University of Bergen
Department of Informatics

Multiway Dataflow Constraint System and UI Programming

Author: Mathias Skallerud Jacobsen
Supervisors: Mikhail Barash and Jaakko Järvi

UNIVERSITETET I BERGEN
Det matematisk-naturvitenskapelige fakultet

May, 2022

Contents

List of Figures

List of Tables

Listings

Chapter 1

Introduction

This report is part of my INF319 project at the University of Bergen (UiB). The goal of the project was to understand event-based Graphical User Interface (GUI) programming and the limitation to such programming. Specify dataflow constraints and understand how they connect to GUI widgets. Understand the possibilities and limitations of constraint systems based on GUIs. The multiway dataflow constraint system used in the development of this project is HotDrink. Before developing this project I had no prior experience with HotDrink.
TODO: Skriv hva rapporten inneholder.

1.1 Svelte

Svelte is a JavaScript framework used for building user interfaces, just like React, Angular, Vue. Where some of these frameworks bulk there work in the browser, Svelte shifts that work onto a compile step [2]. Svelte is written using TypeScript. Instead of using virtual Document Object Model (DOM) svelte uses build time to covert the code into JavaScript [1].

1.2 Constraint systems

A constraint system can be seen as a tuple $\langle V, C \rangle$, where V is a set of *variables* and C a set of *constraints*. Each variable in V has a associated value of a given type (string, integer, boolean, object, etc.). Each constraint in C is a tuple $\langle R, r, M \rangle$. The variables involved in the constraint is given by $R \subseteq V$, r is some n-ary relation among variables in R, where $n = |R|$. M is a set of non-empty set of *constraint system methods*. Executing any method m in M enforces the constraint by computing values for some subset og R, using another disjoint subset of R as inputs, such that the relation R is satisfied [4].

1.3 HotDrink

In this project I used the multiway dataflow constraint system library HotDrink [3], witch is a JavaScript-based library for multiway dataflow constraint systems in GUIs. Instead of writing explicit event handlers, the programmer writes declarative specification of data dependencies, from which the library derives the GUI behavior. This library features a DSL for defining constraint systems. The DSL allows one to specify *components*, *constraints*, *methods* and *variables*. A component in HotDrink holds a set of constraints and variables, as described in section 1.2. Variables often depend on each other, in that case it gets into a setting of multi-way dataflow. The HotDrink DSL is implemented JavaScript tagged template literals [1], which can be seen in Listing **??**. This lets the programmer integrate the library with frontend JavaScript frameworks such as React[2] and Svelte[3].

[1] Template literals
[2] For more information about the framework can be found at reactjs.org
[3] For more information about the framework can be found at svelte.dev

Listing 1.1: Example of the HotDrink DSL

```
import { component } from 'hot-drink';

const comp = component`
    var f=1337, c;

    constraint c1 {
        m1(c -> f) => c * (9/5) + 32;
        m2(f -> c) => (f -32) * 5/9;
    }
`;
```

Currently there are no integrated methods in HotDrink to bind the constraint system to frontend frameworks. So the programmer have to decide on the best way to integrate the HotDrink to the frontend web application. Listings 1.1, 1.3, 1.4 shows one way of binding HotDrink with Svelte.

Listing 1.2: Function for binding HotDrink and Svelte variable

```
function setHDValue<T>(HDvariable: Variable<T>, n: T) {
    if (n !== HDvariable.value) {
        HDvariable.set(n);
    };
};
```

Listing 1.3: Using the reactive statement, in Svelte [1], to update HotDrink corresopnding value, and trigger HotDrink to enforce the constraint system

```
$: {
    setHDValue(HotDrinkValue, frontendValue);
}
```

Listing 1.4: Using the onMount callback, in Svelte [1], to update the frontend value that correspont to the same value in HotDrink, when the HotDrink value changes.

```
onMount(() => {
    HotDrinkValue.subscribeValue((value: number) => frontendValue = value);
});
```

List of Acronyms and Abbreviations

DOM Document Object Model.

DSL Domain Specific Language.

GUI Graphical User Interface.

UiB University of Bergen.

Bibliography

[1] Svelte api documentation. `http://svelte.dev/docs`, May 2022.

[2] Svelte webpage. `http://svelte.dev/`, May 2022.

[3] Gabriel Foust, Jaakko Järvi, and Sean Parent. Generating reactive programs for graphical user interfaces from multi-way dataflow constraint systems. In *Proceedings of the 2015 ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences*, GPCE 2015, page 121–130, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336871. doi: 10.1145/2814204.2814207.
URL: `https://doi.org/10.1145/2814204.2814207`.

[4] Jaakko Järvi, Mat Marcus, Sean Parent, John Freeman, and Jacob Smith. Algorithms for user interfaces. In *Proceedings of the Eighth International Conference on Generative Programming and Component Engineering*, GPCE '09, page 147–156, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605584942. doi: 10.1145/1621607.1621630.
URL: `https://doi.org/10.1145/1621607.1621630`.