAY à utiliser.
-s, --screen	définir l'écran à utiliser. par défaut. par défaut, ratpoison s'affiche sur tous les écrans connectés et reconnus.
-c, --command	envoyer à ratpoison une 'colon-command' ("C-t :") ce qui permet de contrôler ratpoison depuis la ligne de commande. avec l'option "ratpoison -c", vous pouvez écrire des script intégré à ratpoison. certaines commandes affichent leur sortie dans le terminal au lieu de s'exécuter (windows, help). vous pouvez ainsi récupérer la sortie de la commande pour vos scripts. notez que vous pouvez utiliser l'option "-c" plusieurs fois. ex: "\$ ratpoison -c split -c split" divisera la frame deux fois.

<code>-i, --interactive</code>	force ratpoison à exécuter une commande en mode non-interactif.
<code>-f, --file</code>	d"finir un fichier de configuration différent. voir <u>.ratpoisonrc</u> .

ratpoisonrc

arrivé en fin de page, vous vous demandez certainement où se trouve le fichier de configuration de ratpoison. par défaut, ratpoison lit le fichier `/etc/ratpoisonrc`, mais vous pouvez créer le votre dans `~/.ratpoisonrc`. en fait, ratpoison n'en a pas vraiment besoin mais ce fichier permet de définir quelques variables, des alias et de définir certains programmes à lancer au démarrage de ratpoison.

notez que ce fichier est relu à chaque redémarrage de ratpoison "C-t :restart".

vous pouvez utiliser l'option "ratpoison -f" pour définir un fichier de configuration différent.

vous trouverez une multitude d'exemple de 'ratpoisonrc' sur le net, je vous livre le mien, plutôt minimaliste:

```
# ratpoison config and startup file
# set options and startup applications and commands
#####

## set some variables -----
# font to use on status message
set font "Droid Sans Mono-11"
# status text color
set fgcolor DeepSkyBlue4
# status background color
set bgcolor gray20
# border focus color
set fwcolor DeepSkyBlue4
# border unfocused color
set bwcolor gray20

## bind some keys -----
# dmenu with C-t d
bind d exec dmenu-bind.sh
# livarp ratmenu with C-t m
unbind m
bind m exec .ratpoison/rat_main_menu

## open a terminal on startup/restart -----
exec (sleep 4s && urxvtcd)
```

conclusion

ratpoison est un gestionnaire de fenêtres atypique. il peut se classer dans la catégorie des "tilings wms" car il occupe toute la place sur l'écran, mais son système de gestion des frames (contrairement aux "tags" de dwm, awesome ou spectrwm) reste unique.

je vous conseille de rester 2 ou 3 jours sur ratpoison pour réellement apprécier ses fonctionnalités, le temps de vous habituer aux raccourcis clavier :) sources : [wiki ratpoison](#)