

- [NOM](#)
- [SYNOPSIS](#)
- [DESCRIPTION](#)
- [OPTIONS](#)
- [MODE MAC](#)
- [MODES DE CONVERSION](#)
- [UNICODE](#)
 - [Codages](#)
 - [Conversion](#)
 - [Marque d'ordre des octets](#)
 - [Noms de fichiers unicode sous Windows](#)
 - [Exemples Unicode](#)
- [GB18030](#)
- [EXEMPLES](#)
- [CONVERSIONS RÉCURSIVES](#)
- [PARAMÈTRES LINGUISTIQUES](#)
- [VALEUR DE RETOUR](#)
- [STANDARDS](#)
- [AUTEURS](#)
- [VOIR AUSSI](#)

NOM

dos2unix - Convertit les fichiers textes du format DOS/Mac vers Unix et inversement

SYNOPSIS

```
dos2unix [options] [FICHIER ...] [-n FICHIER_ENTRÉE FICHIER_SORTIE ...]  
unix2dos [options] [FICHIER ...] [-n FICHIER_ENTRÉE FICHIER_SORTIE ...]
```

DESCRIPTION

Le package Dos2unix inclut les utilitaires dos2unix et unix2dos pour convertir des fichiers textes au format DOS ou Mac vers le format Unix et inversement.

Dans les fichiers textes DOS/Windows, un saut de ligne est une combinaison de deux caractères: un retour de chariot (CR) suivi d'un saut de ligne (LF). Dans les fichiers textes Unix, le saut de ligne est un seul caractère: le saut de ligne (LF). Les fichiers textes Mac, avant Mac OS X, utilisaient le retour chariot (CR) comme seul caractère. De nos jours, Mac OS utilise le même style de saut de ligne que Unix (LF).

Outre les sauts de lignes, Dos2unix convertit aussi le codage des fichiers. Quelques codes page DOS peuvent être convertis en Latin-1 sous Unix. L'Unicode des fichiers Windows (UTF-16) peut être converti en Unicode Unix (UTF-8).

Les fichiers binaires sont automatiquement ignorés à moins que la conversion soit forcée.

Les fichiers non réguliers tels que les répertoires et les FIFOs sont automatiquement ignorés.

Les liens symboliques et leur cible sont, par défaut, inchangés. En option, les liens symboliques peuvent être remplacés ou, au choix, la sortie peut être écrite dans la cible du lien symbolique. Écrire dans la cible

d'un lien symbolique n'est pas supporté sous Windows.

Dos2unix a été conçu comme dos2unix sous SunOS/Solaris. Il y a une différence importante avec la version originale de SunOS/Solaris. Cette version effectue les conversions en place (ancien mode de fichier) tandis que la version originale de SunOS/Solaris ne supporte que la conversion par paire (nouveau mode de fichier). Voyez aussi les options `-o` et `-n`. Une autre différence est que SunOS/Solaris utilise par défaut le mode de conversion *iso* tandis que cette version utilise par défaut le mode de conversion *ascii*.

OPTIONS

--

Traitez toutes les options à sa suite comme étant des noms de fichiers. Utilisez cette option si vous voulez convertir des fichiers dont le nom commence par un tiret. Par exemple, pour convertir un fichier nommé « -foo », vous pouvez utiliser cette commande:

```
dos2unix -- -foo
```

Ou dans le style des nouveaux fichiers:

```
dos2unix -n -- -foo sortie.txt
```

--allow-chown

Autoriser le changement de propriétaire dans l'ancien mode de fichier.

Quand cette option est utilisée, la conversion n'est pas interrompue si l'utilisateur ou le groupe propriétaire du fichier original ne peut pas être préservé dans l'ancien mode de fichier. La conversion continuera et le fichier converti aura le même nouveau propriétaire que si il avait été converti par le nouveau mode de fichier. Voyez aussi les options `-o` et `-n`. Cette option est uniquement disponible si dos2unix dispose des fonctionnalités pour préserver l'utilisateur ou le groupe propriétaire des fichiers.

-ascii

Convertit uniquement les sauts de lignes. C'est le mode de conversion par défaut.

-iso

Convertit le jeu de caractères du DOS vers ISO-8859-1. Voyez aussi la section des MODES DE CONVERSION.

-1252

Utilise le code page 1252 de Windows (Europe de l'ouest).

-437

Utilise le code page 437 du DOS (US). C'est le code page par défaut pour les conversions ISO.

-850

Utilise le code page 850 du DOS (Europe de l'ouest).

-860

Utilise le code page 860 du DOS (portugais).

-863

Utilise le code page 863 du DOS (français canadien).

-865

Utilise le code page 865 du DOS (nordique).

-7

Convertit les caractères 8 bits vers l'espace 7 bits.

-b, --keep-bom

Conserve la marque d'ordre des octets (BOM). Si le fichier d'entrée a une BOM, elle est écrite dans le fichier de sortie. C'est le comportement par défaut quand les sauts de lignes sont convertis au format DOS. Consultez aussi l'option `-r`.

-c, --convmode MODE_CONV

Change le mode de conversion. `MODE_CONV` prend l'une des valeurs: *ascii*, *7bit*, *iso*, *mac*. *Ascii* est la valeur par défaut.

-D, --display-enc ENCODAGE

Choisi l'encodage des textes affichés. L'`ENCODAGE` peut être : *ansi*, *unicode*, *unicodebom*, *utf8*, *utf8bom*. La valeur par défaut est *ansi*.

Cette option est uniquement disponible dans dos2unix pour Windows avec support pour les noms de fichiers Unicode. Cette option n'a aucun effet sur les noms de fichiers lus et écrits. Son effet se limite à leur affichage.

Il existe plusieurs méthodes pour afficher du texte dans une console Windows selon l'encodage du texte. Elles ont toutes leurs propres avantages et désavantages.

ansi

La méthode par défaut de dos2unix est d'utiliser du texte encodé en ANSI. Elle a l'avantage d'être rétro compatible. Elle fonctionne avec des polices raster ou TrueType. Dans certaines régions, vous pouvez avoir besoin d'utiliser la commande `chcp` pour remplacer le code page DOS OEM actif par le code ANSI système de Windows car dos2unix utilise le code page système de Windows.

Le désavantage de *ansi* est que les noms de fichiers internationaux avec des caractères en dehors du code page système par défaut ne sont pas affichés correctement. Vous verrez un point d'interrogation ou un mauvais symbole à leur place. Cette méthode est acceptable si vous ne travaillez pas avec des noms de fichiers étrangers.

unicode, unicodebom

L'avantage de l'encodage *unicode* (le nom de Windows pour UTF-16) est que le texte est habituellement affiché correctement. Il n'est pas nécessaire de changer le code page actif. Vous pouvez avoir besoin de remplacer la police de la console par une police TrueType pour afficher les caractères internationaux correctement. Lorsqu'un caractère n'est pas inclus dans la police TrueType, il sera généralement remplacé par un petit carré, parfois avec un point d'interrogation à l'intérieur.

Lorsque vous utilisez la console ConEmu, les textes sont affichés correctement car ConEmu sélectionne automatiquement une bonne police.

Le désavantage de unicode est qu'il n'est pas compatible avec ASCII. La sortie n'est pas facile à gérer quand vous la redirigez vers un autre programme.

Quand la méthode `unicodebom` est utilisée, le texte Unicode est précédé d'une BOM (Byte Order Mark=marque d'ordre des octets). Une BOM est nécessaire pour la redirection correcte ou le pipelining dans PowerShell.

utf8, utf8bom

L'avantage de `utf8` est qu'il est compatible avec ASCII. Vous devez utiliser une police TrueType dans la console. Avec une police TrueType, le texte est affiché comme avec un encodage unicode.

Le désavantage est que, si vous utilisez la police raster par défaut, tous les caractères non ASCII sont mal affichés. Pas uniquement les noms de fichiers unicode ! Les messages traduits deviennent inintelligibles. Sous Windows configuré pour une région de l'est de l'Asie, vous pouvez observer énormément de scintillements dans la console quand des messages sont affichés.

Dans une console ConEmu, l'encodage `utf8` fonctionne bien.

Quand la méthode `utf8bom` est utilisée, le texte UTF-8 est précédé d'une BOM (Byte Order Mark=marque d'ordre des octets). Une BOM est nécessaire pour la redirection correcte ou le pipelining dans PowerShell.

L'encodage par défaut peut être changé en assignant la valeur `unicode`, `unicodebom`, `utf8` ou `utf8bom` à la variable d'environnement `DOS2UNIX_DISPLAY_ENC`.

-e, --add-eol

Ajoute un saut de ligne à la dernière ligne si elle n'en a pas déjà un. Cela fonctionne avec toutes les conversions.

Un fichier converti de DOS vers le format Unix peut ne pas avoir de saut de ligne à la dernière ligne. Il existe des éditeurs de texte qui écrivent le fichier texte sans saut de ligne à la dernière ligne. Certains programmes Unix ont des difficultés à traiter ces fichiers car le standard POSIX définit que chaque ligne d'un fichier texte doit être terminée par le caractère de nouvelle ligne. Par exemple, concaténer des fichiers peut ne pas donner le résultat attendu.

-f, --force

Force la conversion de fichiers binaires.

-gb, --gb18030

Sous Windows, les fichiers UTF-16 sont convertis en UTF-8 par défaut sans considération pour les paramètres de la localisation. Utilisez cette option pour convertir UTF-16 en GB18030. Cette option n'est disponible que sous Windows. Consultez aussi la section GB18030.

-h, --help

Affiche l'aide et s'arrête.

-i[FANIONS], --info[=FANIONS] FICHER ...

Affiche les informations du fichier. Aucune conversion n'est réalisée.

Les informations suivantes sont affichées dans cet ordre: le nombre de sauts de ligne DOS, le nombre de sauts de ligne Unix, le nombre de sauts de ligne Mac, la marque d'ordre des octets, texte ou binaire, nom du fichier.

Exemple de sortie :

```

6      0      0  no_bom   text   dos.txt
0      6      0  no_bom   text   unix.txt
0      0      6  no_bom   text   mac.txt
6      6      6  no_bom   text   mixed.txt
50     0      0  UTF-16LE text   utf16le.txt
0      50     0  no_bom   text   utf8unix.txt
50     0      0  UTF-8    text   utf8dos.txt
2      418    219 no_bom   binary dos2unix.exe
```

Notez qu'un fichier binaire peut parfois être considéré à tort comme un fichier texte. Voyez aussi l'option -s.

Des fanions facultatifs peuvent être ajoutés pour changer la sortie. Un ou plusieurs fanions peuvent être ajoutés.

0

Afficher les lignes d'information du fichier suivies d'un caractère nul au lieu d'un saut de ligne. Cela permet d'interpréter correctement les noms de fichiers avec des espaces ou des guillemets quand le fanion c est utilisé. Utilisez ce fanion avec les options -0 ou --null de xargs(1).

d

Affiche le nombre de sauts de ligne DOS.

u

Affiche le nombre de sauts de ligne Unix.

m

Affiche le nombre de sauts de ligne Mac.

b

Afficher la marque d'ordre des octets.

t

Affiche si le fichier est texte ou binaire.

c

Affiche uniquement les fichiers qui seraient convertis.

Avec le fanion c, dos2unix n'affichera que les fichiers contenant des sauts de ligne DOS alors que unix2dos n'affichera que les noms des fichiers aillant des sauts de ligne Unix.

h

Afficher un en-tête.

p

Montrer les noms des fichiers sans le chemin.

Exemples:

Afficher les informations pour tous les fichier *.txt :

```
dos2unix -i *.txt
```

Afficher uniquement le nombre de sauts de ligne DOS et Unix :

```
dos2unix -idu *.txt
```

Montrer uniquement la marque d'ordre des octets :

```
dos2unix --info=b *.txt
```

Liste les fichiers qui ont des sauts de ligne DOS :

```
dos2unix -ic *.txt
```

Liste les fichiers qui ont des sauts de ligne Unix :

```
unix2dos -ic *.txt
```

Ne converti que les fichiers qui ont des sauts de lignes DOS et laisse les autres fichiers inchangés:

```
dos2unix -ic0 *.txt | xargs -0 dos2unix
```

Trouve les fichiers texte qui ont des sauts de ligne DOS :

```
find -name '*.txt' -print0 | xargs -0 dos2unix -ic
```

-k, --keepdate

La date du fichier de sortie est la même que celle du fichier d'entrée.

-L, --license

Affiche la licence du programme.

-l, --newline

Ajoute des sauts de lignes additionnels.

dos2unix: Seuls les sauts de lignes du DOS sont changés en deux sauts de lignes de Unix. En mode Mac, seuls les sauts de lignes Mac sont changés en deux sauts de lignes Unix.

unix2dos: Seuls les sauts de lignes Unix sont changés en deux sauts de lignes du DOS. En mode Mac, les sauts de lignes Unix sont remplacés par deux sauts de lignes Mac.

-m, --add-bom

Écrit une marque d'ordre des octets (BOM) dans le fichier de sortie. Par défaut une BOM UTF-8 est écrite.

Lorsque le fichier d'entrée est en UTF-16 et que l'option -u est utilisée, une BOM UTF-16 est écrite.

N'utilisez jamais cette option quand l'encodage du fichier de sortie n'est ni UTF-8 ni UTF-16 ni GB18030. Consultez également la section UNICODE.

-n, --newfile FICHIER_ENTRÉE FICHIER_SORTIE ...

Nouveau mode de fichiers. Convertit le fichier FICHIER_ENTRÉE et écrit la sortie dans le fichier FICHIER_SORTIE. Les noms des fichiers doivent être indiqués par paires. Les caractères de remplacement *ne* doivent *pas* être utilisés ou vous *perdrez* vos fichiers.

La personne qui démarre la conversion dans le nouveau mode (paire) des fichiers sera le propriétaire du fichier converti. Les permissions de lecture/écriture du nouveau fichier seront les permissions du fichier original moins le umask(1) de la personne qui exécute la conversion.

--no-allow-chown

Ne pas autoriser le changement du propriétaire du fichier dans l'ancien mode de fichier (par défaut).

Interrompt la conversion si l'utilisateur ou le groupe propriétaire du fichier original ne peuvent pas être préservés dans l'ancien mode de fichier. Voyez aussi les options -o et -n. Cette option est uniquement présente si dos2unix dispose des fonctionnalités pour préserver l'utilisateur ou le groupe propriétaire des fichiers.

--no-add-eol

N'ajoute pas de saut de ligne à la dernière ligne s'il n'y en a pas.

-O, --to-stdout

Écrit vers la sortie standard, comme un filtre Unix. Utilisez l'option -o pour revenir au mode de l'ancien fichier (en place).

Combiné avec l'option -e, les fichiers peuvent être concaténés correctement. Les première et dernière lignes ne sont pas fusionnées et il n'y a pas de marque d'ordre des octets au milieu du fichier concaténé. Exemple :

```
dos2unix -e -O fichier1.txt fichier2.txt > sortie.txt
```

-o, --oldfile FICHIER ...

Ancien mode de fichiers. Convertit le fichier FICHIER et écrit la sortie dedans. Le programme fonctionne dans ce mode par défaut. Les noms avec des caractères de remplacement peuvent être utilisés.

Dans l'ancien mode (en place) des fichiers, les fichiers convertis ont le même propriétaire, groupe et permissions lecture/écriture que le fichier original. Idem quand le fichier est converti par un utilisateur qui a la permission d'écrire dans le fichier (par exemple, root). La conversion est interrompue si il n'est pas possible de conserver les valeurs d'origine. Le changement de propriétaire pourrait signifier que le propriétaire original n'est plus en mesure de lire le fichier. Le changement de groupe pourrait être un risque pour la sécurité. Le fichier pourrait être rendu accessible en lecture par des personnes à qui il n'est pas destiné. La conservation du propriétaire, du groupe et des permissions de lecture/écriture n'est supportée que sous Unix.

Pour vérifier si dos2unix dispose des fonctions pour préserver l'utilisateur et le groupe propriétaire du fichier, tapez `dos2unix -v`.

La conversion est toujours réalisée via un fichier temporaire. Quand une erreur survient au milieu de la conversion, le fichier temporaire est effacé et le fichier original reste inchangé. Quand la

conversion réussi, le fichier original est remplacé par le fichier temporaire. Vous pourriez avoir la permission d'écrire dans le fichier original mais ne pas avoir la permission de remplacer les propriétés de l'utilisateur et du groupe propriétaires sur le fichier temporaire telles qu'elles sont définies sur le fichier original. Cela signifie que vous n'êtes pas en mesure de préserver l'utilisateur ou le groupe propriétaire du fichier original. Dans ce cas, vous pouvez utiliser l'option `--allow-chown` pour continuer la conversion.

```
dos2unix --allow-chown toto.txt
```

Une autre option consiste à utiliser le nouveau mode de fichier:

```
dos2unix -n toto.txt toto.txt
```

L'avantage de l'option `--allow-chown` est que vous pouvez utiliser des caractères de remplacement et les propriétaires seront préservés dans la mesure du possible.

-q, --quiet

Mode silencieux. Supprime les avertissements et les messages. La valeur de sortie est zéro sauf quand de mauvaises options sont utilisées sur la ligne de commande.

-r, --remove-bom

Supprime la marque d'ordre des octets (BOM). N'écrit pas la BOM dans le fichier de sortie. Ceci est le comportement par défaut lorsque les sauts de lignes sont convertis au format Unix. Consultez aussi l'option `-b`.

-s, --safe

Ignore les fichiers binaires (par défaut).

Ignorer les fichiers binaires sert à éviter les erreurs accidentelles. Attention que la détection de fichiers binaires n'est pas fiable à 100%. Les fichiers en entrée sont analysés pour y trouver des symboles binaires qui ne sont habituellement pas rencontrés dans des fichiers textes. Il est cependant possible qu'un fichier binaire ne contienne que des caractères textes normaux. Un tel fichier serait erronément traité comme un fichier texte.

-u, --keep-utf16

Conserve l'encodage UTF-16 original du fichier d'entrée. Le fichier de sortie sera écrit dans le même encodage UTF-16 (petit ou grand boutien) que le fichier d'entrée. Ceci évite la transformation en UTF-8. Une BOM UTF-16 sera écrite en conséquent. Cette option peut être désactivée avec l'option `-ascii`.

-ul, --assume-utf16le

Suppose que le fichier d'entrée est au format UTF-16LE.

Quand il y a un indicateur d'ordre des octets dans le fichier d'entrée, l'indicateur a priorité sur cette option.

Si vous vous êtes trompé sur le format du fichier d'entrée (par exemple, ce n'était pas un fichier UTF16-LE) et que la conversion réussi, vous obtiendrez un fichier UTF-8 contenant le mauvais texte. Vous pouvez récupérer le fichier original avec `iconv(1)` en convertissant le fichier de sortie UTF-8 vers du UTF-16LE.

La présupposition de l'UTF-16LE fonctionne comme un *mode de conversion*. En utilisant le mode

ascii par défaut, UTF-16LE n'est plus présupposé.

-ub, --assume-utf16be

Suppose que le fichier d'entrée est au format UTF-16BE.

Cette option fonctionne comme l'option `-u1`.

-v, --verbose

Affiche des messages verbeux. Des informations supplémentaires sont affichées à propos des marques d'ordre des octets et du nombre de sauts de lignes convertis.

-F, --follow-symlink

Suit les liens symboliques et convertit les cibles.

-R, --replace-symlink

Remplace les liens symboliques par les fichiers convertis (les fichiers cibles originaux restent inchangés).

-S, --skip-symlink

Ne change pas les liens symboliques ni les cibles (par défaut).

-V, --version

Affiche les informations de version puis arrête.

MODE MAC

En mode normal, les sauts de lignes sont convertis du DOS vers Unix et inversement. Les sauts de lignes Mac ne sont pas convertis.

En mode Mac, les sauts de lignes sont convertis du format Mac au format Unix et inversement. Les sauts de lignes DOS ne sont pas changés.

Pour fonctionner en mode Mac, utilisez l'option en ligne de commande `-c mac` ou utilisez les commandes `mac2unix` ou `unix2mac`.

MODES DE CONVERSION

ascii

En mode *ascii*, seuls les sauts de lignes sont convertis. Ceci est le mode de conversion par défaut.

Bien que le nom de ce mode soit ASCII, qui est un standard 7 bits, ce mode travail en réalité sur 8 bits. Utilisez toujours ce mode lorsque vous convertissez des fichiers Unicode UTF-8.

7bit

Dans ce mode, tous les caractères 8 bits non ASCII (avec des valeurs entre 128 et 255) sont remplacés par une espace 7 bits.

iso

Les caractères sont convertis entre un jeu de caractères DOS (code page) et le jeu de caractères ISO-8859-1 (Latin-1) de Unix. Les caractères DOS sans équivalent ISO-8859-1, pour lesquels la conversion n'est pas possible, sont remplacés par un point. La même chose est valable pour les caractères ISO-8859-1 sans équivalent DOS.

Quand seule l'option `-iso` est utilisée, `dos2unix` essaie de déterminer le code page actif. Quand ce n'est pas possible, `dos2unix` utilise le code page CP437 par défaut qui est surtout utilisé aux USA. Pour forcer l'utilisation d'un code page spécifique, utilisez les options `-437` (US), `-850` (Europe de l'ouest), `-860` (portugais), `-863` (français canadien) ou `-865` (nordique). Le code page CP1252 de Windows (Europe de l'ouest) est également supporté avec l'option `-1252`. Pour d'autres codes pages, utilisez `dos2unix` avec `iconv(1)`. `Iconv` supporte une longue liste de codages de caractères.

N'utilisez jamais la conversion ISO sur des fichiers textes Unicode. Cela va corrompre les fichiers encodés en UTF-8.

Quelques exemples:

Convertir du code page par défaut du DOS au Latin-1 Unix :

```
dos2unix -iso -n entrée.txt sortie.txt
```

Convertir du CP850 du DOS au Latin-1 Unix :

```
dos2unix -850 -n entrée.txt sortie.txt
```

Convertir du CP1252 de Windows au Latin-1 de Unix :

```
dos2unix -1252 -n entrée.txt sortie.txt
```

Convertir le CP1252 de Windows en UTF-8 de Unix (Unicode) :

```
iconv -f CP1252 -t UTF-8 entrée.txt | dos2unix > sortie.txt
```

Convertir du Latin-1 de Unix au code page par défaut de DOS :

```
unix2dos -iso -n entrée.txt sortie.txt
```

Convertir le Latin-1 de Unix en CP850 du DOS :

```
unix2dos -850 -n entrée.txt sortie.txt
```

Convertir le Latin-1 de Unix en CP1252 de Windows :

```
unix2dos -1252 -n entrée.txt sortie.txt
```

Convertir le UTF-8 de Unix (Unicode) en CP1252 de Windows :

```
unix2dos < entrée.txt | iconv -f UTF-8 -t CP1252 > sortie.txt
```

Consultez aussi <http://czyborra.com/charsets/codepages.html> et <http://czyborra.com/charsets/iso8859.html>.

UNICODE

Codages

Il existe plusieurs codages Unicode. Sous Unix et Linux, les fichiers sont généralement codés en UTF-8. Sous Windows, les fichiers textes Unicode peuvent être codés en UTF-8, UTF-16 ou UTF-16 gros boutien mais ils sont majoritairement codés au format UTF-16.

Conversion

Les fichiers textes Unicode peuvent avoir des sauts de lignes DOS, Unix ou Mac, tout comme les fichiers textes normaux.

Toutes les versions de dos2unix et unix2dos peuvent convertir des fichiers codés en UTF-8 car UTF-8 a été conçu pour être rétro-compatible avec l'ASCII.

Dos2unix et unix2dos, avec le support pour l'Unicode UTF-16, peuvent lire les fichiers textes codés sous forme petit boutien ou gros boutien. Pour savoir si dos2unix a été compilé avec le support UTF-16 tapez `dos2unix -V`.

Sous Unix/Linux, les fichiers encodés en UTF-16 sont convertis vers l'encodage des caractères de la localisation. Utilisez `locale(1)` pour découvrir quel encodage de caractères est utilisé. Lorsque la conversion n'est pas possible, une erreur de conversion est produite et le fichier est abandonné.

Sous Windows, les fichiers UTF-16 sont convertis par défaut en UTF-8. Les fichiers textes formatés en UTF-8 sont bien supportés sous Windows et Unix/Linux.

Les codages UTF-16 et UTF-8 sont parfaitement compatibles. Il n'y a pas de pertes lors de la conversion. Lorsqu'une erreur de conversion UTF-16 vers UTF-8 survient, par exemple, quand le fichier d'entrée UTF-16 contient une erreur, le fichier est ignoré.

Quand l'option `-u` est utilisée, le fichier de sortie est écrit dans le même encodage UTF-16 que le fichier d'entrée. L'option `-u` empêche la conversion en UTF-8.

Dos2unix et unix2dos n'ont pas d'option pour convertir des fichiers UTF-8 en UTF-16.

Les modes de conversion ISO et 7 bits ne fonctionnent pas sur des fichiers UTF-16.

Marque d'ordre des octets

Les fichiers textes Unicode sous Windows ont généralement un indicateur d'ordre des octets (BOM) car de nombreux programmes Windows (y compris Notepad) ajoutent cet indicateur par défaut. Consultez aussi http://fr.wikipedia.org/wiki/Indicateur_d'ordre_des_octets.

Sous Unix, les fichiers Unicodes n'ont habituellement pas de BOM. Il est supposé que les fichiers textes sont codés selon le codage de l'environnement linguistique.

Dos2unix ne peut détecter que le fichier est au format UTF-16 si le fichier n'a pas de BOM. Quand le fichier UTF-16 n'a pas cet indicateur, dos2unix voit le fichier comme un fichier binaire.

Utilisez l'option `-u1` ou `-ub` pour convertir un fichier UTF-16 sans BOM.

Dos2unix, par défaut, n'écrit pas de BOM dans le fichier de sortie. Avec l'option `-b`, Dos2unix écrit une BOM quand le fichier d'entrée a une BOM.

Unix2dos écrit par défaut une BOM dans le fichier de sortie quand le fichier d'entrée a une BOM. Utilisez l'option `-r` pour supprimer la BOM.

Dos2unix et unix2dos écrivent toujours une BOM quand l'option `-m` est utilisée.

Noms de fichiers unicode sous Windows

Dos2unix supporte, en option, la lecture et l'écriture de noms de fichiers Unicode dans la ligne de commande de Windows. Cela signifie que dos2unix peut ouvrir des fichiers qui ont, dans leur nom, des caractères n'appartenant pas au code page système ANSI par défaut. Pour voir si dos2unix pour Windows a été compilé avec le support des noms de fichiers Unicode, tapez `dos2unix -v`.

Il y a quelques soucis avec l'affichage de noms de fichiers Unicode dans une console Windows. Voyez l'option `-D`, `--display-enc`. Les noms de fichiers peuvent être mal affichés dans la console mais les fichiers seront écrits avec les bons noms.

Exemples Unicode

Convertir de l'UTF-16 Windows (avec BOM) vers l'UTF-8 de Unix :

```
dos2unix -n entrée.txt sortie.txt
```

Convertir de l'UTF-16LE de Windows (sans BOM) vers l'UTF-8 de Unix :

```
dos2unix -ul -n entrée.txt sortie.txt
```

Convertir de l'UTF-8 de Unix vers l'UTF-8 de Windows avec BOM :

```
unix2dos -m -n entrée.txt sortie.txt
```

Convertir de l'UTF-8 de Unix vers l'UTF-16 de Windows :

```
unix2dos < entrée.txt | iconv -f UTF-8 -t UTF-16 > sortie.txt
```

GB18030

GB18030 est un standard du gouvernement chinois. Tout logiciel vendu en Chine doit officiellement supporter un sous ensemble obligatoire du standard GB18030. Consultez http://fr.wikipedia.org/wiki/GB_18030.

GB18030 est entièrement compatible avec Unicode et peut être considéré comme étant un format de transformation unicode. Comme UTF-8, GB18030 est compatible avec ASCII. GB18030 est aussi compatible avec le code page 936 de Windows aussi connu comme GBK.

Sous Unix/Linux, les fichiers UTF-16 sont convertis en GB18030 quand l'encodage de l'environnement linguistique est GB18030. Notez que cela ne fonctionnera que si l'environnement linguistique est supporté par le système. Utilisez la commande `locale -a` pour obtenir la liste des environnements linguistiques supportés.

Sous Windows, vous avez besoin de l'option `-gb` pour convertir UTF-16 en GB18030.

Les fichiers encodés en GB18030 peuvent avoir une marque d'ordre des octets, comme les fichiers Unicode.

EXEMPLES

Lire l'entrée depuis « stdin » et écrire la sortie vers « stdout » :

```
dos2unix < a.txt  
cat a.txt | dos2unix
```

Convertir et remplacer a.txt. Convertir et remplacer b.txt :

```
dos2unix a.txt b.txt  
dos2unix -o a.txt b.txt
```

Convertir et remplacer a.txt en mode de conversion ascii :

```
dos2unix a.txt
```

Convertir et remplacer a.txt en mode de conversion ascii. Convertir et remplacer b.txt en mode de conversion 7 bits :

```
dos2unix a.txt -c 7bit b.txt  
dos2unix -c ascii a.txt -c 7bit b.txt  
dos2unix -ascii a.txt -7 b.txt
```

Convertir a.txt depuis le format Mac vers le format Unix :

```
dos2unix -c mac a.txt  
mac2unix a.txt
```

Convertir a.txt du format Unix au format Mac :

```
unix2dos -c mac a.txt  
unix2mac a.txt
```

Convertir et remplacer a.txt tout en conservant la date originale :

```
dos2unix -k a.txt  
dos2unix -k -o a.txt
```

Convertir a.txt et écrire dans e.txt :

```
dos2unix -n a.txt e.txt
```

Convertir a.txt et écrire dans e.txt. La date de e.txt est la même que celle de a.txt :

```
dos2unix -k -n a.txt e.txt
```

Convertir et remplacer a.txt. Convertir b.txt et écrire dans e.txt :

```
dos2unix a.txt -n b.txt e.txt  
dos2unix -o a.txt -n b.txt e.txt
```

Convertir c.txt et écrire dans e.txt. Convertir et remplacer a.txt. Convertir et remplacer b.txt. Convertir d.txt et écrire dans f.txt :

```
dos2unix -n c.txt e.txt -o a.txt b.txt -n d.txt f.txt
```

CONVERSIONS RÉCURSIVES

Dans un shell Unix, les commandes `find(1)` et `xargs(1)` peuvent être utilisées pour exécuter `dos2unix` récursivement sur tous les fichiers textes dans une arborescence de répertoires. Par exemple, pour convertir tous les fichiers `.txt` dans les répertoires sous le répertoire courant, tapez:

```
find . -name '*.txt' -print0 |xargs -0 dos2unix
```

L'option `-print0` de `find(1)` et l'option correspondante `-0` de `xargs(1)` sont nécessaires quand il y a des fichiers avec des espaces ou des guillemets dans leur nom. Sinon, ces options peuvent être omises. Une autre possibilité est d'utiliser `find(1)` avec l'option `-exec`:

```
find . -name '*.txt' -exec dos2unix {} \;
```

En ligne de commande sous Windows, la commande suivante peut être utilisée :

```
for /R %G in (*.txt) do dos2unix "%G"
find /R %G in
```

Les utilisateurs de PowerShell peuvent utiliser la commande suivante dans le PowerShell de Windows :

```
get-childitem -path . -filter '*.txt' -recurse | foreach-object {dos2unix $_.Fullname}
```

PARAMÈTRES LINGUISTIQUES

LANG

La langue principale est sélectionnée par la variable d'environnement `LANG`. La variable `LANG` est composée de plusieurs parties. La première partie est le code de la langue en minuscules. La deuxième partie est le code du pays en majuscules précédé d'un souligné. Elle est facultative. Il y a aussi une troisième partie facultative qui est le codage des caractères précédé par un point. Voici quelques exemples pour un shell au standard POSIX:

<code>export LANG=fr</code>	Français
<code>export LANG=fr_CA</code>	Français, Canada
<code>export LANG=fr_BE</code>	Français, Belgique
<code>export LANG=es_ES</code>	Espagnol, Espagne
<code>export LANG=es_MX</code>	Espagnol, Mexique
<code>export LANG=en_US.iso88591</code>	Anglais, USA, codage Latin-1
<code>export LANG=en_GB.UTF-8</code>	Anglais, UK, codage UTF-8

La liste complète des codes de langues et de pays est dans le manuel de `gettext`: http://www.gnu.org/software/gettext/manual/html_node/Usual-Language-Codes.html

Sur les systèmes Unix, vous pouvez utiliser la commande `locale(1)` pour obtenir des informations sur l'environnement linguistique.

LANGUAGE

Avec la variable d'environnement `LANGUAGE`, vous pouvez spécifier une liste de langues prioritaires séparées par des deux-points. `Dos2unix` fait passer `LANGUAGE` avant `LANG`. Par exemple, pour utiliser le français avant l'anglais: `LANGUAGE=fr:en`. Vous devez d'abord activer l'environnement linguistique en assignant une valeur autre que « C » à `LANG` (ou `LC_ALL`). Ensuite, vous pourrez utiliser la liste de priorité avec la variable `LANGUAGE`. Voyez également le manuel de `gettext`: http://www.gnu.org/software/gettext/manual/html_node/The-LANGUAGE-variable.html

Si vous sélectionnez une langue qui n'est pas disponible, vous obtiendrez des messages en anglais standard.

DOS2UNIX_LOCALEDIR

Grâce à la variable d'environnement `DOS2UNIX_LOCALEDIR`, la variable `LOCALEDIR` compilée dans l'application peut être remplacée. `LOCALEDIR` est utilisée pour trouver les fichiers de langue. La valeur par défaut de GNU est `/usr/local/share/locale`. L'option `--version` affiche la

valeur de LOCALEDIR utilisée.

Exemple (shell POSIX):

```
export DOS2UNIX_LOCALEDIR=$HOME/share/locale
```

VALEUR DE RETOUR

Zéro est retourné en cas de succès. Si une erreur système se produit, la dernière erreur système est retournée. Pour les autres erreurs, 1 est renvoyé.

La valeur de sortie est toujours zéro en mode silencieux sauf quand de mauvaises options sont utilisées sur la ligne de commande.

STANDARDS

http://fr.wikipedia.org/wiki/Fichier_texte

http://fr.wikipedia.org/wiki/Retour_chariot

http://fr.wikipedia.org/wiki/Fin_de_ligne

<http://fr.wikipedia.org/wiki/Unicode>

AUTEURS

Benjamin Lin - <blin@soecs.uts.edu.au>, Bernd Johannes Wuebben (mode mac2unix) - <wuebben@kde.org>, Christian Wurll (ajout de saut de ligne supplémentaire) - <wurll@ira.uka.de>, Erwin Waterlander - <waterlan@xs4all.nl> (Mainteneur)

Page du projet: <http://waterlan.home.xs4all.nl/dos2unix.html>

Page SourceForge: <http://sourceforge.net/projects/dos2unix/>

VOIR AUSSI

`file(1)` `find(1)` `iconv(1)` `locale(1)` `xargs(1)`