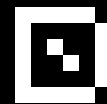


Next.js CONF25

# Type-Safe URL State in Next.js with `?n=u&q=s`



FRANÇOIS BEST  
FREELANCE / 47NG



oi'



Cal

?n=u&q=s

**Type-Safe  
Search Params  
State Manager  
For React**

# **Making URL State Enjoyable**

# URL State 101

/talks?q=Next.js+Conf&year=2025

Search params

Query string

# Why URL State?

# Reload resilience



Not sufficient

Side effect

**Other solutions**

localStorage, IndexedDB

# Superpowers





# Teleportation

## Sharing

Moving memory across devices

## State sync

See the same thing

## Deep Linking

Pick up where you left off



# Time Travel

## Bookmarks

Share forward in time

## Browser History

Recall from the past

## Back / Forward buttons

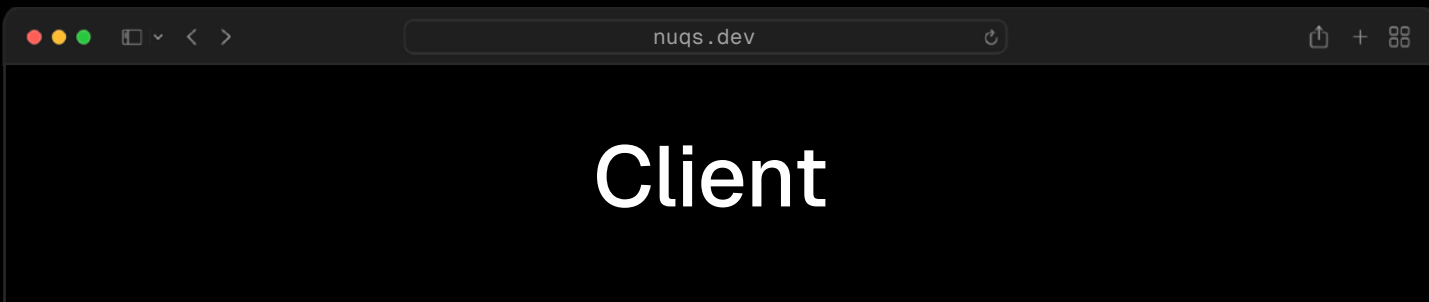
Undo / Redo

```
OCT 22 14:02:47.42 GET 200 nuqs.dev
OCT 22 14:02:47.40 GET 200 nuqs.dev
OCT 22 14:02:47.38 GET 200 nuqs.dev
OCT 22 14:02:47.36 GET 200 nuqs.dev
OCT 22 14:02:47.35 GET 200 nuqs.dev
OCT 22 14:02:47.32 GET 200 nuqs.dev
```

# Server

```
🌐 /
🔍 /api/search
🌐 /docs
🌐 /
🔍 /api/search
🌐 /docs
```

# URL



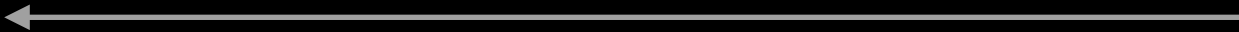
# Client

# Server

GET /search?q=Hello



302 /login?from=/search



# Drawbacks



<https://nuqs.dev/blog/beware-the-url-type-safety-iceberg>

# Size Limits

## Server

8 kB

## Transport

2 to 4 kB

## Humans

Much less

https://example.com/path?  
tableState=%7B%22pagination%22%3A%7B%22pageIndex%22%3A1%2C%22  
pageSize%22%3A50%7D%2C%22filters%22%3A%5B%7B%22id%22%3A%22  
genre%22%2C%22value%22%3A%22fantasy%22%7D%5D%2C%22orderBy%2  
2%3A%7B%22id%22%3A%22releaseYear%22%2C%22desc%22%3Afalse%7D  
%7D&search=SGVsbG8sIFJlYWNoIFBhcmlzIQ

https://example.com/path?  
filters=genre:fantasy&sort=releaseYear:asc&page=1&size=50  
&search=The+Lord+of+the+Rings



**The URL is the first UI  
your users will see**

**URLs are immutable**

**Your app is not**

**“Cool URIs don't change”**

— Sir Tim Berners-Lee

**URL = Database**

# Schema migrations

# **Proxy / Middleware**

## **Redirect**

# Rate Limits

Firefox, Chrome

50 ms

Safari

120 ms

Throttle & Debounce

Built-in nugs

# Time Safety

## Decoupling

UI vs URL

## Optimistic UI

Instant local updates

## Declarative & composable

Isolated timelines, abort on navigation



# Entrypoints

## Validation

URL = untrusted input

## Public vs Private

Search params vs hash

## Interaction bypass

State machine shortcuts

# SEO

Duplicate without user-selected  
canonical

Critical vs optional search params

youtube.com/watch?v=dQw4w9WgXcQ&t=42s

```
export const metadata = {  
  alternates: {  
    canonical: '/watch?v=...',  
  },  
} satisfies Metadata
```

```
function UrlState_TheHardWay() {
  const searchParams = useSearchParams()
  const router = useRouter()
  const pathname = usePathname()
  const search = searchParams?.get('search') ?? ''
  const onUpdate = (value: string) => {
    const usp = new URLSearchParams(searchParams ?? '')
    usp.set('search', value)
    router.replace(`${pathname}?${usp.toString()}`)
  }
  return (
    <input
      type="search"
      value={search}
      onChange={e => onUpdate(e.target.value)}
    />
  )
}
```

```
function UrlState_TheEasyWay() {  
  const [search, setSearch] = useQueryState('search', { defaultValue: '' })  
  return (  
    <input  
      type="search"  
      value={search}  
      onChange={e => setSearch(e.target.value)}  
    />  
  )  
}
```



**Demo**

	App Router	Pages Router
Server	<code>router.replace()</code>	<code>router.replace({   shallow: false })</code>
Client	<code>history.replaceState()</code>	<code>router.replace({   shallow: true })</code>

	App Router	Pages Router
Server	<code>router.replace()</code>	<code>router.replace({   shallow: false })</code>
Client	<code>history.replaceState()</code>	<code>router.replace({   shallow: true })</code>



# Recap

## **Declarative**

Type-safe & time-safe behaviours

## **Composable**

No hand-stitching necessary

## **Un-opinionated**

Adapts to your codebase & style

? **n = u & q = s**

Next.js CONF25

# Thank you

[nuqs.dev/nextjs-conf-25](https://nuqs.dev/nextjs-conf-25)



FRANÇOIS BEST  
FREELANCE / 47NG

