

**NILS HARTMANN**

<https://nilshartmann.net>

<https://react.schule/enterjs-react-2024-slides>



# State of React 2024

ENTERJS | MAINZ, 7. MAI 2024 | @NILSHARTMANN

# NILS HARTMANN

nils@nilshartmann.net

**Freiberuflicher Entwickler, Architekt, Trainer aus Hamburg**

**Java, Spring, GraphQL, React, TypeScript**



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



<https://reactbuch.de>

[HTTPS://NILSHARTMANN.NET](https://nilshartmann.net)

## Beispiel-Anwendung

Source-Code (GitHub):

<https://react.schule/enterjs-2024-react>

Nach dem Vortrag checke ich dort den geschriebenen Code auf einem neuen Branch ein

# Neues aus dem Ökosystem

oder

*Ein "moderner" React Stack*

# Neues aus dem Ökosystem

## *TanStack Router*

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

Typsicheres Routing

# Neues aus dem Ökosystem

## *TanStack Router*

Demo:

Link in RecipeCard

Zugriff auf Params in RecipePage

# Neues aus dem Ökosystem

## *TanStack Router*

Persönliche Bewertung:

Für neue Projekte in Betracht ziehen

Migration von React Router schwer zu rechtfertigen

# Neues aus dem Ökosystem

*zod*

<https://zod.dev/>

Typsicher zur Entwicklungszeit  
Validierung zur Laufzeit



# Neues aus dem Ökosystem

**zod**

Demo:

RecipeListRouteParams

# Neues aus dem Ökosystem

*zod*

Persönliche Bewertung:

Sehr gutes Kosten-Nutzen-Verhältnis

Einsatz auf jedem Fall prüfen

Kann gut in bestehenden Code integriert werden

# Neues aus dem Ökosystem

## *TanStack Query + Suspense*

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

Modernes DataFetching  
Bye, bye, useEffect

# Neues aus dem Ökosystem

## *TanStack Query + Suspense*

Demo:

\$recipeId-Route: useSuspenseQuery

FeedbackListLoader in RecipePageContent

# Neues aus dem Ökosystem

## *TanStack Query + Suspense*

Geladene Daten werden gecacht!

TS Query kann als globales Statemanagement verwendet werden  
(für serverseitige asynchrone Daten)

# Neues aus dem Ökosystem

## *TanStack Query + Suspense*

Persönliche Bewertung:

"Die" Standard-Lösung für modernes Data Fetching  
(für REST-like APIs)

Kann gut in bestehende Anwendungen integriert werden

# React 19

REACT 19

## React 19 beta

erschienen 25. April

<https://react.dev/blog/2024/04/25/react-19>



## React 19: use Hook

- Vereinfacht arbeiten mit Context und Promises

```
function Button( { color } ) {  
  const btnColor = color || use(ThemeContext).color  
  return <button className={btnColor}>...</button>  
}
```

- Unterliegt nicht den "Rules of Hooks"

## React 19

- "use client" / "use server"
- Actions

## React 19

- "use client" / "use server"
- Actions
- Demo: Next.js LikesWidget
  - <http://localhost:8100>
  - nextjs-Workspace

**NILS HARTMANN**

<https://nilshartmann.net>

# Vielen Dank!

Slides: <https://react.schule/enterjs-react-2024-slides>

Source-Code: <https://react.schule/enterjs-2024-react>

E-Mail: [nils@nilshartmann.net](mailto:nils@nilshartmann.net)

Social Media etc.: <https://nilshartmann.net/kontakt>