

# Documentation de la mise à niveau de Windows 10

## Conditions préalables

Les clients sur lesquels vous souhaitez effectuer une mise à niveau doivent être conformes à la [Configuration minimale](#) de Windows 10. La configuration minimale requise pour Windows 10 Release 2004 est la suivante :

- RAM : 2GB de RAM pour les ordinateurs 64-bit et 1GB de RAM pour les ordinateurs 32-bit.
- Stockage : 20Go d'espace libre sur les systèmes 64-bit et 16Go d'espace libre sur les systèmes 32-bit.
- Résolution d'écran : 800×600
- Graphique : Microsoft DirectX 9 ou plus récent avec WDDM 1.0
- Intel CPU : Tous les processeurs jusqu'à 10th-gen, Intel Xeon E-22xx, Atom, Celeron, et Pentium.
- AMD CPU : Tous les processeurs jusqu'à la septième génération.
- Qualcomm CPU : Snapdragon 850 et 8cx.

Un système d'exploitation compatible avec la mise à niveau doit être installé sur le client. La [Chemins de mise à niveau Windows10](#) de Windows 10 offre un aperçu des chemins de la mise à niveau possibles.

## Préparatifs

### Téléchargement du médium d'installation

Pour pouvoir effectuer une mise à niveau vers une version plus récente, il faut disposer d'un support d'installation adéquat. Il faut tenir compte des points suivants :

1. Le médium d'installation doit être téléchargé pour la bonne architecture (32Bit/64Bit).
2. Le médium d'installation doit être téléchargé pour la bonne édition de Windows (Home/Pro/Education/Enterprise).
3. Le médium d'installation doit être téléchargé pour la langue correcte.
4. Le médium d'installation doit être téléchargé pour l'ID de version souhaitée (2004/20H2).

Le médium d'installation peut être obtenu à partir de différentes sources :

- [Téléchargement de l'image disque de Microsoft Windows 10.](#)
- [Media Creation Tool](#)
- [Volume Licensing Center](#)
- [MSDN](#)
- [Windows Insider Website](#)
- [Website for Education Editions \(Product Key required\)](#)

## Intégration du médium d'installation

Pour obtenir les fichiers nécessaires, il faut décompresser le fichier ISO téléchargé.

Les fichiers décompressés doivent être copiés dans le dossier `localsetup\installfiles` du paquet de la mise à niveau de windows10. Ensuite, il faut exécuter un `sudo opsi-set-rights`. Il est également possible d'intégrer plusieurs médiums d'installation dans le paquet, par exemple pour pouvoir mettre à niveau vers différentes ID de version ou pour couvrir différentes architectures ou langues. Il faut alors tenir compte des points suivants :

- Le nom du médium doit toujours commencer par `installfiles` !
- Pour pouvoir utiliser la propriété de produit `automode`, l'ID de version doit faire partie du nom du dossier !

Exemples :

- `installfiles2004`.
- `installfiles-20h2`.
- `installfiles_21h1-32Bit` (à partir de la version 20.09-3)
- `installfiles_21h2-en` (à partir de la version 20.09-3)

Pour que le paquet reconnaisse les différents dossiers `Installfiles`, il doit être réinstallé par `opsi-package-manager -i` après la création et le remplissage des dossiers, car l'intégration des dossiers `Installfiles` se fait dans les scripts `preinst/postinst`. Ensuite le répertoire `Installfiles` à utiliser peut être sélectionné par la propriété de produit `installfiles_dir`.

## Vérification du médium d'installation

Les outils `wiminfo` ou `wimlib` permettent de lire les images contenues dans le fichier `install.wim`. Cela permet de déterminer les éditions, les langues et l'architecture contenues dans le médium d'installation.

### wiminfo

Cet outil est mis à disposition sous Linux via le paquet serveur `opsi-windows-support`. Avec `wiminfo`, les informations peuvent être lues directement sur la console du serveur OPSI. Exemple de commande pour lire les informations:\\.

- `wiminfo /var/lib/opsi/depot/windows10-upgrade/localsetup/installfiles/sources/install.wim`.

### wimlib

Cet outil se trouve dans le paquet sous les chemins suivants :

- `localsetup\wimlib\x64\wimlib-imagex.exe`.
- `localsetup\wimlib\x86\wimlib-imagex.exe\\.`

Exemple de commande pour lire les informations (`\\<opsi-server>\opsi_depot` partage comme lecteur

p:\ mounted):

- `p:\windows10-upgrade\localsetup\wimlib\x64\wimlib-imageex.exe" info "p:\windows10-upgrade\localsetup\installfiles2009\sources\install.wim"`

Pendant la mise à niveau, l'image contenue dans le répertoire qui est sélectionné par la propriété `produit installfiles_dir` est lue avec `wimlib` et la sortie est enregistrée dans le fichier `journal`.

## Symlinks

A partir de la version 21h1-1, les dossiers `Installfiles` ne doivent pas être placés directement dans le dossier `localfiles`, mais peuvent également être intégrés par symlink. Exemple de création d'un symlink vers le dossier `Installfiles` du paquet `netboot win10-x64`:

- `cd /var/lib/opsi/depot/windows10-upgrade/localsetup.`
- `ln -s ../../win10-x64/installfiles/ .`

Après avoir créé le symlink, le paquet doit être réinstallé par `opsi-package-manager -i`, car l'intégration des dossiers `Installfiles` se fait dans les scripts `preinst/postinst`. Le répertoire `Installfiles` à utiliser peut ensuite être sélectionné avec la propriété de produit `installfiles_dir`.

## Propriétés du produit

- `automode.`
  - `False` (par défaut)
    - L'ID de la version vers laquelle la mise à niveau doit être effectuée doit être sélectionnée avec la Product Property `upgrade_to_version`.
  - `True`
    - L'ID de la version est tenté d'être lu à partir du nom du dossier sélectionné avec la Product Property `installfiles_dir`.
    - Exemple : Si le dossier `installfiles2004` a été sélectionné dans la Product Property `installfiles_dir`, 2004 sera utilisé comme Release ID. L'ID de version sélectionné sous `upgrade_to_version` est dans ce cas ignoré !
    - Si aucun Release ID valide ne peut être lu, la valeur sélectionnée sous la propriété de produit `upgrade_to_version` est utilisée comme fallback.
- `copy_files_locally.`
  - `False` (par défaut)
    - Les fichiers d'installation sont obtenus du `opsi_depot` pendant la mise à niveau.
  - `True`
    - Les fichiers d'installation sont copiés localement dans le dossier `%SystemDrive%\Nopsi.org\Nousertmp\windows10-upgrade` avant la mise à niveau.
- `debug.`
  - `False` (par défaut)
    - Verrouille les entrées clavier et souris pendant l'auto-logon de l'`opsiSetupUser` pour éviter l'interaction de l'utilisateur.
  - `True`
    - Le clavier et la souris restent actifs pendant la connexion automatique pour le débogage en cas d'erreur.

- `delete_windows_old`.
  - False (par défaut)
    - Le dossier Windows.old est conservé après la mise à niveau.
  - True
    - Après une mise à niveau réussie, le dossier Windows.old est supprimé.
- `delete_windows_update_cache`.
  - False (par défaut)
    - Le cache de Windows Update dans le dossier SystemRoot\SoftwareDistribution est conservé.
  - True
    - Le cache de Windows Update dans le dossier SystemRoot\SoftwareDistribution est effacé avant la mise à jour.
- `encryption_driver`.
  - Spécification d'un chemin contenant des pilotes pour des logiciels de chiffrement tiers.
- `execution_method`
  - `loginOpsiSetupUser`
    - Le service opscientd crée l'utilisateur opsiSetupUser et le connecte.
    - L'explorer.exe est lancé comme shell.
    - Le bureau de l'utilisateur opsiSetupUser est visible.
    - Après le login, la mise à niveau démarre après un court délai.
    - Les fichiers d'installation peuvent être récupérés sur le serveur, ou enregistrés localement sur le client (voir `copy_files_locally`).
  - `runAsOpsiSetupUser`
    - Le service opscientd crée l'utilisateur opsiSetupUser et le connecte.
    - Le shell powershell.exe est lancé et déclenche le script opsi local qui exécute la mise à niveau.
    - Le bureau de l'opsiSetupUser n'est pas visible.
    - Les fichiers d'installation doivent être enregistrés localement sur le client.
    - La valeur définie sous `copy_files_locally` est ignorée et automatiquement mise à true.
  - `runOpsiScriptAsOpsiSetupUser` (par défaut)
    - Le service opscientd crée l'utilisateur opsiSetupUser et le connecte.
    - Le shell powershell.exe est lancé et déclenche `action_processor_starter.exe`, qui établit une connexion de service et exécute ensuite la mise à jour.
    - Le bureau de l'utilisateur opsiSetupUser n'est pas visible.
    - Les fichiers d'installation peuvent être récupérés sur le serveur, ou enregistrés localement sur le client (voir `copy_files_locally`).
  - Si un client se trouve en mode WAN/VPN (détection automatique), cette propriété de produit est ignorée et l'installation est effectuée avec les options suivantes : (voir aussi le chapitre Mode WAN/VPN).
    - Le service opscientd crée l'utilisateur opsiSetupUser et le connecte.
    - Le shell powershell.exe est démarré et déclenche le script local opsi qui exécute la mise à jour.
    - Le bureau de l'utilisateur opsiSetupUser n'est pas visible.
    - Les fichiers d'installation présents localement dans le cache sont utilisés.
- `installfiles_dir`
  - Sélectionner le dossier qui contient les fichiers d'installation.
  - Le nom du dossier doit toujours commencer par `installfiles` !
  - Si la propriété `produit_automode` doit être utilisée, l'ID de la version doit être ajouté au nom du dossier (par ex. `installfiles2004`).

- Pour que tous les répertoires installfiles contenus dans le paquet soient listés dans la propriété produit `installfiles_dir`, il faut éventuellement réinstaller le paquet avec `opsi-package-manager -i` après la création des dossiers, car la propriété est remplie par les scripts `preinst/postinst`.
- `language_from_language_pack` (voir chapitre Language Packs)
  - False (par défaut)
    - Sélectionner si le client n'a pas été installé avec un média d'installation contenant des Language Packs.
  - True
    - Sélectionner si le client a été installé avec un médium d'installation contenant des packs linguistiques et si la langue du client est fournie par un pack linguistique.
- `language_installation_media` (Voir chapitre Language Packs)
  - Indication du code de langue du médium d'installation original du client.
  - Exemples :
    - 0407 pour l'allemand
    - 0409 pour l'anglais
- `mode`.
  - `upgrade` (par défaut)
    - Mise à niveau vers le Release ID sélectionné.
  - `validate_before_upgrade`
    - Exécution d'essai pour déterminer les problèmes de compatibilité. Si aucun problème n'est détecté, la mise à niveau est lancée.
  - `validate_only`
    - Exécution d'essai pour détecter les problèmes de compatibilité.
- `post_oobe`.
  - Chemin vers `setupcomplete.cmd`
  - Plus d'infos sur :
    - <https://learn.microsoft.com/fr-fr/windows-hardware/manufacture/desktop/add-a-custom-script-to-windows-setup?view=windows-11>
- `productkey`.
  - Clé de licence qui doit être utilisée pour l'installation.
- `proxy_ip`.
  - Permet d'indiquer l'IP d'un serveur proxy.
- `proxy_port`.
  - Permet d'indiquer le numéro de port d'un serveur proxy.
- `quiet`.
  - False
    - L'installation démarre en mode interactif et nécessite la saisie d'un utilisateur.
    - Utile pour obtenir un message d'erreur significatif en cas d'erreur.
  - True (par défaut)
    - L'installation démarre en mode silencieux sans intervention de l'utilisateur.
- `restore_health` (restauration de la santé)
  - False (par défaut)
    - „SFC /scannow“ et „DISM /online /cleanup-image /restorehealth“ ne sont pas exécutés avant la mise à niveau.
  - True
    - „SFC /scannow“ and „DISM /online /cleanup-image /restorehealth“ sont exécutés avant la mise à niveau.
- `setup_after_install`.
  - Sélection des produits OPSI qui doivent être installés après une mise à niveau réussie.
  - Produits sélectionnables (extensibles manuellement) :

- config-win10
- mshotfix
- opsi-client-agent
- showoobe.
  - Full
    - Les utilisateurs doivent passer par le dialogue Out Of Box Experience (OOBE) lorsqu'ils se connectent après la mise à niveau.
  - None (par défaut)
    - Ignore le dialogue Out Of Box Experience (OOBE) lorsqu'un utilisateur se connecte après la mise à niveau.
- telemetry (télémétrie)
  - Disable (par défaut)
    - Aucune donnée télémétrique de la mise à niveau n'est transmise à Microsoft.
  - Enable
    - Les données de télémétrie de la mise à niveau sont transmises à Microsoft.
- unset\_after\_install.
  - Sélection des produits OPSI qui doivent être mis à not\_installed après une mise à niveau réussie.
- upgrade\_to\_version.
  - Sélection de l'ID de la version vers laquelle la mise à niveau doit être effectuée.
  - Ignoré si automode est true.
    - Si aucun Release ID valide n'a pu être lu via Automode, la valeur sélectionnée ici est utilisée comme repli.
  - latest (par défaut)
    - Renvoie toujours à l'ID de version la plus récente, qui était en vigueur au moment de la création du paquet.

## Mode WAN/VPN

Le paquet détecte automatiquement si un client se trouve en mode WAN/VPN. Les fichiers d'installation mis en cache localement sont utilisés.

### Particularités

**Important!** Après chaque modification des dossiers Installfiles, le paquet doit être réinstallé par `opsi-package-manager -i` pour que le fichier `.files` soit régénéré!

Les paquets qui ont été sélectionnés par `setup_after_install` ne peuvent être mis en cache que lorsque le client peut contacter le serveur avec succès la fois suivante. La plupart du temps, cela ne se produit que lorsqu'une connexion VPN est établie après le login d'un utilisateur.

## Language Packs

Les installations Windows avec des Language Packs installés représentent un cas particulier. Pour mettre à jour avec succès une installation qui utilise des Language Packs, il faut tenir compte de certains points.

## Préparation du support d'installation

Tous les Language Packs nécessaires doivent être intégrés dans le fichier install.wim du support d'installation pour la mise à niveau. Il faut veiller à ce que les Language Packs soient téléchargés en fonction de la version du support d'installation. Les Language Packs de la version 1909 ne peuvent par exemple pas être intégrés dans un support d'installation 21H1. L'intégration des Language Packs s'effectue via l'outil DISM. Des instructions et des informations supplémentaires sont disponibles dans les liens suivants :

[Ajouter ou supprimer des packages hors connexion à l'aide de DISM](#)

[How to Add Language Packs Offline Using DISM](#)

[Langues DISM et options de maintenance internationale Command-Line](#)

[Langues disponibles pour Windows](#)

## Déroulement

Le système d'exploitation à mettre à niveau doit être réinitialisé à la langue du médium d'installation d'origine avant la mise à niveau.

Si, par exemple, un client a été installé à partir d'un support d'installation anglais contenant un Language Pack allemand et que l'allemand a été configuré comme langue principale du client, la langue doit être réglée sur l'anglais avant la mise à niveau. \\NLa mise à niveau doit être effectuée par le client lui-même. Après la mise à niveau, la langue peut être réinitialisée sur l'allemand si le support d'installation utilisé lors de la mise à niveau intègre un Language Pack allemand correspondant à son ReleaseID.

Ce processus a été intégré de manière automatisée dans la version 21h1-1 des paquets windows-upgrade. Avant la mise à niveau, la langue actuelle est lue et enregistrée. Ensuite, la langue est réglée sur la langue du support d'installation utilisé pour l'installation du client. Après la mise à niveau, la langue est rétablie sur les valeurs initiales.

## Paramètres du client

- La propriété produit `language_from_language_pack` doit être définie sur `true`.
- La Product Property `language_installation_media` doit être définie sur le code langue à quatre chiffres du support d'installation.
  - Exemples :
    - 0407 pour l'allemand
    - 0409 pour l'anglais

## Mise à niveau des versions précédentes de Windows 10

Les versions précédentes de Windows 10 suivantes peuvent être mises à niveau vers Windows 10 :

- Windows 7
- Windows 8
- Windows 8.1

Voir aussi : [Chemins de mise à niveau Windows10](#)

Avant la mise à niveau, toutes les mises à jour Windows actuelles doivent être appliquées.

## Débogage

De nombreuses circonstances peuvent entraîner l'échec d'une mise à niveau :

- Le client ne répond pas à la configuration minimale requise pour Windows 10.
- Partition réservée au système trop petite
- Le scanner de virus empêche la mise à niveau
- Pilotes incompatibles sur le client
- Logiciel incompatible sur le client
- Mauvais support d'installation
  - Mauvaise architecture
  - Mauvaise langue
  - Mauvaise édition
- État d'activation du système d'exploitation avant la mise à niveau
- Interaction de l'utilisateur pendant la mise à niveau (par ex. redémarrage ou extinction du client)

## Evaluation du code de sortie

A partir de la version 20.09-1, le code de sortie de la mise à niveau est évalué de manière plus détaillée et un message d'erreur significatif avec des conseils pour y remédier est écrit dans le fichier journal. La liste des codes de sortie n'est cependant certainement pas complète, car il n'existe pas de liste exhaustive de tous les codes de sortie de la part de Microsoft.

## Fichiers journaux de mise à niveau inclus

Les fichiers journaux de mise à niveau suivants sont inclus dans le fichier journal du paquet en cas d'erreur (si disponible) et peuvent être facilement analysés dans le Configed :

- „%SystemDrive%\\$WINDOWS.~BT\Sources\Panther\*APPRAISER\*.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\*CompatData\*.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Rollback\setupact.log“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Rollback\setuperr.log“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Rollback\Setupapi\Setupapi.dev.log“
- „%SystemDrive%\\$WINDOWS.~BT\Sources\Panther\UnattendGC\diagerr.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\NUnattendGC\Setupact.log“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\NUnattendGC\Setuperr.log“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\diagerr.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\MigLog.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\ScanResult.xml“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\setupact.log“
- „%SystemDrive%\NWINDOWS.~BT\Sources\Panther\setuperr.log“
- „%SystemRoot%\Panther\*APPRAISER\*.xml“
- „%SystemRoot%\Panther\*CompatData\*.xml“
- „%SystemRoot%\Panther\Rollback\setupact.log“



- „%SystemRoot%\Panther\Rollback\setuperr.log“
- „%SystemRoot%\Panther\Rollback\setupapi\setupapi.dev.log“
- „%SystemRoot%\Panther\UnattendGC\diagerr.xml“
- „%SystemRoot%\Panther\UnattendGC\setupact.log“
- „%SystemRoot%\Panther\UnattendGC\setuperr.log“
- „%SystemRoot%\Panther\diagerr.xml“
- „%SystemRoot%\Panther\MigLog.xml“
- „%SystemRoot%\Panther\ScanResult.xml“
- „%SystemRoot%\Panther\setupact.log“
- „%SystemRoot%\Panther\setuperr.log“
- „%SystemRoot%\Logs\MoSetup\Bluebox.log“
- „%SystemRoot%\Logs\SetupDiag\setupdiagresults.xml“
- „%opsiLogDir%\opsisetupuser.log“
- „%opsiLogDir%\SetupDiagResults.log“
- „%opsiLogDir%\sfcdetails.txt“
- „%SystemRoot%\logs\dism\dism.log“

## Solutions possibles pour les problèmes liés à la mise à niveau de Windows

Pour détecter les logiciels et pilotes incompatibles, il faut rechercher les chaînes suivantes dans le fichier journal de mise à niveau de windows10 ou directement dans „%SystemDrive%\\$WINDOWS.~BT\Sources\Panther\Scanresult.xml“ et %SystemDrive%\\$WINDOWS.~BT\Sources\Panther\Compatdata\*.xml (plusieurs fichiers journaux individuels) :

- CompatibilityInfo BlockingType="Hard"
- CompatibilityInfo BlockMigration="True"

Dans le cas d'un pilote incompatible, le fichier .inf correspondant devrait s'afficher (par ex. oem11.inf). Les fichiers .inf se trouvent dans le dossier %SystemDrive%\NWindows\NINF. Il faut les ouvrir avec un éditeur de texte pour trouver le nom du pilote. Désinstaller le ou les pilotes trouvés sur le client et relancer la mise à jour.

**Attention!** Deux pilotes sont **toujours** trouvés avec le statut CompatibilityInfo BlockMigration="True" (selon le système, dans différents fichiers oemxx.inf) Les périphériques correspondants sont les suivants :

- Microsoft Print to PDF
- Microsoft XPS Document writer

Bien que ces deux pilotes soient affichés avec CompatibilityInfo BlockMigration="True", ils n'empêchent **pas** la mise à niveau et peuvent être ignorés.

Pour obtenir un message d'erreur significatif, on peut lancer la mise à niveau avec les propriétés de produit quiet = false et debug = true. **Attention** : la mise à niveau démarrera alors en mode interactif et nécessitera une intervention de l'utilisateur. Pendant le processus de mise à niveau, on obtient **la plupart du temps** un message d'erreur plus précis dans l'interface graphique.

La mise à niveau peut être lancée avec la propriété produit mode = valider ou mode = valider\_before\_upgrade. Dans ce cas, seul un test de la mise à niveau est lancé. Les erreurs éventuelles peuvent ainsi être détectées avant la mise à niveau proprement dite.

## **Solutions possibles aux problèmes rencontrés avec le paquet windows10-upgrade**

Il arrive malheureusement que des problèmes surviennent avec le paquet de mise à niveau de windows10 lui-même. En règle générale, il est recommandé de toujours utiliser la version la plus récente du paquet windows10-upgrade !

### **Problèmes de connexion automatique**

Il peut y avoir des problèmes avec la connexion automatique de l'opsiSetupUser. La plupart du temps, les stratégies de groupe empêchent l'opsiSetupUser temporaire de se connecter. Dans ce cas, il faut vérifier si la connexion n'est autorisée que pour certains utilisateurs locaux ou si la politique de mot de passe empêche la connexion.

Les boîtes de dialogue Windows qui nécessitent une interaction de l'utilisateur lors de la connexion peuvent également empêcher la connexion automatique. Le paquet de mise à niveau de windows10 supprime toutes les boîtes de dialogue connues, mais de nouvelles boîtes de dialogue empêchant la connexion peuvent toujours être ajoutées en fonction de l'ID de la version. Dans ce cas, une capture d'écran de la boîte de dialogue correspondante est toujours nécessaire afin qu'une solution de contournement puisse être intégrée dans la prochaine version du paquet windows10-upgrade.