JavaScript, Node.js-Apps und Exponential

Umzüge zwischen Exponential 6 und JavaScript-Anwendungen – Next.js und andere statische oder serverseitig gerenderte Frontends, Headless-Setups, Markdown- und MDX-Inhalte in einem Repository – einschließlich der Nutzung…

Source file: `doc/ger-de/installation/migration/javascript-anwendungen.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/install/migration/javascript-apps.md

Eine JavaScript-Site ist ein Frontend; ihre Inhalte liegen woanders: in Markdown- oder MDX-Dateien im Repository, in einem Headless-CMS, in einer Datenbank hinter einer API oder im Code selbst. „Eine Next.js-Site migrieren“ bedeutet also zwei Entscheidungen: **wohin die Inhalte gehen** (nach Exponential) und **was die Site rendert** (die Templates von Exponential oder das bestehende Frontend, das Exponential ausliest).

Übersicht Migration · Checkliste · Abbildung · Werkzeuge

Inhalt

- 1. Wo die Inhalte einer JavaScript-Site liegen
- 2. Inhalte nach Exponential
- 3. Das JavaScript-Frontend behalten: Exponential als seine Inhaltsquelle
- 4. Adressen und Weiterleitungen
- 5. Von Exponential zu einem JavaScript-Frontend
- 6. Checkliste

1. Wo die Inhalte einer JavaScript-Site liegen

Inhaltsquelle	Wie Sie sie auslesen
Markdown- oder MDX-Dateien (Blogs, Dokumentationssites)	die Dateien selbst: Front Matter (YAML) plus Inhalt
Ein Headless-CMS (Contentful, andere)	der Export oder die API dieses CMS; zu Contentful siehe Contentful
JSON- oder YAML-Datendateien	die Dateien
Eine Datenbank hinter der eigenen API der App	die API oder direkt die Datenbank (andere Anwendungen)
Fest in Komponenten codiert	das gerenderte HTML (Crawl, statisches HTML)

Next.js speichert selbst keine Inhalte. Laut seiner Dokumentation lässt `output: 'export'` den Befehl `next build` statisches HTML, CSS und JavaScript nach `out/` schreiben (`next export` wurde in Version 14 entfernt). Diese Ausgabe ist ein vollständiges Crawl-Ziel für die Methode für statisches HTML, wenn sich die Quellen nur schwer auslesen lassen.

2. Inhalte nach Exponential

1. **Sammeln Sie die Quellen**: Listen Sie die Inhaltsverzeichnisse, Datendateien und APIs auf, die die App zur Build- oder Anfragezeit liest (sehen Sie sich `getStaticProps`/ `generateStaticParams`, Loader oder die Datenschicht Ihres Frameworks an).
2. **Wandeln Sie Markdown/MDX um**: Lesen Sie in Node.js das Front Matter ein und rendern Sie Markdown mit der Bibliothek, die die Site bereits verwendet, zu HTML, damit das Ergebnis dem entspricht, was die Besucher gesehen haben. MDX-Komponenten (`<Callout>`, `<Video>`) haben keine HTML-Entsprechung: Bilden Sie jede auf einen Custom Tag oder eine Einbettung von Exponential ab oder rendern Sie sie zu einfachem HTML.
3. **Schreiben Sie JSON-Zeilen**: Remote-ID aus dem Dateipfad oder Slug (`md-blog/2024/hello`), Titel, Datum, Schlagwörter, Autor, das HTML, die Route.
4. **Importieren Sie** mit dem Skript-Grundgerüst auf dem Server oder mit dem Importer von xrowextract mit `--map`. Medien aus `public/` werden zu Bild-/Dateiobjekten.
5. **Templates**: Bauen Sie TPL-Templates oder behalten Sie das Frontend (3).

3. Das JavaScript-Frontend behalten: Exponential als seine Inhaltsquelle

Exponential kann die Headless-Inhaltsquelle eines bestehenden Frontends sein; Redakteure arbeiten in der Administration von Exponential, das Frontend holt die Inhalte zur Build- oder Anfragezeit.

Schnittstelle	Verwendung	Referenz
REST v1/v2 (`ezprestapi`)	Knoten, Unterknoten und Felder als JSON lesen: `GET /api/ezp/v2/content/node/<id>`, `.../list/offset/<n>/limit/<n>`, `?ResponseGroups=Fields`, `?OutputFormat=xhtml` für den gerenderten Inhalt	REST-API v1 und v2
Persönliche API-Schlüssel	ein Build-Server liest mit `Authorization: Bearer expk_...`, beschränkt auf die Scopes des Schlüssels und die Richtlinien seines Eigentümers	Persönliche API-Schlüssel
OAuth2	eine App, die für angemeldete Benutzer handelt	die REST-Anleitung, Abschnitt 3.1
`expservices`	die Funktionen der Administration als benannte HTTP-Dienste (Inhalte, Benutzer, Medien, Tags, Layouts ...)	Remote-Dienste und Apps

Schreibzugriffe über REST sind eingeschränkt (Anlegen und Löschen ab v2; Aktualisieren antwortet derzeit mit 501), deshalb werden Massenimporte auf dem Server ausgeführt, nicht vom Frontend aus.

4. Adressen und Weiterleitungen

- Behalten Sie die Routen: Geben Sie jedem Knoten den Alias seiner alten Route, damit das Frontend (oder die eigenen Templates von Exponential) dieselben Pfade ausliefern kann.
- Laut der Dokumentation von Next.js unterstützt ein statischer Export **keine** `redirects`, `rewrites` oder `headers` aus `next.config.js`, ebenso wenig Route Handlers, die die Anfrage verwenden, ISR, Server Actions oder die standardmäßige Bildoptimierung: Weiterleitungen

einer statisch exportierten Site gehören zum Hoster oder Webserver.

- Wenn Exponential die Site rendert, sind die Weiterleitungen URL-Aliase und Platzhalter (Checkliste, 4).

5. Von Exponential zu einem JavaScript-Frontend

- **Behalten Sie Exponential als Inhaltsquelle** und bauen Sie das neue Frontend gegen seine REST-API (3).
- **Oder ziehen Sie die Inhalte heraus**: Exportieren Sie mit xrowextract nach JSON (Exponential verlassen), rendern Sie XmlText zu HTML oder Markdown und schreiben Sie die Dateien oder das Headless-CMS Ihrer Wahl.
- Bild-Aliase (skalierte Varianten) erzeugt Exponential; eine statische Site braucht ihre eigene Bildverarbeitung.

6. Checkliste

☐ Jede Inhaltsquelle der App aufgelistet (Dateien, Daten, APIs, fest codiert).

☐ Markdown mit dem eigenen Renderer der Site gerendert; MDX-Komponenten abgebildet.

☐ Mit Remote-IDs aus Pfaden oder Slugs importiert; Medien importiert.

☐ Entscheidung getroffen: Templates von Exponential oder das Frontend, das Exponential ausliest (Scopes der API-Schlüssel gesetzt).

☐ Routen erhalten oder beim Hoster weitergeleitet.

Quellen

- Statische Exporte in Next.js: https://nextjs.org/docs/app/guides/static-exports

> **Werden Sie Exponential-Partner**
>
> Partner teilen ihre Informationen, stellen Mittel bereit, geben Beiträge zurück und entwickeln die Produkte mit uns, und so wächst Exponential. Eine Partner-Website und ein Dashboard sind für Mitte 2027 geplant: Werden Sie noch heute Partner, damit wir schneller wachsen und neue Funktionen früher liefern.
>
> Partner: hier beginnen | Wie Sie sich bewerben

Diese Seite ist Teil der Exponential-Dokumentation und freie Dokumentation unter der GNU Free Documentation License 1.3 oder einer späteren Version (die Lizenz dieser Dokumentation).

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.