

WHY ANGULARJS IS THE FUTURE OF WEB DESIGN

MICHAEL MEDING

CONTENTS

1	Background & Motivation	2
2	What is AngularJS?	2
3	Angular Modules	2
4	Controllers & Data Binding	3
5	Factories & Services	3
6	Directives	3
7	Discussion	4

ABSTRACT

The internet as we commonly know it has gone through many changes since its popularization in the late 80's to early 90's. In the time since its inception few things have radically changed the way we think about the web like JavaScript. It has opened up the web to a much broader audience and further "democratized" the web. Since the invention of JavaScript many libraries have been written to further aid the simplicity of its use in the browser. AngularJS is one such library however, it may represent a major turning point in web design as we know it with the introduction of web based frameworks. This paper serves to explore and explain what a web framework is and why it will revolutionize the web as we know it.

* Department of Computer Science, University of Massachusetts Lowell, United States

1 BACKGROUND & MOTIVATION

JavaScript was invented in 1995 by Brendan Eich and it allowed the web to suddenly go from lifeless static pages to in depth dynamic content. JavaScript was originally written for the Netscape browser but was quickly adopted by all major browsers of the time. JavaScript also setup the groundwork for creating a framework on the web as AngularJS is written in pure JavaScript.

2 WHAT IS ANGULARJS?

AngularJS is a framework which allows for easy development of dynamic web pages. It contains many ways to logically separate your code similar to popular object oriented programming languages like Java or C++. AngularJS also adds many missing features that make JavaScript development difficult such as dependency injection from one component to another.

One interesting feature of AngularJS is that it implements an old method in computer science called the Model-View-Controller architecture. At the heart of the Model-View-Controller architecture is an idea called separated presentation. [1] Separated presentation is a strict division between objects that exist in the model and those items which are shown in the view. AngularJS supports this by creating their version of scope for JavaScript variables in the controller. This allows you strict separation between what is seen and how it got that way.

3 ANGULAR MODULES

One major problem with JavaScript is the lack of any kind of code packaging. In Java or Python code is easily written into libraries so that others may take advantage of someone else's code. Until AngularJS this was not possible with JavaScript as there was no way to properly control the scope of variables. JavaScript also has no control over the local file system so it only knows what has been loaded to the browser. This means that there is no way to reference another file unless it is loaded into memory with is highly inefficient.

Angular modules solve this by allowing you to extend the functionality of your application at will. Similar to python when a module is needed you simply have to reference it and give it an alias if needed. From there its just a matter of using a similar call pattern to python to reference what you need from that library. This feature is quite powerful in AngularJS as you are able to easily create large and complex applications by simply including what you need. Additionally, Angular modules allow you to easily package your own applications so that others may use it in the same way.

4 CONTROLLERS & DATA BINDING

From the Model-View-Controller architecture the controller is the main logic of the application and one of the core components to Angular. Controllers are like the glue that binds together models and views and is what allows you to have dynamic data.[3] Only within a controller are you allowed to have access to "scope" variables which are like normal JavaScript variables but have a special two way data binding property.

Two way data binding is one of the coolest features of AngularJS as it allows you to have real time updates for all the variables of your application. In traditional JavaScript programming you must track a series of events such as clicking the mouse or pressing a button to trigger changes in your application. In AngularJS the variable changes happen in real time through the use of built in watch statements and a digest cycle. The watch statements detect any changes in your variables from the view and then propagate those changes using a digest cycle to all of the functions that use that variable.[4] This means that every time a user changes a variable those changes propagate all the way up to show dynamic output as they type.

5 FACTORIES & SERVICES

A major pitfall of traditional JavaScript development is the lack of any kind of real language constructs such as OOP (Object Oriented Programming). The reason OOP is not a default construct in JavaScript is that JavaScript is a functional programming language. Functional programming languages can intrinsically support OOP design but as aforementioned JavaScript was written quickly and this aspect was considered of lesser importance at the time. As web based applications have become more and more complicated the need for more logical design patterns has become more pressing. AngularJS is the answer to this problem as it extends JavaScript in such a way as to allow for OOP design patterns through the use of Factories and Services.

Angular factories and services follow the same design patterns as singletons and standard objects in Java. An Angular factory is like a standard object in Java which when called will build an instance of that object with object fields independent of other objects of the same type. An Angular service is like a singleton as no matter how many times it is referenced the fields always remain the same. The power to be able to package your application in a design pattern such as OOP is what gives Angular its modular power.

6 DIRECTIVES

With the introduction of HTML5 web developers were inundated with a swath of new tags to use while writing HTML. Angular takes this yet another step further by allowing you write your own custom HTML tags. This is the pinnacle of what Angular is capable of as it brings together all of the prior items into one powerful tag.

A directive contains 2 major components, a template, and a linker. The template is an HTML snippet which is loaded in place of the main tag when the page is loaded. For example, if your directive has an embedded canvas element in the template your tag which you place in your main view will be replaced with canvas when the page is loaded to the browser. The linker is exactly the same as a controller except that instead of having access to the DOM of your main page it only has access to your tag and the template. [2]

7 DISCUSSION

Even from the brief overview of what AngularJS is it can easily be seen how powerful of a tool it can be. The speed at which things move on the web is growing day after day and the tools that web developers use need to evolve at a similar pace. AngularJS represents a significant step in the right direction and with the introduction of Angular2 will continue to see this evolution take place.

REFERENCES

- [1] Martin Fowler. *Gui architectures*, 2006.
- [2] Kennedy Kambona, Elisa Gonzalez Boix, and Wolfgang De Meuter. An evaluation of reactive programming and promises for structuring collaborative web applications. In *Proceedings of the 7th Workshop on Dynamic Languages and Applications*, DYLA '13, pages 3:1–3:9, New York, NY, USA, 2013. ACM.
- [3] Dan Wahlin. *AngularJS Fundamentals In 60-ish Minutes*. Youtube, <https://www.youtube.com/watch?v=i9MHigUZKEM>, 2013.
- [4] Sandeep Panda. *AngularJS : novice to ninja*. Collingwood, Victoria, Internet Resource, Computer File, 2014.