Beyond Interop
Baseline and more

Kadir Topal
Why Baseline?
### Keeping up with changes (to APIs, best practices, features) is a common challenge for WebDevs

#### Challenges to the Web Platform

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Heavy coders</th>
<th>Non-heavy coders</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keeping up with changes to existing APIs in web standards</td>
<td>21%</td>
<td>20%</td>
</tr>
<tr>
<td>Testing end-to-end user flows on multiple browsers</td>
<td>21%</td>
<td>19%</td>
</tr>
<tr>
<td>Keeping up with changes to best practices for building for web</td>
<td>20%</td>
<td>19%</td>
</tr>
<tr>
<td>Keeping up with new features in web standards</td>
<td>20%</td>
<td>21%</td>
</tr>
<tr>
<td>Understanding and implementing security measures</td>
<td>20%</td>
<td>19%</td>
</tr>
<tr>
<td>Making a design/experience work the same across browsers</td>
<td>18%</td>
<td>17%</td>
</tr>
<tr>
<td>Developer tools</td>
<td>17%</td>
<td>18%</td>
</tr>
<tr>
<td>Keeping up with a large number of tools/frameworks</td>
<td>17%</td>
<td>18%</td>
</tr>
<tr>
<td>Achieving the performance I want</td>
<td>16%</td>
<td>17%</td>
</tr>
<tr>
<td>Conducting performance, unit, or component tests</td>
<td>16%</td>
<td>16%</td>
</tr>
<tr>
<td>Achieving my desired UI</td>
<td>15%</td>
<td>16%</td>
</tr>
<tr>
<td>Creating experiences I want because capability is not there</td>
<td>15%</td>
<td>18%</td>
</tr>
<tr>
<td>Automation of frequently performed small tasks</td>
<td>15%</td>
<td>15%</td>
</tr>
<tr>
<td>Prioritization of web efforts within my organization</td>
<td>15%</td>
<td>13%</td>
</tr>
<tr>
<td>Supporting old browsers</td>
<td>15%</td>
<td>15%</td>
</tr>
<tr>
<td>Understanding how to use existing tools/frameworks</td>
<td>14%</td>
<td>13%</td>
</tr>
<tr>
<td>Deprecations and removals of language features or Web APIs</td>
<td>1%</td>
<td>0%</td>
</tr>
<tr>
<td>None of the above</td>
<td>2%</td>
<td>2%</td>
</tr>
</tbody>
</table>
Web Platform Feature Phases

State of JS/CSS Web DNA

Research

Blink, WebKit, Gecko, browsers, W3C, WHATWG, IETF, etc

Standards & Implementation

MDN, DevRel, Dev Marketing, eng teams.

Awareness & Adoption
Web Platform Feature Phases

- **State of JS/CSS Web DNA**
- **Research**
- **Blink, WebKit, Gecko, browsers, W3C, WHATWG, IETF, etc**
- **Standards & Implementation**
- **MDN, DevRel, Dev Marketing, eng teams.**
- **Awareness & Adoption**
- **Interop**
Web Platform Feature Phases

State of JS/CSS Web DNA

Research

Standards & Implementation

Blink, WebKit, Gecko, browsers, W3C, WHATWG, IETF, etc

MDN, DevRel, Dev Marketing, eng teams.

Awareness & Adoption

Interop
WEBDX COMMUNITY GROUP

The mission of the WebDX Community Group is to facilitate coordinated approaches to improve the overall experience of developing for the Web platform when such coordination provides unique opportunities for these improvements. See its proposed charter.

web-platform-dx

Group's public email, repo and wiki activity over time

<table>
<thead>
<tr>
<th>Year</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2022</td>
<td></td>
</tr>
<tr>
<td>2023</td>
<td>![Activity Chart]</td>
</tr>
</tbody>
</table>

Note: Community Groups are proposed and run by the community. Although W3C hosts these conversations, the groups do not necessarily represent the views of the W3C Membership or staff.

Tools for this group

- Mailing List
- Matrix
- Github repos
- RSS
- Contact This Group

Get involved

Anyone may join this Community Group. All participants in this community are subject to the W3C Community Code of Conduct.
“We want to make it easier for developers to track the list of features that are widely available and those that are under development”

- Dominique Hazaël-Massieux
What is Baseline?
Baseline

A line in the sand indicating which web platform features are widely available.

Features in Baseline have cross-browser support, they are interoperable, with no major issues.
CSS Grid Layout

CSS Grid Layout excels at dividing a page into major regions or defining the relationship in terms of size, position, and layer, between parts of a control built from HTML primitives.

Like tables, grid layout enables an author to align elements into columns and rows. However, many more layouts are either possible or easier with CSS grid than they were with tables. For example, a grid container’s child elements could position themselves so they actually overlap and layer, similar to CSS positioned elements.

Basic example

The example below shows a three-column track grid with new rows created at a minimum of 100 pixels and a maximum of auto. Items have been placed onto the grid using line-based placement.
A Point
And an interval
Open Discussion

- Feature Set -> New opportunities
- More profiles
- Collecting developer feedback
  - On Baseline
  - On Features
- Baseline in the ecosystem
  - Tooling?
  - 