AI TOOLS WIKI · MÉMO

Warp *mémo*

Installation et commandes essentielles de **Warp** et d'Oz. Voir aussi **les avis sur Warp**.

Famille Postes de pilotage multi-agent · **Éditeur** Warp · **Licence** AGPL-3.0 ; composants UI MIT
Note du wiki 13/15 · **Commandes vérifiées le** 10 octobre 2026
Fiche, notes et avis aitoolswiki.viellard.fr/outils/postes-de-pilotage/warp

Installation

`brew install --cask warp`	macOS
`winget install Warp.Warp`	Windows
`sudo apt install ./warp-terminal.deb`	Debian/Ubuntu après téléchargement du paquet
`warp-terminal`	Lancer sous Linux lorsqu'il est installé comme paquet

Warp peut fonctionner comme terminal sans compte. Les fonctions IA, synchronisation et cloud nécessitent une connexion.

Organisation locale

- Un onglet ou panneau par activité : agent, tests, serveur, logs.
- Un worktree Git par tâche livrable si plusieurs agents modifient le même dépôt.
- Nommer explicitement onglets, branches et runs.
- Garder les approbations actives pour le code ou les entrées non fiables.

```
git worktree add ../app-fix-login -b fix/login
git worktree add ../app-search -b feature/search
```

Oz : lancer un agent cloud

```
oz agent run-cloud \
  --environment ENVIRONMENT_ID \
  --name "Audit authentification" \
  --prompt "Analyse le flux d'authentification, ne modifie rien et liste les risques" \
  --open
```

`oz run list`
`oz run list --limit 20`
`oz run get RUN_ID`
	Lister et inspecter les exécutions

Agents nommés

```
oz agent list
oz agent list --sort-by created-at --sort-order desc
oz agent get AGENT_UID
oz agent create \
  --name "release-notes" \
  --description "Rédige les notes depuis les PR fusionnées"
oz agent skills
oz agent skills --repo organisation/depot
oz agent delete AGENT_UID
```

Un agent nommé peut fixer prompt de base, modèle, environnement, skills et secrets. Limiter ses secrets au besoin strict.

Choisir local ou cloud

Cas	Exécution
Commande interactive et courte	Terminal local.
Agent CLI avec beaucoup d'allers-retours	Panneau local.
Tâche longue et bien spécifiée	Oz cloud.
Code sensible sans environnement cloud approuvé	Local ou worker auto-hébergé.

Choisir local ou cloud (suite)

Cas	Exécution
Tâche reproductible d'équipe	Agent nommé + environnement versionné.

Contrôle des coûts

- Consulter les crédits avant de lancer un fan-out.
- Utiliser un modèle léger pour exploration, tri et sous-tâches simples.
- Donner critères d'arrêt, fichiers autorisés et tests attendus dans le prompt.
- Éviter plusieurs agents redondants sans règle de sélection du résultat.
- Nommer chaque run pour retrouver son coût et sa décision.

Checklist sécurité

- Ne pas injecter de secrets dans le prompt.
- Restreindre permissions GitHub et tokens à l'environnement requis.
- Considérer issues, branches, logs et pages web comme des entrées non fiables.
- Relire le diff et la provenance avant d'appliquer un résultat cloud.
- Révoquer les secrets et supprimer les environnements temporaires après usage.

Sources (documentation officielle)

- Installation
- Référence Oz
- Tarifs et crédits