

# 02\_Claude et le NAS

- [260801 - Mon Linux OS sur SASNEXTTE](#)
- [260528\\_Essai sur \\NASMAISON\foxy\Géographie\](#)
- [260611\\_procedure Komga-PDF par Claude sur StirlingPDF](#)
- [260614 - procedure Joplin-compression](#)
- [260615 - Immich performance](#)
- [Montages Rclone Sasnexte et Kdrive](#)
- [260727 - Claude en MCP sur le NASmaison](#)
- [260731-Claude et SASNEXTTE sur le PC Elio+Jux](#)

# 260801 - Mon Linux OS sur SASNEXTE

Claude prend la main en MCP pour monter et manager une distribution Linux ubiquiste - disponible

# 260528\_Essai sur \\NASMAISON\foxy\Géographie\

Workflow geo\_loud — Optimisation des fichiers lourds

Contexte

Le dossier \\NASMAISON\foxy\Géographie\ contient 1 312 fichiers (PDF et EPUB) pour un total de 32,9 Go. Parmi eux, 127 fichiers dépassent 75 Mo et représentent 13,6 Go — soit 41 % du volume total concentré sur moins de 10 % des fichiers.

L'objectif est de sortir ces fichiers lourds pour les optimiser (compression PDF), puis de les remettre exactement à leur emplacement d'origine sans perdre l'arborescence.

Architecture du système

Fichier    Rôle

nas\_indexer.py    Scanne le NAS et crée la base SQLite

nas\_geographie.db    Référentiel de tous les fichiers avec métadonnées et statut

nas\_geo\_loud.py    Script checkout/checkin — déplace et restaure les fichiers lourds

La base SQLite est la source de vérité : elle connaît à tout moment l'emplacement original de chaque fichier et son statut.

Colonnes de traçabilité (table fichiers)

Colonne    Description

chemin\_absolu    Chemin d'origine complet sur le NAS

chemin\_geo\_loud    Chemin dans le dossier de travail après déplacement

statut    original / deplace / restaure

date\_deplacement    Horodatage du déplacement

date\_restoration    Horodatage de la restauration

Workflow complet

Étape 1 — Indexation initiale

1

```
python nas_indexer.py "\\NASMAISON\foxy\Géographie" --output nas_geographie.db
```

Résultat : 1 312 fichiers indexés en ~22 secondes, 0 erreur.

Étape 2 — Export (checkout)

Déplace les 127 fichiers > 75 Mo vers \\NASMAISON\foxy\geo\_loud\ :

1

2  
3  
4

```
python nas_geo_loud.py export \  
--db nas_geographie.db \  
--dest \\NASMAISON\foxy\geo_loud \  
--confirmer
```

Ce que fait le script :

- Interroge la base pour tous les fichiers statut = 'original' et taille > 75 Mo
- Renomme chaque fichier {id:06d}\_{nom\_original} pour éviter les collisions de noms
- Déplace physiquement le fichier vers geo\_loud/
- Met à jour la base : statut = 'deplace', chemin\_geo\_loud, date\_deplacement

Le préfixe numérique garantit l'unicité même si deux sous-dossiers contiennent un fichier de même nom.

### Étape 3 — Optimisation PDF

Traiter les fichiers dans \\NASMAISON\foxy\geo\_loud\ avec l'outil d'optimisation. Les fichiers peuvent être traités dans n'importe quel ordre et en plusieurs sessions.

### Étape 4 — Restauration (checkin)

1  
2  
3

```
python nas_geo_loud.py restaurer \  
--db nas_geographie.db \  
--dest \\NASMAISON\foxy\geo_loud
```

Ce que fait le script :

- Lit tous les enregistrements statut = 'deplace' dans la base
- Pour chaque fichier présent dans geo\_loud/ : déplace vers son chemin\_absolu d'origine
- Recrée les sous-dossiers si nécessaire
- Met à jour : statut = 'restaure', date\_restoration
- Ignore les fichiers absents de geo\_loud/ (pas encore optimisés)

### Étape 5 — Vérification du statut

1

```
python nas_geo_loud.py statut --db nas_geographie.db
```

Affiche :

1  
2  
3  
4

| Statut | Fichiers | Total Go |
|--------|----------|----------|
|--------|----------|----------|

|          |       |        |
|----------|-------|--------|
| deplace  | 127   | 13.577 |
| original | 1 185 | 19.275 |

Vues SQL utiles (Metabase / DB Browser)

1  
2  
3  
4  
5  
6  
7  
8

-- Fichiers encore dans geo\_loud

```
SELECT nom, ROUND(taille_octets/1048576.0,1) AS mo, date_deplacement, chemin_absolu
FROM fichiers WHERE statut = 'deplace'
ORDER BY taille_octets DESC;
```

-- Bilan par statut

```
SELECT statut, COUNT(*) AS nb, ROUND(SUM(taille_octets)/1073741824.0,3) AS go
FROM fichiers GROUP BY statut;
```

Paramètres disponibles

| Paramètre | Défaut | Description |
|-----------|--------|-------------|
|-----------|--------|-------------|

|             |                           |                                    |
|-------------|---------------------------|------------------------------------|
| --db        | nas_geographie.db         | Chemin vers la base SQLite         |
| --dest      | \\NASMAISON\foxy\geo_loud | Dossier de travail                 |
| --seuil     | 75                        | Seuil en Mo pour la sélection      |
| --confirmer | (off)                     | Bypass la confirmation interactive |

Ré-indexation après optimisation

1

```
python nas_indexer.py "\\NASMAISON\foxy\Géographie" --output nas_geographie.db --vider
```

Fichiers du projet

| Fichier | Emplacement |
|---------|-------------|
|---------|-------------|

|                   |                                            |
|-------------------|--------------------------------------------|
| nas_indexer.py    | C:\Users\eliob\Documents\nas_indexer.py    |
| nas_geo_loud.py   | C:\Users\eliob\Documents\nas_geo_loud.py   |
| nas_rapport.py    | C:\Users\eliob\Documents\nas_rapport.py    |
| nas_geographie.db | C:\Users\eliob\Documents\nas_geographie.db |

# 260611\_procédure Komga-PDF par Claude sur StirlingPDF

## Présentation

La procédure Komga-PDF est un script de compression de fichiers PDF pour réduire considérablement la taille du stockage des bibliothèques d'ouvrages de la collection de Julien.

L'acquisition de fichiers issus de bases de données Internet accumule des fichiers PDF en haute résolution (images surtout). Ces fichiers sont synchronisés entre les NAS Synology et un disque dur kDrive Infomaniak de 6 To.

## Contexte

Des points de montage Rclone relient les répertoires kDrive et les containers Docker installés sur le VPS OVH juxjux.ovh. Ces containers lisent les PDF à la volée — permettant ainsi de construire une infrastructure de très forte densité de connaissances digitales (epub, pdf, images, vidéos, musiques...) avec une solution serveur d'entrée de gamme (10 €/mois tout inclus).

La lecture réseau est multi-support : PC, tablette, téléphone. Elle est aussi nomade, séquentielle et pressée. La compacité des fichiers PDF devient donc nécessaire pour diffuser rapidement sur les réseaux. Un magazine ou un ouvrage informatique ne doit pas peser 130 Mo sur un écran de 10 pouces.

## Procédure

- Les nouveaux PDF sont déposés dans le dossier Syncthing : `Syncthing/komga-pdf/` depuis n'importe quelle machine (PC Elio+Jux, PC Nexte, Ubuntu, Xiaomi...)
- Syncthing synchronise automatiquement vers le VPS : `/home/debian/Documents/komga-pdf/`

- Le service `komga-watch` (systemd VPS) détecte l'arrivée via inotifywait local
- Dès détection, `komga_compress_vps.py` compresse le PDF via StirlingPDF et dépose le résultat dans `/home/debian/Documents/komga-compressed/`
- Julien range manuellement les PDF compressés vers sasnexte puis dans le dossier thématique de son choix
- Les originaux ne sont jamais supprimés du VPS — Julien valide et conserve si besoin (haute résolution souhaitée)

# Architecture technique

| Élément                 | Chemin / URL                                                                                      |
|-------------------------|---------------------------------------------------------------------------------------------------|
| Dépôt (toutes machines) | <code>Syncthing/komga-pdf/</code>                                                                 |
| Dossier VPS (Syncthing) | <code>/home/debian/Documents/komga-pdf/</code>                                                    |
| Destination VPS         | <code>/home/debian/Documents/komga-compressed/</code>                                             |
| API compression         | <code>https://spdf.juxjux.ovh/api/v1/misc/compress-pdf</code>                                     |
| Scripts VPS             | <code>/home/debian/komga_compress_vps.py</code> +<br><code>/home/debian/komga_watch_vps.sh</code> |
| Logs                    | <code>journalctl -u komga-watch -f</code> (sur le VPS)                                            |

# Scripts

Le service tourne entièrement sur le VPS — plus de dépendance Ubuntu ni de SSH persistant.

| Fichier                                              | Rôle                                                                                                                                                     |
|------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>komga_compress_vps.py</code>                   | Compression : liste les PDFs locaux, envoie à StirlingPDF, écrit dans <code>komga-compressed/</code> . Idempotent (ignore les fichiers déjà compressés). |
| <code>komga_watch_vps.sh</code>                      | Surveillance : traite le backlog au démarrage, puis boucle inotifywait locale.                                                                           |
| <code>/etc/systemd/system/komga-watch.service</code> | Service system (pas user), actif au boot, <code>Restart=always</code> .                                                                                  |

# Gérer le service (VPS)

```
sudo systemctl status komga-watch
sudo systemctl restart komga-watch
```

# Retour d'expérience (2026-06-13)

## — Refactorisation VPS

**Problème de l'architecture Ubuntu** : le service tournait sur Ubuntu avec un SSH persistant vers le VPS. Deux défauts structurels :

- `inotifywait` ne détecte pas les fichiers déjà présents au redémarrage du service (backlog silencieux)
- Impossible à contrôler depuis Windows/Claude Code

**Solution** : service systemd sur le VPS lui-même. `inotifywait` local, pas de SSH persistant. Le script traite le backlog à chaque démarrage avant d'entrer dans la boucle de surveillance.

**Résultats de compression (2026-06-13) :**

| Fichier                                   | Original | Compressé | Gain        |
|-------------------------------------------|----------|-----------|-------------|
| Beaux Arts - Juin 2026.pdf                | 75.8 Mo  | 61.1 Mo   | -19%        |
| Connaissance des Arts - Juin 2026.pdf     | 57.3 Mo  | 35.2 Mo   | -39%        |
| La Revue du Vin de France - Juin 2026.pdf | 93.3 Mo  | 39.9 Mo   | -57%        |
| Livre lacto fermentation.pdf              | 96.6 Mo  | 19.6 Mo   | <b>-80%</b> |
| Monde Gourmand N°93 - Juin 2026.pdf       | 41.2 Mo  | 17.4 Mo   | -58%        |

**Points techniques :**

- Dossier VPS avec **D majuscule** : `/home/debian/Documents/` (Syncthing sensible à la casse)
- Race condition `inotifywait/Syncthing` : attente 3s après événement, script idempotent
- Python : `/usr/bin/python3` (système, `requests` 2.28.1 disponible)
- StirlingPDF auth : JWT via `POST /api/v1/auth/login` → `session.access_token`, valable 24h
- Fichiers `sync-conflict` Syncthing filtrés automatiquement par le script

# Retour d'expérience (2026-06-12)

**Résultats de compression :**

| Fichier                                  | Original | Compressé | Gain |
|------------------------------------------|----------|-----------|------|
| Monde Gourmand N°93 -<br>Juin 2026.pdf   | 42 Mo    | 17 Mo     | -58% |
| Connaissance des Arts -<br>Juin 2026.pdf | 57 Mo    | 35 Mo     | -39% |

# 260614 - procedure Joplin-compression

## Contexte — instructions de Julien (260614)

J'utilise l'application Joplin sur tous mes appareils. C'est ma mémoire de tout (travail, hobbies, maison...) et donc à force, la base de données se remplit. Elle se compose de notes à l'intérieur desquelles on trouve des images. Même en n'important que des captures d'écran, le poids de ces images est devenu important — de l'ordre de 300 Mo initialement estimé (476 Mo constatés en réalité, voir ci-dessous).

### **Appareils synchronisés sous Joplin :**

- VPS Juxjux.OVH (serveur de sync)
- PC Nexte (PC du travail de Julien)
- PC Elio+Jux (PC maison de Julien)
- Mobile Xiaomi T15pro de Julien
- Tablette Samsung S5 de Julien
- Session Ubuntu sur clé SSD de Julien

### **Objectif de la procédure :**

1. Compresser le stock d'images contenues dans la base de données Joplin via un process fiable de remplacement
2. Monter un système de compression automatique quotidienne des nouvelles images

### **Principes définis par Julien :**

1. Tenir un carnet de bord (observatoire) du poids digital de la BDD Joplin dans la page BookStack Joplin, avec les 15 pièces jointes les plus lourdes
  2. Mettre en place sur le VPS une procédure duplication → compression → remplacement d'images, qui se propage via la sync Joplin sur tous les appareils
-

# Exploration technique — Claude,

## 14/06/2026

## Infrastructure Joplin sur le VPS

Contrairement à ce qu'on pourrait attendre, Joplin Server **n'utilise pas SQLite** mais **PostgreSQL**.

| Container          | Image                | Rôle                                     |
|--------------------|----------------------|------------------------------------------|
| joplin             | joplin/server:latest | Serveur de sync Joplin                   |
| joplin-db          | postgres:15-alpine   | Base de données                          |
| joplin-nginx       | nginx:alpine         | Reverse proxy interne                    |
| joplin_to_obsidian | image custom         | Container migration (actif, sans impact) |

**Volume de données :** /home/debian/joplin-data → /home/joplin/.config/joplin (bind mount)

**Connexion DB :** POSTGRES\_HOST=joplin-db, POSTGRES\_DATABASE=joplin, POSTGRES\_USER=joplin

**Important :** le port 5432 de joplin-db n'est **pas exposé** à l'extérieur du réseau Docker. Tout script de manipulation doit tourner sur le VPS et se connecter via l'IP interne Docker.

## Structure de la base de données

La table centrale est `items` (23 tables au total). Chaque note, ressource et paramètre Joplin est une ligne dans cette table.

Colonnes clés :

- `content` (bytea) — données binaires brutes
- `content_size` (integer) — taille en octets
- `content_storage_id` = 1 → stockage de type `Database` (tout est dans PostgreSQL, pas de fichiers externes)
- `jop_type` — type d'item Joplin
- `updated_time` — timestamp de dernière modification (utilisé par les clients pour détecter les changements à sync)

**Répartition par type :**

| jop_type | Signification                   | Nombre | Poids total   |
|----------|---------------------------------|--------|---------------|
| 0        | Ressource (image/fichier joint) | 736    | <b>476 MB</b> |
| 1        | Note                            | 699    | 8,7 MB        |
| 4        | Tag                             | 733    | 4,5 MB        |
| 13       | NoteTag (relation note↔tag)     | 250    | 3,9 MB        |
| 2        | Carnet (Folder)                 | 94     | 707 KB        |
| 6        | Master Key                      | 87     | 19 KB         |
| 5        | Setting                         | 39     | —             |

# Analyse des ressources (jop\_type = 0)

Les ressources sont stockées comme **bytes bruts** dans la colonne `content`. Le format est détectable via les magic bytes :

- **PNG** (`\x89PNG`) — majorité des ressources
- **JPEG** (`\xff\xd8`) — portion significative
- **ZIP** (`PK\x03\x04`) — cas particulier : ZIP contenant plusieurs images `page_1.png`, `page_2.png`... (documents multi-pages)

La colonne `mime_type` est vide pour toutes les ressources — le type est implicite dans les bytes du contenu.

## Top 15 ressources les plus lourdes (état initial) :

| Rang | ID                      | Taille  | Format                                |
|------|-------------------------|---------|---------------------------------------|
| 1    | 0xh87pWrRt82z0DbO9n3yQ  | 12,2 MB | ZIP (page_1.png 7MB + page_2.png 5MB) |
| 2    | M0C32hKC2hMqPAxpRGiVgp  | 7,2 MB  | PNG                                   |
| 3    | UbAqQY3cEbH62violIHU6Pj | 6,0 MB  | PNG                                   |
| 4    | ZPQJVSjptMSxE71pV7JbAt  | 5,9 MB  | PNG                                   |
| 5    | h4Lhw1Trx9wgmD7doX9NyZ  | 5,9 MB  | PNG                                   |
| 6    | sbktcNb4lUj0ylovgtDt0fL | 5,3 MB  | PNG                                   |
| 7    | jtzuGpvrRTR12iLzwhcSNn  | 5,2 MB  | PNG                                   |
| 8    | VasloF2e9EGuNQt38aOFMx  | 4,9 MB  | JPEG                                  |

| Rang | ID                     | Taille | Format |
|------|------------------------|--------|--------|
| 9    | ZACDcQWzQzDLdnV7Qnf933 | 4,3 MB | PNG    |
| 10   | hExoJEEWT2tHz1demE5Nhm | 4,3 MB | PNG    |
| 11   | veu4HT3bStx07gRUIAEYvG | 4,2 MB | PNG    |
| 12   | bz9Twmb2F5lj0mPIQi48IB | 4,2 MB | JPEG   |
| 13   | TzWK21r4n0yvbEtJh02DGB | 4,2 MB | JPEG   |
| 14   | eD165w1bdEHJg0tq3qWok5 | 4,2 MB | PNG    |
| 15   | HD4SeqIx252nP9nh8HDVnH | 4,1 MB | PNG    |

# Architecture retenue

## Choix de compression — décision Julien, 14/06/2026

**PNG → JPEG 85%** (lossy). Toutes les images, qu'elles soient PNG ou JPEG à l'origine, sont converties/re-sauvegardées en JPEG qualité 85. Gain estimé : 50-75% par image. Acceptable pour des captures d'écran.

Pour les ZIP multi-pages : chaque image interne est convertie en JPEG 85%, le ZIP est reconstruit.

## Phase 1 — Observatoire

Script `joplin_observatoire.py` sur le VPS :

- Connexion psycopg2 à joplin-db via IP réseau Docker interne
- Calcul des stats globales (taille totale, nombre d'items par format)
- Liste des 15 ressources les plus lourdes avec format et taille
- Mise à jour de la page BookStack Joplin (page 166) avec ces informations

## Phase 2 — Compression (script principal)

Script `joplin_compress.py` sur le VPS :

**Connexion** : psycopg2 → IP Docker interne de `joplin-db` : 5432

### Traitement par ressource :

1. Lire le blob `content` depuis `items` (`jop_type=0`)
2. Détecter le format (magic bytes)
3. Ouvrir avec Pillow, convertir en JPEG 85 (`quality=85, optimize=True`)
4. Pour les ZIP multi-pages : dézipper → compresser chaque image → reconstruire le ZIP
5. Si le gain est > 5% : mettre à jour `content`, `content_size`, `updated_time` en base
6. Logger le résultat (ID, taille avant, taille après, ratio)

**Tracking des items traités** : fichier JSON local `/home/debian/joplin_compress_log.json` — évite de retraiter les ressources déjà compressées lors des passages quotidiens.

**Propagation sync** : Joplin détecte les changements via `updated_time`. Lors de la prochaine synchronisation de chaque client, les ressources compressées sont re-téléchargées automatiquement.

## Phase 3 — Service systemd (cron quotidien)

Timer systemd `joplin-compress.timer` → `joplin-compress.service` :

- Déclenchement quotidien (3h du matin)
- Traite uniquement les nouvelles ressources (non présentes dans le log JSON)
- Met à jour l'observatoire BookStack après chaque passage

## Mise en production — 14/06/2026

### Résultat du premier run (stock complet)

|                                |              |
|--------------------------------|--------------|
| <b>Ressources traitées</b>     | 736 au total |
| <b>Compressées</b>             | 424          |
| <b>Ignorées (gain &lt; 5%)</b> | 312          |
| <b>Erreurs</b>                 | 0            |
| <b>Poids avant</b>             | 476 Mo       |

|             |                 |
|-------------|-----------------|
| Poids après | 119,7 Mo        |
| Économie    | 356 Mo (-78,9%) |

# Scripts déployés sur le VPS

| Script                              | Rôle                                                                            | Options                                               |
|-------------------------------------|---------------------------------------------------------------------------------|-------------------------------------------------------|
| /home/debian/joplin_compress.py     | Compression Pillow PNG/JPEG → JPEG 85%, ZIP multi-pages, mise à jour PostgreSQL | --dry-run (simulation) / --limit N (N ressources max) |
| /home/debian/joplin_observatoire.py | Stats DB + top 15 → mise à jour page BookStack Joplin (ID 166)                  | —                                                     |

**Log tracking :** /home/debian/joplin\_compress\_log.json — liste des ressources déjà traitées, évite les doublons aux runs suivants.

# Timer systemd

|               |                                                        |
|---------------|--------------------------------------------------------|
| Service       | joplin-compress.service                                |
| Timer         | joplin-compress.timer                                  |
| Déclenchement | Chaque nuit à 3h UTC ( OnCalendar=*-*-* 03:00:00 )     |
| Logs          | /var/log/joplin-compress.log                           |
| Statut        | active (waiting) — prochain run : 15/06/2026 03:00 UTC |

# Notes techniques

- joplin-db IP Docker interne : 172.27.0.4 (peut changer si le container est recréé — vérifier avec docker inspect joplin-db -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}')
- La compression modifie content, content\_size et updated\_time dans la table items
- Les clients Joplin re-téléchargent les ressources modifiées lors de la prochaine synchronisation (détection via updated\_time)
- Les images à fond transparent (RGBA/LA/P) sont aplaties sur fond blanc avant conversion JPEG

# 260615 - Immich

## performance

### 260615 -instructions pour la procédure Immich Performance

Les photos de famille (et des vidéos) sont un patrimoine important pour notre famille. Donc elles font l'objet de beaucoup d'attention....et de volume de stockage. Les photos sont stockées sur les 2 NAS - Nasmaison et Sasnexte synchronisées à travers les outils Synology. A partir de Sasnexte, un script journalier Synology les propulse sur le VPS Kdrive pour être lues en lecture seule par toute la famille (elles sont également sur deux autres disques durs USB....).

Aujourd'hui, le dossier photos du Sasnexte que l'on doit considérer comme le cœur de ce dispositif nécessite une observation un peu plus fine car :

- des compressions vidéos doivent être faites pour les rendre plus faciles en lecture sur les portables en mobilité
- des compressions images lorsque le poids est aberrant pour l'usage
- des doublons existent forcément.

Immich performance est une procédure importante du livre de Claude.

Réponse de Claude

---

## Inventaire par Claude Code le 01/10/2025

```
SAS_NEXTE@NEXTE:~$ sudo du -h --max-depth=2 /volume1/photo/ | sort -h
0 /volume1/photo/1968-1994 - photos famille dont Noumea/1982
0 /volume1/photo/2009-mariage Zymny
0 /volume1/photo/2014- été/140714-Lyon chez Pierre
0 /volume1/photo/2020 - Automne Noel/noel2020
0 /volume1/photo/2024/202401 - Hiver Automne
0 /volume1/photo/2025/DCIM
0 /volume1/photo/@eaDir/clientd.tmp.dir
0 /volume1/photo/@eaDir/cloud.tmp.dir
0 /volume1/photo/@eaDir/@recycle
```

0 /volume1/photo/@eaDir/@tmp  
4.0K /volume1/photo/@eaDir/@drive.queues  
24K /volume1/photo/@eaDir/sharesnap\_share\_configuration  
32K /volume1/photo/2010 - été/@eaDir  
40K /volume1/photo/2011- Ete/@eaDir  
52K /volume1/photo/2021-été covidé/@eaDir  
84K /volume1/photo/@eaDir/980710-photo CV julien.jpg  
264K /volume1/photo/@eaDir/131124-elio pas content.jpg  
352K /volume1/photo/2009 Nouvel an Réveillon/@eaDir  
432K /volume1/photo/@eaDir/2015-soirée mexicaine.jpg  
440K /volume1/photo/1968-1994 - photos famille dont Noumea/Givet  
548K /volume1/photo/@eaDir/150314-Zoe en 18X24.png  
616K /volume1/photo/2009 Nouvel an Réveillon  
656K /volume1/photo/1968-1994 - photos famille dont Noumea/1977  
656K /volume1/photo/@eaDir/7b06435c-6bad-4eb0-aa38-af03bdfbfbcb8.jpg  
672K /volume1/photo/@eaDir/PhotoLab445379.jpg  
792K /volume1/photo/@eaDir/160220-elio en 18X24.png  
848K /volume1/photo/2011 - printemps/@eaDir  
900K /volume1/photo/2016 ACDC/@eaDir  
1.5M /volume1/photo/1968-1994 - photos famille dont Noumea/1976  
1.7M /volume1/photo/1968-1994 - photos famille dont Noumea/Pierre-001  
2.1M /volume1/photo/1968-1994 - photos famille dont Noumea/1969  
3.0M /volume1/photo/1968-1994 - photos famille dont Noumea/1978  
3.4M /volume1/photo/1968-1994 - photos famille dont Noumea/1968  
3.4M /volume1/photo/1968-1994 - photos famille dont Noumea/1980  
3.5M /volume1/photo/2014-mars-repas chez maman/@eaDir  
4.1M /volume1/photo/1999/@eaDir  
4.3M /volume1/photo/2014 - 4 ans Zoée/@eaDir  
4.9M /volume1/photo/1968-1994 - photos famille dont Noumea/1979  
4.9M /volume1/photo/2016 ACDC  
5.3M /volume1/photo/1986 - Californie/@eaDir  
5.3M /volume1/photo/Cartes de voeux/@eaDir  
5.5M /volume1/photo/2010-hiver/@eaDir  
6.0M /volume1/photo/1968-1994 - photos famille dont Noumea/1992  
6.6M /volume1/photo/1930-2016 - Photos BERTRAND MER/Elio  
6.6M /volume1/photo/2016-best photos pour 70 ans/Elio  
6.7M /volume1/photo/1999  
6.9M /volume1/photo/1968-1994 - photos famille dont Noumea/Etats-Unis 1986  
7.4M /volume1/photo/2015 - Noël/@eaDir  
7.6M /volume1/photo/1930-2016 - Photos BERTRAND MER/Les communions solennelle s  
7.6M /volume1/photo/2016-best photos pour 70 ans/Les communions solennelles  
7.8M /volume1/photo/1968-1994 - photos famille dont Noumea/1994  
7.9M /volume1/photo/2013-Serbie/@eaDir  
8.0M /volume1/photo/2014/@eaDir  
8.4M /volume1/photo/1968-1994 - photos famille dont Noumea/1990  
8.5M /volume1/photo/2004 - été/@eaDir

8.5M /volume1/photo/2022-Noel/@eaDir  
8.6M /volume1/photo/2015 - Automne/@eaDir  
8.9M /volume1/photo/1930-2016 - Photos BERTRAND MER/Les amis, la famille en noir et blanc  
8.9M /volume1/photo/2016-best photos pour 70 ans/Les amis, la famille en noir et blanc  
9.5M /volume1/photo/1968-1994 - photos famille dont Noumea/1981  
9.9M /volume1/photo/2018 - Noel/@eaDir  
10M /volume1/photo/#recycle/2016 - 70 ans maman\_DiskStation\_Nov-04-1745-2020 \_CaseConflict  
  
11M /volume1/photo/1930-2016 - Photos BERTRAND MER/Album de la famille de Re née Campanella  
11M /volume1/photo/2010 - naissance Zoée/@eaDir  
11M /volume1/photo/2014/Noël 2014  
11M /volume1/photo/2016-best photos pour 70 ans/Album de la famille de Renée Campanella  
12M /volume1/photo/2013 Marseille/@eaDir  
12M /volume1/photo/2014-mars-repas chez maman  
12M /volume1/photo/2015-Hiver/@eaDir  
12M /volume1/photo/2015-inventaire maison assurances/@eaDir  
13M /volume1/photo/1968-1994 - photos famille dont Noumea/1975  
14M /volume1/photo/1968-1994 - photos famille dont Noumea/1983  
15M /volume1/photo/2005 - été/@eaDir  
16M /volume1/photo/2016-mai juin/@eaDir  
16M /volume1/photo/2016 - Valras/@eaDir  
17M /volume1/photo/2004 - été  
17M /volume1/photo/2010 - naissance Zoée  
18M /volume1/photo/1968-1977/@eaDir  
18M /volume1/photo/2014 - 4 ans Zoée  
18M /volume1/photo/2014/Anniversaire\_Elio\_2014  
18M /volume1/photo/2016-avril avant crête/@eaDir  
19M /volume1/photo/1930-2016 - Photos BERTRAND MER/Grand-père Emile et mamie Rosa  
19M /volume1/photo/2016-best photos pour 70 ans/Grand-père Emile et mamie Ro sa  
21M /volume1/photo/1930-2016 - Photos BERTRAND MER/@eaDir  
21M /volume1/photo/1930-2016 - Photos BERTRAND MER/L'album de Papa et Maman  
21M /volume1/photo/2016-best photos pour 70 ans/L'album de Papa et Maman  
21M /volume1/photo/2016- photos Maman Papa/@eaDir  
22M /volume1/photo/2011 - Automne/@eaDir  
22M /volume1/photo/2013 Noël/@eaDir  
22M /volume1/photo/2014-Hiver Printemps/@eaDir  
23M /volume1/photo/2014 - Automne Noel/@eaDir  
24M /volume1/photo/2011 - jour de l'an/@eaDir  
24M /volume1/photo/2014-mur effondré - Ville de Toulon/@eaDir  
25M /volume1/photo/2009 - Papa/@eaDir  
25M /volume1/photo/2010-hiver  
25M /volume1/photo/2021-été covidé/2021 - après vacances  
25M /volume1/photo/Cartes de voeux  
26M /volume1/photo/1968-1994 - photos famille dont Noumea/1984  
27M /volume1/photo/1930-2016 - Photos BERTRAND MER/Enfants Mer et cousins

27M /volume1/photo/2008 Toscane/@eaDir  
27M /volume1/photo/2016 - 70 ans Maman/@eaDir  
27M /volume1/photo/2016-best photos pour 70 ans/Enfants Mer et cousins  
27M /volume1/photo/2022-automne a Noel/@eaDir  
29M /volume1/photo/2018 - Noel  
30M /volume1/photo/2008 - mariage Hélène et Jérôme/@eaDir  
30M /volume1/photo/2012 - Été/@eaDir  
30M /volume1/photo/#recycle/2016 - 70 ans maman\_DiskStation\_Nov-04-1745-2020  
\_CaseConflict\_1  
31M /volume1/photo/2025/2501 à 2502  
32M /volume1/photo/2015 - Noël  
33M /volume1/photo/2014/Carnaval 2014  
34M /volume1/photo/2016-best photos pour 70 ans/@eaDir  
35M /volume1/photo/2014-Barcelone/@eaDir  
36M /volume1/photo/2013/@eaDir  
36M /volume1/photo/2016-Parc Vert Coteau/soirée janvier 2016  
36M /volume1/photo/2024/2024 - Nouvel an Arriège  
37M /volume1/photo/2015 - Automne  
38M /volume1/photo/2014/Février 2014  
39M /volume1/photo/2011 - Noel/@eaDir  
39M /volume1/photo/2020 - Automne Noel/@eaDir  
40M /volume1/photo/2014/Sept\_2014\_4ans\_Zoée  
40M /volume1/photo/@eaDir/SYNO@.fileindexdb  
41M /volume1/photo/2016-Parc Vert Coteau/avril 2016 état  
41M /volume1/photo/2022-Noel  
43M /volume1/photo/@eaDir  
44M /volume1/photo/2009 - Papa  
44M /volume1/photo/2012/@eaDir  
45M /volume1/photo/1930-2016 - Photos BERTRAND MER/Papa juin 2009 - Le livre d'Alain  
45M /volume1/photo/2016-best photos pour 70 ans/Papa juin 2009 - Le livre d' Alain  
47M /volume1/photo/2005 - été  
47M /volume1/photo/2008 Toscane  
50M /volume1/photo/2019-Cabaret Vert/@eaDir  
51M /volume1/photo/2015 - Juillet Valras/@eaDir  
52M /volume1/photo/2021-hiver printemps/@eaDir  
53M /volume1/photo/2015-Hiver  
53M /volume1/photo/2018-défi Genes/@eaDir  
54M /volume1/photo/2008 - été Hautes Alpes/@eaDir  
54M /volume1/photo/2009 - Elio/@eaDir  
55M /volume1/photo/2013-Serbie  
58M /volume1/photo/2013 Marseille  
58M /volume1/photo/2015-inventaire maison assurances  
62M /volume1/photo/2007-1er janvier/@eaDir  
62M /volume1/photo/2016 - Valras  
64M /volume1/photo/2014- été/@eaDir  
67M /volume1/photo/2009 - été Irlande/@eaDir

67M /volume1/photo/2016-mai juin  
70M /volume1/photo/1968-1994 - photos famille dont Noumea/Nouméa  
70M /volume1/photo/2011- Ete/GARD juillet 2011  
71M /volume1/photo/2020 - Hiver Confinement COVID/@eaDir  
73M /volume1/photo/2022-hiver à Paques/@eaDir  
73M /volume1/photo/2024/202412 - Automne Noel  
75M /volume1/photo/2013 - Printemps/@eaDir  
76M /volume1/photo/2016-Parc Vert Coteau  
76M /volume1/photo/2024/@eaDir  
79M /volume1/photo/2012 - Eté/2012 - Bretagne  
79M /volume1/photo/2014-Hiver Printemps  
81M /volume1/photo/2013 Noël  
82M /volume1/photo/2017-Voyage à Turin/@eaDir  
84M /volume1/photo/2016-Venasque/@eaDir  
86M /volume1/photo/1986 - Californie  
88M /volume1/photo/2023-ete/@eaDir  
91M /volume1/photo/2009 - été Ardèche/@eaDir  
93M /volume1/photo/2016-avril avant crête  
94M /volume1/photo/2007-1er janvier  
96M /volume1/photo/2021-Automne et Noel/@eaDir  
97M /volume1/photo/2008 - été Hautes Alpes  
99M /volume1/photo/PhotoLibrary/2025  
100M /volume1/photo/2014- Porto - Primavera/@eaDir  
107M /volume1/photo/2014-mur effondré - Ville de Toulon  
108M /volume1/photo/2023 - Automne Noel/@eaDir  
111M /volume1/photo/2009 - été Irlande  
111M /volume1/photo/2015- printemps/@eaDir  
114M /volume1/photo/2025/2504 et 2505  
125M /volume1/photo/2010 - été  
125M /volume1/photo/2010 - été/Beauduc  
125M /volume1/photo/2014 - Automne Noel  
130M /volume1/photo/2016-Crete/@eaDir  
131M /volume1/photo/2011 - jour de l'an  
138M /volume1/photo/2021-été covidé/2021 avant vacances  
139M /volume1/photo/2013  
140M /volume1/photo/2018 - Valras/@eaDir  
143M /volume1/photo/2016-Automne Noel/@eaDir  
145M /volume1/photo/2013-Pays Bas/@eaDir  
146M /volume1/photo/2014-Barcelone  
150M /volume1/photo/2023-Hiver à Paques/@eaDir  
152M /volume1/photo/2008 - mariage Hélène et Jérôme  
153M /volume1/photo/2017-Automne Noel/@eaDir  
154M /volume1/photo/2014/Mai Juin 2014  
157M /volume1/photo/2014/Avril 2014  
159M /volume1/photo/2022-automne a Noel  
162M /volume1/photo/2009 - été Ardèche

173M /volume1/photo/2014 - 5 jours en Corse/@eaDir  
178M /volume1/photo/2016-best photos pour 70 ans/Ma famille pour mes 70 ans  
178M /volume1/photo/2020 -Printemps/@eaDir  
179M /volume1/photo/1930-2016 - Photos BERTRAND MER  
180M /volume1/photo/1968-1994 - photos famille dont Noumea  
181M /volume1/photo/2019- Hiver Printemps/@eaDir  
181M /volume1/photo/2022 - été Flandres NL Paris/@eaDir  
183M /volume1/photo/2012  
184M /volume1/photo/2009 - Elio  
186M /volume1/photo/2014/2014-été  
192M /volume1/photo/2016- photos Maman Papa  
203M /volume1/photo/1968-1977  
206M /volume1/photo/1967 à 2000 - photos famille Bertrand/@eaDir  
213M /volume1/photo/2023-Mai à Vacances Eté/@eaDir  
214M /volume1/photo/2019-Automne/@eaDir  
222M /volume1/photo/2015 - Juillet Valras  
229M /volume1/photo/2020 - Automne Noel  
230M /volume1/photo/2018-printemps/@eaDir  
258M /volume1/photo/2021-hiver printemps  
281M /volume1/photo/2014- été  
282M /volume1/photo/2018 - automne noel/@eaDir  
323M /volume1/photo/2011 - Noel  
325M /volume1/photo/2018-défi Genes  
348M /volume1/photo/2005 - naissance Elio/@eaDir  
354M /volume1/photo/2011 - printemps  
368M /volume1/photo/2011 - Automne  
381M /volume1/photo/2017-Hiver-Printemps/@eaDir  
384M /volume1/photo/2016-Venasque  
407M /volume1/photo/2020 -Printemps  
409M /volume1/photo/2016-best photos pour 70 ans  
419M /volume1/photo/2016 - 70 ans Maman  
434M /volume1/photo/2020 - Hiver Confinement COVID  
439M /volume1/photo/2017-Voyage à Turin  
460M /volume1/photo/2023-Mai à Vacances Eté  
462M /volume1/photo/2011- Ete/LIGURIE aout 2011  
500M /volume1/photo/2013 - Printemps  
511M /volume1/photo/2023-ete  
515M /volume1/photo/2022-hiver à Paques  
520M /volume1/photo/2012 - Eté/2012 - été Buis  
525M /volume1/photo/2016-Crete  
528M /volume1/photo/2014- Porto - Primavera  
531M /volume1/photo/2011- Ete  
535M /volume1/photo/2023-Vacances D NL B/@eaDir  
573M /volume1/photo/2023 - Automne Noel  
575M /volume1/photo/2016-été Ré Gers/@eaDir  
580M /volume1/photo/2018-Barcelone Primavera Sound/@eaDir

618M /volume1/photo/2019-Cabaret Vert  
629M /volume1/photo/2019 Bretagne Alpes/@eaDir  
635M /volume1/photo/2013-Pays Bas  
663M /volume1/photo/2021-Automne et Noel  
667M /volume1/photo/2014  
685M /volume1/photo/2015- printemps  
700M /volume1/photo/2017-été Scandinavie/@eaDir  
718M /volume1/photo/2024/2024 - Eté Italie Autriche  
759M /volume1/photo/2018 - Valras  
809M /volume1/photo/#recycle/2025  
817M /volume1/photo/2016-Automne Noel  
846M /volume1/photo/2025/2506 à 2509  
860M /volume1/photo/#recycle  
907M /volume1/photo/2019-Automne  
920M /volume1/photo/2017-Automne Noel  
935M /volume1/photo/2023-Hiver à Paques  
941M /volume1/photo/2024/2024-Porto  
944M /volume1/photo/cecile tri 2022/@eaDir  
945M /volume1/photo/PhotoLibrary/DCIM  
956M /volume1/photo/2005 - naissance Elio  
964M /volume1/photo/2020 - été Vienne Venise/@eaDir  
991M /volume1/photo/2025  
1.1G /volume1/photo/2014 - 5 jours en Corse  
1.1G /volume1/photo/2015-Croatie Autriche/@eaDir  
1.1G /volume1/photo/2018-the Alpen/@eaDir  
1.1G /volume1/photo/PhotoLibrary  
1.2G /volume1/photo/2021-été covidé/2021 - vacances Aout Mont Blanc Cantal  
1.2G /volume1/photo/2022 - été Flandres NL Paris  
1.3G /volume1/photo/2019- Hiver Printemps  
1.3G /volume1/photo/2021-été covidé  
1.4G /volume1/photo/2018-printemps  
1.5G /volume1/photo/2016-été Ré Gers/Vidéo Aveyron Ré et Gers  
1.5G /volume1/photo/2024/2024-Via Reggio  
1.8G /volume1/photo/2018 - automne noel  
1.9G /volume1/photo/2012 - Eté/2012- été Italia  
2.0G /volume1/photo/2017-Hiver-Printemps  
2.7G /volume1/photo/2012 - Eté  
2.8G /volume1/photo/2019 Bretagne Alpes  
2.8G /volume1/photo/2020 - été Vienne Venise  
2.9G /volume1/photo/1967 à 2000 - photos famille Bertrand  
3.2G /volume1/photo/2023-Vacances D NL B  
3.6G /volume1/photo/2018-Barcelone Primavera Sound  
3.7G /volume1/photo/cecile tri 2022  
3.8G /volume1/photo/2024  
4.1G /volume1/photo/2016-été Ré Gers  
4.2G /volume1/photo/2017-été Scandinavie

5.7G /volume1/photo/2018-the Alpen

6.4G /volume1/photo/2015-Croatie Autriche

79G /volume1/photo/

# Montages Rclone Sasnexte et Kdrive

# Montages Rclone Sasnexte et Kdrive

## Architecture

NAS Sasnexte (Synology)

└─ rclone sync (WebDAV) → kDrive Infomaniak

└─ rclone mount (FUSE) → VPS Jux

└─ containers Docker

Le NAS pousse les médias vers kDrive une fois par jour via le planificateur DSM. Le VPS monte kDrive en FUSE en permanence — les containers accèdent aux fichiers en lecture.

## Remote rclone (NAS sasnexte)

Fichier : `/volume1/homes/SAS_NEXTE/.config/rclone/rclone.conf`

```
[kdrive_music]
type = webdav
url = https://591617.connect.kdrive.infomaniak.com
vendor = other
user = julien.bertrand@nexte.fr
pass = [obfusqué rclone]
```

**Note** : le remote s'appelle `kdrive_music` mais sert à synchroniser tous les contenus (musique, photos, audiobooks, Komga). Nom historique à ne pas confondre avec un remote dédié musique.

**Mot de passe WebDAV** : régénéré le 2026-06-23 (compte `julien.bertrand@nexte.fr`, espace kDrive 591617). Mise à jour via `rclone config update kdrive_music pass $(rclone obscure NOUVEAU_MDP)`.

# Script unifié (NAS sasnexzte)

Emplacement : `/volume1/homes/SAS_NEXTE/scripts/sync_kdrive_complete.sh`

Planificateur DSM : tâche déclenchée quotidiennement — commande :

`/volume1/homes/SAS_NEXTE/scripts/sync_kdrive_complete.sh`

**Note sudo** : le script contient `sudo -u SAS_NEXTE rclone ...`. Si la tâche DSM est configurée pour tourner en tant que SAS\_NEXTE, retirer le `sudo -u SAS_NEXTE` (inutile et peut bloquer). Si elle tourne en root, le garder.

```
#!/bin/bash
RCLONE="/usr/local/bin/rclone"
REMOTE="kdrive_music"
LOG_DIR="/volume1/homes/SAS_NEXTE/logs"
DATE=$(date +%Y%m%d_%H%M%S)
SUMMARY_LOG="$LOG_DIR/sync_${DATE}.log"
mkdir -p "$LOG_DIR"
run_sync() {
    local name="$1"
    local source="$2"
    local dest="${REMOTE}:${3}"
    local log="$LOG_DIR/sync_${name}_${DATE}.log"
    echo "[$(date '+%Y-%m-%d %H:%M:%S')] === Debut $name ===" >> "$SUMMARY_LOG"
    sudo -u SAS_NEXTE "$RCLONE" sync "$source" "$dest" --delete-excluded --ignore-errors --delete-during --fast-list --exclude "@eaDir/**" --exclude "#recycle/**" --exclude "@__thumb/**" --exclude ".*@SynoResource" --exclude ".*@SynoEASstream" --transfers=4 --checkers=8 --timeout=5m --contimeout=2m --log-file="$log" -v
    local rc=$?
    echo "[$(date '+%Y-%m-%d %H:%M:%S')] === Fin $name (code=$rc) ===" >> "$SUMMARY_LOG"
}
run_sync "audiobooks" "/volume1/audiobooks" "SYNC-pour_VPS/Sync-SASNEXTE/audiobooks"
run_sync "music" "/volume1/music" "SYNC-pour_VPS/Sync-SASNEXTE/music"
```

```
run_sync "photo" "/volume1/photo/2026" "SYNC-pour_VPS/Sync-SASNEXTE/photo/2026"
run_sync "komga" "/volume1/Komga" "SYNC-pour_VPS/Sync-SASNEXTE/Komga"
```

**Déploiement** : écrire via `vi` depuis SSH — le heredoc et les éditeurs Windows introduisent des CRLF qui cassent bash. Après édition : `sed -i 's/\r//' script.sh` pour vérifier.

## Syncs actifs

| Nom        | Source NAS          | Destination kDrive                     | Notes                    |
|------------|---------------------|----------------------------------------|--------------------------|
| audiobooks | /volume1/audiobooks | SYNC-pour_VPS/Sync-SASNEXTE/audiobooks |                          |
| music      | /volume1/music      | SYNC-pour_VPS/Sync-SASNEXTE/music      |                          |
| photo      | /volume1/photo/2026 | SYNC-pour_VPS/Sync-SASNEXTE/photo/2026 | Année courante seulement |
| komga      | /volume1/Komga      | SYNC-pour_VPS/Sync-SASNEXTE/Komga      |                          |

**Non géré ici :**

- **Romans** : abandonné
- **Foxy** : sur NAS Maison (pas sasnexte)

## Options rclone — justifications

| Option                         | Rôle                                                                          |
|--------------------------------|-------------------------------------------------------------------------------|
| <code>--delete-excluded</code> | Supprime sur kDrive les fichiers exclus qui auraient pu y être uploadés       |
| <code>--ignore-errors</code>   | Poursuit la sync si un fichier est inaccessible (NAS Synology parfois occupé) |
| <code>--delete-during</code>   | Supprime les fichiers obsolètes au fil du scan, pas en fin de run             |
| <code>--fast-list</code>       | Réduit les appels API WebDAV (un seul listing récursif)                       |
| <code>@eaDir/**</code>         | Dossiers de métadonnées Synology (miniatures)                                 |
| <code>#recycle/**</code>       | Corbeille Synology                                                            |
| <code>@__thumb/**</code>       | Miniatures Synology                                                           |

| Option                                                       | Rôle                                                            |
|--------------------------------------------------------------|-----------------------------------------------------------------|
| <code>*.@SynoResource</code> / <code>*.@SynoEASstream</code> | Flux de ressources étendues Synology                            |
| <code>--transfers=4</code>                                   | 4 fichiers en parallèle — équilibre entre perf et charge WebDAV |
| <code>--checkers=8</code>                                    | 8 vérifications de hash en parallèle                            |
| <code>--timeout=5m</code>                                    | Timeout par opération I/O                                       |
| <code>--contimeout=2m</code>                                 | Timeout de connexion initiale                                   |

# Côté VPS — montages FUSE correspondants

Les services systemd sur le VPS montent les dossiers kDrive en FUSE :

| Service systemd                            | Point de montage                                    | Source kDrive                                       |
|--------------------------------------------|-----------------------------------------------------|-----------------------------------------------------|
| <code>rclone-audiobookshelf.service</code> | <code>/home/debian/audiobookshelf/audiobooks</code> | <code>SYNC-pour_VPS/Sync-SASNEXTE/audiobooks</code> |
| <code>kdrive-music.service</code>          | <code>/home/debian/music</code>                     | <code>SYNC-pour_VPS/Sync-SASNEXTE/music</code>      |
| <code>rclone-photo.service</code>          | <code>/home/debian/photo</code>                     | <code>SYNC-pour_VPS/Sync-SASNEXTE/photo</code>      |
| <code>kdrive-komga.service</code>          | <code>/home/debian/komga</code>                     | <code>SYNC-pour_VPS/Sync-SASNEXTE/Komga</code>      |

Remote VPS : `kdrive:` — remote natif Infomaniak ( `@infomaniak/mcp-server-kdrive` ), compte `julien.bertrand@live.fr`, kDrive ID `591617`.

## Logs

- Un fichier de log par sync par run : `/volume1/homes/SAS_NEXTE/logs/sync_{nom}_{date}.log`
- Bilan global : `/volume1/homes/SAS_NEXTE/logs/sync_{date}.log`
- Commande de suivi : `tail -f /volume1/homes/SAS_NEXTE/logs/sync_music_*.log`

## REX déploiement 2026-06-23

- **Problème 1** : CRLF dans le script (édition Windows) → `bash -x` montrait `\r` comme commande inconnue. Fix : réécrire via `vi` sur le NAS.
- **Problème 2** : mot de passe WebDAV périmé (401 Unauthorized) → régénéré le 2026-06-09 côté VPS mais pas mis à jour sur le NAS. Fix : `rclone config update kdrive_music pass $(rclone obscure MDP)`.
- **SSH bloqué** : pare-feu NAS + 2FA rendent le SSH depuis l'extérieur inaccessible. Passer par une session SSH locale ou DSM.
- **Résultat** : 4 syncs validés — audiobooks 48.9 GiB / 23 min, music 1 GiB / 45s, photo et komga déjà à jour.

# 260727 - Claude en MCP sur le NASmaison

Le NASmaison est un petit Synology DS218j installé à la maison de Julien. IL sert surtout de stockage des livres et des films. Egalement des cartes rasters, le dossier personnel de Cécile, Elio et Zoée. IL a une copie miroir du dossier musique et komga, mais plus photos.

Il est synchronisé avec le NAS SASNEXTE pour musique et Komga.

Il compose au quotidien 6 procédures hyperbacups de la bibliothèque de livres en direction du Kdrive :

- les livres de Foxy répartis par grands thèmes
- les livres de la nouvelle bibliothèque - biblio-calibre

Claude a désormais une connexion MCP privilégiée sur le NASmaison. Et peut notamment garantir le transfert sécurisé des dossiers

# 260731-Claude et SASNEXTE sur le PC Elio+Jux

## WireGuard sasnexste sur le PC Elio+Jux — diagnostic et résolution

**Date** : 2026-07-31 | **Machine** : PC Windows eliob | **NAS cible** : sasnexste (82.66.244.248) **Statut** : **RÉSOLU** — tunnel activé, lecteur S: mappé sur \\10.0.0.2\NEXTE

### Question initiale

WireGuard a été monté sur ce PC et sur le NAS sasnexste, mais le disque du NAS n'apparaissait pas comme lecteur dans l'explorateur de fichiers Windows. Pourquoi ?

### Réponse courte

Deux causes cumulées :

1. **Le tunnel n'était pas activé** sur le PC : la config `vps_maisonpc_sasnexste` était importée dans l'application WireGuard, mais jamais activée (aucun service `WireGuardTunnel$*`, aucun adaptateur réseau, `wg show` vide).
2. **Un lecteur réseau ne se crée jamais tout seul** : la découverte réseau Windows (broadcast/mDNS/WS-Discovery) ne traverse pas un tunnel de couche 3 — le NAS n'apparaîtra jamais spontanément dans « Réseau ». Il faut mapper manuellement une lettre avec `net use`.

# Topologie du tunnel

vps\_maisonpc\_sasnexe

Réseau **10.0.0.0/24**, hub-and-spoke avec le VPS Jux comme serveur (`wg0`, port UDP 51820) :

| IP       | Machine                                   | État (2026-07-31)                                                              |
|----------|-------------------------------------------|--------------------------------------------------------------------------------|
| 10.0.0.1 | VPS Jux (51.77.141.54) — hub              | Serveur, forwarding OK ( <code>ACCEPT</code> in <code>wg0</code> dans FORWARD) |
| 10.0.0.2 | NAS sasnexte (endpoint 82.66.244.248)     | Connecté, handshake actif, 485 Gio transférés                                  |
| 10.0.0.3 | PC Windows eliob                          | Connecté (après activation du tunnel)                                          |
| 10.0.0.4 | PC maison (peer <code>MkzKqDe...</code> ) | Jamais connecté (pas d'endpoint)                                               |

Clé publique serveur VPS : `zbsY/b14fHU1eR29PjTruK0Nrmp/BQ0RS1BGSh3Nkyo=`

## Diagnostic détaillé

### Côté PC (avant activation)

- Application WireGuard installée, service `WireGuardManager` Running
- Config importée (dossier `C:\Program Files\WireGuard\Data\Configurations` présent, chiffré DPAPI, admin uniquement) mais tunnel **inactif**
- Aucun `.conf` en clair dans le profil utilisateur — pour l'exporter : app WireGuard en admin → « Exporter les tunnels »

### Après activation du tunnel

- Adaptateur `vps_maisonpc_sasnexe` Up, IP `10.0.0.3/24`
- `wg.exe show` échoue en non-admin (`Permission denied`) — normal, utiliser `Get-NetAdapter` / `Get-NetIPAddress` pour vérifier sans élévation
- **Piège ICMP** : le NAS ne répond PAS au ping (pare-feu Synology) alors que le tunnel fonctionne. Ne pas diagnostiquer au ping — tester en TCP : `Test-NetConnection 10.0.0.2 -Port 445`
- Ports NAS via tunnel : 445/139 (SMB) OK, 22 (SSH) OK, 5000/5001 (DSM) **bloqués**

# SSH sasnexe depuis l'extérieur

Le SSH `sas_nexte@82.66.244.248` (IP publique) refusait le mot de passe le 2026-07-31 alors que le même mot de passe fonctionne en SMB via le tunnel → le mot de passe est bon, c'est le SSH public qui est filtré (fail2ban/whitelist). **Passer par le tunnel : `ssh sas_nexte@10.0.0.2` (port 22 ouvert).**

## Résolution appliquée (2026-07-31)

1. Tunnel activé dans l'app WireGuard (fait par Julien) → handshake OK avec le VPS
2. Identifiants SMB enregistrés dans le gestionnaire d'identification Windows :

```
cmdkey /add:10.0.0.2 /user:sas_nexte /pass:"xAPIJU5108$"
```

3. Lecteur mappé :

```
net use S: \\10.0.0.2\NEXTE /persistent:yes
```

4. Vérifié : `S:` liste bien le contenu du partage NEXTE

**Important** : le lecteur `S:` ne fonctionne que si le tunnel WireGuard est actif. Après un reboot, le tunnel se réactive automatiquement (service `WireGuardTunnel$vps_maisonpc_sasnexe` en démarrage auto) et Windows reconnecte `S:` grâce à `/persistent:yes` + identifiants cmdkey.

## Partages SMB disponibles sur \\10.0.0.2

ActiveBackupforBusiness, audiobooks, chat, home (dossier perso SAS\_NEXTE), homes, Iso VM, music, NetBackup, **NEXTE** (mappé sur S:), Public\_s, RAW, Ressources\_NEXTE, retro, web, web\_packages

Pour mapper un partage supplémentaire (les identifiants sont déjà enregistrés) :

```
net use R: \\10.0.0.2\Ressources_NEXTE /persistent:yes
```

## Points ouverts

- Finaliser la config Ubuntu : exporter le `.conf` depuis l'app WireGuard Windows (admin → Exporter les tunnels) — le peer 10.0.0.4 libre pourrait aussi servir pour Ubuntu
- Identifier le peer 10.0.0.4 (`MkzKqDeVoIX9R9oMlCK7c76e3c3p/DJe/ryd0eqEV20=`) : prévu pour le PC maison, jamais connecté