

NOM

diff – Trouver les différences entre des fichiers.

SYNOPSIS

diff [options] source cible

DESCRIPTION

Dans le cas le plus simple, *diff* compare le contenu du fichier *source* et celui du fichier *cible*. Un nom de fichier remplacé par un tiret – correspond à la lecture depuis l'entrée standard. Un cas particulier est représenté par "**diff --**" qui compare une copie de l'entrée standard avec elle-même.

Si la *source* est un répertoire, et pas la *cible*, alors *diff* compare le fichier *cible* avec le fichier placé dans le répertoire *source* et dont le nom soit identique à la *cible*. La situation réciproque est également possible. Le fichier indiqué (pas le répertoire) ne peut pas être -. En effet, l'entrée standard n'a pas de nom et la correspondance « nom à nom » ne pourrait pas s'appliquer.

Si les deux fichiers *source* et *cible* sont des répertoires, *diff* compare les fichiers correspondant dans les deux répertoires, dans l'ordre alphabétique. Cette comparaison n'est pas récursive, à moins d'employer l'option **-r** ou **--recursive**. *diff* ne compare jamais le contenu « réel » d'un répertoire en le considérant comme un fichier.

Les options de **diff** commencent par un tiret -, ainsi les fichiers *source* et *cible* ne devraient pas commencer par un tiret -. Toutefois, le double tiret -- est un argument spécial signifiant « fin des options », ainsi des noms de fichiers commençant par un tiret peuvent être mentionnés à sa suite.

Options

Voici une liste de l'ensemble des options que le *diff* GNU accepte. La plupart des options ont deux noms équivalents, l'un constitué d'une seule lettre précédée par -, et l'autre composé d'un nom long plus explicite, précédé par --.

Plusieurs options avec lettre unique (sauf si elle prennent un argument) peuvent être combinées en un seul mot de la ligne de commande ; ainsi : **-ac** est équivalent à **-a -c**.

Les options avec un nom long peuvent être abrégées s'il n'y a pas d'ambiguïté.

Les crochets ([et]) indiquent qu'une option prend éventuellement un argument.

- nb** (nb est un nombre entier) Afficher *nb* lignes de contexte. Cette option ne précise pas le format de sortie par elle-même, elle n'a pas d'effet si elle n'est pas combinée avec **-c** ou **-u**. Cette option est obsolète. Pour fonctionner correctement, *patch* nécessite typiquement deux lignes de contexte.
- a** Traiter tous les fichiers comme du texte, et les comparer ligne-à-ligne, même s'ils semblent contenir des données binaires.
- b** Ne pas tenir compte des différences concernant des espaces blancs.
- B** Ne pas tenir compte des différences qui concernent des lignes blanches.
- brief** Indiquer seulement si les fichiers diffèrent, pas les différences elles-mêmes.
- c** Utiliser le format de sortie contextuel.
- C nb**
- context[=nb]** (nb est un nombre entier) Utiliser le format de sortie contextuel en affichant *nb* lignes de texte. Par défaut *nb* vaut 3. Pour fonctionner correctement, *patch* nécessite typiquement au moins deux lignes de contexte.

- changed-group-format=***format*
Utiliser le *format* pour afficher un groupe de lignes contenant des différences entre les deux fichiers. L'affichage est du type si-alors-sinon (voir option -D)
- d** Modifier l'algorithme pour trouver éventuellement des ensembles de changements plus petits. Ceci rend *diff* plus lent (parfois beaucoup plus lent).
- D** *nom*
Sortir les deux textes originaux regroupés. La différenciation se fait par des directives de préprocesseur *ifdef-then-else* concernant l'existence, ou non, de la macro ayant le *nom* indiqué. Par exemple on utilisera '*diff -D Nouvelle_version ancien_fichier.c nouveau_fichier.c*' pour obtenir un fichier pouvant être compilé au choix avec la constante symbolique *Nouvelle_version*.
- e**
- ed** Sortir un script valide pour *ed*.
- exclude=***motif*
Lors de la comparaison de répertoires, ignorer les fichiers et les sous-répertoires dont le nom correspond au *motif*.
- exclude-from=***fichier*
Lors de la comparaison de répertoires, ignorer les fichiers et les sous-répertoires dont le nom correspond à n'importe quel motif indiqué dans le *fichier*.
- expand-tabs**
Développer les tabulations en espaces en sortie, afin de préserver l'alignement des tabulations des fichiers d'entrée.
- f** S'arranger pour que la sortie ressemble vaguement à un script *ed*, mais en préservant l'ordre des modifications du fichier.
- F** *regexp*
Dans les formats contextuel et unifiés, pour chaque bloc de différences, afficher quelques lignes précédentes qui correspondent à l'expression rationnelle *regexp*.
- forward-ed**
S'arranger pour que la sortie ressemble vaguement à un script **ed** mais en préservant l'ordre des modifications du fichier.
- h** Cette option n'a aucun effet, il s'agit simplement de préserver la compatibilité Unix.
- H** Utiliser une heuristique accélérant le traitement des gros fichiers ayant de nombreux petits changements dispersés.
- horizon-lines=***nb*
Ne pas éliminer les *nb* dernières lignes du préfixe commun, ni les *nb* premières lignes du suffixe commun.
- i** Ignore les différences entre majuscules et minuscules.
- I** *regexp*
Ignorer les changements consistant seulement en insertion ou suppression de lignes correspondant à l'expression régulière *regexp*.
- ifdef=***nom*
Sortir les deux textes originaux regroupés. La différenciation se fait par des directives de préprocesseur *ifdef-then-else* concernant l'existence, ou non, de la macro ayant le *nom* indiqué.
- ignore-all-space**
Ignorer les espaces lors des comparaisons de lignes
- ignore-blank-lines**
Ignorer les changements consistant uniquement en insertion ou suppression de lignes blanches.

- ignore-case**
Ignorer les différences entre majuscules et minuscules.
- ignore-matching-lines=*regex***
Ignorer les changements consistant seulement en insertion ou suppression de lignes correspondant à l'expression rationnelle *regex*.
- ignore-space-change**
Ignorer les changements concernant le nombre d'espaces blanches.
- initial-tab**
Sortir une tabulation plutôt qu'une espace avant le texte dans le format normal ou contextuel. Ceci améliore la présentation des alignements.
- l** Envoyer la sortie à *pr* pour la paginer.
- L étiquette**
- label=*étiquette***
Utiliser l'*étiquette* à la place du nom de fichier dans les en-têtes des formats contextuels ou unifiés.
- left-column**
N'afficher que la colonne de gauche des lignes communes dans le format side-by-side.
- line-format=*format***
Utiliser le *format* pour afficher toutes les lignes d'entrée dans le format if-then-else.
- minimal**
Modifier l'algorithme pour trouver éventuellement des ensembles de changements plus petits. Ceci rend *diff* plus lent (parfois beaucoup plus lent).
- n** Sortir les différences au format RCS. Comme **-f** sauf que chaque commande précise le nombre de lignes affectées.
- N**
- new-file**
Dans les comparaisons de répertoires, si un fichier ne se trouve que dans un seul répertoire, considérer qu'il est présent, mais vide dans l'autre répertoire.
- new-group-format=*format***
Utiliser le *format* pour sortir un groupe de lignes ne se trouvant que dans le second fichier, dans le format if-then-else.
- new-line-format=*format***
Utiliser le *format* pour sortir une ligne ne se trouvant que dans le second fichier, dans le format if-then-else.
- old-group-format=*format***
Utiliser le *format* pour sortir un groupe de lignes ne se trouvant que dans le premier fichier, dans le format if-then-else.
- old-line-format=*format***
Utiliser le *format* pour sortir une ligne ne se trouvant que dans le premier fichier, dans le format if-then-else.
- p** Indiquer dans quelle fonction C se trouve chaque changement.
- P** Dans les comparaisons de répertoires, si un fichier ne se trouve que dans le second répertoire, considérer qu'il est présent, mais vide dans l'autre répertoire.
- paginate**
Envoyer la sortie à *pr* pour la paginer.
- q** Indiquer seulement si les fichiers diffèrent, pas les différences elles-mêmes.
- r** Lorsque l'on compare des répertoires, comparer également tous les sous-répertoires récursivement.

- rcs** Sortir les différences au format RCS. Comme **-f** sauf que chaque commande précise le nombre de lignes affectées.
- recursive**
Lorsque l'on compare des répertoires, comparer également tous les sous-répertoires récursivement.
- report-identical-files**
- s** Indiquer lorsque deux fichiers sont identiques.
- S fichier**
Lorsque l'on compare des répertoires, commencer par le *fichier* indiqué. Ceci permet une reprise après une comparaison interrompue.
- sdiff-merge-assist**
Afficher des informations supplémentaires pour aider *sdiff*. Celui-ci utilise cette option lorsqu'il invoque *diff*. L'utilisateur n'a aucune raison d'appeler cette option.
- show-c-function**
Indiquer dans quelle fonction C se trouve chaque changement.
- show-function-line=regex**
Dans les formats contextuel et unifiés, pour chaque bloc de différences, afficher quelques lignes précédentes correspondant à l'expression rationnelle *regex*.
- side-by-side**
Utiliser le format de sortie côte-à-côte.
- speed-large-files**
Utiliser une heuristique accélérant le traitement des gros fichiers ayant de nombreux petits changements dispersés.
- starting-file=fichier**
Lorsque l'on compare des répertoires, commencer par le *fichier* indiqué. Ceci permet une reprise après une comparaison interrompue.
- suppress-common-lines**
Ne pas afficher les lignes communes dans le format side-by-side.
- t** Développer les tabulations en espaces en sortie, afin de préserver l'alignement des tabulations des fichiers d'entrée.
- T** Sortir une tabulation plutôt qu'un espace avant le texte dans le format normal ou contextuel. Ceci améliore la présentation des alignements.
- text** Traiter tous les fichiers comme du texte, et les comparer ligne-à-ligne, même s'ils semblent contenir des données binaires.
- u** Utiliser le format de sortie unifié.
- unchanged-group-format=format**
Utiliser le *format* pour sortir un groupe de lignes communes aux deux fichiers, dans le format if-then-else.
- unchanged-line-format=format**
Utiliser le *format* pour sortir une ligne commune aux deux fichiers, dans le format if-then-else.
- unidirectional-new-file**
Dans les comparaisons de répertoires, si un fichier ne se trouve que dans le second répertoire, considérer qu'il est présent, mais vide dans l'autre répertoire.
- U nb**
- unified[=nb]**
Utiliser le format unifié, en montrant *nb* lignes de contexte, ou 3 lignes si *nb* n'est pas précisé. Pour fonctionner correctement, *patch* nécessite typiquement deux lignes de contexte

- v**
- version**
Afficher le numéro de version de *diff*.
- w** Ignorer les espaces blancs lors de la comparaison de lignes.
- W *nb***
- width=*nb***
Utiliser une sortie ayant *nb* colonnes de texte dans le format side-by-side.
- x *motif***
Lors de la comparaison de répertoires, ignorer les fichiers et les sous-répertoires dont le nom correspond au *motif*.
- X *file*** Lors de la comparaison de répertoires, ignorer les fichiers et les sous-répertoires dont le nom correspond à l'un quelconque des motifs indiqués dans le *fichier*.
- y** Utiliser l'affichage côte-à-côte (side-by-side).

VOIR AUSSI

cmp(1), comm(1), diff3(1), ed(1), patch(1), pr(1), sdiff(1)

DIAGNOSTIC

Un code de retour valant 0 signifie qu'aucune différence n'a été trouvée, 1 signifie que des différences sont apparues, 2 indique une erreur.

TRADUCTION

Ce document est une traduction réalisée par Christophe Blaess <<http://www.blaess.fr/christophe/>> le 15 décembre 1998 et révisée le 14 août 2006.

L'équipe de traduction a fait le maximum pour réaliser une adaptation française de qualité. La version anglaise la plus à jour de ce document est toujours consultable via la commande : « **LANG=C man 1 diff** ». N'hésitez pas à signaler à l'auteur ou au traducteur, selon le cas, toute erreur dans cette page de manuel.