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 grass and small yellow flowers. Beyond the path is a large, turquoise-blue lake. In the distance, 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

CAN I USE REACT WITHOUT A "FRAMEWORK"?

Can I use React without a framework?

[^ Hide Details](#)

You can definitely use React without a framework—that's how you'd [use React for a part of your page](#). However, if you're building a new app or a site fully with React, we recommend using a framework.

Here's why.

Even if you don't need routing or data fetching at first, you'll likely want to add some libraries for them. As your JavaScript bundle grows with every new feature, you might have to figure out how to split code for every route individually. As your data fetching needs get more complex, you are likely to encounter server-client network waterfalls that make your app feel very slow. As your audience includes more users with poor network conditions and low-end devices, you might need to generate HTML from your components to display content early—either on the server, or during the build time. Changing your setup to run some of your code on the server or during the build can be very tricky.

<https://react.dev/learn/start-a-new-react-project#can-i-use-react-without-a-framework>

TanStack

<https://tanstack.com/>



TANSTACK

**High-quality open-source
software for web developers.**

Headless, type-safe, & powerful utilities for Web Applications, Routing, State Management, Data Visualization, Datagrids/Tables, and more.

TanStack Query

(formerly known as React Query)

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

TanStack Query

(formerly known as React Query)

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

All-inclusive Data Fetching solution for React (and others)

TanStack Query

(formerly known as React Query)

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

All-inclusive Data Fetching solution for React (and others)

Suspense, Caching, TypeSafety, Error Handling

TanStack Query

(formerly known as React Query)

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

All-inclusive Data Fetching solution for React (and others)

Suspense, Caching, TypeSafety, Error Handling

TanStack Router

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

TanStack Router

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

Filebased, typesafe Routing

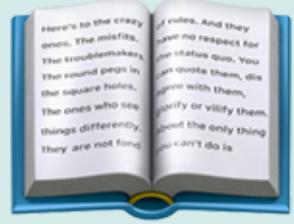
TanStack Router

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

Filebased, typesafe Routing

Search Params as global State

Code



Examples

HTTP://LOCALHOST:24080

Lazy Routes

- 🕵️ about.lazy.tsx ↗
- 🕵️ __root.tsx: Add Link
- 🕵️ Network traffic

Typesafe Routes

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

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?

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>

