**Svelte and Sapper in Action**

**R. Mark Voklmann**

**1. Tell us about yourself.**

Here is the bio I typically use when proposing conference talks:

"Mark Volkmann is a partner at Object Computing, Inc. (OCI) in St. Louis

where he has provided software consulting and training since 1996.

As a consultant, Mark has assisted many companies with JavaScript,

Node.js, React, Angular, and Java application development.

Mark has created and taught many courses on topics including React, Vue,

Angular, Node.js, jQuery, JavaScript, HTML5, CSS3, Ruby, Java, and XML.

Mark is a frequent presenter at St. Louis area user groups. He has

written for XML Journal and the No Fluff Just Stuff magazine.

He has presented at many conferences including Nordic.js, MidwestJS,

Jfokus, Strange Loop, NDC Oslo, No Fluff Just Stuff, and XML DevCon."

---

I have given four talks on Svelte. The most recent was at the

Nordic.js conference on 10/10/2019. Another was at the MidwestJS

conference on 8/9/2019.

Here is a link to video of the Nordic.js talk:

https://www.twitch.tv/videos/492603232?t=02h42m43s&fbclid=IwAR1rdkIxXgBMYWksr1qnvnUSDJfTYLrsnigGiKKLbKd\_0UGwoJAQ139PjRQ

I have a fairly long history of writing articles about software topics

and giving talks on the same topics. Here is a link to a page with many

of my articles and videos:

<https://objectcomputing.com/resources/publications/mark-volkmann>

Here is an article I wrote on Svelte: <https://objectcomputing.com/resources/publications/sett/july-2019-web-dev-simplified-with-svelte>

**2. Tell us about the book’s topic.**

Svelte is a tool for creating web applications that takes a fundamentally

different approach than other popular web frameworks. It compiles web

app code to a JavaScript bundle that is significantly smaller than those

produced by frameworks such as React, Vue, and Angular. This reduces

first page load time for users.

Svelte significantly simplifies development of web UI components through

features such as reactivity, reactive statements, two-way data binding,

stores, and context.

Sapper provides a framework around Svelte that supports server-side

rendering, code splitting, easy routing, easy API development,

static site generation, and more.

Together Svelte and Sapper are poised to become popular choices for

developing web applications.

Svelte was created by Rich Harris, now at the New York Times.

Development started a few years ago, but it has only recently

begun garnering serious attention.

**3. Tell us about the book you plan to write.**

After reading this book, the reader will be able to:

• Use Svelte and Sapper to implement web applications.

• Write far less code to accomplish tasks than with other web frameworks.

• Benefit from fast load times enabled by Svelte’s small bundle sizes and Sapper’s code splitting

• My book is designed to both teach and serve as a reference.

• I believe my book fall into the Manning “In Action”.

**4. Q&A**

What are the three or four most commonly-asked questions about this technology?

• How does Svelte differ from frameworks like React, Vue, and Angular?

• How does state management work in Svelte?

• What capabilities does Sapper add to Svelte?

• Can I use Svelte to implement mobile applications**?**

**5. Tell us about your readers.**

Your book will teach your readers how to accomplish the objectives you’ve established for the book. It’s critical to be clear about the minimum qualifications you’re assuming of your reader and what you’ll need to teach them.

• What skills do you expect the minimally-qualified reader to already have?

o Some experience with writing HTML, CSS, and JavaScript.

o Some exposure to concepts in web development.

• What are the typical job roles for the primary reader? Be specific: e.g.

o 1 year of experience as a front-end software engineer.

• What will motivate the reader to learn this topic?

o Desire to make web applications easier to write and maintain.

o Desire to make web applications faster, especially in download speed.

**6. Tell us about the competition.**

• There are currently no books on Svelte or Sapper. This is an opportunity for Manning to be first to market!

• What resources would you currently recommend to someone wanting to learn this subject?

o “Rethinking Reactivity” talk by Rich Harris, creator of Svelte - <https://www.youtube.com/watch?v=AdNJ3fydeao>

o Svelte web site – https://svelte.dev

o My Nordic.js conference talk on Svelte - https://www.twitch.tv/videos/492603232?t=02h42m43s

**7. Book size and illustrations**

Please estimate:

• The approximate number of published pages to within a 50-page range

200 – This may seem like a small number of pages. However, Svelte is much simpler to use than other frameworks, so it should not require as many pages to describe as other web frameworks.

• The approximate number of diagrams and other graphics

50

• The approximate number of code listings

100

**8. Contact information**

Name: Mark Volkmann

Mailing Address: 644 Glen Summit, St. Charles, MO 63304

Preferred email: r.mark.volkmann@gmail.com

Preferred phone: 314-398-6537

Website/blog: https://mvolkmann.github.io

GitHub: mvolkmann

Twitter: @mark\_volkmann

**9. Schedule**

• Most authors require 2-4 weeks to write each chapter. Please estimate your writing schedule

Chapter 1: November 30, 2019

1/3 manuscript: January 31, 2020

2/3 manuscript: March 31, 2020

3/3 manuscript: June 30, 2020

• Are there any critical deadlines for the completion of this book

o At some point support for TypeScript will be added, but the date is unknown. I will want to add information on that when it is available.

o There is no known competition yet.

**10. Table of Contents**

I will gladly format the table of contents any way you see fit. But before putting in that effort I would like to learn whether there is interest in my book proposal. For now I have this outline:

\* Section 1 - Introduction

\* Chapter 1 - Svelte Overview

\* What is Svelte?

\* web application compiler

\* no virtual DOM like React and Vue

\* developed by Rich Harris

\* small bundle sizes

\* CSS scoped by default

\* clear place for global CSS

\* easy component state management - reactivity

\* reactive statements (destiny operator)

\* easy app state management (stores)

\* easy way to pass data to descendant components (context)

\* two-way data binding

\* easy animations built-in

\* runtime warnings for accessibility issues

\* Why consider Svelte?

\* write less code

\* code is easier to understand

\* all the features in previous section

\* History of Svelte

\* Does Svelte disappear?

\* discuss amount of Svelte library code included in bundles

\* Chapter 2 - Sapper Overview

\* What is Sapper?

\* pages are components

\* easy routing based on directory and file names

\* code splitting

\* offline support through Service Workers

\* server-side rendering (SSR) (similar to Next.js for React)

\* static site generation (similar to Gatsby for React)

\* server routes for implementing server APIs (REST services)

\* prefetching so components render faster

\* Why consider Sapper?

\* History of Sapper

\* Section 2 - Svelte

\* Chapter 3 - Getting Started With Svelte

\* npx

\* REPL download

\* Chapter 4 - Defining Components

\* in .svelte files with <script>, <style>, and HTML sections

\* component Names

\* component Props

\* importing Components

\* inserting HTML from strings

\* Chapter 5 - Styling

\* component-based styles

\* global styles

\* using CSS preprocessors (like Sass)

\* Can Svelte identify unused CSS without Prettier?

\* Chapter 6 - Reactivity

\* component state management

\* Chapter 7 - Reactive Statements

\* a.k.a. destiny operator

\* reactive declarations

\* Chapter 8 - Logic in Markup

\* If Markup

\* Each Markup

\* Await Markup

\* Chapter 9 - Slots

\* default and named

\* Chapter 10 - Binding Form Elements

\* to Variables

\* to custom props

\* Chapter 11 - Event Handling

\* DOM events

\* dispatching custom events

\* Chapter 12 - Lifecycle Functions

\* onMount

\* beforeUpdate

\* afterUpdate

\* onDestroy

\* Chapter 13 - Actions

\* call a function when an element is added to DOM

\* Chapter 14 - Sharing Data

\* Props

\* Context

\* Stores

\* readable, writable, derived, and custom

\* Chapter 15 - Module Context

\* Chapter 16 - Batched DOM Updates

\* Chapter 17 - Animation

\* Chapter 18 - Special Elements

\* Chapter 19 - Debugging

\* Chapter 20 - Compiling to Custom Elements

\* Section 3 - Sapper

\* Chapter 21 - Getting Started With Sapper

\* npx

\* Chapter 22 - Provided Files

\* package.json

\* client.js

\* server.js

\* service-worker.js

\* template.html

\* static directory

\* bundler configuration

\* Chapter 23 - Routes

\* Chapter 24 - Server Routes

\* for implementing server APIs (REST services)

\* Chapter 25 - Error Page

\* Chapter 26 - Client API

\* start app with `start()`

\* programmatically navigate with `goto()`

\* Chapter 27 - Pre-fetching

\* fetching data ahead of component rendering

\* methods `this.fetch`, `this.error`, and `this.redirect`

\* Chapter 28 - Layouts

\* Chapter 29 - Server-side Rendering

\* Chapter 30 - Sapper Stores

\* Chapter 31 - Building dynamic apps for deployment

\* Chapter 32 - Exporting static apps for deployment

\* Section 4 - Issues

\* Chapter 33 - Svelte Issues

\* no support for TypeScript yet, but it is coming

\* Chapter 34 - Sapper Issues

\* Section 4 - Tooling for Svelte and Sapper

\* Chapter 35 - ESLint

\* Chapter 36 - Prettier

\* Chapter 37 - VS Code and the Svelte Extension

\* Chapter 38 - Unit Tests

\* Jest and svelte-testing-library

\* Chapter 39 - End-to-End Tests

\* Cypress

\* Chapter 40 - Storybook

\* Appendix - Resources

\* Svelte Resources

\* Sapper Resources