

# 00\_Recapitulatif

## 260802 - Allegelement JVM Komga + StirlingPDF (applique le 2026-08-02 a 22h)

Application des optimisations memoire validees le 2026-08-01, apres verification du backup quotidien (BILAN 5/5 OK). Les deux stacks ont ete redepoyees depuis `/home/debian/{komga,stirling}-compose-new.yml`, les composes precedents sauvegardes en `.bak-260802`.

| Service                | Avant                                                                                                          | Apres                                                   | Resultat                                                              |
|------------------------|----------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|-----------------------------------------------------------------------|
| Komga (stack 7)        | <code>-Xmx4g -Xms2g -XX:MaxRAMPercentage=70.0</code><br>limits 4500M / reservations 3000M<br>707 Mio residents | <code>-Xmx1g</code><br>limits 1500M / reservations 256M | <b>415 Mio</b> (-292 Mio, -41 %)                                      |
| StirlingPDF (stack 89) | JVM sans limite<br>722 Mio residents                                                                           | <code>-Xmx512m</code><br>limits 2g                      | <b>785 Mio</b> — pas de reduction, mais consommation desormais bornee |

**Bilan hote :** swap **3,7 Go -> 2,2 Go** (-1,5 Go), memoire disponible 4,4 -> 4,6 Gi. L'essentiel du gain vient de Komga : c'est le `-Xms2g` qui pre-allouait 2 Go de heap au demarrage.

**Pourquoi StirlingPDF ne baisse pas :** le process `java` conserve 799 Mio de RSS malgre `-Xmx512m` (512 Mio de heap + non-heap : metaspace, code cache, piles de threads), auxquels s'ajoutent LibreOffice (`soffice.bin`, ~98 Mio), `unoserver` et `Xvfb` que Stirling lance pour les conversions. Le benefice obtenu n'est pas une baisse mais un plafond : la consommation ne peut plus deborder.

**Limite portee de 1 Go a 2 Go avant application.** Le plan initial prevoyait 1 Go. Or l'OCR fait tourner ocrmypdf et tesseract comme processus Python *hors JVM* : `-Xmx` ne les borne pas, seul le plafond cgroup du container s'applique. Pic mesure a **894-969 Mio pour une seule page A4 300 dpi** — un plafond de 1 Go aurait provoque un OOM kill des le premier OCR. La version 1 Go est conservee en `/home/debian/stirling-compose-new.yml.origlg` mais ne doit pas etre utilisee.

**Verifications apres deploiement :** Komga repond 200 et son API `/api/v1/libraries` renvoie bien les bibliotheques ; StirlingPDF valide ses deux fonctions critiques, `compress-pdf` (utilise par les workflows komga-pdf et nas\_geo\_compress) et l'**OCR** (page A4 300 dpi, pic 969 Mio, aucun OOM kill, aucun redemarrage).

**Piege de deploiement :** les projets compose s'appellent `7` et `89` (les IDs de stack Portainer), pas `komga` / `stirlingpdf`. La commande correcte est `sudo docker compose -p 7 -f /var/lib/docker/volumes/portainer_data/_data/compose/7/docker-compose.yml up -d`.

**Rollback :** restaurer le `.bak-260802` de la stack concernee et redeployer de la meme facon.

# Vue d ensemble - VPS Jux + Claude Code

34 containers en production sur le VPS. 25 pages de documentation, une par service. 13 services disposent d un MCP actif dans Claude Code. Mis a jour le 2026-05-30.

| Page                              | URL                                                                               | MCP Claude Code                         | Statut integration |
|-----------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------|--------------------|
| <a href="#">01_Portainer</a>      | <a href="https://portainer.juxjux.ovh">https://portainer.juxjux.ovh</a>           | <code>mcp__portainer__*</code>          | MCP actif          |
| <a href="#">02_Bookstack</a>      | <a href="https://bookstack.juxjux.ovh">https://bookstack.juxjux.ovh</a>           | <code>bookstack-mcp-server (npm)</code> | MCP actif          |
| <a href="#">03_Homarr</a>         | <a href="https://homarr.juxjux.ovh">https://homarr.juxjux.ovh</a>                 | <code>mcp__homarr__*</code>             | MCP actif          |
| <a href="#">04_Audiobookshelf</a> | <a href="https://audiobookshelf.juxjux.ovh">https://audiobookshelf.juxjux.ovh</a> | <code>mcp__audiobookshelf__*</code>     | MCP actif          |
| <a href="#">05_BentoPDF</a>       | -                                                                                 | -                                       | API directe        |
| <a href="#">06_Cadvisor</a>       | -                                                                                 | -                                       | via Prometheus     |
| <a href="#">07_Calibre-web</a>    | -                                                                                 | -                                       | OPDS               |
| <a href="#">08_Dozzle</a>         | -                                                                                 | -                                       | via Portainer MCP  |
| <a href="#">09_FileBrowser</a>    | <a href="https://files.juxjux.ovh">https://files.juxjux.ovh</a>                   | <code>mcp__filebrowser__*</code>        | MCP actif          |
| <a href="#">10_FreshRSS</a>       | <a href="https://freshrss.juxjux.ovh">https://freshrss.juxjux.ovh</a>             | <code>mcp__freshrss__*</code>           | MCP actif          |
| <a href="#">11_Gitea</a>          | -                                                                                 | -                                       | API REST directe   |
| <a href="#">12_Grafana</a>        | <a href="https://grafana.juxjux.ovh">https://grafana.juxjux.ovh</a>               | -                                       | API REST directe   |

| Page                                              | URL                                                                     | MCP Claude Code                          | Statut integration                      |
|---------------------------------------------------|-------------------------------------------------------------------------|------------------------------------------|-----------------------------------------|
| <a href="#">13 Immich</a>                         | <a href="https://immich.juxjux.ovh">https://immich.juxjux.ovh</a>       | mcp__immich__*                           | MCP actif (5 containers)                |
| <a href="#">14 Jellyfin</a>                       | <a href="https://jellyfin.juxjux.ovh">https://jellyfin.juxjux.ovh</a>   | mcp__jellyfin__*                         | MCP actif                               |
| <a href="#">15 Joplin</a>                         | -                                                                       | -                                        | sync server uniquement                  |
| <a href="#">16 Kavita</a>                         | <a href="https://kavita.juxjux.ovh">https://kavita.juxjux.ovh</a>       | mcp__kavita__*                           | MCP actif                               |
| <a href="#">17 Komga</a>                          | <a href="https://komga.juxjux.ovh">https://komga.juxjux.ovh</a>         | mcp__komga__*                            | MCP actif                               |
| <a href="#">18 Mealie</a>                         | <a href="https://mealie.juxjux.ovh">https://mealie.juxjux.ovh</a>       | mcp__mealie__*                           | MCP actif                               |
| <a href="#">19 Navidrome</a>                      | <a href="https://navidrome.juxjux.ovh">https://navidrome.juxjux.ovh</a> | mcp__navidrome__*                        | MCP actif                               |
| <a href="#">20 NodeExporter</a>                   | -                                                                       | -                                        | via Prometheus                          |
| <a href="#">21 Prometheus</a>                     | -                                                                       | -                                        | API REST / PromQL                       |
| <a href="#">22 Readeck</a>                        | <a href="https://readeck.juxjux.ovh">https://readeck.juxjux.ovh</a>     | mcp__readeck__*                          | MCP actif                               |
| <a href="#">23 Syncthing</a>                      | <a href="https://syncthing.juxjux.ovh">https://syncthing.juxjux.ovh</a> | mcp__syncthing__*                        | MCP actif                               |
| <a href="#">24 Yourls</a>                         | -                                                                       | -                                        | API REST directe                        |
| <a href="#">25 StirlingPDF</a>                    | <a href="https://spdf.juxjux.ovh">https://spdf.juxjux.ovh</a>           | -                                        | API directe                             |
| <a href="#">A liens entre Claude et le KDRIVE</a> | <a href="https://kdrive.infomaniak.com">kdrive.infomaniak.com</a>       | @infomaniak/mcp-server-kdrive + API REST | MCP actif (kdrive_search) + API directe |

# Scripts de maintenance

|                         |                                                                     |                                                |
|-------------------------|---------------------------------------------------------------------|------------------------------------------------|
| fill_bookstack_pages.py | Remplit les pages avec les infos containers (image, ports, volumes) | C:\Users\eliob\.claude\fill_bookstack_pages.py |
| fill_bookstack_mcp.py   | Ajoute la section integration Claude Code / MCP a chaque page       | C:\Users\eliob\.claude\fill_bookstack_mcp.py   |

# Configuration MCP

Scripts dans C:\Users\eliob\.claude\, enregistres dans C:\Users\eliob\.mcp.json. Projet Claude Code principal : C:\Users\eliob. Endpoint Portainer Docker local : ID 3.

# 260611-REX VPS par Claude

Releve du 2026-06-11 via Prometheus (PromQL sur localhost:9090) + Portainer MCP.

## Etat global

|      |                             |
|------|-----------------------------|
| RAM  | 8.4 / 11.41 GB (73.6%)      |
| Swap | 3.99 / 4.0 GB (quasi plein) |
| CPU  | 5.2%                        |

## Top consommateurs RAM (RSS)

| Container            | RAM RSS |
|----------------------|---------|
| komga                | 1059 MB |
| Immich-SERVER        | 1011 MB |
| Immich-MICROSERVICES | 520 MB  |
| stirling-pdf         | 442 MB  |
| homarr2026           | 355 MB  |
| jellyfin             | 272 MB  |
| prometheus           | 259 MB  |
| jdownloader          | 220 MB  |
| Kavita               | 196 MB  |
| joplin               | 179 MB  |
| calibre-web-eyrolles | 132 MB  |
| mealie               | 129 MB  |
| grafana              | 125 MB  |
| cadvisor             | 91 MB   |
| navidrome            | 86 MB   |

Swap sature : komga + Immich representent ~2.5 GB a eux deux. StirlingPDF (442 MB) reste charge entre les appels.

# 260801-REX VPS par Claude

Releve du 2026-08-01 via docker stats + /proc en SSH (PC eliob).

## Etat global

|      |                                         |
|------|-----------------------------------------|
| RAM  | 8.2 / 11.41 GB (71.7%)                  |
| Swap | 4.0 / 4.0 GB (sature a 100%)            |
| CPU  | 28% (compression 7z du backup en cours) |

## Top consommateurs RAM (RSS)

| Container            | RAM RSS |
|----------------------|---------|
| Immich-SERVER        | 1147 MB |
| komga                | 791 MB  |
| Immich-MICROSERVICES | 618 MB  |
| stirling-pdf         | 566 MB  |
| jellyfin             | 380 MB  |
| homarr2026           | 313 MB  |
| Immich-DB            | 310 MB  |
| prometheus           | 282 MB  |
| joplin               | 241 MB  |
| Kavita               | 240 MB  |
| cadvisor             | 195 MB  |
| grafana              | 164 MB  |
| joplin-db            | 158 MB  |
| yourls-db            | 157 MB  |
| audiobookshelf       | 140 MB  |

Swap sature : komga (760 MB) + StirlingPDF (406 MB) en tete des processus swappes, Immich ~330 MB. Des limites memoire ont ete posees depuis juin sur les gros containers (komga 4.4G, Immich 2G, Kavita 4G, joplin 2G). Decouvertes du jour : GNOME/gdm3 tourne a vide sur tty1 depuis le 8 juillet (98 paquets installes, ~150 MB RAM, aucun acces distant configure) ; sauvegardes Immich vides depuis l'origine (pg\_dump sans -d immich, archives de 4 Ko) — corrigees le 2026-08-

01, dump reel 307 MB.

# Actions realisees le 2026-08-01 (soir)

| Action                         | Detail                                                                                                                                                                                                                                                                                                                                             | Gain                        |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|
| Fix backup Immich              | <code>-d immich</code> ajoute dans <code>/opt/backups/vps_backup.sh</code> + run manuel complet 5/5 OK (archive 87 MB sur kDrive, premiere sauvegarde Immich valide)                                                                                                                                                                               | donnees enfin sauvegardees  |
| Fusion Immich                  | 5 containers → 3 : SERVER unique (API + workers, <code>NODE_OPTIONS=-max-old-space-size=768</code> ), MICROSERVICES et LEARNING supprimees (recherche semantique et visages non utilises). Ancien compose : <code>compose/29/docker-compose.yml.bak-260801</code>                                                                                  | ~900 MB RAM                 |
| Arret cAdvisor                 | <code>docker update --restart=no cadvisor</code> + stop. Plus de metriques par container dans Prometheus/Grafana (node-exporter conserve le host)                                                                                                                                                                                                  | ~200 MB RAM + 4-8% CPU      |
| GDM/GNOME desactive puis purge | gdm3 + gnome-shell + Xorg tournaient a vide sur tty1 depuis le 2026-07-08. <code>systemctl disable --now gdm</code> puis purge apt de 263 paquets (bureau complet + LibreOffice, jeux...). Verifie avant : ens3 gere par systemd-networkd, NetworkManager non critique. Verifie apres : 30 containers up, nginx/docker/ssh/wg actifs, sites en 200 | ~150 MB RAM + 3,5 GB disque |

**Resultat global du 2026-08-01** : swap 4,0/4,0 GB (sature depuis juin) → 2,6/4,0 GB ; disque 62% → 59% ; ~1,3 GB de RAM liberes. Immich verifie fonctionnel par Julien.

# Plan du 2026-08-02 a 19h (tache planifiee Claude Code)

Apres verification du backup nocturne (BILAN 5/5 OK), appliquer les deux fixes JVM prepares et valides (fichiers en attente dans `/home/debian/`) :

| Service         | Avant                                                                               | Apres                                            | Gain attendu                   |
|-----------------|-------------------------------------------------------------------------------------|--------------------------------------------------|--------------------------------|
| Komga (stack 7) | <code>-Xmx4g -Xms2g</code> (2 GB pre-alloues, ~760 MB en swap), limites 4500M/3000M | <code>-Xmx1g</code> sans Xms, limites 1500M/256M | ~400-500 MB RAM + ~700 MB swap |

| Service                | Avant                                                  | Apres                                                  | Gain attendu    |
|------------------------|--------------------------------------------------------|--------------------------------------------------------|-----------------|
| StirlingPDF (stack 89) | JVM sans plafond (566-643 MB residents + ~406 MB swap) | <code>JAVA_TOOL_OPTIONS=-Xmx512m</code><br>, limite 1g | ~300-400 MB RAM |

Verifications prevues : komga.juxjux.ovh en 200, test compress-pdf sur spdf.juxjux.ovh (fonction critique des workflows komga-pdf et geo\_loud), docker stats, free -h (swap attendu sous ~2 GB).  
Rollback : restaurer les `.bak-260802`. Nota : BentoPDF n'existe plus sur le VPS (container supprime) — il ne peut de toute facon pas remplacer StirlingPDF (pas d'API serveur, traitement 100% navigateur).