

The REBOL Documentation Project

-- FR - Documentation REBOL - Divers --

Divers

Assouplissement des alias pour les équivalences symboliques

Philippe Le Goff

Première publication : 9 mai 2006, et mis en
ligne le mardi 9 mai 2006

Résumé :

Traduction de l'article 13 du blog Rebol 3.0, au sujet des alias

Assouplissement des alias pour les équivalences symboliques

Carl Sassenrath, CTO REBOL Technologies 28-Apr-2006 23:33 GMT Article #0013

En REBOL 3.0, nous allons assouplir les conditions relatives aux alias.

Rappel : un alias est un équivalent symbolique pour un mot. Normalement, les alias sont utilisés pour mettre en majuscules ou minuscules l'équivalent des mots (ce qui est fait en interne et automatiquement dans REBOL).

Par exemple, vous pouvez écrire :

```
if time > now [...]
```

ou

```
IF Time > now [...]
```

Ces expressions sont identiques. La comparaison renvoie "true" :

```
'if = 'IF true
```

Dans les versions précédentes, si vous vouliez définir un alias pour un mot, vous deviez y faire attention. Le mot ne devait pas être déjà utilisé (de n'importe quelle manière). Vous pouviez créer l'alias d'un mot avec une ligne comme celle-ci :

```
alias 'if "fi"
```

Après cela, fi signifie la même chose que if. L'expression suivante est vraie :

```
'if = 'fi true
```

et vous pouvez écrire :

```
fi time > now [...]
```

Les débutants doivent noter que l'usage d'alias n'est pas la même chose que le partage de la valeur des mots. L'exemple ci-dessus n'est pas la même chose que :

```
fi : if
```

Voir le manuel pour plus d'information sur cela.

Comme les experts le savent, les équivalences avec des alias sont rarement nécessaires, mais c'est important dans certains cas, quand les mots sont utilisés symboliquement, comme avec :

```
'if = first [fi]
```

ou dans un path :

path/fi

REBOL 2.0 possède des restrictions sur la façon dont les alias doivent être définis. Un alias peut être refusé si le mot a déjà été utilisé.

En REBOL 3.0, nous allons assouplir ces contraintes et vous pourrez écrire :

alias 'if 'fi

même si fi est déjà apparu auparavant.

Notez, cependant, que vous devrez encore utiliser les alias avec beaucoup d'attention. Si vous définissez des contextes (modules, objets, fonctions) qui utilisent le mot avant que ne soit défini son alias, eh bien... alors, vous risquez d'avoir quelques problèmes parce que vous aurez changé quelque chose d'extrêmement fondamental : le sens le plus basique du symbole.

Ce changement dans REBOL 3.0 n'est pas définitif. Nous allons observer comme cela fonctionne, et prévoir de revenir à la méthode de REBOL 2.0. Mais, je voulais pour le signaler. Faisons un essai.

(Traduction : Philippe Le Goff)