
Windows 10 - Variables d'environnement

BTS SIO - Bloc 1 - Support et mise à disposition des services informatiques

U4 - 1.2 Répondre aux incidents et aux demandes d'assistance et d'évolution

1. Qu'est-ce qu'une variable d'environnement	3
1.1. Sous Windows	3
1.1.1. Liste non exhaustive des principales variables d'environnement sur les systèmes Windows	3
1.1.2. La commande set	5
1.2. Sous Unix	5
1.2.1. Liste non exhaustive des principales variables d'environnement sur les systèmes Unix	6

1. Qu'est-ce qu'une variable d'environnement

Une variable d'environnement est une valeur dynamique, chargée en mémoire, pouvant être utilisée par plusieurs processus fonctionnant simultanément. Sur la plupart des systèmes d'exploitation, les emplacements de certaines librairies, voire des principaux exécutables du système peuvent avoir un emplacement différent selon l'installation.

Ainsi, grâce aux variables d'environnement, il est possible, à partir d'un programme, de faire référence à un emplacement en s'appuyant sur les variables d'environnement définissant ces données.

1.1. Sous Windows

Sous Windows, les variables d'environnement sont entourées du caractère « % ». Ainsi, pour afficher la valeur d'une variable d'environnement, il suffit de taper une commande du type :

```
echo %NOM_DE_LA_VARIABLE%
```

Sous Windows, on distingue les variables globales des variables locales, valables uniquement pour l'utilisateur courant.

Pour accéder à la liste des variables d'environnement, faites un clic droit sur le Poste de travail puis choisissez Propriétés -> Paramètres système avancés -> Variables d'environnement.

1.1.1. Liste non exhaustive des principales variables d'environnement sur les systèmes Windows

Variable d'environnement	Description
%APPDATA%	Retourne un chemin d'accès au répertoire par défaut contenant les programmes utilisateur.
%CMDCMDLINE%	Retourne la commande exacte utilisée pour accéder à l'interpréteur de commandes (<i>cmd.exe</i>).
%COMPUTERNAME%	Retourne le nom attribué à l'ordinateur.
%DATE%	Retourne la date actuelle.
%ERRORLEVEL%	Contient le code d'erreur de la dernière commande utilisée.
%HOMEDRIVE%	Contient la lettre de lecteur sur laquelle le répertoire de l'utilisateur courant est situé.
%HOMEPATH%	Contient le chemin d'accès complet au répertoire de l'utilisateur courant.

%NUMBER_OF_PROCESSOR%	Contient le nombre de processeurs présents sur le système.
%OS%	Contient une chaîne de caractères décrivant le système d'exploitation installé. Noter que Windows 2000 se déclare comme étant <i>WINDOWS_NT</i> .
%PATH%	Spécifie le chemin d'accès vers les principaux exécutables du système.
%PATHEXT%	Décrit les extensions que le système considère comme exécutables.
%PROCESSOR_ARCHITECTURE%	Décrit l'architecture du processeur (x86 pour les machines Intel, IA64 pour les architectures RISC).
%RANDOM%	Retourne un entier compris entre 0 et 32167 choisi aléatoirement par le système.
%SYSTEMDRIVE%	Contient la lettre de lecteur sur laquelle le système est situé (généralement C:).
%SYSTEMROOT%	Contient le chemin d'accès au répertoire racine du système.
%TEMP%	Contient le chemin d'accès du répertoire temporaire pour les applications.
%TMP%	Contient le chemin d'accès du répertoire temporaire pour les applications.
%TIME%	Retourne l'heure actuelle.
%USERDOMAIN%	Contient le domain auquel appartient le compte courant.
%USERNAME%	Contient le nom d'utilisateur correspondant au compte courant.
%USERPROFILE%	Contient l'emplacement du profil utilisateur du compte courant.
%WINDIR%	Contient le chemin d'accès au répertoire du système (généralement <i>C:WINDOWS</i> ou <i>C:WINNT</i>).

1.1.2. La commande set

Sous Windows, la commande set permet de créer, de modifier et d'afficher les variables d'environnement. Néanmoins, les modifications interviennent uniquement dans la session en cours.

Pour afficher une variable, la commande est la suivante :

```
set NomVariable
```

Pour créer une variable, la commande est la suivante :

```
set NomVariable=valeur
```

Pour supprimer une variable, la commande est la suivante :

```
set NomVariable=
```

Il est possible d'utiliser n'importe quel caractère comme valeur, y compris des espaces. Néanmoins, pour pouvoir utiliser des caractères spéciaux (<, >, |, & ou ^), il suffit de les faire précéder du caractère d'échappement (^) ou bien les entourer de guillemets, qui feront alors partie de la valeur car tous les caractères après le signe = sont pris en compte. Par exemple, pour définir « noir&blanc » comme valeur :

```
set couleur=noir^&blanc
```

ou bien

```
set varname="new&name"
```

1.2. Sous Unix

Sous les systèmes UNIX, les variables d'environnement sont précédées du caractère « \$ » et encadrées par des accolades, même si les accolades ne sont pas obligatoires. Compte tenu de la variété des systèmes de type UNIX (en particulier les différentes distributions de Linux), les variables d'environnement. Ainsi, pour afficher la valeur d'une variable d'environnement, il suffit de taper une des commandes suivantes :

```
echo $VARIABLE
```

ou bien

```
echo ${VARIABLE}
```

1.2.1. Liste non exhaustive des principales variables d'environnement sur les systèmes Unix

Variable d'environnement	Description
\$ARCH	Contient la description de l'architecture de la machine.
\$DISPLAY	Contient l'identifiant du terminal d'affichage à utiliser dans le gestionnaire de fenêtres (X11).
\$HOME	Retourne le chemin d'accès vers le répertoire de l'utilisateur courant.
\$HOST	Retourne le nom de l'ordinateur.
\$LANG	Retourne le code de langue par défaut.
\$PATH	Retourne une liste de chemins d'accès vers des répertoires contenant les exécutables, séparés par des points-virgules.
\$PRINTER	Contient le nom de l'imprimante par défaut.
\$SHELL	Indique le chemin de l'interpréteur de commande utilisé.
\$USER	Retourne l'identifiant de l'utilisateur courant.