TECHNICAL NOTE

The Human Process of Web Standards Generation: A Journey from Idea to Implementation

Paola Di Maio, W3C AI KR CG

30 October 2025

Abstract

The human-centered, multi-year process required to generate and implement web standards, primarily within the World Wide Web Consortium (W3C) framework (see W3C "Process Document") typically spans 4 to 7 years, begins with Problem Identification by developers and proceeds through formal stages like Community Group Formation, Specification Drafting, and Working Group Adoption. The core of the process involves Iterative Refinement and rigorous Horizontal Review (W3C "Horizontal Review"), ensuring compatibility, security, and broad usability. The paper stresses that the extended timelines reflect not just technical difficulty, but the fundamentally human challenge of consensus-building among diverse stakeholders (Rescorla). While slow, this deliberative nature ensures standards are robust and compatible. The conclusion suggests that the future lies in an AI-human symbiotic relationship that augments human judgment and wisdom while accelerating mechanical tasks.

Introduction

Web standards are the invisible foundation of the internet's interoperability, yet their creation follows a remarkably human process—one characterized by deliberation, consensus-building, and painstaking attention to edge cases. Understanding this traditional process is essential for appreciating both its strengths and the opportunities for augmentation through AI-assisted tools.

The Lifecycle of a Web Standard

Phase 1: Problem Identification and Incubation (3-12 months)

Standards typically begin not in committee rooms, but in the trenches of web development. A developer encounters a limitation, a company identifies a market need, or a community recognizes a pattern of workarounds that signals missing platform capabilities. This recognition phase can span months or years as the problem crystallizes across multiple implementations.

The incubation period involves informal discussions in forums, blog posts articulating the problem space, and experimental libraries or polyfills that prototype solutions. For example, the need for responsive images simmered in the developer community for years before formal standardization began, with developers sharing increasingly complex JavaScript workarounds for serving appropriately sized images to different devices.

During this phase, champions emerge—individuals or organizations willing to invest significant time in pursuing standardization. These champions must build a compelling case that the problem is widespread enough to warrant platform-level solution rather than library-level innovation.

Phase 2: Community Group Formation (2-6 months)

Once sufficient interest coalesces, participants typically form a W3C Community Group (CG) or WHATWG workstream. This stage involves:

• Writing an initial charter defining scope and goals

• Recruiting diverse stakeholders (browser vendors, developers, accessibility experts) • Establishing communication channels (mailing lists, GitHub repositories, regular calls) • Creating initial use cases and requirements documents

The Community Group phase is deliberately lightweight, allowing rapid exploration without the formal obligations of a Working Group. However, this informality can also lead to unclear intellectual property commitments and uncertain paths to standardization.

Phase 3: Specification Drafting (6-18 months)

The first draft of a specification represents a crucial transition from problem description to solution proposal. This phase demands particular expertise:

Technical Writing: Specifications must be precise enough for independent implementation yet readable enough for review. Authors learn to write in "spec-ese"—a peculiar dialect that defines behavior through algorithms, state machines, and normative requirements.

Abstraction Design: Authors must anticipate use cases beyond those initially considered, designing APIs that remain coherent as they evolve. This requires imagining how web development might change over the specification's multi-decade lifetime.

Cross-Specification Integration: New specifications must integrate seamlessly with existing standards. Authors spend countless hours reading related specifications, understanding their models and terminology, and ensuring consistent behavior at intersection points.

A typical drafting process involves:

1. **Initial proposal** (1-2 months): Basic API shape and primary algorithms

2. **Iteration based on feedback** (3-6 months): Weekly calls discussing edge cases 3. **Implementation prototyping** (2-4 months): Browser vendors create experimental implementations 4. **Specification refinement** (2-6 months): Discoveries from implementation fed back into spec **Phase 4: Working Group Adoption (3-6 months)**
Transitioning from Community Group to Working Group represents a significant commitment escalation.
This requires:

• Demonstrating implementer interest (usually 2+ browser vendors)

• Securing organizational support for multi-year participation

• Formal charter development with precise scope and timelines

• W3C Advisory Committee review and approval

• Patent commitment gathering from all participants

Working Groups operate under the W3C Process Document, which mandates specific stages and review requirements. Participants must officially represent their organizations, and decisions carry legal implications through patent policies.

Phase 5: Iterative Refinement (1-3 years)

The Working Group phase involves intensive iteration between specification text and implementation reality:

Weekly/Biweekly Calls: Groups discuss issues, often spending entire hours on single algorithm steps or API names. These discussions must balance competing concerns: web compatibility, implementation complexity, author ergonomics, security, privacy, and accessibility.

Issue Tracking: A successful specification might accumulate hundreds of GitHub issues. Each requires triage, discussion, and resolution. Edge cases multiply exponentially—what happens when features X and Y interact during network failure while the user navigates away?

Test Suite Development: The Web Platform Tests project requires comprehensive test coverage. Writing tests often reveals specification ambiguities, leading to further refinement cycles. A mature specification might have thousands of tests.

Implementation Feedback: As browsers implement features behind flags, real-world testing reveals unexpected interactions. The Fetch specification, for example, underwent major revisions as implementation revealed security model complexities.

Phase 6: Horizontal Review (2-4 months)

W3C requires horizontal review from specialized groups:

• **Technical Architecture Group (TAG)**: Architectural consistency and web platform coherence • **Accessibility**: WCAG compliance and usability with assistive technologies

• **Internationalization**: Proper handling of world languages and writing systems • **Security and Privacy**: Threat model analysis and mitigation strategies

Each review can trigger substantial specification changes. The Payment Request API, for instance, underwent significant redesign following privacy review concerns about merchant tracking capabilities.

Phase 7: Candidate Recommendation (6-12 months)

Candidate Recommendation (CR) signals specification stability and calls for implementation experience. Requirements include:

• Feature complete with no known major issues

• At least two independent, interoperable implementations

• Comprehensive test suite with implementation reports

• Formal objection resolution for any dissenting opinions

• Wide review including public feedback period

During CR, the specification is essentially frozen except for changes based on implementation experience. This phase tests whether the specification actually works in practice across diverse implementations.

Phase 8: Recommendation Track (3-6 months)

The final stages involve:

• **Proposed Recommendation**: Final review by W3C membership

• **W3C Recommendation**: Official standard status

• **Maintenance**: Ongoing errata and eventual version updates

Even after Recommendation, specifications require maintenance. The HTML specification, despite being a "living standard" in WHATWG, continues evolving with dozens of changes monthly.

The Human Timeline

The complete journey from problem identification to implemented standard typically spans **4-7 years**: • Year 0-1: Problem recognition and community discussion

• Year 1-2: Community Group formation and initial drafting

• Year 2-3: Working Group chartering and early implementations

• Year 3-5: Iterative refinement and horizontal review

• Year 5-6: Candidate Recommendation and testing

• Year 6-7: Final standardization and broad implementation

However, complex specifications can take much longer. CSS Grid Layout took over a decade from initial proposal to widespread implementation. WebRTC required nearly eight years to reach Recommendation status.

The Hidden Costs

Beyond the visible timeline lie substantial hidden costs:

Expertise Development: Becoming an effective specification author requires years of experience. Understanding the subtle interplay between specifications, learning historical context for design decisions, and developing intuition for web compatibility takes significant time investment.

Coordination Overhead: A typical Working Group call might have 20 participants, each preparing for

hours beforehand. Multiply this across dozens of calls yearly, and the human-hours become staggering.

Implementation Burden: Browser engineers must review specifications, participate in discussions, implement features, and fix bugs—all while maintaining existing systems. A single engineer might spend months implementing a seemingly simple feature.

Opportunity Cost: While standards development proceeds deliberately, the web ecosystem continues evolving. JavaScript frameworks solve problems in userland while platform features slowly gestate, leading to fragmentation and redundancy.

The Consensus Challenge

Perhaps the most human aspect of standards development is consensus-building. Unlike traditional software development where a single team makes decisions, web standards must satisfy:

• Multiple browser vendors with different architectures

• Web developers with varying skill levels and use cases

• Accessibility communities ensuring universal usability

• Security researchers identifying threat models

• Privacy advocates protecting user rights

• International communities with diverse needs

Reaching consensus often requires careful diplomacy, creative compromises, and occasionally painful trade-offs. The debate over HTML5's parsing algorithm, for instance, required years of negotiation between idealized behavior and web compatibility reality.

Institutional Knowledge and Tribal Wisdom

Significant specification knowledge remains uncodified, passed through mentorship and experience:

• Understanding which patterns have failed historically

• Knowing key individuals' positions and concerns

• Recognizing when consensus is impossible versus merely difficult

• Intuiting which compromises preserve future extensibility

• Navigating the social dynamics of standards organizations

This tribal knowledge creates barriers for newcomers and concentrates influence among longtime participants, potentially limiting diverse perspectives.

Conclusion: The Human Element

The traditional web standards process embodies both the strengths and limitations of human-centered design. Its deliberative nature ensures thorough consideration of edge cases, broad stakeholder input, and careful attention to compatibility. The consensus requirement, while slow, creates buy-in necessary for implementation investment. Yet this process also exhibits distinctly human limitations: cognitive load constraints that miss interactions between features, social dynamics that can override technical merit, and expertise barriers that limit participation. The multi-year timelines reflect not just technical complexity but also the fundamentally human challenge of building consensus among diverse stakeholders with competing interests.

Understanding this human process—its careful deliberation, its hidden complexities, its social dynamics—is essential for designing AI systems that augment rather than replace human judgment in standards development. The goal is not to eliminate the human element but to amplify human capacity, allowing standards bodies to tackle more complex problems, include more diverse voices, and respond more quickly to web platform needs while maintaining the careful consideration that makes web standards robust and enduring. As we consider AI-assisted standards development, we must preserve what makes the human process valuable: the wisdom accumulated through experience, the empathy that considers diverse use

cases, and the judgment that balances competing concerns. The challenge lies not in replacing human deliberation but in augmenting it—maintaining the thoughtfulness while accelerating the mechanical, preserving consensus while reducing redundancy, and expanding participation while ensuring quality.

The future of web standards likely lies not in choosing between human and artificial intelligence, but in crafting a symbiotic relationship where each amplifies the other's strengths—a true co-evolution of human wisdom and machine capability in service of a better web.

REFERENCES

Andrew, Rachel. *The Story of CSS Grid Layout*. Smashing Magazine, 2017.

Marquis, Mat. "Responsive Images: How They Got Here." *A List Apart*, 2013.

Responsive Images Community Group. *W3C Community Group Charter*. W3C, 2012. https://www.w3.org/community/respimg/

Rescorla, Eric. "Consensus and the IETF." *IETF Blog*, 2016.

W3C. *WebRTC 1.0: Real-Time Communication Between Browsers*. W3C Recommendation, 2021–2025 updates. https://www.w3.org/TR/webrtc/

W3C. *Process Document*. W3C, 2025. https://www.w3.org/policies/process/

W3C Privacy Interest Group. *Privacy Review of Payment Request API*. W3C Draft Review Notes, 2017–2020.

WHATWG. *HTML Living Standard*. WHATWG, 2025. https://html.spec.whatwg.org/

WHATWG. *Fetch Standard*. WHATWG, 2025. https://fetch.spec.whatwg.org/