

Adult.js

JavaScript ist erwachsen
geworden!

Christian Mayer & Marc Jansen

FOSSGIS 2018, Bonn, 22.03.2018

ROCKINGBUENISE



Nobody likes you when you're 23

Gliederung

- Über...
- Motivation & Ziel
- kiddie.js
- adult.js
- Best Practices

Christian Mayer



✉ chris@meggsum.de

📍 [@chrismayer](https://www.instagram.com/chrismayer)

🐦 [@geofootballer](https://www.twitter.com/geofootballer)

- Geoinformatiker
- Softwareentwickler & -architekt
- Kernentwickler und PSC Mitglied GeoExt
- Sprecher & Trainer national & international
- OSGeo Foundation Charter Member

meggsum



 @meggsum
 @meggsum

- meggsum.de
- Dienstleistungen im Bereich GIS, Webmapping & GDI
- Maßgeschneiderte WebGIS-Lösungen
- Softwarekonzepte und Softwareentwicklung
- Geodaten
- Beratung und Schulung

Marc Jansen



✉ jansen@terrestris.de

👤 [@terrestris](https://github.com/terrestris)

🐦 [@selectoid](https://twitter.com/selectoid)



- Geschäftsführer @terrestris
- Kernentwickler OpenLayers
- Kernentwickler GeoExt
- Buchautor "OpenLayers"
- Sprecher & Trainer
national & international
- OSGeo Foundation Charter
Member

terrestris



 @terrestris
 @terrestrisde

- terrestris.de
- OpenSource GIS aus Bonn
- Entwicklung, Projekte & Support/Schulung
- Beratung, Planung, Implementierung & Wartung

Motivation & Ziel

Motivation & Ziel

- Wir beide machen seit vielen Jahren JavaScript-Projekte
- Zunehmende Komplexität
- Zunehmende Funktionalität
- Zeiten von Spielsprache lange vorbei
- Ziel: Professionalisierung aufzeigen

kiddie.js

JavaScript damals

Entstehung



Brandon Eich, [Quelle](#)

- 1995
- Netscape
- Mocha → LiveScript → JavaScript
- Browser-Skriptsprache
- Interaktivität in Webseiten
- Kurzzeitig serverseitige Unterstützung (Netscape-Webservern)

Script-Tag und ein paar Zeilen Code

```
<div id="map" class="smallmap"></div>
<script src="../OpenLayers.js"></script>
<script>
var map = new OpenLayers.Map("map");
var olWms = new OpenLayers.Layer.WMS(
    "OpenLayers WMS",
    "http://vmap0.tiles.osgeo.org/wms/vmap0",
    {layers: "basic"}
);
map.addLayers([olWms]);
</script>
```

Script-Tag und ein paar Zeilen Code

- Einfache kleine Programme
- STRG + U → Entwicklers bester Freund
- Editieren auf dem Server
- Meist in Kombination mit serverseitigen Skripten
- steigende Komplexität der Anwendung → Chaos in der Organisation der Eingangsdateien



Bastian Allgeier

@bastianallgeier



„View source“ embodies the very essence of the web I learned to love so much. The magic of writing something into a text file, opening it in a browser and being able to see and learn from the source of every single site: twitter.com/tomdale/status...

11:17 - 21. Feb. 2018

♡ 19  Weitere Tweets von Bastian Allgeier ansehen



Tom Dale

@tomdale



Can we agree that, in 2018, human-readable "View Source" is a constraint the web can discard? I benefitted from "View Source" too, but today we have an embarrassment of resources and open source examples I would have killed for as a kid.

21:18 - 19. Feb. 2018

♡ 333  110 Nutzer sprechen darüber



adult.js

JavaScript heute

Everything.js & Nothing.js

npm i vec2-copy

```
module.exports = function vec2Copy(out, a) {  
  out[0] = a[0]  
  out[1] = a[1]  
  return out  
}
```

Heute

- Frameworks & Libraries en masse
- Tools und Paradigmen dito
- Boilerplates / Generatoren etc.
- Typescript & Flow
- Jedes Jahr ein neues EcmaScript
- JS überall: Browser & Node.js
- `<noscript>`?

“

*Ultimately, the problem is that by choosing React (and inherently JSX), you've unwittingly opted into a confusing nest of build tools, boilerplate, linters, & time-sinks to deal with before you ever get to **create** anything.*

Hervorhebung von uns, <https://medium.com/@ericclemmons/javascript-fatigue-48d4011b6fc4>

Professionalität
bekommt man nicht
umsonst

Best Practices

Node.js – JavaScript auf dem Server

- Serverseitige JavaScript Plattform
- Verarbeitung von großen Datenmengen durch nicht blockierendes IO-System
- Event-basiert
- Ideal für Webanwendungen
- Open Source (MIT)
- Vortrag "Serverseitiges JavaScript und GIS Fossgis 2015"

Node.js - für die Cliententwicklung

- Meist im Zusammenspiel mit npm
- Oft im Bereich der Frontend-Entwicklung
 - Webserver
 - Dependency-Management
 - Build-Tools
 - Unit Testing

JavaScript vs. EcmaScript vs. TypeScript

Findet Euren Favoriten

Build Tools



Alexis King

@lexi_lambda



“I don’t need macros, they’re too complicated and not useful,” says the programmer as they use Flow with JSX with Babel with two dozen plugins and maintain two hundred line webpack configs for code with machine-checked comments that parses CSS in template strings at runtime and—

07:06 - 3. Jan. 2018

♡ 1.201 💬 438 Nutzer sprechen darüber



Fazit

- Sehr hoher Professionalisierungsgrad
- Unglaubliches Momentum in der Weiterentwicklung
- Durchaus komplexe Konstrukte
- Manchmal zu viel Overhead
- ...jedoch auch reichlich Dokumentation
- JavaScript unbekannt? Lern es, es lohnt sich
- JavaScript Pro? Hinterfrag Dich und lerne weiter

Vielen Dank

Fragen &
Anmerkungen?

Impressum

Impressum

Autor

Christian Mayer

meggsum Geoinformatik

Hauptstraße 165a

67125 Dannstadt-

Schauernheim

chris@meggsum.de

Marc Jansen

terrestris GmbH & Co. KG

Kölnstr. 99

53111 Bonn

jansen@terrestris.de

Lizenz

Diese Folien sind unter **CC BY-SA** veröffentlicht.

Vortragsfolien, **TODO PDF-Version**, **git repository**