The background of the slide features a wide-angle photograph of a natural landscape. In the foreground, there's a dirt path leading through a field of dry, yellowish-green grass. To the left, a large, turquoise-blue lake stretches towards the horizon. In the background, a range of mountains with snow-capped peaks rises against a bright blue sky with scattered white clouds.

NILS HARTMANN

<https://nilshartmann.net>

Modern React SPAs with TanStack Router and Query

NILS HARTMANN

nils@nilshartmann.net

Freelance Software Developer, Architect and Trainer from Hamburg

Java, Spring, GraphQL, React, TypeScript



<https://graphql.schule/video-kurs>

<https://reactbuch.de>

HTTPS://NILSHARTMANN.NET

TanStack Query

(formerly known as React Query)

<https://tanstack.com/query/latest>

All-inclusive DataFetching solution for React (and others)

Bye, bye, useEffect

Suspense, Caching, TypeSafety, Error Handling

TanStack Router

<https://tanstack.com/router/latest>

Filebased, typesafe Routing

Search Params as global State

TanStack Query

Seems to be de-facto standard nowadays

(often in combination with zod)

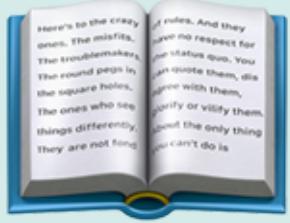
I really like it and would use it in almost all cases

(Exceptions: GraphQL, Redux)

Code

Examples

[HTTP://LOCALHOST:24080](http://localhost:24080)



Typesafe Routes

- 🕵️ \$bookId/index.tsx
- 🕵️ <Link> in BookList.tsx
- 🤑 useParams bri

Lazy Routes

-  about.tsx
-  Network traffic

Loading data with suspense

- 🕵️ book-query.ts
- 🕵️ \$bookId/index.tsx

Pending state

- 🕵️ book-query.ts - slowdown
- 🕵️ \$bookId/index.tsx - pendingComponent

Validation using zod

- 🧑 book-types.ts: add zod
- 🧑 book-query.ts – TBookDetails.parse

Router: State as search params

- 🕵️ books/index.tsx – define search params
- 🕵️ BookmarkButton.ts – consume search params
- 🕵️ event-store.ts – enabledLogEvents = ["Rendered"]

Query cache

- 🧑 add 40_query_cache/00_material
- 🧑 BookDetails.tsx add <ReviewLink />
- 🧑 ReviewLink.tsx implement
- 🧑 reviews/index.tsx: Load book data
- 🧑 What is happening when on the network? What is happening in the cache?
 - Navigate from bookId -> reviews
 - Reload reviews

Router: Early loading

- 🧑 add 45_before_load/00_material with route.tsx
- 🧑 Implement route.tsx
- 🧑 What is happening when on the network? What is happening in the cache?
 - Navigate from bookId -> reviews
 - Reload reviews
 - Still slow, but Requests in parallel
 - There are other ways to achieve the same

Mutations

- 🧑 Remove slowdown in bookByIdQueryOpts
- 🧑 __root.tsx: enable BookOfTheMonth
- 🧑 BookOfTheMonth: Load book
 - Note: again the queryOpts
 - Note: again it's coming from the cache
- 🧑 Enable LikeButton in BookDetails
- 🧑 like-mutation: implement mutation
- 🧑 LikeButton: implement mutation

Summary

Should I use it?

TanStack Query

Seems to be de-facto standard nowadays

(often in combination with zod)

I really like it and would use it in almost all cases

(Exceptions: GraphQL, Redux)

SUMMARY

TanStack Query

TanStack Query

Seems to be de-facto standard nowadays

(often in combination with zod)

TanStack Query

Seems to be de-facto standard nowadays

(often in combination with zod)

I really like it and would use it in almost all cases

(Exceptions: GraphQL, Redux)

SUMMARY

TanStack Router

SUMMARY

TanStack Router

Would I use it? Probably, but...

TanStack Router

Would I use it? Probably, but...

Quite new (less than one year!)

TanStack Router

Would I use it? Probably, but...

Quite new (less than one year!)

Very active development (can be hard to keep up)

TanStack Router

Would I use it? Probably, but...

Quite new (less than one year!)

Very active development (can be hard to keep up)

Not as widely used (as React Router)

Example application

Source-Code and Slides:

<https://react.schule/ijs-tanstack>

I will commit my live coding source code on an own branch later

Thank you very much!

Slides & Code: <https://react.schule/ijs-tanstack>

Questions and contact:

nils@nilshartmann.net

<https://nilshartmann.net/contact>

