

HERCULES: Substitutions

Table of Contents

- [HERCULES: Substitutions](#)
 - [Substitutions de symboles](#)
 - [Symboles spéciaux](#)
 - [Symboles de groupe d'appareils](#)
 - [Symboles non définis](#)
 - [Substitutions de symboles améliorées](#)
 - [Syntaxe](#)

Substitutions de symboles

Dans les instructions de configuration et de périphérique, ainsi que dans les commandes du panneau et les fichiers OAT, des symboles peuvent être substitués au texte. Syntaxe

Pour substituer le symbole `symbole` à son contenu, le symbole doit être placé entre parenthèses et précédé du signe `$`. Par exemple, si le symbole `FOO` contient la chaîne de texte `"BAR"`, `$(FOO)` sera remplacé par la chaîne `"BAR"`; Les noms de symbole sont sensibles à la casse. Exemple

```
DEFSYM  TAPEDIR  "/home/hercules/tapes"  
...  
0380    3420    $(TAPEDIR)/scratch.aws  
...
```

Dans cet exemple, le périphérique 0380 sera changé en 3420 avec le fichier de bande AWS dans `/home/hercules/tapes/scratch.aws`

Symboles spéciaux

Symboles de groupe d'appareils

Lorsque plusieurs périphériques sont définis avec une seule déclaration de définition de périphérique, les symboles

CUU	(Numéro d'appareil à 3 chiffres, chiffres hexadécimaux en majuscule)
CCUU	(Numéro d'appareil à 4 chiffres, chiffres hexadécimaux en majuscule)
cuu	(Numéro d'appareil à 3 chiffres, chiffres hexadécimaux en minuscule)
ccuu	(Numéro d'appareil à 4 chiffres, chiffres hexadécimaux en minuscule)

sont définis pour contenir pour chaque périphérique l'adresse de périphérique correspondante.

Par exemple:

```
0200,0201 3340 /home/hercules/dasds/myvols.$(CUU)
```

définira deux packs 3340, le périphérique 0200 étant chargé avec le fichier myvols.200 et le périphérique 0201 défini avec myvols.201. Variables d'environnement

Si un symbole n'est pas explicitement défini par une instruction DEFSYM et qu'il existe une variable d'environnement du même nom, le contenu de la chaîne de cette variable d'environnement sera utilisé pour la substitution.

Symboles non définis

Si un symbole n'est pas défini par un DEFSYM explicite, n'est pas un symbole généré automatiquement et n'est pas une variable d'environnement, une chaîne vide sera remplacée. Échapper à la substitution, récursivité

Pour pouvoir spécifier la chaîne '\$ (' sans substitution, un signe '\$' supplémentaire doit être utilisé. Par exemple, \$\$ (FOO) ne sera pas substitué. Si une substitution est requise, le texte précédent doit contenir un Le signe '\$' est le dernier caractère, puis \$\$\$ (FOO) doit être spécifié. Ainsi, si le symbole FOO contient "BAR", \$\$ (FOO) restera "\$\$ (FOO)" pendant que \$\$\$ (FOO) deviendra "\$ BAR".

La substitution n'est pas récursive (une seule passe de substitution est effectuée).

Substitutions de symboles améliorées

La substitution de symbole améliorée diffère de la substitution de symbole supérieure à la normale de plusieurs manières très importantes:

- Tout d'abord, la syntaxe est différente. Les noms de symbole de substitution améliorés sont spécifiés avec \$ {var} (dollar + accolade) plutôt que \$ (var) (dollar + parenthèse).
- Deuxièmement, la syntaxe améliorée prend en charge la spécification d'une valeur par défaut à utiliser chaque fois que le symbole du nom n'est par ailleurs pas défini. La valeur par défaut est placée entre les accolades d'ouverture et de fermeture, exactement comme le nom du symbole, mais en est séparée par un seul signe égal '=' ou un signe deux-points égal ':='. Par exemple, en spécifiant "\$ {DASD_PATH=dasd/}" dans le fichier de configuration, on devra remplacer la valeur du symbole "DASD_PATH" ou utiliser la valeur "si la variable est indéfinie". dasd/ "à la place. Si aucune valeur par défaut n'est spécifiée, une chaîne vide est utilisée.
- Enfin, la substitution de symbole améliorée se produit uniquement à partir de variables d'environnement définies par l'hôte et non à partir de tout symbole DEFSYM portant le même nom, le cas échéant. Par exemple, si la variable d'environnement 'FOO' est définie avec la valeur "bar", l'instruction de fichier de configuration "DEFSYM FOO myfoo" suivie immédiatement de l'instruction "\$ {FOO}" entraîne le remplacement de la valeur "bar". 'myfoo', comme on pourrait le croire autrement, alors que l'instruction "\$ (FOO)", puisqu'il s'agit d'une séquence de substitution de symbole normale, est remplacée par "myfoo" (puisque'il s'agissait de la valeur qui lui avait été définie via l'instruction DEFSYM précédente).

En d'autres termes, chaque technique de substitution de symbole est prise en charge complètement séparément les unes des autres. DEFSYM permet de définir / annuler la définition / l'utilisation de symboles privés (définis en interne) distincts du pool de variables d'environnement du système d'exploitation hôte, tandis que la substitution de symboles améliorée ne permet pas et ne permet qu'un accès en lecture seule au pool de variables d'environnement de l'hôte sans prise en charge. modifier un symbole déjà défini (variable d'environnement), mais un moyen pratique et simple de définir une valeur par défaut à utiliser si la variable d'environnement hôte spécifiée est actuellement indéfinie.

Syntaxe

En outre les noms de symbole, en tant que noms de variables d'environnement, sont soumis à la sensibilité à la casse ou à l'insensibilité à la casse que le système d'exploitation hôte doit appliquer / autoriser. Sous Windows, les variables d'environnement ne sont pas sensibles à la casse, alors que sur d'autres systèmes d'exploitation, elles peuvent l'être. Ainsi, "\$ {FOO}", "\$ {foo}", "\$ {Foo}", etc. provoquent la substitution de la même valeur sous Windows, alors que les symboles DEFSYM \$ (FOO) et \$ (foo) sont deux symboles complètement différents et uniques pourraient être remplacés par deux valeurs complètement différentes (étant donné que DEFSYM est sensible à la casse sur toutes les plateformes prises en charge, y compris Windows).

Pour substituer le symbole symbole à la valeur de la variable d'environnement actuelle, le symbole doit être placé entre accolades et précédé du signe \$. Par exemple, si une variable d'environnement nommée FOO contient la valeur "BAR", \$ {FOO} sera alors remplacé par la chaîne "BAR". Si la variable d'environnement "FOO" n'est pas définie, une chaîne nulle (vide) est remplacée.

Si la chaîne "\$ {FOO: = myfoo}" est utilisée à la place, la valeur "BAR" sera toujours remplacée si la valeur "BAR" a bien été précédemment attribuée à FOO, mais sera remplacée par la valeur "myfoo" si la variable d'environnement FOO est actuellement indéfinie.

La valeur par défaut est une chaîne littérale et qu'aucune substitution ne lui est appliquée. Tenter ainsi d'utiliser la syntaxe "\$ {foo = \$ {bar}}" ne donnera pas les résultats attendus. Il ne sera pas remplacé par la valeur actuellement définie de la variable d'environnement "bar", mais sera toujours remplacé par la chaîne littérale "\$ {bar}" suivie immédiatement du caractère littéral '}'.

Les noms de symboles (noms de variables d'environnement) ne sont pas sensibles à la casse sous Windows, alors qu'ils peuvent figurer sur d'autres systèmes d'exploitation hôtes.

From:
<http://www.ouarte.garden/> - **dwndoc**

Permanent link:
<http://www.ouarte.garden/doku.php?id=ibm:hercules-390-conf-substitution>

Last update: **2025/02/19 10:59**

