Beyond Adaptive Web Design

Ján ANTALA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 2, 842 16 Bratislava, Slovakia
`hello@janantala.com`

Extended Abstract

The Web is continually evolving and we need to evolve with it. It used to be easier to manage browsers when there were just a few of them on the desktop. Today we not only have to deal with a wide range of desktop browsers but mobile devices, tablets, televisions, weareable devices and more. Even for the average web site things have changed a lot over four years: browser share, operating systems, screen resolutions, and more [3]. The basic approach how to provide an optimal viewing experience is to use "Responsive web design". While creating flexible layouts is important, there is a lot more that requires remarkable adaptive web experiences.

Adaptive web design is fundamentally "Progressive enhancement", but it is being applied to a much larger and more diverse landscape. We now have Web-enabled smartphones, tablets, e-readers, netbooks, watches, TVs, phablets, notebooks, game consoles, cars and more. We also have many types of internet networks with different speed, latency and quality. However there are many more factors we need to think about. It is also important to consider as well ergonomics, input methods, internet connections and other features that can be detected. We consider "adaptive web design" as an equal with creating a single Web experience. We can adjust it based on the capabilities of the device and browser. Website can access sensors in devices and use them to enhance user experience.

We have detected several web components [1] and input methods and tried to verify which of them provide better experience for the website users and are also useful for the web developers. There have been made series of experiments on both adaptive input methods and adaptive web components. To prove the usefulness of the proposed concept we have designed and implemented several reusable modules and components and we have used them in propotype web projects. We have tracked amount of saved web traffic and web requests, webpage rendering time and interest in alternative input methods. We have also received a lot of feedback from the community which helped us to verify the proposed concept. The experiments have been attended by thousands of visitors.

Adaptive input methods provide alternative way of web application control and extend current approach. We have been experimenting to control web applications using voice commands, motion detected by a gyroscope and a video camera.

We have created an experiment based on our module which provides voice control over to-do list. There were 7 voice commands using exact expressions and 9 using regular expressions. This has been the most popular input method and the experiment has been attended by thousands of users. One of the biggest problems of the voice commands is however incorrect speech recognition. To solve this issue we can conditionalize regular expressions or use utterance error correction.

The another experiment based on gyroscope rotation which allows to scroll in the application has been attended by thousands of users. However some problems have been detected. The biggest one is that browser vendors do not use specification correctly and use own orientation ranges and directions. This causes some scroll issues.

Almost every modern device contains a video camera. This is a great opportunity to control web sites using motion gestures. This experiment has been attended by hundreds of users. Using video motion detection we can enhance experience in websites and games. We can also detect device motion and orientation changes so it can be used as a supplementary input method to the accelerometer and gyroscope rotation. However we cannot take a full control over the web app and need a proper ambient lighting. There can be detection issues in too dark and too bright scenes.

There are many commonly used web services that produce many requests and unnecessary traffic even when the website user does not want to use them. There is lot of studies that highlight the importance of speed. More than half of people with a bad loading experience on mobile, will not come back [2]. 73% of mobile internet users say they have encountered Web pages that are too slow [4]. There are also native mobile application for that services with better perfomance which provide full user experience. This is a great opportunity to utilize conditional loading to serve the best experience for the right context. We have used the Mobile First principle to develop Google Maps and Youtube videos web components. Using adaptive web components we reduce initial traffic by 400 kB and a page load times by 350 ms for a youtube video in the average. The amount of saved traffic and load times are various for the map elemets and depend on element size. This approach works quite well for simple use cases. We can show multiple map types, markers or videos. However, more interactive elements require additional consideration. Even then use of an embedded elements directly still might not make sense because of unnecessary traffic. We can still use adaptive techniques to reduce it and replace basic static image with richer elements.

We have received a lot feedback and have usage and popularity results of adaptive input methods. As we have expected, the most popular input method is speech input since we can control an entire web application using only voice commands. Gyroscope and video motion detection are also useful input methods, but there are limited possibilities where to use them and can be used as supplementary input method only. All of them however provide a great opportunity to enhance user experience. Adaptive web components are a great to save an unwanted web traffic. They also increase user experience because the web page produces less requests and loads faster. However the limitations also exist because many websites need custom elements and design. As a result the adaptive web components produce less interest than adaptive input methods.

More changes are coming. So we have to prepare the Web to this evolution and provide the best user experience, design for many inputs and save the inessential web traffic.

References

[1] Bidelman, E.: Web Components: A Tectonic Shift for Web Development. In: *Google IO*, San Francisco, CA, USA, 2013.

[2] Grigorik, I.: Optimizing the Critical Rendering Path. In: *Breaking Development*, San Diego, CA, USA, July 22-24, 2013.

[3] Gustafson, A.: Building Adaptive Designs Now. In: *User Interface 17 Conference*, Boston, MA, USA, November 5-7, 2012.

[4] McLachlan, P.: Page Speed is Only the Beginning. In: *Breaking Development*, San Diego, CA, USA, July 22-24, 2013.