All your spec are belong to us!

Irrigating dev resources from specs

W3C TPAC - October 2020

François Daoust (W3C) - @tidoust
Dominique Hazaël-Massieux (W3C) - @dontcallmedom
Outline

- Vision
- Ecosystem
- Examples
- Discussion
Vision

● The Open Web Platform is defined by at least
  ○ 26 W3C Working Groups and 9 Community Groups
  ○ 15 WHATWG workstreams
  ○ TC39
  ○ Khronos WebGL Working Group

● Lots of interdependencies that cross groups and organizations barriers
  ○ *Any organization that designs a system will produce a design whose structure is a copy of the organization's communication structure*

● Asynchronous decentralized communication is hard
● Let's help the tools help us!
Learning from software dependencies management
Automatic spec references

§ C. References

§ C.1 Normative references

[dom] DOM Standard. Anne van Kesteren. WHATWG. Living Standard. URL: https://dom.spec.whatwg.org/
[HTML] HTML Standard. Anne van Kesteren; Domenic Denicola; Ian Hickson; Philip Jägerstedt; Simon Priester. WHATWG. Living Standard. URL: https://html.spec.whatwg.org/multipage/
[infra] Infra Standard. Anne van Kesteren; Domenic Denicola. WHATWG. Living Standard. URL: https://infra.spec.whatwg.org/
Automatic definitions references

- Currently based on data from Bikeshed collected by Shepherd
- Integrated in authoring tools
  ```javascript
  {{RTCPeerConnection}}, [= browsing context =]
  ```
- Allows groups to distinguish between internal/exported dfns
  - Internal: definition may disappear at any time
  - Exported: group commits to maintain the definition over time
  - Forces groups to coordinate!
  - Maintain consistent view of the Web
- Serves as anchors to where a feature is defined
  - Useful e.g. for MDN Browser Compat annotations
- Current limitation: DB does not contain data from ED
  - webref to the rescue, new crawls every 6 hours
  - not yet integrated in Bikeshed/Respec
Integration with WPT

- Authoring tools create IDL blocks and indexes
- IDL extracts in webref updated every 6 hours
- **Semi-automated** integration in Web Platform Tests
  - Pull requests automatically generated, e.g. [Add interfaces/screen-fold.idl (PR #25742)](https://github.com/w3c/webref/blob/7583df5/ed/idl/interfaces/screen-fold.idl)
  - Need human validation to update [web-platform-tests/wpt/interfaces](https://github.com/w3c/webref/blob/729569825/builds/729569825)
Integration with MDN Browser Compat Data / CanIUse

- MDN Browser Compat Data helps developers know what features work
  - Also used now as source of caniuse.com
- WebIDL and CSS formal definitions help identify what features need to be documented
  - MDN BCD collector uses webref
Propagate WebIDL updates

- **WebIDL updater** tool
- Based on browser-specs
- Generates pull requests automatically, e.g. w3c/remote-playback PR #138
TypeScript typing for Web APIs

- WebIDL contains useful type information
- TypeScript ❤️ type information
- → Use WebIDL to generate TypeScript descriptions
- Based on webref
Exploring the platform

- [Webidlpedia](#) to explore naming practices across the platform
- Formal grammars helps with automated generation and parsing
  - csstree
  - E.g. useful for validators, fuzzers

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>EventTarget</td>
<td>96</td>
</tr>
<tr>
<td>HTMLIElement</td>
<td>74</td>
</tr>
<tr>
<td>Event</td>
<td>69</td>
</tr>
<tr>
<td>Element</td>
<td>44</td>
</tr>
<tr>
<td>SVGElement</td>
<td>43</td>
</tr>
<tr>
<td>Node</td>
<td>31</td>
</tr>
<tr>
<td>AudioNode</td>
<td>21</td>
</tr>
<tr>
<td>BASEAudioContext</td>
<td>20</td>
</tr>
<tr>
<td>Blob</td>
<td>19</td>
</tr>
<tr>
<td>SVGAnimatedString</td>
<td>19</td>
</tr>
<tr>
<td>CSSRule</td>
<td>18</td>
</tr>
<tr>
<td>SVGAnimatedEnumeration</td>
<td>18</td>
</tr>
<tr>
<td>SVGAminatedLength</td>
<td>18</td>
</tr>
<tr>
<td>XRRIgIDTransform</td>
<td>17</td>
</tr>
<tr>
<td>SVGAminatedNumber</td>
<td>16</td>
</tr>
<tr>
<td>MediaStream</td>
<td>15</td>
</tr>
<tr>
<td>XRSpace</td>
<td>15</td>
</tr>
<tr>
<td>NodeList</td>
<td>14</td>
</tr>
<tr>
<td>HTMLICollection</td>
<td>14</td>
</tr>
<tr>
<td>Document</td>
<td>14</td>
</tr>
<tr>
<td>ReadableStream</td>
<td>14</td>
</tr>
<tr>
<td>HTMILFormElement</td>
<td>13</td>
</tr>
</tbody>
</table>
What does that mean for editors?

- Better definition tracking requires work from spec editors to ensure consistency
  - Avoid proxy definitions
    - `<dfn><a href="...">RTCPeerConnection</a></dfn>`
  - Use only exported definitions from other specs
  - Export the definitions that other specs need

- Formalize input / output of spec algorithms

- Expected new tools
  - Detect which specs depend on your definitions (useful when you to change it - automate in PR reviews?)
  - More fine-grained dependency map across specs (useful e.g. for transition requests)
  - Automatic tracking with MDN BCD
Formalizing support for dev tools

- Webref will provide NPM packages for:
  - IDL
  - CSS definitions
  - ... (as needed)
Structural resources

- So far, mostly driven by individuals as side projects
- SpecInfra OpenCollective (Specref)
- ReSpec OpenCollective

https://xkcd.com/2347/ CC-BY-NC
Questions

- Any other tools that you are working on / aware of?
- Are there other useful scenarios to consider for spec data?
- How would you want to keep track of dependencies to and from your specs?
- Are benefits (spec quality, dev ecosystem) worth the editing efforts?
- How to coordinate? Is spec-prod@w3.org the right tool?
- Data model for definitions: do we need new types, e.g. for algorithms?