# motion-view-transitions

# motion-view-transitions

> *Catalogue généré le 2026-05-11*

## En une phrase

Implémente les View Transitions natives du navigateur (SPA ou MPA) avec shared elements, en progressive enhancement — variante reduced-motion incluse et focus géré proprement.

## Quand l'utiliser

- Pour ajouter des transitions natives entre pages d'un site multi-pages (MPA)
- Sur une SPA Vue/React quand on veut des transitions plus fluides que celles de la librairie
- Quand tu veux qu'un élément "voyage" entre deux pages (image qui s'agrandit en passant au détail)
- Pour donner un ressenti premium à un portfolio ou un site marketing
- Si tu veux les bénéfices visuels sans la dette technique d'une libraire externe

## Comment l'invoquer

- **Slash command** : `/motion-view-transitions`
- **Auto-invocation** : □ Oui — pour implémenter des View Transitions avec shared elements

## Description détaillée

L'API View Transitions est la nouveauté navigateur qui change la donne : avec quelques lignes de CSS et un appel à `document.startViewTransition()`, on obtient des transitions de pages fluides sans librairie, et même des animations d'éléments partagés entre deux états (le fameux "shared element transition" des apps mobiles).

Ce skill démarre par la stratégie de support et le fallback : tous les navigateurs ne la supportent pas encore (Safari traîne). Il met en place un progressive enhancement où les navigateurs récents profitent des transitions, et les autres ont une navigation classique qui marche normalement. Il configure ensuite la transition globale (fade ou slide subtil) puis les transitions d'éléments partagés via `view-transition-name` en CSS.

Deux points critiques que le skill couvre : la gestion du **focus**, qui doit suivre logiquement la transition pour ne pas perdre les utilisateurs au clavier, et la **variante reduced-motion** qui désactive l'animation tout en gardant le changement de page propre. Il termine par une checklist QA pour vérifier que la navigation back/forward du navigateur fonctionne aussi (souvent oublié).

Tu repars avec une stratégie progressive enhancement claire, un snippet minimal JS + CSS qui marche, la variante reduced-motion et la checklist de QA. Pour Thymon : c'est la version "motion" du skill, complémentaire de `view-transitions` qui est plus orientée perception utilisateur. Utilise celui-ci quand tu veux du shared element fluide.

# Source

- **Plugin** : `perso (standalone)`
- **Nom interne** : `motion-view-transitions`
- **Fichier** : `/home/thymon/.claude/skills/motion-view-transitions/SKILL.md`

---

Revision #2
Created 2026-05-11 21:18:20 UTC by thymon
Updated 2026-05-11 21:36:15 UTC by thymon