

博士論文 平成 26(2014)年度

実装主義プロセスの標準仕様収斂効果：
W3C におけるマルチステークホルダー協働形態確立
によるイノベーション創出事例分析

慶應義塾大学大学院 政策・メディア研究科

深見嘉明

実装主義プロセスの標準仕様収斂効果： W3C におけるマルチステークホルダー協働形態確立による イノベーション創出事例分析

本論文は World Wide Web Consortium(W3C)において、創設以来 20 年にわたる仕様策定プロセス運営方針の変化と、標準仕様が確立した実効性を時系列比較することにより、標準仕様収斂に寄与する要因に関する仮説を導出した。

表計算やワードプロセッサは、これまでデバイスにインストールされた OS 上で動作するネイティブアプリケーションによって提供されていた。W3C 標準仕様である HTML5 は、アプリケーションの実行環境がサーバに移行したウェブアプリケーションと呼ばれるサービスを実現した。

ウェブのように多様なモジュール間の相互可用性を前提とする分野においては、合意形成を通じた仕様の収斂・統一が必要である。つまり、ドミナントデザインの移行をステークホルダーの合意のもとに実現する「協調型イノベーション」の実現が必要である。HTML5 は、ネイティブアプリケーションから巨大な利潤を得ていた Microsoft と、Google、Apple 等他のブラウザ供給者とが協力して策定され、ドミナントデザインを移行させた「協調型イノベーション」の事例であるといえる。

合意形成を通じた仕様の収斂と、標準の普及には、1) 仕様のカバーするスコープを「機能」で定義する、2) 複数の実装事例が存在する仕様のみを標準として認証するという「実装主義」とよばれる方針の採用、3) プロセスのオープン化を通じた多様なステークホルダーの参画が協調型イノベーションの実現に有効であるという仮説の抽出に至った。実装主義は互換性標準の特徴であるネットワーク外部性を逆手にとって実装事例の創出と、実装事例を元にしたフィードバックを踏まえた洗練化を促進する。この際、策定仕様のスコープを実現する機能によって定義することにより、同一機能を提供する独自仕様の乱立を抑制することができる。また、提案者による外部開発者向け実装事例創出支援により仕様の受容性が拡大する。

加えて、4) ロイヤルティ・フリー・ポリシーが明文化されることにより、人材の流動性が高いウェブ技術者が組織の垣根を超えて連携することが促進されるとともに、提案者の所属が変わっても、議論・開発を継続することが容易になる。また、イノベーションの源泉を外部に広く求めることも可能となる。

キーワード：実装主義、標準化、マルチステークホルダー、イノベーション、ドミナントデザイン

慶應義塾大学 政策・メディア研究科
深見嘉明

Doctoral Dissertation Academic Year 2014

Convergence Promoting Effect of Implementation-Oriented Process

: An Analysis of Innovation by Multi-Stakeholder Collaborative
Standardization at W3C

This thesis analyzes and proposes hypotheses regarding how the World Wide Web Consortium (W3C) developed a process that realized the convergence of standards among fiercely competitive players. This analysis involved a study of internal documents and more than 230,000 emails from archives since the standard-setting organization was established.

Particular focus was placed on standards that realized web applications. HTML5, the newest version of W3C's standard, realizes a dominant design of business tools referred to as "web applications" and allows applications such as spreadsheets to run on servers instead of on client hardware regardless of the type of browser the client uses. It works only when all components run together according to the standards. Architectural innovation with standardization for multiple modules can be realized only with "concerted innovation" among all players in the Web domain.

Thus, HTML5 is a novel example of a concerted innovation by Microsoft that earns substantial profits from existing dominant designs, Google, Apple and other browser vendors. This standard has encouraged the transition of dominant design from native applications to web applications.

We identify and propose as hypotheses four factors for the success of standardization: 1) defining scope of specifications to be developed by functions instead of technical structures, 2) development management policy based on feedback on implementations, referred to as "implementation-oriented policy," 3) inclusion of diverse stakeholders in open standardization processes that facilitates consensus formation and diffusion of developed standards.

Implementation-oriented policy takes advantage of network externality and enjoys the benefits of accelerated implementation cycles, feedback that leads to the sophistication of specifications, and spread of market receptivity. Defining scope of specifications by functions restrains emergence of competing proprietary specifications.

Additionally, 4) loyalty-free-policy encourages cooperation among engineers from different organizations and enables the procurement of diverse sources of innovation.

Keywords: implementation-oriented process, standardization, multi-stakeholder,
concerted innovation, dominant design

Yoshiaki Fukami
Graduate School of Media and Governance
Keio University

目次

第1章. 背景	15
1.1. 市場競争を通じた普及から、協調型イノベーションへの移行.....	15
1.2. ウェブアプリケーションのアーキテクチャと必要とされる技術.....	18
1.3. 利害対立があるステークホルダー間の協働	21
1.4. 標準を巡る競争と協働の歴史	25
1.5. 本論文のテーマと構成.....	26
第2章. 先行研究レビューを通じた制約の特定	28
2.1. ウェブアプリケーションを成立させる標準の特徴と制約	28
2.1.1. 互換性標準	28
2.1.2. ネットワーク外部性	28
2.1.2.1. 直接的効果と間接的効果、過剰慣性	28
2.1.2.2. スイッチングコストとロックイン	30
2.1.3. レイヤード・モデュラー・アーキテクチャにおけるプラット フォーム	30
2.1.4. コーディネーションとイノベーション遅滞のトレードオフ	33
2.1.5. 互換性標準のもつ制約をいかにして乗り越えるか	33
2.2. 標準化プロセスに関与するステークホルダーとそれぞれのイン センティブ	34
2.2.1. プラットフォーム・リーダーシップ確立を通じた利潤獲得 機会の創出	34
2.2.2. 時間をかけて協調するよりも、素早く独自仕様を普及させる	35
2.2.3. 紐付け戦略による支配	36
2.2.4. 水平分業を通じた価値創出コストの分担とイノベーションの 源泉拡大	36
2.2.5. 貢献へのインセンティブとコーディネーションメカニズムの 設計という課題.....	39
第3章. 調査課題と調査手法	40
3.1. 調査課題の導出	40
3.2. 標準化の成否を規定する測定基準	41
3.3. 分析対象ならびに調査設計	42

3.4. 調査手法と活用データ	45
3.4.1. W3C オフィスにおけるフィールドワーク	45
3.4.2. メーリングリスト・アーカイブ分析	47
3.4.2.1. 分析対象リストの選定	47
3.4.2.2. アーカイブ分析手法	50
3.4.3. 非構造化インタビュー	50
3.4.4. その他資料、ならびにウェブ上のツール分析	51
第 4 章. Phase 0: W3C 発足以前のウェブ標準を取り巻く状況	52
4.1. 初期のウェブ標準化に対する試み	52
4.1.1. ブラウザ開発者間の独自機能追加と仕様の分裂	52
4.1.2. IETF における標準化	53
4.1.3. W3C 設立と HTML3.0 仕様策定の失敗	55
4.2. 当初プロプライエタリ技術として登場したウェブアプリケーションのための仕様群	57
第 5 章. Phase 1: 議論の場作りが手探りで始められた勃興期 - HTML3.2/ 4.0	59
5.1. 策定作業の W3C への移行と手探りでのプロセス開始	59
5.2. 現状追認としての HTML3.2	60
5.3. 策定プロセスの整備と HTML Working Group 始動	63
5.4. 独自実装による分裂状況が解消できなかった HTML4	64
5.5. 既に存在する独自実装を、後追いで統合する困難さ	66
第 6 章. Phase 2: プロセス運営方針の明確化が進んだ過渡期 - DOM Level 1/ 2	69
6.1. Document Object Model の概略と WG 発足に至る経緯	69
6.2. 議論のチャンネル増加とオープン化	71
6.2.1. Draft 公開の迅速化と公開メーリングリストの導入	71
6.2.2. Interest Group の設置	73
6.3. Process Document における Technical Report ステップの整備	73
6.4. 未実装の機能をスコープとして定めるマイルストーン設定	73
6.5. Level 1: 市場競争の追認としての統一	75

6.6. 実装主義の導入	79
6.6.1. Last Call for Comment の導入	79
6.6.2. Call for Implementation による実装主義の導入	79
6.6.3. 実装主義導入の意図	82
6.7. Level 2: 実効性のある標準仕様策定成功	83
6.7.1. モジュールに分割されて進められた仕様策定	83
6.7.2. 実装主義導入と 2 社の実装	84
6.7.3. Call for Implementation への外部ステークホルダーの関与を 進める動き	85
6.7.4. テストプログラム開発・供給の担い手における変化	85
6.7.4.1. 外部開発者コミュニティによるテストサービス	85
6.7.4.2. W3C による公式プログラムの開始	88
6.7.5. 市場占有率による影響が抑制された状態での仕様の確立と 普及	89
第 7 章. Phase 3: 仕様案の競合を通じた運営方針の確立- HTML5/ XHTML.....	92
7.1. 2 つの次世代 HTML コンセプト	92
7.1.1. XHTML: アプリケーションで処理できるデータを 埋め込める HTML.....	92
7.1.2. HTML5: 既存仕様と後方互換性を保ちつつ、アプリケー ションとしての機能を充実させることを主眼においた仕様	94
7.2. クローズドな投票による HTML5 仕様提案の棄却	95
7.2.1. Workshop 決議による XHTML 開発開始	95
7.2.2. HTML5 の提案と否決	96
7.2.3. Workshop であらためて XHTML が選ばれる	101
7.3. 市場動向に後押しされての HTML5 策定プロセス開始	102
7.3.1. W3C 外部におけるオープンな仕様開発活動	102
7.3.2. Web Forms 2.0 仕様提案再提出	105
7.3.3. ブラウザ供給者とコミュニティ主導のコンセンサス形成	106
7.3.3.1. XHTML の普及状況	108
7.3.3.2. WHATWG HTML5 (Web Forms 2.0 + Web Applications 1.0)実装状況	110

7.4. 更なるプロセスのオープン化とロイヤルティ・フリー・ポリシー の確立	111
7.4.1. 新 HTML WG の活動内容と目標設定	111
7.4.2. 新 HTML WG 発足と体制固め	112
7.4.3. ロイヤルティ・フリー・ポリシーの明文化	115
7.5. 次世代 HTML 仕様の HTML5 への一本化	115
7.5.1. XHTML の一部機能を HTML5 に残すための試み	115
7.5.2. XHTML2 WG の機能不全	117
7.5.3. 2009 年 3 月 AC Meeting	118
7.5.4. Mozilla による XHTML の実装除外	119
7.6. 機能開発ならびに普及促進施策の新たな形態	119
7.6.1. 独自仕様の実装を元にした標準仕様提案 : Google のオープン標準戦略	119
7.6.1.1. ブラウザ参入計画と Mozilla Firefox 開発への貢献	119
7.6.1.2. アドオンによる独自先行実装と標準化プロセスへの持ち込み	120
7.6.1.3. 自社製ブラウザ、Chrome の市場投入	122
7.6.1.4. Gears 開発中止、仕様のブラウザ実装と標準化提案	123
7.6.2. オープン標準仕様に関するディベロッパー・リレーション 施策の活用	126
7.6.2.1. ディベロッパー・リレーション施策とは	126
7.6.2.2. 外部開発者を通じた普及施策	127
7.6.2.3. 中立性を保つことによる訴求効果の拡大	131
7.6.2.4. コミュニティによる自律的な活動へ	134
7.6.3. マルチステークホルダーによる協働形態の確立	137
7.6.3.1. Social Web Incubation Group の活動と、外部開発者 コミュニティとの連携強化	137
7.6.3.2. オープンなテストプログラム開発プロジェクト	140
7.6.3.2.1. スポンサーから公式活動へ	140
7.6.3.2.2. OSS 開発ツールの積極的活用	141
7.6.3.3. webplatform.org を通じた普及活動における協働	143
7.7. ステークホルダーの協調体制確立	144
7.7.1. Microsoft が HTML5 を実装できない事情	144

7.7.2. 最新の(しかし策定が終了していない)標準に対応しない Microsoft への反発	145
7.7.3. Microsoft の方針転換	146
第 8 章. 仮説導出.....	150
8.1. 各仕様の実効性獲得状況	150
8.2. 仕様策定プロセス運営方針の変化と仮説導出	151
8.2.1. 仕様がカバーするスコープを「機能」で定義する	151
8.2.2. 仕様策定プロセスにおける実装主義の導入.....	155
8.2.3. プロセスのオープン化	160
第 9 章. イノベーションの源泉拡張と、組織間の垣根を超えた 協働	165
9.1. 同一企業内での方針不統一	165
9.1.1. 焦点となった機能と競合した 2 つの仕様	165
9.1.1.1. ウェブアプリケーション間データ引き渡しプロトコル.....	165
9.1.1.2. 片方向のデータ引き渡しのみに対応した HTML5 に含まれる仕 様: registerProtocolHandler()、registerContentHandler()..	167
9.1.1.3. 往復のデータ引き渡しに対応した対案: Web Intents.....	168
9.1.2. ソーシャルメディアと OSS ツールを用いた仕様提案	169
9.1.3. W3C 内での支持拡大と、企業の垣根を超えた機能開発	170
9.1.4. Google 社内で支持拡大するも、HTML5 仕様への組み込みは 失敗に	175
9.1.5. 企業間の垣根を超えた協働の継続	177
9.2. ロイヤルティ・フリー・ポリシーの明文化による効果	180
9.2.1. 組織の垣根を超える協働の促進	180
9.2.2. 法人内で集約されていれば日の目を見なかった提案の 取り込み	183
第 10 章. 考察と今後の課題.....	185
10.1. 思想としての W3C 標準化プロセス	185
10.1.1. オープンなプロセス、オープンな仕様	185
10.1.2. 学術コミュニティを基盤とした協働プラットフォーム.....	188
10.1.3. “Rough consensus, running code”の制度化としての 実装主義	190

10.2.プラットフォーム・ガバナンスと競争政策に関する議論	191
10.2.1. 垂直統合に対する政府の介入は必要か	191
10.2.2. 互換性標準策定を担う主体は誰が担うべきか	192
10.2.3. 民間標準化団体の経済的自立を成立させる施策の必要性.....	193
10.3.実装主義の適用範囲についての検討.....	194
10.3.1. ソフトウェアの特徴と相性がいい実装主義.....	194
10.3.2. 財のサービス化による適用範囲拡大の可能性	197
10.4.本論文の貢献と今後の課題	198
謝辞	201
付録 I. W3C の組織形態	206
I.i. 会員制の産業コンソーシアム	206
I.ii. 組織形態.....	206
I.iii. Technology Architecture Group と Director.....	207
I.iv. Advisory Committee と Advisory Board	207
I.v. Management Team	208
I.vi. de Jure 標準との関係.....	208
付録 II. HTML というウェブの基幹技術.....	209
II.i. ウェブ登場時のコンセプト	209
II.ii. 構造化文書としての HTML.....	210
II.iii. 構造化の機能に特化した XML の登場	214
参考文献	216

目次

図 1. ネイティブアプリケーションとウェブアプリケーションの違い	16
図 2. ウェブアプリケーションの他サービスとの連携事例 (Google Drive, Spreadsheet)	17
図 3. ウェブアプリケーションに必要とされる技術領域とアーキテクチャ	19
図 4. HTML5 の技術領域	21
図 5. Microsoft の供給する表計算アプリケーション (ネイティブアプリ ケーションとウェブアプリケーション)	24
図 6. 利害関係が対立する主体間の協働によるドミナントデザインの移行	25
図 7. ウェブアプリケーションでの協働作業は HTML5 対応ブラウザがイン ストールされたデバイス間のみで可能	29
図 8. ウェブアプリケーションのレイヤード・モデュラー・アーキテクチャ	32
図 9. 推奨ブラウザを示すバナー	42
図 10. W3C で策定されている HTML ならびに HTML5 の先行仕様の 策定時期と Process Document 制定・改定時期	43
図 11. HTML5 の先行仕様・競合仕様	44
図 12. W3C/Keio オフィスがおかれていた 慶應義塾大学湘南藤沢 キャンパスタウ館	46
図 13. W3C における電話会議の様子。IRC でリアルタイムに速記録を 作成する。	47
図 14. IETF の標準化プロセス (RFC1602)	54
図 15. プロプライエタリな仕様として提供されていたウェブアプリ ケーション 実現のための技術仕様群	58
図 16. W3C Markup Validation Service	66
図 17. HTML3.2/ HTML4 策定の構図	67
図 18. ウェブアプリケーションアーキテクチャにおける DOM の位置づけ	69
図 19. Document Object Model の仕様策定手法	74

図 20. Process Document によって定められた標準化プロセスにおける 仕様書(Technical Report)のステータス	80
図 21. 実装主義プロセス	81
図 22. Acid 1 の結果画面 (仕様が正しく実装されているとこのように 表示される)	87
図 23. DOM Level 2 仕様策定への Netscape の関与 (左：2000 年 11 月 13 日版 WD、右：2001 年 12 月 7 日版 WD).....	90
図 24. XML ベースで HTML を再開発するメリット	93
図 25. HTML 中の情報をプログラム処理した例.....	93
図 26. ウェブサイト(HTML)内に設置された Form の例	95
図 27. 当初の XHTML と HTML がスコープに入れていた機能、 ならびにアーキテクチャの差異.....	95
図 28. 2003 年時点におけるウェブアプリケーション実現のための 機能開発・標準化動向.....	98
図 29. WHATWG メーリングリストにおける 一般エンジニアからの フィードバック	105
図 30. public-html@w3.org メーリングリストにおける 非会員からの フィードバック	114
図 31. 2009 年 3 月 AC ミーティングの様子	118
図 32. Test the Web Forward Tokyo 2013 の様子.....	141
図 33. GitHub の Test the Web Forward 等、W3C 標準仕様テストの リポジトリ	142
図 34. Twibbon IE6 Must Die.....	147
図 35. Internet Explorer 6 の葬式というイベントの公式ウェブサイト	148
図 36. 各バージョンの策定時期と主要ブラウザの実装時期.....	151
図 37. W3C における仕様策定プロセス運営ルールの変化	152
図 38. 仕様のスコープを「機能」で定義した場合のプロセス	154
図 39. 実装主義導入による仕様策定プロセスの変化.....	156
図 40. ネイティブ OS におけるアプリケーション間データ引き渡しの プロセス.....	166
図 41. ユースケース 1: 片方向のデータ引き渡し	166
図 42. ユースケース 2: 往復でのアプリケーション連携.....	167

図 43 . Android OS における Intent のしくみ (画像は Version 1.6 のもの)	168
図 44 Web Intents における引き渡し先アプリケーション選択ウィンドウ	169
図 45. これまでの一般的なソフトウェア互換性標準のライフサイクル モデル.....	195
図 46. ウォーターフォール型開発モデルのステップ(Royce, 1970; pp. 329)	195
図 47. W3C の組織形態図	207
図 48. Tim Berners-Lee が CERN に提出したウェブの概念図.....	210
図 49. 文書を構造化する利点の一例	213
図 50. 同一サイトを同一ベンダのブラウザ(Apple Safari)を用い、 ラップトップマシン(左)とスマートフォン(右)で表示させた例	214

表 目次

表 1. ネイティブアプリケーションとウェブアプリケーションとの間の競 合関係	22
表 2. 分析対象作業部会リスト	48
表 3. 本論文で分析したメーリングリスト	49
表 4. インタビュー実施リスト	50
表 5. 1995 年 11 月に Dan Connolly が結成したグループの参加者	60
表 6. HTML4.01 とそれ以前の HTML 標準化動向	68
表 7. DOM Level 1 の標準化動向	78
表 8. DOM Level 2 仕様策定プロセス初期のモジュールと担当 Editor	85
表 9. DOM Level 2 の標準化動向	91
表 10. 2006 年 5 月時点における XHTML/ XForms 標準化プロセス進捗状況	108
表 11. W3C に設置される作業部会(Group).....	139
表 12. ウェブアプリケーション間データ受け渡し プロトコル仕様の収斂 に至る経緯	178

第1章. 背景

1.1. 市場競争を通じた普及から、協調型イノベーションへの移行

創造的破壊 (Schumpeter, 1950)、破壊的イノベーション(Christensen, 1997) という言葉が表すとおり、イノベーションによる新たな製品・サービス形態の導入、つまりドミナントデザイン (Abernathy and Utterback, 1978; Utterback; 1996; Tushman and Murmann, 2002) の移行は競争に敗れた旧来型製品・サービスの市場退出をひきおこしてきた。コンピュータが機械式タイプライターを、CD がアナログレコードを、LED 照明が白熱灯を市場から駆逐するといった破壊的イノベーション (Christensen, 1997) が生じた際、それぞれの市場において支配的な地位を占める企業も変化してきた。イノベーションの普及のために重要視されてきたのは新・旧両タイプの競合製品・サービスに打ち勝つことであった (Utterback, 1996)。

情報通信の分野においても、多くのイノベーションは市場競争を経て普及してきた。メインフレームがワークステーションに、ワークステーションがパーソナルコンピュータによって市場を侵食されてきた。ワープロ専用機にとって変わるべく登場した PC 用ワードプロセッサソフトウェアは、急速にワープロ専用機市場を破壊した後に、Word Perfect と Microsoft Word という新タイプの製品間による激しく市場競争が繰り広げられた。ところが、「イノベーションの普及には市場“競争”を前提とした戦略が最も重要である」といった常識は、クラウドの普及によって時代遅れになろうとしている。

Word Perfect との競争に勝利した Microsoft Word を脅かす存在として登場したのが、Google Document や Microsoft Online といったウェブアプリケーションという形態で提供されるサービスである。Google Drive と Microsoft Online は、現在 Microsoft Office というパッケージ名で提供されているワードプロセッサ (Word)、表計算 (Excel)、プレゼンテーション (PowerPoint) 等の“製品”を代替する“サービス”である。

Microsoft Word や Excel は Windows PC、もしくは Machintosh マシンにインストールし、各 OS 上で動作する。ハードウェアにインストールされ、OS 上で動作するアプリケーションはネイティブアプリケーションと呼ばれる。

パーソナルコンピュータや Machintosh 等においては、アプリケーションはローカルデバイスのハードディスクにインストールされて動作している。電子メ

ールのようなインターネットアプリケーションであっても、本文の執筆や受信メールの表示などとデータ生成、処理はローカルデバイスにインストールされた電子メールソフトが担い、データはハードディスクに保存される。

一方 Google Drive などのウェブアプリケーションは、サービス提供者が自社のサーバ上でデータ生成・処理を行い、その結果はインターネットを介してローカルデバイスとやりとりされる。ユーザはウェブブラウザをインターフェイスとしてサービスを利用するという形態となる。つまり、ウェブアプリケーションとは、アプリケーションの実行環境が（手元の）デバイスからサーバに移行したものとなる(図 1)。

このようなアーキテクチャをもつため、ウェブアプリケーションはインターネットに接続されたデバイスであれば、複数のデバイスからシームレスに機能を利用することができる。例えば、デスクトップコンピュータで作成した表計算のファイルを、タブレットで編集し、スマートフォンで閲覧するといったことが容易にできる。また、複数のデバイスから同時にファイルにアクセスし、共同で編集することも可能である。

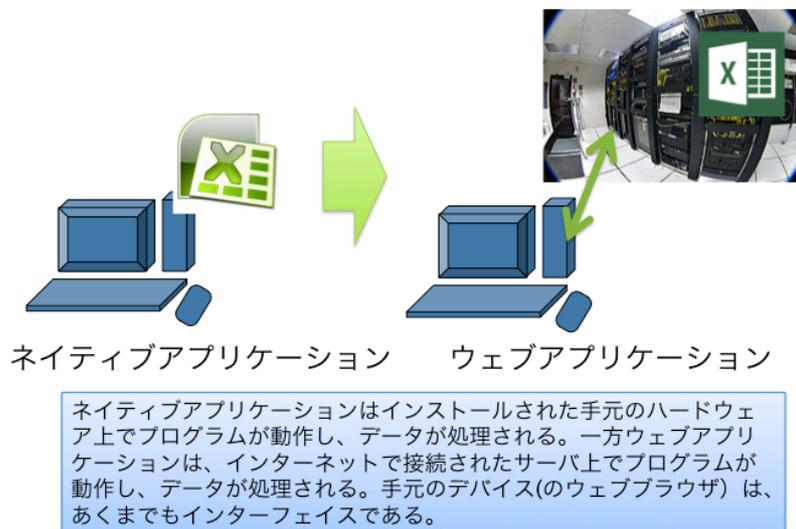


図 1. ネイティブアプリケーションとウェブアプリケーションの違い

更に、ウェブ上で提供されているデータベースと連携することにより、リアルタイムでデータを更新することも可能である(図 2)。ウェブアプリケーションはネイティブアプリケーションが実現できない機能を提供できる新たなドミナントデザインであり、ネイティブアプリケーション供給者の立場から見ると創造的破壊 (Schumpeter, 1950)にほかならない。そしてサーバ/ブラウザ/(ローカル)デバイスの役割分担を変える、言い換えるとアーキテクチャの改変を伴うイノベーション (Architectural innovation) (Abernathy and Clark, 1985; Henderson and Clark, 1990)ともいえる。

ネイティブアプリケーションは PC 等のユーザが直接操作するデバイスにインストールされ、デバイスの OS 上で動作する。ネイティブアプリケーションの設計や機能は OS に依存するため、Microsoft Word を含む多くの製品は OS の種別に対応した幾つかのバージョンを設けている。一方ウェブアプリケーションの場合手元のデバイスはインターフェイスに過ぎず、プログラムが動作するのはサーバ上である。サーバとのやりとりはインターネットを介し、のウェブブラウザを通じて操作する。そのためウェブアプリケーションの機能はウェブブラウザやウェブ標準、インターネット標準仕様に依存する。Google Drive は Microsoft Windows 上の Internet Explorer (IE) から、Microsoft Office Online は MacOS 上の Google Chrome から利用可能である。

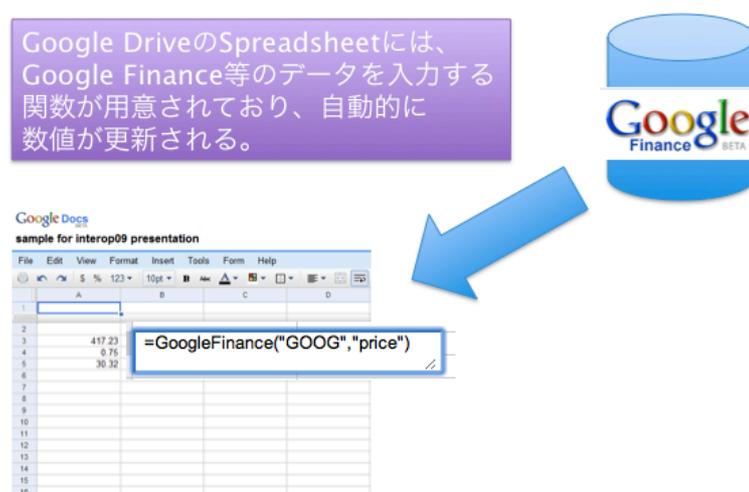


図 2. ウェブアプリケーションの他サービスとの連携事例
(Google Drive, Spreadsheet)

IE や Chrome といったウェブブラウザの機能ならびに設計を規定する大きな要素がウェブ標準である。中でも重要な標準仕様が、国際標準化団体である World Wide Web Consortium (W3C) において策定される Hyper Text Markup Language (HTML) である。HTML はウェブサイトを記述するための言語仕様であり、ウェブの機能を規定する重要な要素となっている。HTML に機能が追加されたことによって、ウェブサイトに画像や動画が組み込まれるなど、ウェブの機能が拡張されてきた。ウェブアプリケーションが普及した要因の1つとして、HTML の最新バージョン、HTML5 の開発と普及を挙げることができる。

HTML5 は W3C において、Google や Microsoft だけではなく、Apple、Mozilla、Opera など、様々な企業・団体の協働によって開発・策定されてきた。ウェブアプリケーションの登場とネイティブアプリケーションの代替という現象は、「標準仕様の策定・バージョンアップという、多数の主体の協働を通じて実現されたイノベーション」であり、「ドミナントデザインの移行」といえる。

これまでドミナントデザインの移行は製品イノベーションによって生じた、新旧デザインベースの製品・サービス供給者間の競争によって実現されることが多かった (Utterback; 1996)。HTML5 の策定によるウェブアプリケーションの実現は、利害が対立するステークホルダーの協働を通じた協調型イノベーションという形態が生まれていることを示している。

1.2. ウェブアプリケーションのアーキテクチャと必要とされる技術

ウェブアプリケーションの実現には、サーバとローカルデバイスとの間で効率的にデータをやりとりするためのプロトコル、サーバ内の処理に応じて表示が更新されたり、データ入力を受け付けたりする高機能なインタラクティブ機能、ローカルデバイスで生成したデータを取得するための Application User Interface (API)、キーボードやタッチパネルなど多様なデバイスからの操作を可能とするユーザーインターフェイスなどの技術開発が必要となる。

ウェブアプリケーションは、様々な OS のデバイス、ウェブブラウザを通じて利用される。言い換えれば、ウェブアプリケーションのアーキテクチャは、モジュール化 (Simon, 1962; Sanchez and Mahoney, 1996; Baldwin and Clark, 2000; Langlois, 2002) されたコンポーネントが連携して動作する必要がある。そのため、ウェブアプリケーションの実現、機能向上には共通して実装される標準仕様の確立が不可欠である。特にブラウザの機能を規定するウェブ標準は、ウェブアプリケーション構築において重要な役割を果たす。

図 3 は、ウェブアプリケーションのアーキテクチャと、実装が必要とされる仕様群を示している。ウェブというインフラを構成する主要な要素は、ファイルを蓄積・送信するサーバ、サーバから送信されたファイル (Hyper Text Markup Language; HTML 等) を表示するブラウザ、ブラウザが動作する (ローカル) デバイスの 3 つである¹。

ウェブアプリケーションではデータの生成・処理をサーバが担い、その結果をリアルタイムでブラウザ/デバイスに送信する。ブラウザ/デバイスは、サーバに対し送信するデータを生成し、サーバから送られてきた処理済みのデータを表示する、というアーキテクチャによって実現されている。

この役割分担を可能とするには、ブラウザも継続的なデータのやり取りを受けて動的に表示を変えることが必要になる。これを実現するためには、サーバとブラウザ間で継続的かつ効率的に通信を行うことができるプロトコル、サーバから送信されたデータによって HTML を操作するためのサーバサイド API、リアルタイムに動的処理を行うためのプログラミング言語が必要となる。

サーバサイドでデータの生成・処理が完結する場合もあるが、デバイスサイドから入力されたデータやウェブページの中に埋め込まれたデータを処理するといったユースケースも想定される。ウェブページの中に計算機可読性のあるデータ (セマンティックデータ) を埋め込むための仕様も重要である。

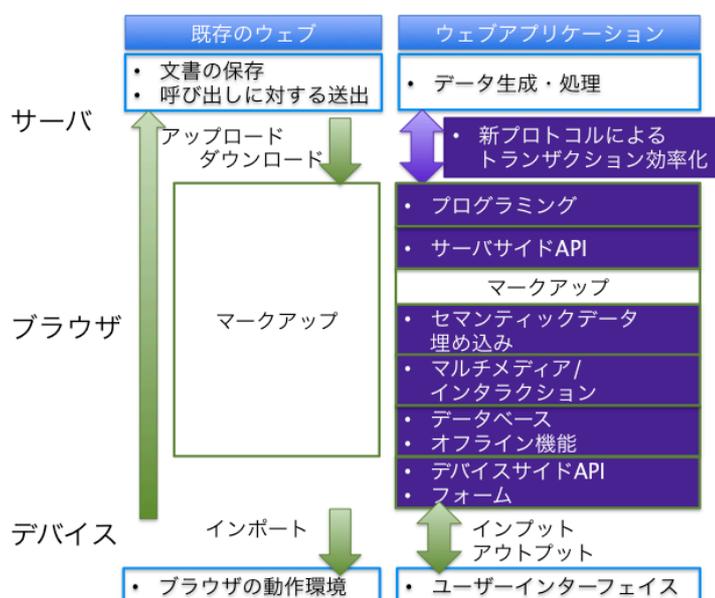


図 3. ウェブアプリケーションに必要な技術領域とアーキテクチャ

¹ サーバとブラウザ/デバイス間の通信はインターネットが担うが、インターネット以下の層についてはここでは省略する。

ウェブアプリケーションのコンセプトでは、(ローカル) デバイスの役割はインプット/アウトプットのインターフェイスに限定される。しかしコンピュータ、特にスマートフォンの普及により、デバイスが生成するデータの種別は急激に拡大している。カメラによって撮影された画像や動画、GPS から取得した緯度経度情報、地軸センサーから取得した傾きなどである。と同時にデバイスのアウトプット形式も多様になっている。ディスプレイでの表示のみならず、スピーカーからの音声、バイブレーション、LED ランプの点灯・点滅パターンなどによるアウトプットを活用するアプリケーションも数多く存在する。サーバサイドで処理された結果に基づき、デバイスに搭載された様々な部品を制御するための多様なデバイスサイド API も必要となる。また、多様なウェブアプリケーションが処理する多様な種別のデータを入力するためのフォームの取り揃えも増やさなくてはならない。

スマートフォンやタブレットデバイスの普及により、ウェブアプリケーションが利用される場所は屋外へと広がりを見せている。屋外では携帯電話網が普及したとはいえ、オフラインになることも頻繁にある。この状態に対応するため、ブラウザ内にデータベースを構築し、オフライン時でもアプリケーションが実行可能となる機能も必要である。

ウェブアプリケーションを構築するために必要とされる技術領域は多岐にわたるが、HTML5 仕様はその多くの領域をカバーしている (図 4)。よって、ウェブアプリケーションというイノベーションが実現した要因の中でも、特に重要なものの 1 つといえるであろう。ウェブアプリケーションは「標準仕様のバージョンアップを通じて実現したイノベーション」なのである。

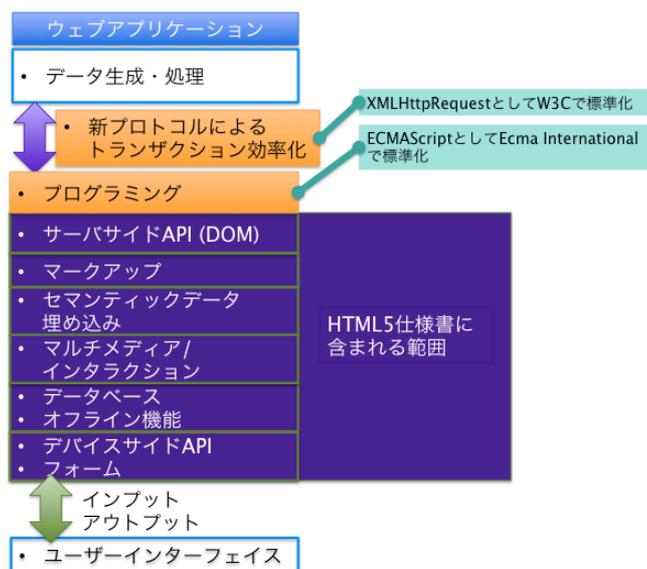


図 4. HTML5 の技術領域²

1.3. 利害対立があるステークホルダー間の協働

ウェブアプリケーションを実現した重要な要素である HTML5 は、Microsoft や Google を含むブラウザ供給者や、他の事業者・団体によって協調的に開発されたとともに、両社を含む全ての主要なブラウザ事業者が自社製品に実装している。しかし、HTML5 の策定に参加し、自社製品に実装した各社の利害は必ずしも一致していたわけではなかった。ここからは、HTML5 という標準仕様の策定、ならびにウェブアプリケーションというイノベーションの実現の背景について分析しておきたい。

ワードプロセッサの Word、表計算ソフトの Excel、プレゼンテーションソフトの PowerPoint など、オフィス業務で多用されるアプリケーションをパッケージした製品はオフィススイートと呼ばれる。

これまで Microsoft が寡占していたこの市場に、Google はウェブアプリケーションという形態で参入した。表 1 のように、Google が提供する Google Drive は、Microsoft 以外の企業が供給するものも含む幅広いネイティブアプリケーションと競合関係にある。

² 2014 年 4 月 29 版 Candidate Recommendation (Berjon, R. et al. (eds.) (2014, April 29). 1.4 History. *HTML5: A vocabulary and associated APIs for HTML and XHTML: W3C Candidate Recommendation 29 April 2014*. Retrieved May 25, 2014, from <http://www.w3.org/TR/2014/CR-html5-20140429/introduction.html>)を基にしている。

表 1. ネイティブアプリケーションとウェブアプリケーションとの間の競合関係

	ネイティブアプリケーション	ウェブアプリケーション
メール	Microsoft Outlook、 Mozilla Thunderbird 等	Gmail (Google)等
表計算	Microsoft Excel	Google Spreadsheet (現 Google Drive の一部)
ワードプロセッサ	Microsoft Work、 Justsystem 一太郎	Google Docs (現 Google Drive の一部)

ネイティブアプリケーションはパッケージソフトウェアの販売により収益を上げるビジネスモデルである。一方ウェブアプリケーションの多くは無償で提供されている。Google Drive はクラウドストレージ(保存領域)に対して課金するというビジネスモデルであり、表計算やワードプロセッサの機能を利用すること自体に対する課金は発生しない。

しかも、一定量以下の保存領域は無料で利用することができる。ウェブアプリケーションの利用にはアカウントの登録が必要であり、同一アカウントによる多岐に渡るウェブアプリケーションやコンテンツ利用時に、蓄積した個人情報を利用して広告を配信することによって収益を上げることにつながる。

W3C に HTML5 の仕様策定が初めて提案された 2004 年 6 月の W3C Workshop on Web Applications and Compound Documents 直後の 7 月に発表された Microsoft の 2003 年度決算では、全社売上高が 368 億ドルに対し Excel、Word などを含むオフィススイートパッケージである Microsoft Office を事業領域に含む Information Worker 部門の売上高は 108 億ドル、全社の営業利益額が 90 億ドルに対し Information Worker 部門の営業利益額は 72 億ドルである³。これは Microsoft にとって Microsoft Office を中心としたネイティブアプリケーションが重要な収益源であることを意味する。

W3C において HTML5 の開発が始まる直前の 2006 年 6 月に、Google は Google Spreadsheet という表計算を、10 月には Google Docs & Spreadsheet と名称変更し、ワードプロセッサをウェブアプリケーションとして供給し始めている。これらは Microsoft Excel と Word の競合サービスである。HTML5 ならびに関連仕様の

³ Microsoft Corporation. (2004). *2004 annual report*. Retrieved December 14, 2013, from http://www.microsoft.com/investor/reports/ar04/downloads/MSAR_10K_091404.doc

機能が向上、拡大すれば、Google Docs & Spreadsheet⁴の機能も向上する可能性が高い。それはそのまま Microsoft Office の利用機会を減ずることとなる。

ネイティブアプリケーションとウェブアプリケーションはアーキテクチャ間での市場競争、つまり Architectural competition (Morris & Ferguson, 1993)を展開している状況である。

サービスへのアクセス無償で提供し、クラウドストレージの提供や広告などを通じて収益を上げるビジネスモデルの Google と異なり、OS とネイティブアプリケーションの販売収入が重要な収益源である Microsoft にとってウェブアプリケーションの普及はビジネスモデルの毀損につながる。しかし Microsoft は 2010 年 7 月に、オフィススイートのウェブアプリケーションである Office Online サービスを開始している⁵。

ウェブアプリケーションは提供可能な機能を向上させ続けており、徐々にネイティブアプリケーションに近づいている。Microsoft が提供するネイティブアプリケーションの Excel とウェブアプリケーションの Excel Online を比較すると見た目のデザインや多くの機能は既に同一水準に近づいている(図 5)。

サービスが提供されていても、各社のブラウザが HTML5 を実装しなければウェブアプリケーションは機能しない。しかし 2014 年 8 月時点で未だ中身が確定していないものの、Microsoft Internet Explorer (IE)、Google Chrome、Mozilla Firefox、Apple Safari、Opera という主要ブラウザの最新バージョン全てが既に実装している。これにより Google Drive は、Microsoft IE を含む全ての主要ブラウザ上で動作するという状況になっている。そして Microsoft Office Online も Google Drive 同様に Google Chrome や Mozilla Firefox など IE に競合する製品上でも動作するのである。

Google は、オフィススイートをウェブアプリケーションという形態で供給することにより、新たな収益源を創出した。そして Google は HTML5 仕様書の Editor を排出するなど、W3C において HTML5 開発の中心的な役割を果たしている。

⁴ Google Docs & Spreadsheet は、2012 年 4 月に Google がオンラインストレージサービス Google Drive を始める際に統合されており、以降 Google Drive で作成・編集するワードプロセッサファイルは Google Drive Document、表計算は Google Drive Spreadsheet と呼称されている。

⁵ Finley, K. (2010, June 8). Microsoft Rolls Out Office Web Apps. *ReadWrite Enterprise*. Retrieved May 18, 2014, from <http://readwrite.com/2010/06/08/microsoft-rolls-out-office-web>

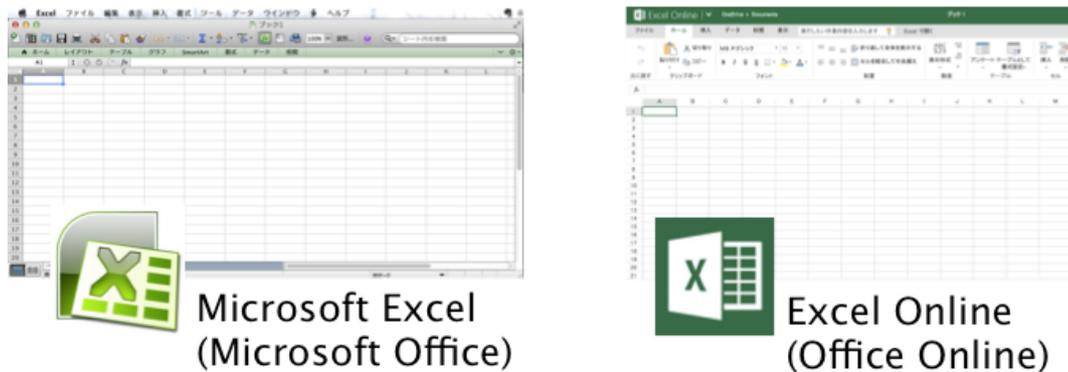


図 5. Microsoft の供給する表計算アプリケーション
(ネイティブアプリケーションとウェブアプリケーション)

一方 Microsoft にとり、ネイティブアプリケーションからウェブアプリケーションへドミナントデザインが移行することは、ネイティブアプリケーションによって多額の利益を得ている以上、好ましい状況ではない。

このように、既存のネイティブアプリケーションによって収益を挙げている企業は、ウェブアプリケーションの実現と普及に対し反対する動機が存在する。しかし、既存のコンピュータならびにスマートフォン OS 市場において寡占的な地位を有する Microsoft、Apple はともに、HTML5 の普及促進のためのコンテンツを、ウェブを通じて提供し⁶、Microsoft も W3C における HTML5 策定作業に貢献しているし、自社ブラウザ IE にも実装している。つまり標準仕様のバージョンアップを通じたイノベーションの実現に貢献し、ドミナントデザインの移行に貢献したのである(図 6)。

⁶ 例としては、Microsoft. (n. d.) *Internet Explorer 9 Test Drive*. Retrieved December 12, 2013, from <http://ie.microsoft.com/testdrive/>、Apple. (n. d.) *HTML5 Showcase - 360°*. Retrieved December 12, 2013, from <http://www.apple.com/html5/showcase/threesixty/> 等

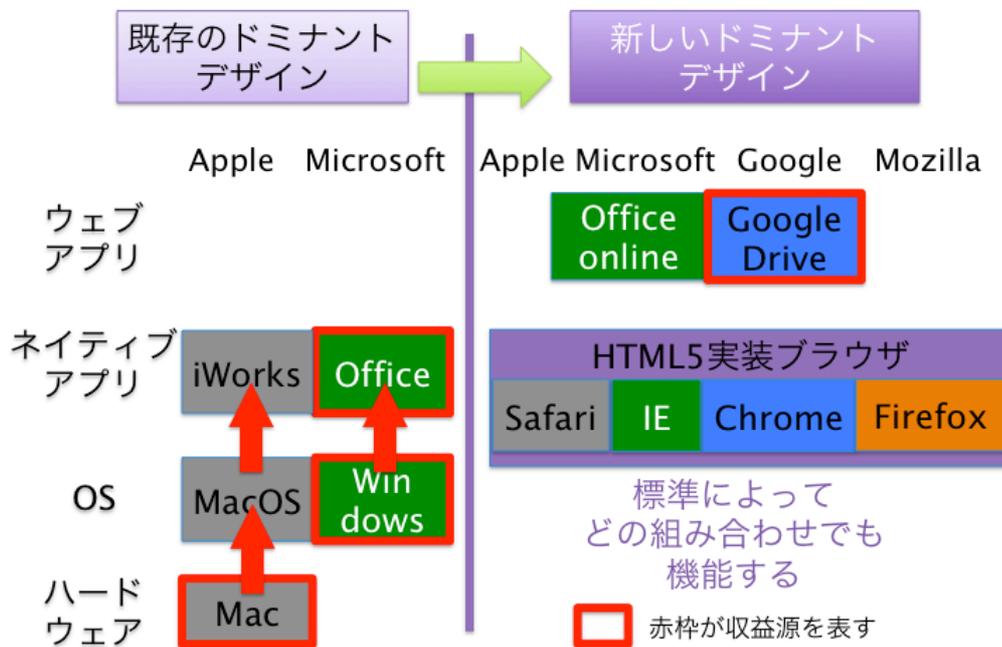


図 6. 利害関係が対立する主体間の協働によるドミナントデザインの移行

1.4. 標準を巡る競争と協働の歴史

HTML5 の策定においては、利害が対立する Google、Microsoft を始めとするステークホルダーによる協働が成立した。一方、情報技術分野における標準仕様を巡っての市場競争事例は、枚挙にいとまがない。

家庭用ビデオカセットレコーダー仕様である VHS と Betamax、光ディスクメディア仕様である Blu-ray Disc と HD DVD など、小型コンピュータ OS である Microsoft Windows/ IBM OS2/ Apple Machintosh など、市場競争を通じて数多くの仕様が標準の地位を獲得し、敗者陣営は少なからず損失を被ってきた。2008 年 2 月 19 日、東芝は自社が主導して開発を進めてきた HD DVD 事業からの撤退を表明したが、これによる損失額は 1085 億円にも及んでいる⁷。

第 3 世代移動通信システム(IMT-2000)仕様の W-CDMA と CDMA2000 のように、公的機関が認証する標準であっても、複数仕様が市場競争するといった事例もある。公的機関が単一の仕様を策定し、ステークホルダーに対し強制的に準拠させる場合を除けば、標準仕様の確立は協働よりも競争によって成立することが多い。

しかし、Betamax のテープを VHS のデッキで再生できないように、市場競争

⁷ Toshiba Corporation. (2008, April 25). *Presentation for FY2007 Results: For Fiscal Year ended March, 2008*. Retrieved October 31, 2014, from <http://www.toshiba.co.jp/about/ir/en/pr/pr2007q4.htm>

を通じて標準仕様が確立されると、多くの場合競争関係にある仕様間で互換性が欠如することとなる。各主体が技術的差別化をはかろうとすれば、互換性が損なわれる事態は避けがたい。

1.5. 本論文のテーマと構成

W3C は 1994 年に設立されて以来、数多くの標準仕様を策定してきた。そもそも、HTML5 は Hyper Text Markup Language (HTML) というウェブサイト記述言語の 1 バージョンであり、過去のバージョンからのアップデート版である。

一方 W3C においては HTML の過去バージョン含め、普及が進まなかった仕様や、独自拡張仕様ができるなど収斂に失敗した仕様もある。1990 年代後半に生じた Microsoft と Netscape による激しい市場シェア争い、通称「ブラウザ戦争」においては、両社が技術的差別化のために W3C で策定した標準である HTML3.2/4 などから外れた独自の拡張仕様を実装したことにより、両社のブラウザ間で互換性が失われた。これにより、ウェブサイト制作者は、どちらのブラウザで閲覧しても問題がないようにするために多大なコストを負担することとなった。

W3C の標準仕様策定プロセスに直接関与するのは、W3C に所属する (Team staff と呼ばれる) 職員、ならびに会員企業・団体に所属する開発者である。ただし、W3C は政府・国際機関から独立した民間の任意団体⁸に過ぎず、参加者間の利害対立を仲裁し、策定された仕様の普及のために強制する権限は持ち得ていない。

にもかかわらず、Google や Microsoft といった利害が対立するステークホルダーによる協働が成立し、HTML5 の策定・普及が成功している。ウェブアプリケーションを生み出した協調型イノベーションは、なぜ成立しているのだろうか。

既に述べているように、W3C でこれまで策定された標準仕様の全てがこの要件を満たしているわけではない。なにより HTML3.2/4 という、HTML5 の先行バージョンですらブラウザ供給者による独自拡張仕様の発生を抑制できず、仕様の収斂に失敗している。

HTML5 の仕様策定プロセスが完了した 2014 年は、W3C 創設 20 周年にあたる。20 年の間、W3C は標準仕様そのものをバージョンアップさせるのみならず、仕様策定プロセスの運営方針・手法もブラッシュアップさせてきた。であれば、

⁸ W3C は法人格を持たず、会員組織や参加者個人は、会費納入や著作権・知的財産権などの契約をホストの研究機関との間で取り交わしている。

W3C における協働の成立ならびに標準仕様の収斂・普及に成功した仕様 (HTML5) と、失敗した過去仕様の策定プロセスを比較することにより、利害の対立する多様なステークホルダーによる協働の成立要因を抽出できるのではないだろうか。

本論文は W3C の標準化プロセス運営ポリシーの確立に伴い、HTML5 ならびに先行仕様の収斂・普及状況がどのように変化したかについて分析することにより、標準仕様の策定・バージョンアップを通じたドミナントデザイン移行、すなわち協調型イノベーションを実現するための、成功要因について仮説導出を行う。

本論文の構成は以下の通りである。第 2 章で先行研究を整理し、それを踏まえて第 3 章で調査課題を導出するとともに、調査手法ならびに活用データを提示する。第 4 章から第 7 章までかけて W3C における標準仕様プロセスの確立状況と、策定された仕様の実効性獲得状況を分析し、第 8 章で仮説を提示する。更に第 9 章では、HTML5 の仕様策定プロセスにおいて見られた機能の追加を伴う競合仕様提案によってもたらされた組織の壁を超えた協働について分析し、イノベーションの源泉を標準仕様開発により効率的に取り込み、より多様なステークホルダーの間で協働を成立させるための要因に関する仮説を抽出している。最後に第 10 章にて、仮説の適用範囲について検討するとともに、今後の課題を導出している。

第2章. 先行研究レビューを通じた制約の特定

本章では標準仕様の策定・バージョンアップを通じた協調型イノベーション成立に必要な要件を導き出すため、標準化理論、標準化に關与するステークホルダーのインセンティブに關する理論に關する先行研究をレビューする。このレビューを通じて、本論文で明らかにすべき調査課題を明確にしていきたい。

2.1. ウェブアプリケーションを成立させる標準の特徴と制約

2.1.1. 互換性標準

一口に標準と言っても、その機能は様々であり、機能によって標準化プロセスやステークホルダーのインセンティブ構造は異なる。機能による分類として Kindleberger (1983) や Grindley (1995, p. 21)は、取引費用(North, 1990; Williamson, 1975)を削減するために最低限達成しなくてはならない品質を定めた品質標準 (Quality standard)と、製品間の相互可用性を通じて規模の経済を実現する (To achieve economies of scale through product interchangeability) 互換性標準 (Compatibility standard)という2種類を提示している。David (1987)は、これら2つに加えて参照標準 (Reference and measurement standard)を提示している。

HTML5 はウェブブラウザ自身の機能、ならびにサーバなど他のモジュールとの連携する機能を規定する仕様であり、分類としては互換性標準にあたる。

2.1.2. ネットワーク外部性

2.1.2.1. 直接的効果と間接的効果、過剰慣性

ウェブアプリケーションが実現した機能の1つに、複数人による1ファイルの同時編集がある。ただし自分だけが保有するコンピュータにHTML5が機能するブラウザをインストールするだけでは、複数人数によるファイルの同時編集という機能を使うことはできない。この機能を活用する場合、1つのファイルで共同作業する両者がともに、HTML5が動作するブラウザを用いる必要がある(図7)。

HTML5が動作するブラウザを利用するユーザが増えるほど、ウェブアプリケーションによる共同作業という機能を活用する機会が増える。このように、その財の利用者が増加すればするほど、その財のもつ価値が増大するという性質をネットワーク外部性 (Katz & Shapiro, 1985)という。ネットワーク外部性は互換性標準のもつ特徴の1つである。

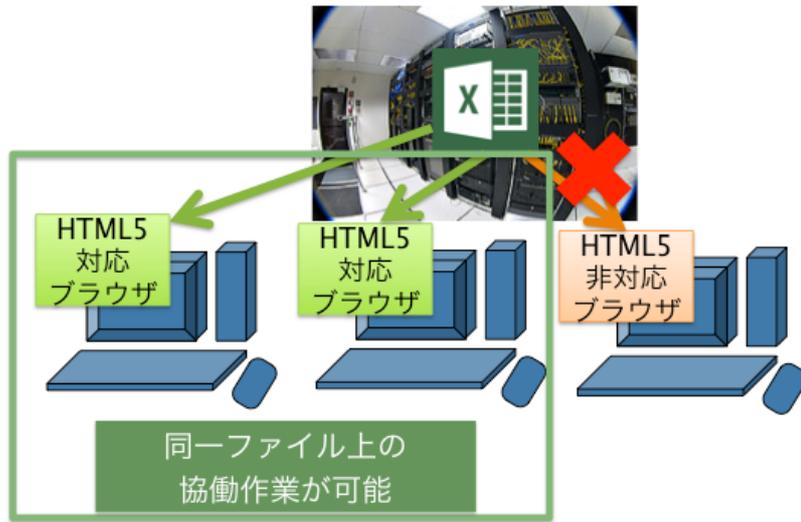


図 7.ウェブアプリケーションでの協働作業は HTML5 対応ブラウザがインストールされたデバイス間のみで可能

ネットワーク外部性は直接的効果と間接的効果に分けられる。直接的効果とは、電話のようにそのネットワークに接続される他者の存在があつて初めて価値が生まれ、接続者が増えれば増えるほど価値が増大するという性質を指す (Katz & Shapiro, 1986)。間接的効果は、ネットワークに接続するユーザの増加は接続に必要な財（電話の場合は電話機）の生産量の増大を招く。その結果、財を安価で提供することが可能となって普及が促進され、結果的に外部性が高まる現象を指す (Farrell & Saloner, 1985)。

ハードウェアと補完財であるソフトウェアが異なる主体によって供給されるという水平分業アーキテクチャの場合、多様なソフトウェアの供給を受けるハードウェアが市場競争を優位に戦い、標準的な地位を占める傾向にある (Church & Gandal, 1992)。

新たに登場した仕様は機能面で現行仕様を上回ろうとも、既に多くの利用者を集めた仕様から顧客を奪うことは難しい。この現象は過剰慣性 (Excess inertia) と呼ばれる (Farrell & Saloner, 1985)。一方マーケティング施策として、機能面で現行の標準からそれほど改善されていないのにもかかわらず、次世代に標準が移行することは過剰転移 (Excess momentum) と呼ばれる (Farrell & Saloner, 1986; Katz & Shapiro, 1992) と呼ばれる。

2.1.2.2. スイッチングコストとロックイン

Windows PC を所有し Microsoft PowerPoint を使っているユーザが、Apple 社のプレゼンテーションソフトである Keynote を使うためには、MacOS を搭載した Apple Machintosh コンピュータを購入しなければならない⁹。

更に既に作成・保存している PowerPoint ファイルを MacOS 上の Keynote で利用する場合、文字や図形のズレや、アニメーションなど一部機能が動作しないためにファイルを修正する必要がある場合がある。また、操作手順を一から習熟しなければならない。その上、Keynote で作成したファイルは、PowerPoint で開くことができないため、Keynote 非保有者に対してファイルを送付する際は PowerPoint など別のファイル形式に変換しなければならない。

このように、ネットワーク外部性のある財を移行する際には様々なコストが発生する。このようなコストはスイッチングコストと呼ばれる (Klemperer, 1987)。

周囲に PowerPoint 利用者が多いのであれば、スイッチングコストを負担してファイルをそのままやりとりできるユーザ数が少ない Keynote に移行するメリットは小さいため、同一の財を使い続けざるを得なくなる傾向がある。つまり、ロックインと呼ばれる現象が生じる傾向にある (Farrell & Shapiro, 1985; Katz & Shapiro, 1986; Arthur, 1989)。そのため寡占的な地位を獲得したプラットフォームに、技術的差別化で対抗することは非常に困難となる。

初めて当該の財を利用する場合にも、初期購入費用、操作手順の習熟といったセットアップコストが発生する。セットアップコストの大きさや、推測可能かどうかによって競争構造は左右される。

市場形成当初の段階においては、セットアップコストが推測できる場合はその段階で最も優れた製品・サービスが大きなシェアを獲得する傾向にある (Farrell & Shapiro, 1989)。

2.1.3. レイヤード・モデューラー・アーキテクチャにおけるプラットフォーム

Microsoft Word は、Windows OS 上にインストールしなければ機能しない。逆に Windows OS は、アプリケーションをインストールしない場合、提供できる機能は限定される。

Windows に対する Word のように、ある財が便益をもたらすために必要とす

⁹ Keynote は MacOS に無償でバンドルされているため、ハードウェアを購入する以上の金銭的支出は生じない。

る別の財は補完財(complement)と呼ばれる。一主体が組み合わせて用いられる複数のモジュールを供給することもあれば、複数の主体が組み合わせて用いるモジュールを供給する場合もある。多様な機能をもつモジュールの組み合わせパターンはアーキテクチャと呼ばれる (Ulrich, 1995)。

Windows には、Word の他に Excel や PowerPoint、他社製品である Adobe Illustrator や Mozilla Firefox といった様々なアプリケーションをインストールすることが可能である。これらアプリケーションは、ウィンドウ・インターフェイスやファイル保存、コピー&ペーストなどのデータ処理などの機能を OS に依存する。

このように、複数の補完財によって共通して利用される機能を提供するコンポーネントはプラットフォームと呼ばれる (Utterback & Meyer, 1993; Gawer & Cusumano, 2002; Meyer & Lehnerd, 2011)。

Microsoft Word の文字列データは、Adobe Illustrator にコピー&ペーストすることが可能である。つまり、Microsoft Windows は複数のコンポーネント間でデータの引き渡しを行っている。この機能もプラットフォームの要素の 1 つである。Rochet and Tirole (2003) は、プラットフォームを 2 者、もしくはそれ以上の主体間の取引を仲介する要素、もしくはサービスとして定義している。

ウェブアプリケーションは、各社が供給するブラウザで共通して動作する HTML5、つまり W3C で策定された標準仕様が実現する機能を活用して設計されている。HTML5 は、Google や Microsoft を始めとする様々な企業が共通して用いるコンポーネントであり、ウェブアプリケーションの多様性や発展性を促進する役割を担っていることから、HTML5 は典型的なプラットフォームであるといえよう。

プラットフォームは、他のコンポーネント間の関係をコントロールすることにより、システムにおける多様性や発展性を促進する安定的なコンポーネント (Baldwin and Woodard, 2009; pp.19) と定義されている。プラットフォームの設計はシステム全体の構造に大きく影響する。そのため設計を規定する仕様策定では、ステークホルダー間の利害対立が生じやすいのである。

HTML5 によって設計が規定されるウェブブラウザは、OS 上で動作するネイティブアプリケーションという側面をもつ。これは、ウェブアプリケーションはウェブブラウザに、ウェブブラウザは OS に、と依存関係が積み重なっていることを意味する(図 8)。このように、依存関係にあるモジュールが積み重なるア

アーキテクチャは、レイヤード・モデューラー・アーキテクチャ (Yoo et al., 2010) と呼ばれる。また、それぞれのコンポーネントはレイヤと呼ばれる。

多くのウェブブラウザは無償で供給されているが、ブラウザ供給者はそれぞれ別のレイヤで利益を創出している。Internet Explorer を供給する Microsoft はネイティブアプリケーションや Windows OS の販売によって、Safari を供給する Apple が Macbook や iPhone の販売によって、それぞれ利益を得ている。異なる収益構造をもつステークホルダーにとって、自社のビジネスに有利となるアーキテクチャは異なる。HTML5 ならびにウェブ標準仕様策定における利害関係の調整は、ブラウザという単一レイヤにおける市場構造のみならず、レイヤード・モデューラー・アーキテクチャを構成する複数レイヤにおける市場構造が影響するのである。

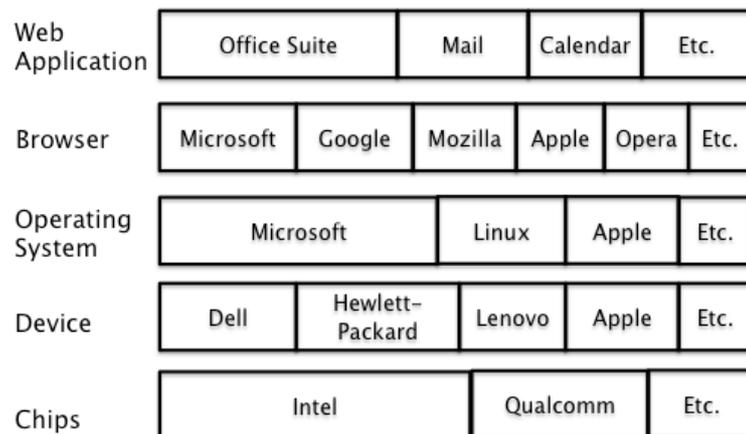


図 8. ウェブアプリケーションのレイヤード・モデューラー・アーキテクチャ

2.1.4. コーディネーションとイノベーション遅滞のトレードオフ

互換性標準は、モジュール化を成立させてより複雑かつ高機能な効用を実現するアーキテクチャを実現し、多様な準拠コンポーネントの供給を促すといった効用がある一方で、過剰慣性やロックインといった現象の原因にもなる。そのため、Baldwin and von Hippel (2011)は標準の存在はイノベーションの機会を阻害する傾向にあると指摘している。

また、標準化によるイノベーションの遅滞とコーディネーションの可能性とはトレードオフの関係にある (Simcoe, 2007; Farrell & Simcoe, 2012)。過剰慣性による技術移行の遅れの例としては、FM ラジオ放送方式の普及遅延がある (Besen, 1992)。また、標準の存在によって供給される財の多様性が小さくなる場合もあることも指摘されている (Farrell & Saloner, 1986)。

HTML5 の場合、レイヤード・モデューラー・アーキテクチャにおける 1 レイヤの標準仕様策定であるため、他のレイヤでビジネスを展開する企業も仕様策定のステークホルダーに含まれる。そのため、コーディネーションに必要なコストは大きくなってしまっている。

2.1.5. 互換性標準のもつ制約をいかにして乗り越えるか

互換性標準はネットワーク外部性をもつプラットフォームである。そのため、イノベーションを実現させることのできる技術を導入しようとしても、既存仕様に対する過剰慣性が制約となる。既存仕様に対するロックインを解消するためにスイッチングコストを低減する必要がある。

既存の技術仕様が存在しない機能については、プロプライエタリな仕様をもとに Sponsored 標準の地位を確立しようとするプレイヤーの First-mover advantage 戦略を抑制しなければならない。なぜなら、過剰慣性の影響により、一度定着したプロプライエタリ仕様を置き換えるのは困難だからである。そのため、ニーズを踏まえた仕様案をいち早く策定し、コーディネーションを迅速に行なって仕様を確立させ、実装事例を普及させなければならない。

ドミナントデザインの移行は、新旧デザイン間の市場競争を通じて行われることが多い。協調型イノベーション成立のためには、いかに旧デザインで競争力を有するプレイヤーを協働に参加させるか、新デザインをベースにしたビジネスモデルを企図するプレイヤー間の協働をいかにして成立させるかが重要である。

2.2. 標準化プロセスに関与するステークホルダーとそれぞれのインセンティブ

標準仕様には特定企業、もしくは Blu-ray Disc フォーマットにおけるブルーレイディスクアソシエーションのような企業連合によって開発・普及が主導されているプロプライエタリな技術仕様 (Sponsored 標準)と、特定企業/企業連合が知的財産を所有しない、非プロプライエタリな技術仕様 (Unsponsored 標準)に区分される (David and Greenstein, 1990)。HTML5 のような Unsponsored 標準仕様に対して競合仕様が登場したり、独自の拡張が開発されたりする背景には、プロプライエタリな技術を Sponsored 標準にしようとする主体との調整に失敗するからである。ステークホルダーが自発的 (Voluntary) に調整を受入れた結果としての標準 (自発的標準; Link, 1983) を成立させるためには、各ステークホルダーのもつインセンティブを理解する必要がある。ここからは、各ステークホルダーに対してどのようなインセンティブが働き、どのような戦略をとる傾向にあるのかについて整理していく。

2.2.1. プラットフォーム・リーダーシップ確立を通じた利潤獲得機会の創出

前述のとおり HTML5 という標準仕様はウェブアプリケーションの動作環境としての性格を有する、典型的なプラットフォームとしての側面をもつ標準仕様である。HTML5 に競合する技術や仕様もまた、プラットフォームとしての性格を有する。よって、競合する技術・仕様を提唱、普及させようとする主体は、HTML5 に対してプラットフォーム間競争を仕掛けてきているといえる。

プラットフォーム間競争を勝ち抜く、もしくはプラットフォームを構築する上でレイヤード・モデューラー・アーキテクチャの中で影響力を高める施策はプラットフォーム・リーダーシップ (Gawer & Cusumano, 2002) を確立する作業である。

Gawer and Cusumano は(2008)において、これまでに存在しない新たなプラットフォームを構築する Coring 戦略と、市場の方向性に影響を与えることを通じてプラットフォーム間競争に打ち勝つ Tipping 戦略という2つの戦略が存在するとしているとしている。

プラットフォーム間競争を戦う場合、競合プラットフォームから顧客を奪うには、スイッチングコストを負担するインセンティブとなるだけの機能的差別化が必要となる (Lieberman & Montgomery, 1998; Bresnahan, 1999; Shapiro & Varian, 1999; Evans & Schmalensee, 2002)。

プラットフォームを自ら供給したり、プラットフォーム・リーダーシップを通じた利潤獲得機会を創出しようとする主体は、機能的差別化を実現することのできる独自仕様を提唱・実装したり、新たな機能や便益を備えたプラットフォームを別途提唱しようとするのである。標準仕様策定を通じた協調型イノベーションを実現するためには、積極的に新たなニーズを取り込み、プロプライエタリな独自拡張や競合仕様の出現を抑制する必要がある。

2.2.2. 時間をかけて協調するよりも、素早く独自仕様を普及させる

ネットワーク外部性が強く働く市場においては、大きなシェアを獲得する寡占的な企業と、ニッチな領域でシェアを確保する複数企業によって構成される Dominant-firm and fringe の状態になる傾向にある (Bresnahan, 1999)。そのため、プロプライエタリなプラットフォームを抱えて市場参入する主体は、自らが標準の地位を獲得するための施策を講じる必要がある (Shapiro & Varian, 1999)。

ネットワーク外部性を活かす戦略としては、いち早く市場参入してシェアを確保する First-mover advantage 戦略 (Lieberman & Montgomery, 1988) が有効である。また、競合に対する参入障壁を設け、規模の経済性を活かさなければならない (Bresnahan, 1999)。競合が類似のアーキテクチャを採用して対抗することを防ぐための知的財産権の保護は重要な課題である (Teece, 1986)。

しかし、他社が先に参入している市場であっても First-mover advantage を覆すことが可能である。家庭用ビデオカセットレコーダー市場においては、ソニーが Betamax という規格で先行して参入していたが、松下電器と日本ビクターは録画可能時間の長さといった機能面での差別化に加え、準拠製品の供給者を増やすことによりネットワーク外部性の間接的効果を創出し、コスト競争力につながることにより VHS 規格によって市場競争を勝ち抜いた (Cusumano et al., 1992)。ビデオカセットレコーダーのように、技術の成熟に長期間かかる領域では、後発が先行者を追い抜いて市場競争に勝利することも可能となる (Rosenbloom & Cusumano, 1987)。

ネットワーク外部性がある市場では、利用者の増加が期待できる仕様がより選好されるという傾向がある (Krugman, 1991)。そのため、ネットワーク外部性のある市場において、利用者の増加につながるような施策に関するアナウンスを事前に行うことにより、シェアの獲得を目指すという戦略が採用される場合がある。

標準のバージョンアップを通じたイノベーションを実現させるためには、プ

ロプライエタリな仕様の出現を抑制するためには、長期化するコーディネーションをいかに短くしつつ、将来の普及可能性をいち早く確立し、訴求することが必要となる。

2.2.3. 紐付け戦略による支配

ウェブアプリケーションを、協働を通じた標準仕様のバージョンアップを通じたイノベーションとして実現するためには、オープンな水平分業によるメリットを活かした形で標準化プロセスを進めなければならない。

一方レイヤード・モデューラー・アーキテクチャにおいて、特定レイヤのプラットフォームの競争力を高める、もしくは競合に対する参入障壁とするために隣接レイヤの補完財を自ら開発・供給するという紐付け (Tying) 戦略 (Whinston, 1989; Nalebuff, 2004) が採用されることがある。また、戦略的に紐付けを進めることでプラットフォームの競争力を高めることにより、競合の参入意欲を阻害し (Choi & Stefanadis, 2001; Choi, 2004)、独占的な立場を築くことも可能となる (Carlton & Waldman, 2002)。この戦略を推し進めると隣接レイヤの買収となる (Gao & Iyer, 2006)。

HTML5 の策定プロセスで中心的な役割を担ったのは、OS を供給する Microsoft や Apple、ウェブアプリケーションを供給する Google など、隣接するレイヤにおいてビジネスを展開する主体も多い。彼らが紐付け戦略をとることは、標準仕様策定における利害対立の先鋭化につながる。仕様策定におけるコーディネーションの対象は、隣接レイヤを含む幅広い分野のプレイヤーを念頭に置く必要があるといえよう。

2.2.4. 水平分業を通じた価値創出コストの分担とイノベーションの源泉拡大

プラットフォームは、紐付けとは逆に第三者に補完財モジュールの供給を認め、彼ら/彼女らによる自発的な機能追加を通じたプラットフォームの価値向上を目指すという、水平分業戦略をとることもできる。これにより価値向上のために必要な節約することができ、補完財供給者は各コンポーネントが共通して用いる機能を開発するコストを節約することができる (Langlois & Robertson, 1992)。

水平分業を選択した場合、プラットフォームはエンドユーザに対し、異なるタイプの顧客と相互作用を通じて価値を提供する必要性が生じる。Windows OS が Mac OS に対して競争力をもつためには、有力なネイティブアプリケーション

が必要であるし、Sony Playstation が任天堂 Wii から市場占有率を奪うためには良質なコンテンツを開発できる企業からタイトル供給の供給を受ける必要がある。このように、エンドユーザに加え他の種類の顧客が同時に存在する市場は多面市場(multi sided market) (Rochet and Tirole, 2003; Parker & Van Alstyne, 2005; Eisenmann Et. Al., 2006) と呼ばれる。HTML5 という仕様の場合、ウェブアプリケーション/ ウェブコンテンツ開発者とエンドユーザという 2 つの市場が存在する。

プラットフォームの競争力向上ならびに多様な補完財供給が成立するための枠組みを確立するにはビジネスエコシステム (Iansiti & Levien, 2004) の考え方が重要である。ウェブアプリケーションの場合はモジュラーがソフトウェア・コンポーネントで構成されるため、ソフトウェアエコシステム (Jansen et al., 2012; Jansen & Cusumano, 2013) の確立が必要である。ソフトウェアエコシステムの代表的な事例としては Google による Android OS (Anvaari & Jansen, 2013) などが存在する。レイヤード・モジュラー・アーキテクチャにおいては、レイヤ毎に協働関係を組み替える Layerd collaboration scheme (Bogers, 2011) が成立する場合がある。

補完財供給者の開発やイノベーション創出を支援するために、開発のためのツールキットを提供する (von Hippel & Katz, 2002) などの施策が講じられることもある。他にも複数主体間の協働を成立させるためのコーディネーションコストや、多数存在する補完財供給者のイノベーションを見つけ出すなどのコストが協働には付随する (Almirall & Casadesus-Masanell, 2010)。

二面市場における競争戦略の 1 つにプラットフォーム包囲(Platform envelopment) がある。プラットフォーム包囲とはレイヤード・モジュラー・アーキテクチャにおいて、他のレイヤが提供する機能を取り込むことにより利益を増やす戦略である (Eisenmann et al., 2011)。具体的には、Microsoft が Windows OS にマルチメディア再生ソフトである Windows Media Player をバンドルすることによって RealNetworks が供給する RealPlayer を市場から放逐した事例が該当する。

紐付け戦略、プラットフォーム包囲戦略は一社で行われるとは限らない。異なる主体が供給するモジュールの組み合わせによって競争優位を創出するのが “Divided technical leadership” (Bresnahan & Greenstein, 1999) 戦略である。代表的な例として、Microsoft Windows OS と Intel X86 アーキテクチャ CPU の組み合わせ

せによるパーソナルコンピュータ市場戦略を挙げることができる。

この組み合わせは広く普及し、両社は Wintel と呼ばれるほど利害を共有する強力なパートナーシップを組んでいると評されることが多い。しかし両社の利害は完全に一致している(いた)わけではない。Intel は顧客を高性能の次世代製品に次々と移行させたいのに対し、Microsoft は OS のインストールベース、つまり OS の市場占有率を維持し続けることが目的であるため、急激な技術進化と移行の機会が登場することは歓迎されない (Casadesus-Masanell & Yoffie, 2007)。“Divided technical leadership” を成立させるためには、パートナーシップを組む者同士で利害調整がなされる必要がある。

レイヤード・モデューラー・アーキテクチャにおいて互換性標準としての実効性を確立するには、ウェブアプリケーション/ウェブコンテンツ開発者とエンドユーザという 2 つの市場に受容され、かつプラットフォーム包囲を受けないような施策を講じる必要がある。そして、仕様策定プロセスに直接参加するステークホルダーのみならず、(将来の)ウェブアプリケーション制作者、エンドユーザに対しても、利害調整を講じる必要がある。更に、(将来の)ウェブアプリケーション制作者やエンドユーザをイノベーションの源泉としようとするならば、彼ら/彼女らたちとの協働を成立させるための施策をある程度コストを払っても講じなければならない。

もっとも、プラットフォーム自体の価値が広く認知されていない状況、または補完財が自然と供給されない状況においては、プラットフォーム自身も補完財を供給するよりほかない。Intel は X86 アーキテクチャ CPU を市場投入する際に自ら補完財を開発し、プラットフォームと補完財市場の価値を認知させるとともに、補完財供給者の参入を促す戦略を実行している (Burgelman & Andrew, 2001)。ウェブアプリケーションという新たな形態のサービスは、当初はユーザにその価値を認識されていない。ウェブアプリケーションを実現するための機能を提供する標準仕様策定においてはユーザ、つまりウェブコンテンツ制作者や将来のウェブアプリケーション制作者、そしてエンドユーザに対し、その仕様が確立することによって提供される便益について訴求することが必要となるのである。

2.2.5. 貢献へのインセンティブとコーディネーションメカニズムの設計という課題

Microsoft の Windows OS、プラットフォーム・リーダーシップを確立し、多様な準拠ネイティブアプリケーションが供給されて独自のソフトウェア・エコシステムを確立すれば、大きな利益を得られる可能性がある。

特定の主体がプロプライエタリな仕様に固執したり、策定された新たな仕様を採用・実装しないといった状態が生じれば、標準の実効性は損なわれる。そのため、できるだけ多くのステークホルダーが、新たな仕様の開発・実装に参加するよう促す必要がある。政府・国際機関による強制によらない標準化の場合、各主体が自発的に参加するためのインセンティブ設計が非常に重要となる。

その一方で幅広いステークホルダーが標準化プロセスに参画した場合、コーディネーションコストが増大し、仕様の確定に時間がかかってしまう。仕様策定や普及の遅延は、プロプライエタリ仕様の出現を誘引することとなる。

協調型イノベーションを標準仕様のバージョンアップを通じて実現しようとするならば、できるだけ多数のステークホルダーを巻き込みながら、迅速にコーディネーションを進めるという両立が難しい施策をとらなければならないのである。

第3章. 調査課題と調査手法

3.1. 調査課題の導出

先行研究のレビューによって明らかになったことは、ウェブアプリケーションは「互換性標準のバージョンアップを通じて成立した」という点で特徴的なイノベーションであるということである。ウェブアプリケーションを実現したHTML5は、互換性標準、またレイヤード・モジュラー・アーキテクチャにおけるプラットフォームという特徴を有している。

本論文では、このような特徴をもつプラットフォームを構築するための「協調型イノベーションを成立させる要因」の仮説抽出を調査課題とする。

ただし、イノベーションの成立、つまり機能実現が技術的可能となるだけでなく、当該機能が標準仕様として認められ、かつ広く普及しなければならない。そのためには単にステークホルダー間の協働が成立するだけでは不十分である。

まず、必要な技術仕様が標準化提案されなければならない。しかし先行研究において指摘されているように、自社で開発した技術を標準化することは、イノベーションによって生み出される利益を独占することができなくなるというリスクが生じる。各社が開発したプロプライエタリ仕様間による市場競争を避け、ステークホルダーによる協働によって仕様が開発されるためには、技術保有者が標準化提案するという環境を整えなければならない。

仕様が標準化提案されたとしても、主要なステークホルダーが揃って仕様策定・普及活動に参加しなければ独自仕様の発生を抑制できない。そのため、仕様策定プロセスへの参加を促すインセンティブが設計されなければならない。その上で、仕様策定プロセス参加者の利害関係を調整し、プロセスからの離脱、独自仕様の開発・実装とその仕様への固執を抑制しなければならない。更に、ステークホルダーが揃って標準仕様を採用、実装しなければ、その仕様は実効性を有することはない。政府・国際機関による強制力が働かないウェブ標準の場合、ステークホルダーが自発的に標準仕様を採用するメカニズムを構築する必要がある。

まとめると、本論文の調査課題 (Research Question) は、

RQ: 標準仕様策定を通じた協調型イノベーションの成立要因

RQ-1: どのようにして革新性を形成したか

(イノベーションの源泉を取り込むメカニズムの確立)

RQ-2: どのようにして協調行動をとらせたか

(協働に対するインセンティブ設計)

RQ-3: どのようにして相互可用性が損なわれる状況を抑制したか

(標準仕様の収斂)

RQ-4: どのようにして仕様が普及したのか

(標準仕様の普及)

となる。

3.2. 標準化の成否を規定する測定基準

本論文が目的とするのは、協調型イノベーションの実現に対する必要要件である。特に、標準仕様のバージョンアップを通じたドミナントデザインの移行を成立させるための要因を抽出することを目的としている。ここからは、議論を厳格にするために、「標準仕様のバージョンアップを通じたドミナントデザインの移行」という状態を定義し、この状態を満たしているかどうかの判別基準を提示したい。

まず、バージョンアップした標準仕様がドミナントデザインとしての性格を有するということは、その仕様が普及し、かつ競合仕様の登場によって相互可用性が欠如するという状況を回避できている状態であると言い換えられる。

W3C は大学や研究機関が共同運営する任意団体であるため、策定された仕様の普及に対して行政による強制力が機能しない。そのため仕様が確定しても、提案者や賛同者以外のステークホルダーが採用、実装しないという状態が起こりうる。ドミナントデザインは、市場の主要プレイヤーのほとんどが採用している状況を指すため、単に仕様が確定するのみならず、市場の主要プレイヤーが揃って当該仕様を採用、実装するという状態となって初めて当該仕様は標準として実効性があると判別できる。

ウェブアプリケーションの登場というドミナントデザインの移行は、HTML4 から HTML5 へのバージョンアップに伴って生じている。過去バージョンの策定の成否を判別するための基準としては標準としての実効性、つまり、1) 市場の主要プレイヤーが揃って採用・実装している(仕様の普及)、2) 競合し、相互可

用性がない仕様が普及しない(仕様の収斂)という 2 点が成立しているとする。

3.3. 分析対象ならびに調査設計

本論文の調査課題は、なぜウェブアプリケーションというイノベーションを成立させる要件を含んだ HTML5 という標準仕様が策定され、実効性を有したのか、その要因に対する仮説を抽出というものである。

HTML5 は W3C において策定された標準ではあるが、全ての W3C 標準が必ずしも実効性を有し、イノベーションに寄与する機能を含んでいるわけではない。

W3C において策定された HTML5 の直接的な先行バージョンは、HTML3.2、HTML4 (4.0 ならびにマイナーバージョンアップ版の 4.01) である。1995 年から 1998 年にかけて Microsoft と Netscape の間でブラウザ戦争と呼ばれる激しい市場シェア争いが勃発し、両社は機能的差別化を図るために当時の標準化が進められていたバージョン(3.2、4.0 ならびに 4.01)の仕様書ならびに Draft に存在しない独自仕様を実装した。両方のブラウザで問題なく表示されるサイトを構築することは難しく、一部のウェブサイトには推奨ブラウザを示すバナー (図 9) が掲出される事態となった。HTML の過去バージョンにおいて、W3C の標準化プロセスは有効に機能しなかったこともあったのである。更には、W3C 内において HTML5 に競合する仕様 (XHTML1.1/2.0) が策定されていた期間がある。



出典：<http://www.jewelion.com/blog/validation-get-it-or-forget-it>

出典：<http://saipanpublicschools.tripod.com/>

図 9. 推奨ブラウザを示すバナー

HTML はその改定の歴史において、策定プロセス運用方針が変化してきた(図 10)。つまり各バージョンの仕様策定プロセスと、プロセス運営方針の時系列分析を行うことにより、HTML5 策定プロセスにおける成功要因の仮説抽出を行うことができる。

なお W3C においては、2013 年の 1 年間において 167 もの Technical Report (標準仕様、Draft、ならびに Note)が公開される¹⁰など、数多くの仕様策定されている。各仕様により、関与するステークホルダーと、各主体の関与の度合いは異なる。そのため、ウェブアプリケーションというドミナントデザインを成立させた HTML5、ならびにその先行バージョンのみを分析対象とし、バージョンごとの実効性を比較することとする。

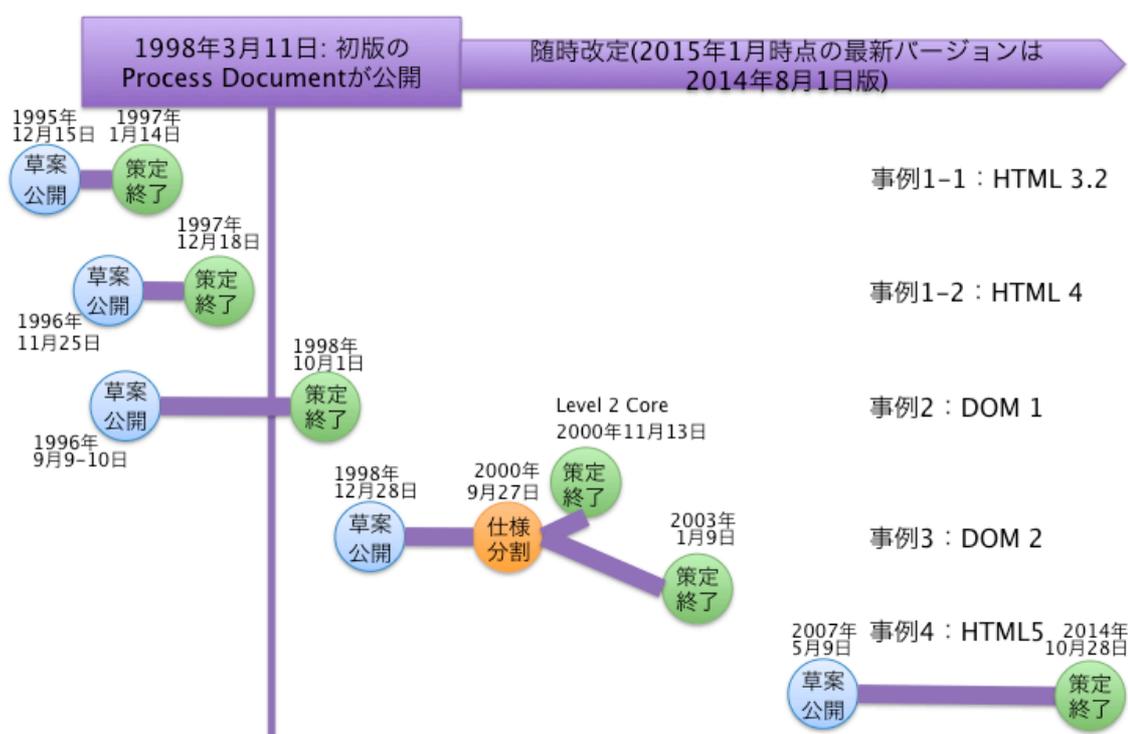


図 10. W3C で策定されている HTML ならびに HTML5 の先行仕様の策定期間と Process Document 制定・改定期間¹¹

¹⁰ World Wide Web Consortium. (n. d.) 2013, *All Documents (sorted by date)*. Retrieved July 16, 2014, from http://www.w3.org/TR/tr-date-all#tr_2013

All Documents (sorted by date) - W3C http://www.w3.org/TR/tr-date-all#tr_2013

¹¹ 2015 年 1 月時点での最新版 Process Document: Neville, C. M. (ed.). (2014, August 1). *World Wide Web Consortium Process Document: 1 August 2014*. Retrieved January 29, 2015, from <http://www.w3.org/2014/Process-20140801/>

ただし HTML5 は、HTML の最新バージョンであるというだけではなく、XHTML1 と DOM2 HTML の後継であると規定されており¹² (図 11)、これらも先行仕様に加えて分析するのが妥当であろう。

そのため本研究では W3C において策定されたバージョンの HTML3.2/ 4 (4.01)/ 5 と DOM1/ DOM2HTML、XHTML1.0 を分析対象事例とする。また W3C 内で HTML5 と競合の関係にあった XHTML 1.1/ 2.0 も分析対象として含めることとする。

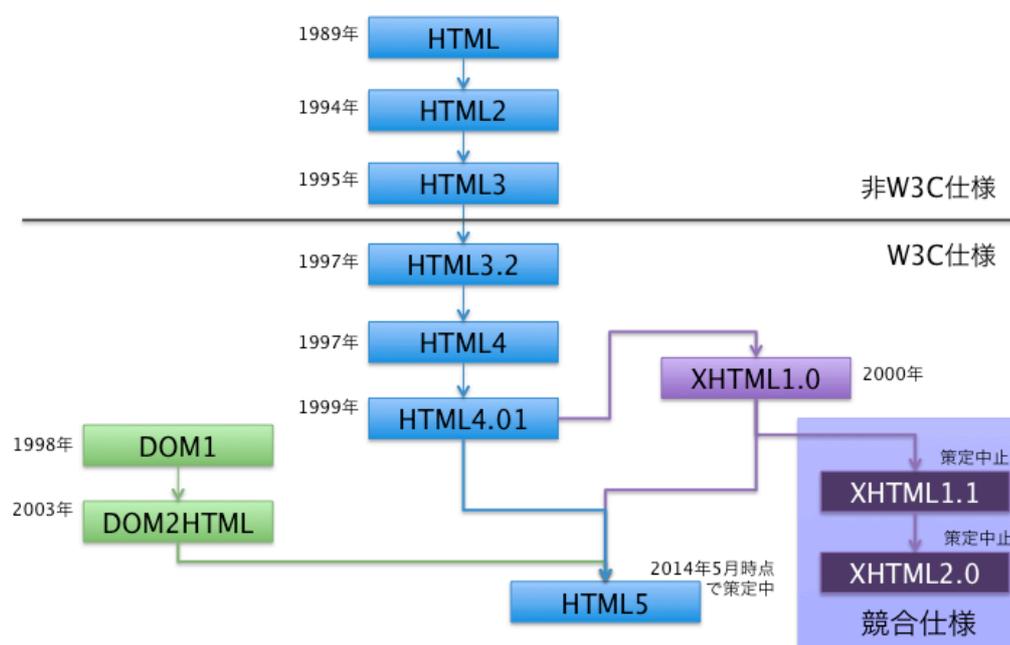


図 11. HTML5 の先行仕様・競合仕様

¹² Berjon, R. and el al. (eds.) (2014, April 29). 1.4 History. *HTML5 A vocabulary and associated APIs for HTML and XHTML: W3C Candidate Recommendation 29 April 2014*. Retrieved May 25, 2014, from <http://www.w3.org/TR/2014/CR-html5-20140429/introduction.html#history-0>

また HTML5 は、2012 年 12 月に HTML5 とそのマイナーバージョンアップ版の HTML5.1 に分割されており¹³、2014 年 10 月 28 日に HTML5 は仕様策定プロセスを完了している¹⁴ (HTML5.1 の仕様策定作業は継続している。2012 年 9 月に公開された W3C の計画¹⁵によると、HTML5.1 の仕様策定作業終了は 2016 年を予定している)。本論文では HTML5 に関しては、メジャーバージョンである「HTML5」の策定終了までを分析対象とし、2014 年 11 月以降の HTML5.1 策定プロセスについては分析対象としないこととする。

3.4. 調査手法と活用データ

本論文は、W3C において行われてきた HTML 各バージョンならびに競合仕様の仕様策定プロセスをケーススタディ (Yin, 2008) として分析する。各バージョンの策定プロセスを、運用ルール (適用されていた Process Document の内容、ならびに実際に採用されていた手法) や市場環境などと照らし合わせながら比較分析を行うことにより、標準としての実効性のある仕様策定が成功する要因の仮説を抽出したい。

3.4.1. W3C オフィスにおけるフィールドワーク

W3C は米国マサチューセッツ工科大学 (MIT)、欧州情報処理数学研究コンソーシアム (ERCIM) 、日本の慶應義塾大学 SFC 研究所、中国の北京航空航天大学 という 4 研究機関によって共同運営されている。

筆者はケーススタディを行うため、2010 年 4 月から 2013 年 3 月まで、W3C の日本拠点である慶應義塾大学 SFC 研究所に設置されている W3C office (W3C/Keio) にインターンとして所属した (図 12)。また、2012 年 7 月から 9 月まで Visiting Student として MIT の W3C office (W3C/ MIT) に滞在している。インターン期間中、また MIT 滞在中は W3C 職員間、W3C 会員企業・組織が参加する会議へ参加している。

¹³ World Wide Web Consortium. (2012, December 17). *HTML5 Definition Complete, W3C Moves to Interoperability Testing and Performance*. Retrieved October 29, 2014, from <http://www.w3.org/2012/12/html5-cr.html>

¹⁴ World Wide Web Consortium. (2014, October 28). *Open Web Platform Milestone Achieved with HTML5 Recommendation*. Retrieved October 29, 2014, from <http://www.w3.org/2014/10/html5-rec.html>

¹⁵ World Wide Web Consortium. (2012, September 17). *Plan 2014*. Retrieved October 29, 2014, from <http://dev.w3.org/html5/decision-policy/html5-2014-plan.html>



図 12. W3C/Keio オフィスがおかれていた
慶應義塾大学湘南藤沢キャンパススタウ館¹⁶

W3C/Keio においては、広報担当として W3C の方針伝達や、HTML5 を始めとする W3C 標準仕様の普及活動に従事した。また本論文において、7.6.2 項で分析している日本の HTML5 開発者コミュニティ html5j.org に対する窓口を務めるとともに、その活動にも参加している。

ただし HTML の策定に深く関与するステークホルダーの多くは北米や欧州、特に米国在住し、米国に所在する企業や団体に在籍している。北米に拠点を置く企業も所在地は西海岸であることが多く、W3C office という物理的空間に所在することで得られるデータは少ない。そのため、本研究の分析はその多くをメーリングリスト・アーカイブに負っている。

しかし、W3C/Keio でインターンとして所属し、W3C/MIT に滞在することにより得られた背景情報は、メーリングリスト・アーカイブ分析作業の大きな助けとなっている。

¹⁶ 2014 年 4 月より、キャンパス内の別校舎に移転

3.4.2. メーリングリスト・アーカイブ分析

3.4.2.1. 分析対象リストの選定

W3C は議論を主にフェイス・トゥ・フェイス (F2F) 会合、メーリングリスト、Internet Relay Chat (IRC) を通じて行う(図 13)。F2F 会合の議事録・速記録は自動的にウェブページとしてアーカイブされている。本論文では重要な方針決定や議決がされる F2F 会合の議事録とメーリングリスト・アーカイブを分析する。

分析対象のメーリングリストは、分析対象仕様を策定する Working Group (WG)によって設けられたものに加え、W3C 全体の運営管理関係に関するもの、関連する API 仕様を策定する WG や、その他の作業部会が設けたものも含む。本研究では W3C 全体の運営方針に関する部会を併せて 21 の作業部会(表 2)、27 (WHATWG を含む)¹⁷のメーリングリストに投稿された 237,243 通 (表 3) を分析対象としている。



図 13. W3C における電話会議の様子。IRC でリアルタイムに速記録を作成する¹⁸。

¹⁷ W3C のメーリングリストが 27、WHATWG のメーリングリストを加えて 27。

¹⁸一色正男. (2010, December 6). W3C (World Wide Web Consortium)へのご案内. Retrieved June 5, 2014, from <http://www.w3.org/2010/Talks/1206-echonet-mi/20101201EchonetForumpre.pdf>

表 2. 分析対象作業部会リスト

HTML5 ならびに先行仕様/競合仕様策定関係

HTML Working Group
XHTML2 Working Group
HTML Editing Task Force
HTML Coordination Group
DOM Working Group
DOM Interest Group
Forms Working Group
XForms Working Group
Forms Task Force
XML Coordination Group

関連 API 仕様策定関係

Web Application Formats Working Group
Web Applications Working Group
Device APIs Working Group
Web Intents Task Force
Web APIs Working Group
Device APIs and Policy Working Group

W3C 全体の運営管理関係

Advisory Committee
Technical Architecture Group
HTML Editorial Review Board
Process Editorial Review Board
Revising W3C Process Community Group

表 3. 本論文で分析したメーリングリスト

合計	237,243 (通)
W3C 全体の運営管理関係	
w3c-ac-members@w3.org	3,932
w3c-ac-forum@w3.org	8,754
tag@w3.org	9,791
public-new-work@w3.org	483
chairs@w3.org	8,717
public-w3process@w3.org	1,844
HTML (3.2/ 4/ 5), XHTML (1/ 1.1/ 2.0)	
html-erb@w3.org	2,553
www-html@w3.org	18,128
public-html@w3.org	44,850
w3c-html-wg@w3.org	14,359
w3c-html-cg@w3.org	7,352
public-hypertext-cg@w3.org	179
public-xhtml2@w3.org	1,852
w3c-xhtml-m12n-review@w3.org	142
Document Object Model	
www-dom@w3.org	8,539
w3c-dom-wg@w3.org	3,633
w3c-dom-ig@w3.org	9,836
XForms	
www-forms@w3.org	6,242
public-forms@w3.org	2,574
public-forms-tf@w3.org	107
デバイスサイド API 関係/ WebIntents	
member-appformats@w3.org	1,123
public-webapps@w3.org	26,768
public-device-apis@w3.org	6,997
public-webapi@w3.org	4,663
public-web-intents@w3.org	1,021
WHATWG	
whatwg@whatwg.org	42,308

3.4.2.2. アーカイブ分析手法

標準化団体のメーリングリストは、OSS 開発プロジェクトとは異なり参加者間で利害が対立する場面が多数出現する。標準化プロセスを進めるための手続きのための事務的連絡を目的とする投稿も多い。そのため、投稿数の度数分布、投稿者プロフィールや返信記録を基にしたネットワーク分析など、定量的手法を用いても、参加者の果たす役割やその背景をきめ細かく把握することが難しい。また策定プロセスがオープンになっている仕様のメーリングリストは、個人が Gmail 等のフリーメールアドレスで投稿する場合も多く、重要な意思決定に関わる投稿を定量的に抽出し、そこに至るまでの経緯をトレースするのは難しい。そのため本研究では、会合の議事録や仕様書の改定記録から、ターニングポイントとなる事象の日付と関係者を特定し、関係する投稿を一通ずつ読み解くという手法を用いている。

3.4.3. 非構造化インタビュー

本研究では、W3C 関係者ならびにウェブサイト開発者等に対するインタビュー調査を 9 件実施している(表 4)。

表 4. インタビュー実施リスト

対象者氏名	所属	実施日
横田昌彦	アビダルマ株式会社	2011 年 2 月 18 日
西村一彦	株式会社ボイスリサーチ	2011 年 4 月 11 日
村岡正和	HTML5west.jp	2011 年 9 月 4 日
小松健作	html5j.org (株式会社 NTT コミュニケーションズ)	2011 年 9 月 7 日
一条美和子	html5j.org (株式会社産経デジタル)	2011 年 9 月 8 日
白石俊平	html5j.org (株式会社オープンウェブプラットフォーム)	2011 年 12 月 14 日
Ralph Swick	World Wide Web Consortium Chief Operating Officer	2012 年 8 月 14 日
Philippe Le Hégarret	World Wide Web Consortium Interaction Domain Leader	2012 年 8 月 22 日
Ian Jacobs	World Wide Web Consortium Head of Communication	2012 年 9 月 12 日

3.4.4. その他資料、ならびにウェブ上のツール分析

メーリングリスト・アーカイブやインタビュー等のデータを補足するために、公式文書、W3C ならびに関係企業・団体のプレスリリース、メディアに掲載された記事などの分析を通じて、W3C や会員企業のとった行動と、その影響を分析している。

また本論文では、W3C が設けた公式の場以外でステークホルダー、特に個人レベルの情報発信や提案も分析対象とすることで、より精緻な仮説構築を目指している。

第4章. Phase 0: W3C 発足以前のウェブ標準を取り巻く状況

HTML5 の登場により、ウェブがアプリケーションの実行環境としての役割を有するようになった。しかし、もともとウェブはそのような機能を果たすものとして開発されたものではなく、初期の HTML が果たしていた機能はもっと単純なものであった。HTML はバージョンアップを通じて少しずつ機能を増やしていき、HTML5 において飛躍的にその役割を拡大させた。

本論文では W3C における HTML5 ならびに先行バージョンが、バージョンアップを通じて機能を増加させていく過程と、各バージョンの策定プロセスを分析する。

ただし HTML 自体は 1989 年 3 月に開発が開始されており¹⁹、1994 年の W3C 設立以前から開発、標準化が始まっている。そのため、W3C における仕様策定プロセス分析を行う前提として、本章では開発当初のウェブならびに HTML の機能や役割とともに、W3C にて HTML を始めとするウェブ標準が策定されるに至った経緯について整理する。

4.1. 初期のウェブ標準化に対する試み

4.1.1. ブラウザ開発者間の独自機能追加と仕様の分裂

1991 年 8 月 6 日、世界最初のウェブサイトである、<http://info.cern.ch/> が開設される。また Berners-Lee は、alt.hypertext ニュースグループに World Wide Web 開発プロジェクトに関するエグゼクティブ・サマリーを投稿²¹する。これにより、ウェブは CERN という一研究所内の所内インフラから、“World Wide” Web という名称通り世界規模の情報インフラへ発展する一歩を踏み出し始めたのである。

Berners-Lee は CERN において、自身とごく少数の同僚と一緒にウェブの基本技術、つまり Hyper Text Transfer Protocol (HTTP)、Hyper Text Markup Language

¹⁹ Jacobs, I. (2014, March 12). *The World Celebrates 25 Years of the Web*. Retrieved October 26, 2014, from <http://www.webat25.org/news/press-release-the-world-celebrates-25-years-of-the-web>

²⁰ The European Organization for Nuclear Research. (2009, March 13). *CERN celebrates 20th anniversary of World Wide Web*. Retrieved October 26, 2014, from <http://press.web.cern.ch/press-releases/2009/03/cern-celebrates-20th-anniversary-world-wide-web>

²¹ Berners-Lee, T. (1991, August 6). *WorldWideWeb: Summary*. Retrieved April 26, 2014, from alt.hypertext ニュースグループアーカイブ(Google グループ): <https://groups.google.com/forum/#!msg/alt.hypertext/eCTkkOoWTAY/bJGhZyooXzkJ>

(HTML)、Uniform Resource Identifier (URI)/ Uniform Resource Locator (URL) に加え、HTTP によって送受信される文書を作成・表示するためのソフト²²である“World Wide Web (ブラウザ)”を開発、提供していた。

多様なロケーションに点在し、異なるマシン環境をもつ研究者間の情報共有という目的を追求すると、ウェブはあらゆる OS が搭載された、様々なスペックのマシンにおいても同様に利用できることが理想となる。そのためには、様々な OS のマシンに対応したブラウザの開発が必要である。

しかし様々な OS に対応するためのウェブブラウザを開発するには多大な資源が必要である。仕様とブラウザのバージョンアップ、多様な OS に対応するブラウザの開発、更にはウェブの普及活動を同時に進めるには手元の資源では不十分であったため、外部のエンジニアによるブラウザ開発を奨励、支援するようになる (Berners-Lee and Fischetti, 1999)。この戦略は、ウェブの普及を迅速にした。

Berners-Lee はウェブを、マシンスペックに依存せずどのような環境でも同じように文書を閲覧・利用できるインフラとする方針であった。しかし、低スペックのマシンにあわせて仕様を定めることは機能追加に対する大きな足枷となった。そのため、独自仕様を実装することで、技術的差別化をはかるブラウザ開発者が登場することとなり、各ブラウザが実装する仕様の多様化を招くこととなった。

4.1.2. IETF における標準化

仕様の分裂状況を避けるべく、HTTP、URL、HTML はインターネット標準を策定する標準化機関である Internet Engineering Task Force (IETF) へ持ち込まれた。

Berners-Lee は 1992 年 7 月に開催された会合である IETF 24 に参加し、Universal Document Identifier (UDI、現在の URL ならびに Uniform Resource Identifier: URI 仕様) に関する Birds of Feather (BoF) セッションを行った^{23,24}。これがウェブ技術を標準化する具体的な動きの第一歩である。

URL と HTTP は、その後も IETF において仕様策定・バージョンアップが進

²² Berners-Lee, T. (n. d.). *The worldwideweb browser*. Retrieved April 12, 2014, from <http://www.w3.org/People/Berners-Lee/WorldWideWeb.html>

²³ BoF セッションの記録は、IETF 24 の proceedings (<http://www.ietf.org/proceedings/24.pdf>) に記録が残っている。

²⁴ その場で議題を提起し、興味関心のある者が自由に集まってディスカッションを行う形式のミーティング。

められた。しかし、ブラウザで表示される文書を記述するための HTML については、後に Berners-Lee が設立する W3C に策定の舞台が移行することとなった。

IETF における当時の標準化プロセス (図 14)²⁵は、IETF 内で標準化の必要性が認められた提案が Internet Draft というステータスとなり、IETF の Internet Draft Directory にて公開される。Internet Engineering Steering Committee (IESG) によって認証されれば Request for Comment (RFC: インターネット標準)となるが、6ヶ月以内に認証を受けない場合、Internet Draft Directory から削除され、Internet Draft としての効力を失う。

HTML は、1993 年 6 月に Internet Draft²⁶として公開されたものが HTML のバージョン 1.0 とされている。しかし、実際に RFC 番号が付与されたインターネット標準は 1995 年 11 月の HTML 2.0 (RFC 1866)²⁷ まで登場しなかった。

この裏側には新規参入してきた商業ブラウザベンダによる、独自拡張の実装があった。HTML3.2 の著者である Dave Ragget は、以下のように記している。



図 14. IETF の標準化プロセス (RFC1602²⁸)

²⁵ Bradner, S. (1996, October). *The Internet Standards Process -- Revision 2. (RFC 1602)*. Retrieved April 18, 2014, from <http://tools.ietf.org/html/rfc1602>

²⁶ Berners-Lee, T. and Connolly, D. (1993, June). *Hypertext Markup Language (HTML): A Representation of Textual Information and MetaInformation for Retrieval and Interchange*. Retrieved 18 September 2010. from <http://www.w3.org/MarkUp/draft-ietf-iiir-html-01.txt>

²⁷ Berners-Lee, T. and Connolly, D. (1995, November). *Hypertext Markup Language- 2.0*. Retrieved 14 April 2014. from <http://tools.ietf.org/html/rfc1866>

²⁸ Bradner, S. (1996, October). *ibid.*

During 1993 and early 1994, lots of browsers had added their own bits to HTML; the language was becoming ill-defined. In an effort to make sense of the chaos, Dan Connolly and colleagues collected all the HTML tags that were widely used and collated them into a draft document that defined the breadth of what Tim Berners-Lee called HTML 2 (Ragget et al., 1998).

HTML2.0 は Internet Standard のステータスまで辿り着いたが、その背景には各社の独自仕様を調べあげ、それを標準仕様提案に落としこむという苦労が存在したのである。

IETF における標準化作業のペースが早まらない一方で、NCSA を始めとするブラウザベンダは独自仕様を次々と実装し、ウェブの機能を拡大させていく。この仕様が分裂する状況を解消するために Tim O'Reilly が開催した (Quittner and Slatalla, 1998, p. 64) のが 1993 年 6 月の WWW Wizard Workshop である。ここでは、Tim Berners-Lee が標準の重要性を説いたのに対し、後に Netscape 社を率いることとなる Marc Andreessen は、独自にかつ迅速に新たな機能を実装することを優先させる姿勢を崩さなかった (Quittner and Slatalla, 1998, p. 69)。その後 Netscape は Andreessen の考え方のもと、自社のブラウザである Navigator の開発において、標準の HTML 仕様に対する互換性を充たすだけでなく、独自の拡張機能を相次いで投入することとなる。

4.1.3. W3C 設立と HTML3.0 仕様策定の失敗

この状態を改善すべく Berners-Lee は W3C を 1994 年に発足させる²⁹。IETF に代わるウェブ技術の標準化団体の設立を探っていた Berners-Lee と、MIT X consortium を外部に切り出した後の新たな産学連携プロジェクト発足を狙っていた MIT 計算機科学ラボラトリー (The Laboratory for Computer Science: LCS) との間で利害が一致したことにより、MIT/ LCS において W3C が設立されることとなった。

W3C がどのような役割を担うかについて、以下の様な議論が交わされてい

²⁹ Massachusetts Institute of Technology. (1994, October 19). LCS announces Web industry consortium. *MIT News*. Retrieved October 26, 2014, from <http://newsoffice.mit.edu/1994/lcs-1019>

³⁰ Tim Berners Lee が W3C を設立するまでの経緯については、(Berners-Lee and Fischetti, 1999) に詳しい。

る。

On December 14 (1994) at LCS the World Wide Web Consortium held the first meeting of its Advisory Committee. The meeting was very friendly and quite small, with only about twenty-five people. Competitors in the marketplace, the representatives came together with concerns over the potential fragmentation of HTML. This was seen as a huge threat to the entire community. There were so many proposed extensions for HTML that a standard really was needed. We wrestled over terms – whether the consortium should actually set a “standard” or stop just short of that by issuing a formal “recommendation”. We chose the latter to indicate the getting “rough consensus and running code” – the Internet maxim for agreeing on a workable program and getting it out there to be tried – was the level at which we would work. We also had to move fast, and didn’t want to be dragged down by the sort of long international voting process that typified the setting of an actual standard (Berbers-Lee and Fischetti, 1999, p. 97).

この場では、W3Cとして発行する技術文書をどのようなものとして位置づけるか、つまり標準化団体としての性格付けについて議論されていたのである。

このような議論が進められている間にも新たな実装が増え続けるという事態に対応し、HTML 2.0がRFCとなる以前の1995年3月28日に、W3Cの設立メンバーであるDave RaggettがHTML 3.0仕様をInternet-DraftとしてIETFに提出する³¹。

このInternet-Draftは、各社の独自仕様の取り込みをはかった結果膨大な分量となり、議論が紛糾することとなった (Raggett et al., 1998)。結局6ヶ月という期限内に意見を集約することができず、HTML 3.0 Internet Draftは失効することとなる。この後、HTML3.0をベースにしたHTML3.2からは、W3Cにおいて標準化が行われることとなった。

標準化においては、コンセンサス形成に必要とされる期間と仕様が実現する

³¹ Raggett, D. (1995, February). *HyperText Markup Language Specification Version 3.0*. Retrieved 14 April 2014, from <http://www.w3.org/People/Raggett/html3/html3.txt>

イノベーションはトレードオフの関係にあるという指摘がある (Simcoe, 2007; Farrell & Simcoe, 2012)。IETF ではコーディネーションの議論に期限を設けることで、意見集約の迅速化を図るとともに、普及可能性の低い仕様提案が標準として採択されることを防ぐ効果をもたらすことが狙われたと推測できるが、既に普及している仕様のバージョンアップに対しても効果を発揮することができなかつたのである。ルールとして期限を設けるだけでなく、合意形成と普及を促進するための施策が盛り込まれた標準化プロセス運営手法が必要とされる状況であった。

4.2. 当初プロプライエタリ技術として登場したウェブアプリケーションのための仕様群

当初ウェブの機能拡張（とそれに伴う仕様分裂）は、画像の挿入やテーブルレイアウトなど「静的な文書の表現力の向上」が主眼であった。しかし、1990年代なかばから、インタラクティブコンテンツやアプリケーションの実行環境としての機能を付加するための技術が相次いで市場投入されるようになった。

1995年にMacromedia（現Adobe Systems）はShockwaveならびにFlashというウェブ上のマルチメディアコンテンツ再生環境ならびにインタラクティブ機能を実現するプラグインソフトを提供し始める。

また、Sun Microsystemsは、これまでインタラクティブテレビ向けに開発されてきたプログラミング言語であるOAKをJavaと改称し、マルチプラットフォーム向け言語として市場投入する。そしてNetscape Communications (Netscape)が自社ブラウザのNavigator 1.1にSun Microsystemsが開発したプログラミング言語であるJavaの実行環境を実装したことにより、ウェブは本格的にアプリケーションの実行環境としての進歩が始まることとなった。同年、Netscapeは、ウェブブラウザ上でウェブページを操作する目的に特化させたプログラム言語であるJavaScriptをNetscape 2.0に実装する。

更に1999年にはMicrosoftがXMLHttpRequestと呼ばれるサーバとブラウザ間でデータを効率的にやりとりするプロトコルをInternet Explorerに実装する。

ウェブの機能拡張は、Adobe、Sun Microsystems、Netscape、Microsoftなどの企業によるプロプライエタリ仕様によってはじまった。特に1990年代後半はNetscapeとSun Microsystems連合とMicrosoftの間による熾烈なブラウザ市場シェア獲得競争は、ブラウザ戦争と呼ばれるに至る。

当初、ウェブアプリケーションを実現するための技術仕様開発は、プロプラ

イェタリな技術として始まったのである(図 15)。

W3C においても、ウェブをアプリケーションの実行環境とするための取り組みは 1995 年には始められている。Director の Tim Berners-Lee は、“Distributed object-oriented system”というコンセプトを打ち出している³²。また、1995 年 7 月 5 日には、Mobile code workshop が開催され、Java や Python など様々なウェブプログラミング、ウェブアプリケーション基盤を構成する要素技術について紹介され、意見交換がなされている³³。

W3C における HTML のバージョンアップと機能拡張は、技術革新のスピードが速い上に、プロプライエタリな競合技術仕様を供給する主体を含めた利害調整が必要という難易度の極めて高い作業であった。

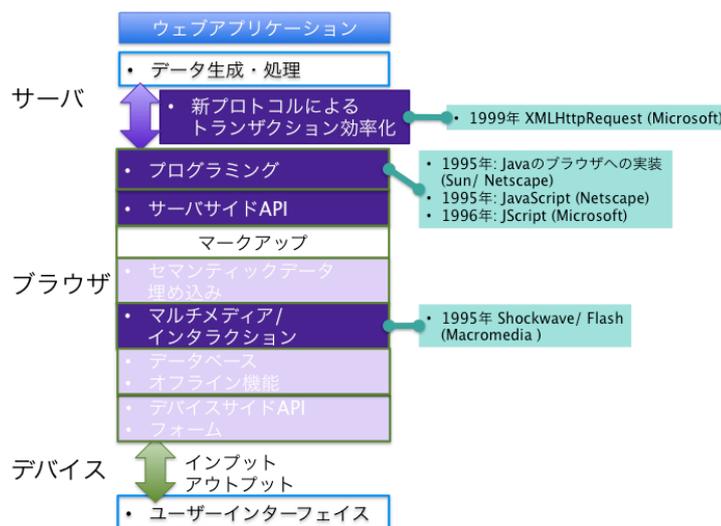


図 15. プロプライエタリな仕様として提供されていたウェブアプリケーション
実現のための技術仕様群

³² Berners-Lee, T. (1995). *Convergence with distributed OO systems*. Retrieved June 3, 2014, from <http://www.w3.org/Bugs/Objects.html>

³³ Miller, J. (1995). *Minutes of the First W3C Mobile Code Workshop*. Retrieved June 3, 2014, from <http://groups.csail.mit.edu/mac/users/jmiller/mobile.html>

第5章. Phase 1: 議論の場作りが手探りで始められた勃興期 – HTML3.2/ 4.0

ウェブは 1 研究所における所内インフラから始まった。当初は開発者の Tim Berners-Lee が単独で主要な仕様 (HTTP/ HTML/ URL) を開発し、ブラウザまでも自製していたが、様々な主体がブラウザの開発と機能拡張を担う事となった。このことは、ウェブが世界規模な情報インフラへと役割を拡大し、迅速に普及していった大きな要因となったものの、仕様の分裂・乱立といった状態を招くに至った。その後 IETF において標準化の試みが進められたものの、独自拡張仕様の出現を抑止することができなかった。

この状況を W3C という標準化団体がどのように改善していくのか、ここからは W3C における HTML ならびに DOM³⁴の仕様策定プロセスを時系列で分析する。本章では IETF による標準化の失敗を受けて急遽 HTML の標準化を開始した W3C が手探りで標準化プロセス運営方針を確立していく過程と、その限界を明らかにしていく

5.1. 策定作業の W3C への移行と手探りでのプロセス開始

1994 年 10 月 19 日に W3C が発足した以降も、IETF における HTML の新バージョン策定作業は進められた。HTTP、URI/ URL は現在(2014 年)に至るまでバージョンアップ作業は、引き続き IETF において行われている。

W3C において HTML の策定が始まったのは HTML 2.0 Internet Draft の効力が切れた 1995 年 9 月前後である。1995 年 10 月 19 日、W3C の Dave Raggett は、“HTML3 Linking and Embedding Model” という HTML3.0 拡張仕様の Draft³⁵ を公開するなど、W3C 設立メンバーは HTML3.0 の拡張仕様を標準化する試みを継続していた。

そして HTML2.0 の筆者である Dave Raggett は、IETF の HTML WG の活動に行き詰まりを感じ、1995 年 11 月にブラウザベンダの代表者を集めた小規模なグループを結成する (表 5)。このグループの活動成果としては、1995 年 12 月 1 日に内部公開³⁶、12 月 21 日に一般公開³⁷された、“Inserting multimedia objects into

³⁴ Document Object Model。ウェブサイトを操作するための Application User Interface 仕様。機能の詳細は第 6 章を参照のこと。

³⁵ Raggett, D. (ed.). (1995, October 19). *HTML3 Linking and Embedding Model: W3C Working Draft 19-Oct-95*. Retrieved May 27, 2014, from <http://www.w3.org/MarkUp/Embed/WD-embed-951019.html>

³⁶ Raggett, D. (ed.) (1995, December 1). *Inserting multimedia objects into HTML3: W3C Working Draft*

HTML3”である。

12月7日に、このグループ参加者等を登録したメーリングリストが開設され (html-erb@w3.org)³⁸、W3C において HTML 仕様バージョンアップの作業が本格的に始まることとなった。

有志によるメーリングリストによる議論から発展し、HTML3.2 仕様策定の議論が行われる場として W3C 内に設置されたのが HTML Editorial Review Board (HTML-ERB)である。

HTML3.2 の策定作業が始まった当時、まだ W3C において策定プロセスの運営方針を明文化するという動きは始まっていない。HTML-ERB では、Draft のバージョン毎に参加企業・団体による投票が行われ、かなり細かく合意形成の手順を積み重ねるといった形をとっていた。

5.2. 現状追認としての HTML3.2

HTML3.0 の策定プロセスが進められていた 1995 年当時は、Microsoft がインターネット接続機能を実装した Windows95 を発売し、かつ Internet Explorer をバンドルすることでブラウザ市場に参入した時期であり、ウェブサイト制作者もエンドユーザも急速に増加する時期であった。

そのような市場環境下で激しく競争していた Microsoft と Netscape は、それぞれ独自の拡張仕様を実装することで、機能的差別化を図るという戦略を採用したのである。

表 5. 1995 年 11 月に Dan Connolly が結成したグループの参加者

氏名	所属
Lou Montulli	Netscape
Charlie Kindel	Microsoft
Eric Sink	Spyglass
Wayne Gramlich	Sun Microsystems
Dave Raggett	W3C
Tim Berners-Lee	W3C
Dan Connolly	W3C
Jonathan Hirschman	Pathfinder

01-Dec-95. Retrieved April 14, 2014, from <https://www.w3.org/MarkUp/Group/WD-insert-951201.html>

³⁷ Raggett, D. (ed.) (1995, December 21). *Inserting multimedia objects into HTML3: W3C Working Draft 21-Dec-95*. Retrieved April 14, 2014, from <http://www.w3.org/TR/WD-insert-951221.html>

³⁸ メーリングリスト最初のメール送信日より: Raggett, D. (1995, December 7). *Welcome to the HTML ERB*. Retrieved May 27, 2014, from html-erb@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/html-erb/1995OctDec/0000.html>

4.1.3 項にあるように、HTML3.0 は IETF において標準化が進められたものの、Internet-Draft としての効力が失われ、激しい競争を続けるブラウザベンダに実装内容を準拠させる強制力は伴っていなかった。1996 年 2 月 13 日から 14 日にかけて、Redmond の Microsoft において行われた HTML-ERB 初の F2F 会合の議事録³⁹によると、ブラウザ供給者が HTML3.0 準拠と謳った製品をリリースしているにもかかわらず、実際には各社が独自に仕様を拡張しているためそれに対する対処が必要であるという議論がなされている。

その対処としてブラウザ供給者が既に独自拡張として実装してしまった機能を追認して標準化することにより、現在ウェブサイト制作者やエンドユーザが利用している機能を損なわないまま、相互可用性の実現を目指したのである。

1996 年 5 月 7 日に公開された HTML3.2 の First Public Working Draft (FPWD) 公開時のプレスリリースには、以下の様な文言が含まれており、主要なステークホルダーが揃って仕様策定に参画していることと、合意形成の元に作業が進められている点が訴求されている。

HTML 3.2 is W3C's new specification for HTML, developed together with industry leaders IBM, Microsoft, Netscape Communications Corporation, Novell, SoftQuad, Spyglass, and Sun Microsystems. Design work on HTML 3.2 is based upon HTML 2.0 and draws from such sources as the HTML+ and HTML 3.0 drafts by Dave Raggett of Hewlett Packard Laboratories, and extensions proposed by W3C member companies, including major contributions popularized by Netscape Communications Corporation.

W3C is continuing work with vendors on extensions to HTML for multimedia objects, scripting, style sheets, layout, forms, higher quality printing and math. W3C plans on incorporating this work in further versions of HTML⁴⁰.

³⁹ Raggett, D. (1996, February). *Meeting Notes for HTML-ERB 13/14 Feb '96*. Retrieved 14 April 2014, from https://www.w3.org/MarkUp/Group/960213_Minutes.html

⁴⁰ World Wide Web Consortium. (1996, May 7). *The W3 Consortium Announces HTML 3.2: providing bright*

しかし、実際には合意形成は困難を極めていた。World Wide Web Journal に掲載された HTML3.2 の策定プロセス参加者に対するインタビュー記事 (World Wide Web Consortium, 1997) において、Netscape から参画していた Lou Montulli は以下のように述べている。

Netscape's Lou Montulli states. "But in fact the history of the Internet is that things have to be working and in place before they get standardized. It allows people to work out problems before setting anything in stone. (World Wide Web Consortium, 1997, pp. 5)

各ブラウザ供給者の独自拡張仕様実装による差別化を図る傾向は、W3C 設立以降も変わらないものであった。

1996年6月20日に実施された W3C HTML Editorial Review Board の F2F 会合議事録⁴¹には、以下の様な記述があり、各社が独自に実装している機能を互換性のある形で仕様化する作業は困難であったことが窺える。

All examples should be concrete and should work in browsers and be in wide use in content. This expert group has difficulty expressing how this decision would be made⁴².

この困難さをストレートに伝えるのが、HTML-ERB Chair である Dan Connolly の「リバース・エンジニアリング」という言葉である。具体的に Connolly は以下のようにインタビューに回答している。

"In a sense it was a reverse-engineering assignment," Dan Connolly, the group's chair adds. "We wanted to take what people were already doing and write it up. Basically we had to decide which elements we were going to bless." (World Wide Web Consortium, 1997, pp. 6)

future for HTML as a non-proprietary standard. Retrieved 14 April, 2014, from <http://www.w3.org/MarkUp/Wilbur/pr7may96.html>

⁴¹ Connolly, D. (ed.) (1996, December 9). HTML ERB Jun 96 Meeting (議事録). Retrieved May 27, 2014, from https://www.w3.org/MarkUp/Group/960620_Minutes.html

⁴² *ibid.*

1997年1月14日、HTML 3.2はRecommendationとなり、初のW3C版HTML標準仕様策定プロセスが完結する。最終的にこの仕様は、最新のウェブ技術を積極的に取り入れるのではなく、激しくブラウザ戦争を争っていたNetscapeとMicrosoftが実装した仕様の中で両社、ならびに他の参加者間で合意ができたものに絞って集めたものとなった。

5.3. 策定プロセスの整備とHTML Working Group 始動

HTML3.2 策定プロセス終結の目処が見えてきた1996年12月、Menlo Parkで開催されたHTML-ERBのF2F会合において、Dave RaggetがHTML4.0の策定作業開始を提案する⁴³。翌1997年2月、HTML-ERBはWGに改組され⁴⁴、HTML WGにおいてCougar、後のHTML4の策定プロセスが始まることとなった。

HTML-ERBのHTML WGへの改組に先駆け、W3Cにおいて標準化プロセス整備のための活動が開始されている。1996年6月11日にPeabodyで行われたAdvisory Committee Meeting (AC meeting)において標準化プロセスをレビューするパネルディスカッションが行われ⁴⁵、そこで標準化プロセスについて検討するための作業部会の成立が求められた。この要請に応じ⁴⁶、Process Editorial Review Board (Process-ERB)の設置方針が決定され⁴⁷、メンバーが募集された。Advisory Committeeによる投票を経て、9月16日にProcess-ERBが正式に発足する。

10月23日に初回のF2F会合が、次いで28日に電話会議が行われ、2会合の議論をベースとして⁴⁸10月30日にProcess DocumentのDraft1⁴⁹が提出される。11月20日に提出されたDraft4⁵⁰において、仕様案のステータスとして、Working Draft/ Proposed Recommendation/ Recommendationの3段階が置かれることが明記

⁴³ Bos, B. (1996, December). *HTML-ERB minutes, 5/6 Dec 1996 (Menlo Park, CA)*. Retrieved 15 April 2014, from <https://www.w3.org/MarkUp/Group/9612/05-minutes>

⁴⁴ Quint, V. (1997, February 21). Working Groups Replace HTML ERB. *W3C news letter*. 3(4). Retrieved 15 April 2014, from <https://www.w3.org/Member/Newsletter/970221.html#Three>

⁴⁵ Greene, T. J. (2006, June). *Agenda for 10 June 1996*. Retrieved 15 April 2014, from https://www.w3.org/Member/Meeting/960611/960610_Agenda.html

⁴⁶ Jacobs, I. (2005). *Long Description of W3C10 Timeline Graphic*. Retrieved 15 April 2014, from <http://www.w3.org/2005/01/timelines/description>

⁴⁷ Berners-Lee, T. (2004, December 1). *How It All Started*. Retrieved May 28, 2014, from <http://www.w3.org/2004/Talks/w3c10-HowItAllStarted/all.html>

⁴⁸ Khudairi, S. (1997, August 9). *W3C Process Document*. Retrieved May 28, 2014, from <https://www.w3.org/Consortium/Process/Group/Document.html>

⁴⁹ Khudairi, S. (ed.) (1996, October 30). *W3C Process Document - Draft 1*. Retrieved May 28, 2014, from <https://www.w3.org/Consortium/Process/Group/Draft1.html>

⁵⁰ Khudairi, S. (ed.) (1996, November 20). *W3C Process Document - Draft 4*. Retrieved May 28, 2014, from <https://www.w3.org/Consortium/Process/Group/Draft4.html>

される。

HTML-ERB においても 1996 年 6 月 20 日に開催された F2F 会合において⁵¹、HTML-ERB の位置付けや運営手続きなどの基本ルールをまとめる必要性が議論され、後に Dan Connolly が“HTML ERB Ground Rules⁵²”を作成している。このような流れに基づき、1997 年 2 月に HTML-ERB は HTML WG に改組されることとなった⁵³のである。

5.4. 独自実装による分裂状況が解消できなかった HTML4

HTML3.2 のコンセプトは、当時の主要ブラウザが独自実装していた機能をできるだけ標準化することで相互可用性を確保することであった。一方 W3C と Director の Tim Berners-Lee は、デザインと構造の分離を推し進めて HTML の機能を文書の構造化に限定したものを次期バージョンとしたいと考えていた。最終的にこのコンセプトが HTML4 として結実する。

構造化された文書とは、文書を構成する内容が役割によって分割・整理され、各部分間の関係性が表されているという状態である。本論文は章節項目に分割されて階層構造をもっている。この章節項目への分割と階層構造（上位概念・下位概念の整理）、を行われている文書が構造化文書である⁵⁴。文書の構造化という、当時の技術的理想を実現するというコンセプトで HTML4 は策定されたのである。

新たなコードネームが与えられたものの、HTML4 も Microsoft と Netscape という 2 大ブラウザベンダによる独自実装を追加する形で策定されていった。その象徴的な仕様が IFRAME 要素である。

Microsoft は 1996 年 8 月にリリースした IE バージョン 3.0 に実装した独自拡張である IFRAME 仕様を WG に提案する⁵⁵。そして激しい議論の末、HTML4.0 の Draft は 1997 年 7 月に一般公開されることとなった⁵⁶。

その後 HTML4.0 は、11 月 7 日に新設された Proposed Recommendation⁵⁷ に、

⁵¹ Connolly, D. (ed.) *HTML ERB Jun 96 Meeting (minutes)*. Retrieved May 28, 2014, from https://www.w3.org/MarkUp/Group/960620_Minutes.html

⁵² Connolly, D. (1996, December 9). *HTML ERB Groundrules – draft*. Retrieved May 28, 2014, from <https://www.w3.org/MarkUp/Group/groundrules>

⁵³ Quint, V. (1997, February 21). 3. Working Groups Replace HTML ERB. *W3C News Letter*, 3(4). Retrieved May 28, 2014, from <https://www.w3.org/Member/Newsletter/970221.html#Three>

⁵⁴ 構造化に関する詳細は、付録 II を参照のこと。

⁵⁵ Isaacs, I. (1997, March 18). *IFrame*. Retrieved 15 April 2014, from w3c-html-wg@w3.org mailing list archive, <https://lists.w3.org/Archives/Member/w3c-html-wg/1997JanMar/0068.html>

⁵⁶ Raggett, D. et al. (eds.). (1997, July 8). *HTML 4.0 Specification - W3C Working Draft 8-July-1997*. Retrieved May 29, 2014, from <http://www.w3.org/TR/WD-html40-970708/>

⁵⁷ Raggett, D. et al. (eds.). (1997, November 7). *HTML 4.0 Specification - W3C Proposed Recommendation*

12月18日に Recommendation⁵⁸ となる。

なお、当時は標準化プロセスの手順、ならびに仕様書 (Technical Report)の段階について定める Process Document は Draft の段階であり、仕様書のステータスに関する明確な規定は公開されていなかった。Process Document の HTML4.0 の策定作業が終了した段階のバージョン⁵⁹では、仕様書のステータスは Working Drafts (WD)/ Proposed Recommendation (PR)/ Recommendation の3段階とされており、WD が PR に進むための要件は、単に “Stable” になればよいだけであった。

HTML4.0 が Recommendation になった直後の 1998 年 4 月には、HTML の開発を HTML4 で終了し、XML ベースの HTML である XHTML の開発を進める方針が決定されている⁶⁰。その後、HTML4 は HTML4.0 (Revised)⁶¹、HTML4.01⁶² というバージョンが公開されるものの、その中身は HTML4.0 の微修正である⁶³。HTML4.0 が Recommendation となった 1997 年 12 月 18 日に仕様はほぼ固定化されたわけである。

W3C は HTML 4.0 が Recommendation となったのを受け、Validator と呼ばれる HTML ファイルの記述内容をチェックするサービス(図 16)⁶⁴を 1998 年 4 月 20 日にリリースしており⁶⁵、オープンソース・ソフトウェア開発コミュニティでは一般的な(von Hippel and Katz, 2002)、開発者をサポートするためのツールキットの提供という普及促進のための施策が講じられている。

一方 Netscape のブラウザが IFRAME 要素を実装したのは、America on Line (AOL) に買収された以降、2000 年 11 月にリリースされた Netscape6 からであった。つまり、ブラウザ戦争が終結した後に、IFRAME は主要ブラウザ全てに実装されたことになる。

7-Nov-1997. Retrieved May 29, 2014, from <http://www.w3.org/TR/PR-html40-971107/>

⁵⁸ Raggett, D. et al. (eds.). (1997, December 18). *HTML 4.0 Specification - W3C Recommendation 18-Dec-1997*. Retrieved May 29, 2014, from <http://www.w3.org/TR/REC-html40-971218/>

⁵⁹ Cargill, C. et al. (1997, November 19). *World Wide Web Consortium Process Document - PR-w3c-process-971119*. Retrieved May 29, 2014, from <https://www.w3.org/Consortium/Process/Group/Draft12-971119>

⁶⁰ 詳細は 7.2 節を参照

⁶¹ Raggett, D. et al. (eds.). (1998, April 24). *HTML 4.0 Specification - W3C Recommendation, revised on 24-Apr-1998*. Retrieved May 29, 2014, from <http://www.w3.org/TR/1998/REC-html40-19980424/>

⁶² Raggett, D. et al. (eds.). (1999, December 24). *HTML 4.01 Specification - W3C Recommendation 24 December 1999*. Retrieved May 29, 2014, from <http://www.w3.org/TR/1999/REC-html401-19991224/>

⁶³ *ibid.*, A.1 Changes between 24 April 1998 HTML 4.0 and 24 December 1999 HTML 4.01 versions. Retrieved May 29, 2014, from <http://www.w3.org/TR/1999/REC-html401-19991224/appendix/changes.html#19991224>

⁶⁴ World Wide Web Consortium. (n. d.). *The W3C Markup Validation Service*. Retrieved November 5, 2014, from <http://validator.w3.org/>

⁶⁵ World Wide Web Consortium. (n. d.). *What's New at The W3C Markup Validation Service*. Retrieved November 5, 2014, from <http://validator.w3.org/whatsnew.html>

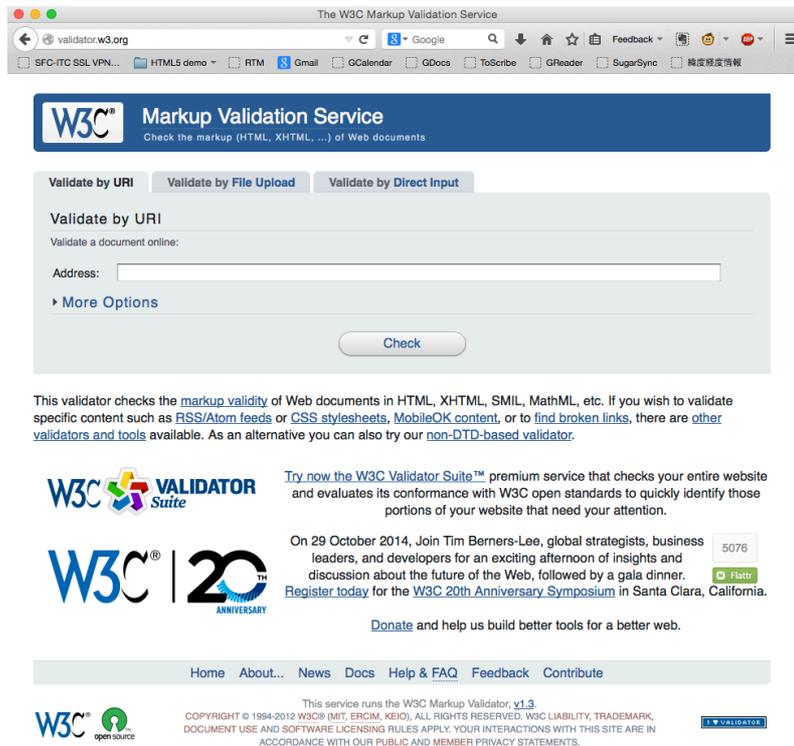


図 16. W3C Markup Validation Service⁶⁶

W3C は、HTML3.2 と HTML4 の策定により、実効性のある標準仕様を確立させることはできず、市場競争の結果として勝者が提唱した仕様が普及したのである。この時、既に HTML4.0 が Recommendation となってから 3 年が経とうとしていた。

5.5. 既に存在する独自実装を、後追いで統合する困難さ

HTML3.2 ならびに HTML4 の仕様策定作業は、W3C の仕様策定プロセス整備とほぼ平行して行われた。HTML3.2 が HTML-ERB の元で策定され、頻繁に投票が行われていたのに対し、HTML4.0 以降は WD/ PR/ Recommendation という仕様書のステータスが明確になり、投票は Recommendation になる直前のみと、手続としては簡略化された。しかし、仕様策定の議論は紛糾を繰り返した。

その大きな原因は、既に各社がバラバラに実装していた独自拡張仕様をリベース・エンジニアリングする形で再構築して仕様を創りあげるという手法にある。この手法のために、1) 標準仕様のスコープにはステークホルダー間の合意が得られた部分のみが含まれる、2) スコープから外れた機能の独自実装を抑制

⁶⁶ ibid.

するしくみがない、という状況となったのである (図 17)。

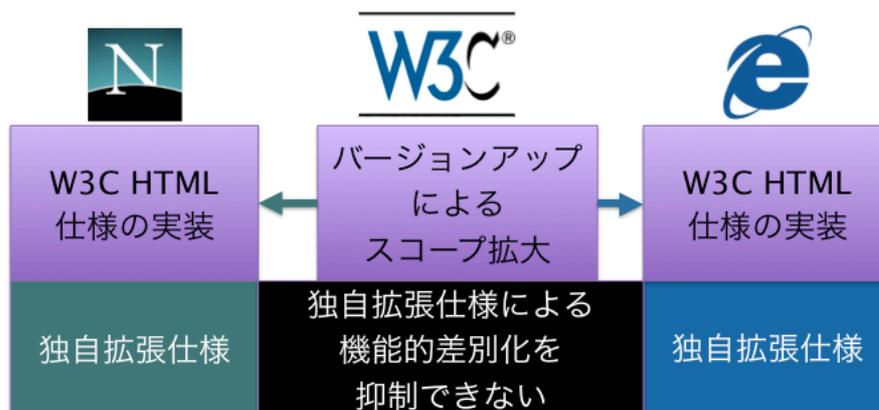


図 17. HTML3.2/HTML4 策定の構図

表 6. HTML4.01 とそれ以前の HTML 標準化動向

日時	事象
1993/6	Tim Berners-Lee と Dan Connolly によって、HTML 1.0 (Internet Draft) が IETF IIR Working Group に提出される。
1993/6/28-30	WWW Wizard workshop 開催
1994/10/1	W3C 発足
1994/12/15	Netscape Navigator 1.0 公開
1995/8/16	Internet Explorer 1.0 公開 (Windows 95 発売)
1995/9/28	HTML 3.0 の Internet Draft としての効力が切れる
1995/11	Dan Connolly が IETF の HTML WG の活動に行き詰まりを感じ、ブラウザベンダの代表者を集めた小規模なグループを結成
1995/12/7	W3C Editorial Review Board (HTML-ERB) ML 開設
1996/2/13-14	HTML-ERB 初めての F2F 会合
1996/5/7	HTML 3.2 の First Public Working Draft が公開される
1996/6/15	W3C の Dave Raggett が Cougar (HTML4)の策定に関する基本方針、とスペックを提案する
1996/9/16	Process-ERB のメンバーが Advisory Board によって選出される
1996/11/20	Process-ERB による Process Document Draft 4 が会員向けに公開。このバージョンから、WG のプロセスが記述される。Working Draft/ Proposed Recommendation/ (Disposition)/ Recommendation の 3 段階
1996/11/25	HTML 4.0 First Public Working Draft 公開
1997/1/14	HTML 3.2 が Recommendation に
1997/2	HTML-ERB が HTML WG に改組される
1997/7/8	HTML 4.0 FPWD 公開
1997/11/7	HTML 4.0 PR
1997/12/18	HTML 4.0 Recommendation
1998/4/24	HTML 4.0 (Revised)が Recommendation
1998/11/24	Netscape が America on Line に買収される
1999/12/24	HTML 4.01 Recommendation
2000/11/14	Netscape 6 公開。IFRAME を実装

第6章. Phase 2: プロセス運営方針の明確化が進んだ過渡期 - DOM Level 1/ 2

ここまで HTML の標準化は、ブラウザベンダ各社が独自に拡張した仕様を組み込む形で進められてきた。この手法とは異なる形でプロセスが進められたのが、サーバで動作するアプリケーションが HTML を操作する際の API である Document Object Model (DOM) 仕様の策定である。

DOM はウェブアプリケーション実現のための重要な要素である (図 18) が、それだけに HTML3.2/ 4.0 同様に独自仕様の乱立などが予測された。本章では、DOM Level 1/2 という 2 バージョンの策定プロセス分析を通じ、どのようにして DOM 仕様の実効性が確立されたか分析していく。

6.1. Document Object Model の概略と WG 発足に至る経緯

W3C における DOM 仕様の策定が始まる以前から、各ブラウザ供給者は先行して HTML (や CSS) を操作するためのプログラミング言語がブラウザに導入している。

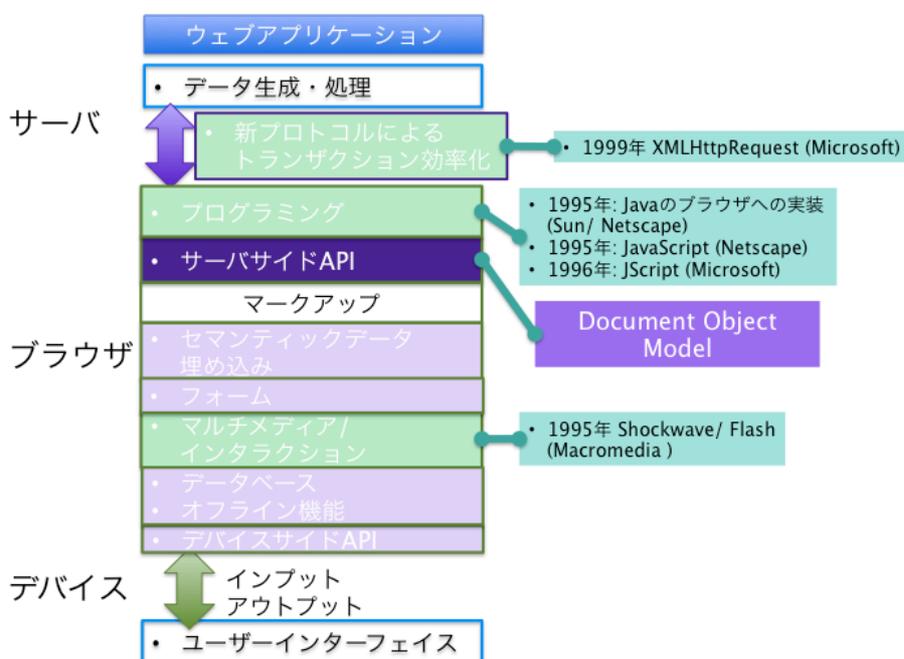


図 18. ウェブアプリケーションアーキテクチャにおける DOM の位置づけ

1995年12月に Netscape が Navigator 2.0 Beta で JavaScript を実装、Java プラグインにも対応している。また Microsoft も、翌年の 1996 年 8 月にリリースした Internet Explorer 3 にて JScript を実装している。これら 2 つの言語は 1996 年 11 月より、Ecma International において ECMAScript という名称⁶⁷にて標準化作業が開始されている⁶⁸。

W3C における API 側の標準の口火を切ったのは Microsoft であった。1996 年 1 月 20 日に Object Linking and Embedding (OLE)⁶⁹ と HTML の専門家である Charlie Kindel が HTML-ERB メーリングリストに Internet Explorer のスクリプト言語対応仕様の紹介を投稿している⁷⁰。この紹介内容と、Netscape Navigator の JavaScript 対応仕様拡張を元に⁷¹、W3C team staff の Dave Raggett が “HTML3 Scripting” という名称で仕様案を作成している⁷²。一方 Netscape は、1996 年 8 月 19 日に “JavaScript-based Style Sheets” という名称の仕様⁷³を W3C に提出。8 月 22 日に Draft として HTML-ERB 内で公開された⁷⁴。

1996 年 9 月 9 日から 10 日にかけて開かれた HTML-ERB の F2F 会合は、2 社による競合する 2 つの仕様が提案された段階で開かれた。そこで Microsoft の Scott Isaacs が API の標準化を改めて提案する。呼びかけの中では “put MS, NS, Spyglass together work through the issues⁷⁵.” と発言するなど、当初から Microsoft は競合ブラウザベンダと協力して標準化を進めていく姿勢を見せていた。その後 9 月 15 日に Isaacs は、“HTML Object Model Overview” draft⁷⁶を HTML-ERB のメー

⁶⁷ 一般的に JavaScript と呼ばれている仕様は ECMAScript を指すことが多い。JavaScript は、本来 Sun Microsystems と Netscape によって開発された言語を指し、競合する JScript との間で標準化された仕様を ECMAScript と称するが、JavaScript の方が広く用いられるようになっている。

⁶⁸ Netscape Communications. (1996, November 15). Industry leaders to advance standardization of Netscape’s JavaScript at standards body meeting – Netscape to post JavaScript specification and licensing information on Internet site. Retrieved June 1, 2014, from Internet archive: <http://web.archive.org/web/19981203070212/http://cgi.netscape.com/newsref/pr/newsrelease289.html>

⁶⁹ Microsoft のネイティブ OS 上でのアプリケーション間のデータ転送技術

⁷⁰ Kindel, C. (1996, January 20). *INSERT and SCRIPT*. Retrieved June 1, 2014, from html-erb@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/html-erb/1996JanMar/0006.html>

⁷¹ Raggett, D. (1996, February 8). *HTML3 Scripting*. Retrieved June 1, 2014, from <http://www.w3.org/TR/WD-script-960208.html>

⁷² Raggett, D. (1996, February 8). *Draft spec for scripting extensions to HTML*. Retrieved June 1, 2014, from html-erb@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/html-erb/1996JanMar/0034.html>

⁷³ Montulli, M. et al. (1996, August 22). *JavaScript-based Style Sheets*. Retrieved June 1, 2014, from <https://www.w3.org/MarkUp/Group/WD-jsss-960822>

⁷⁴ Connolly, D. (1996, August 22). *JavaScript stylesheets docs available*. Retrieved June 1, 2014, from html-erb@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/html-erb/1996JulSep/0375.html>

⁷⁵ Freter, T. & Connolly, D. (1996, September). Sep 96 HTML ERB Meeting. Retrieved 15 April 2014, from <https://www.w3.org/MarkUp/Group/960909-SFO.html>

⁷⁶ Kaplan, L. A. & Isaacs, S. (1996, September 15). *HTML Object Model Overview*. Retrieved June 1, 2014, from <https://www.w3.org/MarkUp/Group/webpageapi/>

リングリストに提出する⁷⁷。この Draft をもとに API、つまり DOM の標準化作業が行われることとなった。

その後も Isaacs は WG 設立に向けた準備を着々と進めていく。1997 年 1 月 13 日には、HTML-ERB のメーリングリストに HTML Object Model Activity Group の目的等を示した文書（いわゆる Charter Draft）を投稿し⁷⁸、2 月 26 日に、Isaacs と SoftQuad 社に所属する Peter Sharpe による最初の Working Draft が提出した^{79,80}。Charter Draft 提出に先立って、Peter Sharpe の同僚である SoftQuad 社の Lauren Wood が WG Chair に就任。HTML-ERB メーリングリストに就任の動機について説明する投稿を行っている⁸¹。Microsoft は、自身が提出した Draft を元にした仕様策定プロセスの開始と、Draft の共同提出者が Chair を輩出するという構図を整えたのである。

一方 Netscape は、その後も Navigator4 に独自仕様の実装をすすめる、1997 年 2 月 10 日に開催された WG 設立準備のための電話会議に参加しない⁸²など、DOM の標準化に対し非協力的な態度であった⁸³。

6.2. 議論のチャンネル増加とオープン化

6.2.1. Draft 公開の迅速化と公開メーリングリストの導入

現在ウェブに残されている DOM WG 公開ウェブサイトの最も古いアーカイブ⁸⁴には、以下の様な文言が記載されており、パブリックコメントを受けて仕様をブラッシュアップするために、安定する以前の段階で Draft を公開する方針が明確にされた。

⁷⁷ Isaacs, S. (1996, September 16). *Object Model Overviews*. Retrieved 15 April 2014, [html-erb@w3.org mailing list archive, https://lists.w3.org/Archives/Member/html-erb/1996JulSep/0371.html](https://lists.w3.org/Archives/Member/html-erb/1996JulSep/0371.html)

⁷⁸ Isaacs, S. (1997, January 13). *Object Model Standardization Objectives*. Retrieved June 1, 2014, from [html-erb@w3.org mailing list archive: https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0026.html](https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0026.html)

⁷⁹ Sharpe, P. (1997, February 26). *Document Object Model: WD-DOM-970226.html*. Retrieved 15 April, 2014, from [w3c-dom-wg@w3.org mailing list archive: https://lists.w3.org/Archives/Member/w3c-dom-wg/1997JanMar/0003.html](https://lists.w3.org/Archives/Member/w3c-dom-wg/1997JanMar/0003.html)

⁸⁰ Isaacs, S. & Sharpe, P. (1997, February 26). *Document Object Model: wd-dom-970219.htm*. Retrieved May 30, 2014, from <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997JanMar/att-0003/wd-dom-970226.html>

⁸¹ Wood, L. (1997, February 10). *motivation for DOM*. Retrieved June 1, 2014, from [html-erb@w3.org mailing list archive: https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0163.html](https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0163.html)

⁸² Wood, L. (1997, February). *minutes from DOM phone meeting Monday Feb 10*. Retrieved 15 April 2014, from [html-erb@w3.org mailing list archive, https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0171.html](https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0171.html)

⁸³ Isaacs, S. (1997, February 7). *events and system info and the object model*. Retrieved 15 April 2014, from [html-erb@w3.org mailing list archive, https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0153.html](https://lists.w3.org/Archives/Member/html-erb/1997JanMar/0153.html)

⁸⁴ Internet Archive に保存された最も古いバージョンの DOM WG の一般向けウェブサイト: World Wide Web Consortium. (1997, December 9). *Document Object Model*. Retrieved May 31, 2014, from Internet archive: <http://wayback.archive.org/web/19971210221128/http://www.w3.org/DOM/>

Note that these documents represent a snapshot of our thoughts and are here for public comment only. Experimental implementations may be made in the realization that specifications will change without regard to compatibility with earlier versions of the specifications⁸⁵.

つまり、パブリックコメントを受けて仕様をブラッシュアップするために、仕様が安定する以前の段階で Draft を公開するという方針が明確にされたのである。

1997 年 4 月には、非会員に対しても公開されたメーリングリスト “www-dom@w3.org” が設置される。初回の投稿は Chair の Wood によって行われ、以下のような呼びかけがなされている。

There is an overview page and a preliminary version of the requirements. We are also working on an FAQ.

Questions for the FAQ as well as comments on the preliminary requirements are welcome⁸⁶.

DOM 仕様の策定は、「全ての文書を 3 ヶ月以内に公開」し、公開メーリングリストの設置によってフィードバックを受けるという、以前と比較してオープンな形でプロセス運営が実施されることとなった。

実際に 1999 年 2 月に当時 Harvard 大学の学生で、始動したばかりの Mozilla のコミッターである⁸⁷⁸⁸ David Baron による Level 2 CSS に関するコメント⁸⁹が採用されている⁹⁰。

⁸⁵ *ibid.*

⁸⁶ Wood, L. (1997, April 25). *first pages about the DOM are available*. Retrieved May 31, 2014, from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/1997AprJun/0000.html>

⁸⁷ Baron, D. (n. d.). *Profile on the personal web site*. Retrieved June 24, 2014, from <http://dbaron.org/>

⁸⁸ 当時はまだ Netscape が存在していたので、Mozilla のコミッターがそのまま W3C 会員にはならない。なお、Baron が Netscape (既に America on Line に買収後) でインターンとして勤務し始めたのは 1999 年 6 月からであり、この時の投稿は Harvard の学生=W3C 非会員としてのものである。Baron の職務経歴は LinkedIn にて確認できる (<http://www.linkedin.com/in/ldavidbaron>, retrieved on June 24, 2014)。

⁸⁹ Baron, D. (1999, February 4). *DOM CSS*. Retrieved June 24, 2014, from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/1999JanMar/0054.html>

⁹⁰ Apparao, V. (1999, February 12). *Re: DOM CSS*. Retrieved June 24, 2014 from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/1999JanMar/0064.html>

6.2.2. Interest Group の設置

仕様策定プロセスへの関与者を増やすという試みは、W3C 外部に対してのみならず、会員企業（団体）に所属するエンジニアに対しても行われている。DOM WG のメーリングリスト “w3c-dom-wg@w3.org” は仕様策定に直接関わる、つまり Chair、仕様書の Editor、仕様書の Author、議決を取る際の投票者のみを登録対象とするものとされている。その後、毎回の電話会議や F2F 会合に出席しないステークホルダーの会員向けに Interest Group (IG) が設置される⁹¹。以降電話会議の議事録は IG 用メーリングリスト “w3c-dom-ig@w3.org” に投稿されるようになり、日常的な議論の参加者が広げられることとなった。なお、DOM Level 1/2 策定当時の IG は、現在の IG とは全く異なる機能を果たしている点を留意されたい。

6.3. Process Document における Technical Report ステップの整備

HTML4.0 ならびに DOM Level 1 の策定と平行して、Process-ERB では Process Document の改訂作業が進められていた。前述のとおり、1998 年 3 月 11 日に初めての外部公開版 Process Document⁹² が公開される。このバージョンの内容は、HTML4.0 の策定作業が終了した段階の Draft⁹³ と大差なく、Technical Report は Working Draft (WD)/ Proposed Recommendation (PR)/ Recommendation という 3 段階が設定されている。WD が PR に進むためには仕様が Stable であると Director (Tim Berners-Lee) が認証すればよく、Recommendation になるのも会員による全会一致は求められていない⁹⁴。

6.4. 未実装の機能をスコープとして定めるマイルストーン設定

プロセス運営方針の確立と平行して、仕様策定作業も進められた。3 月 25 日から 26 日にかけて第 1 回の DOM WG F2F 会合が開催される。仕様の要件として Microsoft Internet Explorer と Netscape Navigator の当時における最新バージョンでの実装済み仕様と互換性を持たせるという提案がされている。しかし、両

⁹¹ Wood, L. (1997, April 24). *DOM IG and WG*. Retrieved June 1, 2004, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0043.html>

⁹² Cargill, C. et al. (1998, May 11). *World Wide Web Consortium Process (version May 11, 1998)*. Retrieved June 5, 2014, from <http://www.w3.org/Consortium/Process/Process-19980311>

⁹³ Cargill, C. et al. (1997, November 19). *World Wide Web Consortium Process Document - PR-w3c-process-971119*. Retrieved May 29, 2014, from <https://www.w3.org/Consortium/Process/Group/Draft12-971119>

⁹⁴ World Wide Web Consortium Process Working Group. (1998, March 11). 1.3 W3C's consensus policy. *World Wide Web Consortium Process (version March 11, 1998, revised on October 22, 2002)*. Retrieved November 2, 2014, from <http://www.w3.org/Consortium/Process/Process-19980311#Consensus>

社の仕様を統一した形での策定は困難であると判断され、この提案は却下されている⁹⁵。

DOM WG ではその代わりに、既の実装されている 2 つの仕様を元にした機能的な定義「Level 0」とする。Microsoft・Netscape 各社の次期バージョン(Internet Explorer 4/ Navigator 4)に実装を予定する機能を仕様としてまとめ上げたものを Level 1、Level 1 から発展させた仕様を Level 2 と定義し、Level 1 ならびに Level 2 の策定を WG の目標とする方針となった (図 19)⁹⁶。つまり、DOM は既存の分裂した仕様をベースにして策定されたものではなく、初めての仕様策定プロセスが先行する形の標準化となったのである。



図 19. Document Object Model の仕様策定手法

⁹⁵ Watson, A. & Wood, L. (1997, March). *Minutes of DOM working group 19970325*. Retrieved 15 April, 2014, from w3c-dom-wg@w3.org mailing list archive, <https://www.w3.org/DOM/Group/meetings/m19970325.html>

⁹⁶ Wood, L. (1997, May 13). *level zero, level one, level two*. Retrieved 15 April 2014, from w3c-dom-ig@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-ig/1997AprJun/0009.html>

6.5. Level 1: 市場競争の追認としての統一

1997年4月24日、MicrosoftのScott Isaacsによって、Level 0のEarly Draft⁹⁷が提出される⁹⁸。以降引き続き、MicrosoftとNetscapeによってLevel 0の策定が続けられていく。

Level 1の仕様書Editorは、ブラウザベンダではなくプログラミング言語であるJavaとJavaScriptの開発企業であるSun MicrosystemsのSteve Byrneが担った。

ウェブページを操作する側の言語は議論の大きな争点となった。Microsoftが最初期に提出したDraft⁹⁹はJScriptやVBScriptといったMicrosoftのプロプライエタリ言語を前提としたものであった。Netscapeが対抗して提出した仕様案¹⁰⁰も自社が推し進めるJavaScriptを前提としている。どちらの言語が採用されるかは、ブラウザの競争力に対して大きな影響力をもつ。

結局DOM仕様はFPWDから特定言語に依存しない仕様 (*Language neutral and platform independent*¹⁰¹) とされた。このような対立構造をCnetは以下のようにまとめている。

Microsoft claims its version of the DOM is more powerful and easier to work with than Netscape's, plus it's language-independent; it supports scripting languages other than JavaScript (notably, Microsoft's own JScript and VBScript).

Netscape counters that its version is "based on open standards and does not rely on ActiveX or VBScript to provide functionality," according to Daniel Klaussen, Netscape group product manager for Communicator¹⁰².

⁹⁷ Isaacs, S. (1997, April 24). *HTML Object Model - Level 0*. Retrieved May 30, 2014, from <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/att-0040/level0.html>

⁹⁸ Isaacs, S. (1997, April 24). *early draft of level 0*. Retrieved May 30, 2014, from [w3c-dom-wg@w3.org mailing list archive: https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0040.html](https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0040.html)

⁹⁹ Kindel, C. (1996, January 20). *INSERT and SCRIPT*. Retrieved June 1, 2014, from [html-erb@w3.org mailing list archive: https://lists.w3.org/Archives/Member/html-erb/1996JanMar/0006.html](https://lists.w3.org/Archives/Member/html-erb/1996JanMar/0006.html)

¹⁰⁰ Montulli, M. et al. (1996, August 22). *JavaScript-based Style Sheets*. Retrieved June 1, 2014, from <https://www.w3.org/MarkUp/Group/WD-jsss-960822>

¹⁰¹ Wood, L. & Sorensen, J. (Eds.) (1997, October 9). General Requirement. *Document Object Model Requirements - W3C Working Draft 09-October-97*. Retrieved June 4, 2014, from <http://www.w3.org/TR/WD-DOM/requirements-971009#General>

¹⁰² Corcoran, C. T. (1997, October 30). Netscape and Microsoft square off. *inside the W3C: how Web standard get made*. Retrieved June 3, 2014, from Internet archive: <http://wayback.archive.org/web/20010424015155/http://coverage.cnet.com/Content/Features/Dlife/Inside/ss02b.html>

DOM の標準化は仕様をモジュール化し、Requirement を含むコンポーネントごとに Editor や Author を分担して進められた。また、Level 0 と Level 1 の策定を同時並行で行うとともに、機能を Level 1 と Level 2 のどちらで含めるかという仕分けも同時に行われている。

Level 1 策定プロセスにおける作業分担は、Level 0/ Requirement/ Level 1 Core/ Level 1 XML/ Level 1 HTML の 4 つであった。Core は XML/ HTML/ XHTML で共通して用いる基礎的な機能、XML/ HTML はそれぞれ XML/ HTML に対応させた拡張仕様という位置付けである。なお Level 1 XML は、最終的に Level 1 Core へ統合されている。

Editor は、プロセスが進むに連れて増員されている。Sun Microsystems の Steve Byrne は、DOM 仕様の根幹である Core の Editor をプロセス開始当初から終了まで継続して担当している。DOM の標準化が円滑に進んだ要因としては、Microsoft が Softquad を巻き込んで作成した Draft をベースに議論を開始しつつも、Core の Editor を Sun Microsystems が担うというバランスのとれた人員配置があったのではないかと考えられる。

状況は意外な原因で変わる事となった。1998 年 3 月 31 日、Netscape はブラウザの次期バージョンである Communicator 5.0 のソースコードをオープンソース化することを発表する¹⁰³。その後 11 月 24 日には、Netscape は America on Line (AOL) に買収され、90 年代半ばに繰り広げられた Netscape と Microsoft によるブラウザ戦争は終結することとなった。

DOM Level 1 の仕様策定は順調に進み、1998 年 8 月 18 日に、PR に、10 月 1 日に DOM Level 1 は Recommendation となる。なお、WD から PR へとステータスを進めるためには、他の WG によるレビューが必要との指摘が Chair の Lauren Wood からなされている¹⁰⁴。当時は既に HTML や XML など、関連する複数の仕様策定が W3C において進められており仕様間の調整メカニズムが仕様策定プロセスにおいて機能し始めていることが伺える。

その後 Microsoft Internet Explorer が 1999 年 3 月にリリースしたバージョン 5.0

¹⁰³ Netscape Communications. (1998, March 31). Netscape accelerates communicator evolution with first release of next-generation communicator source code to developer community via Mozilla.org – Industry leading companies support bold move and will participate in early development of the source code. Retrieved June 5, 2014, from Internet archive: <https://web.archive.org/web/20021105061654/http://wp.netscape.com/newsref/pr/newsrelease591.html>

¹⁰⁴ Wood, L. (1998, July 6). W3C review of DOM draft. Retrieved June 24, 2014, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/1998JulSep/0013.html>

で仕様の一部に対応し、2000年7月リリースのバージョン5.5でようやくほとんどの部分を実装するようになった。策定プロセスでの貢献が殆ど見られなかったにも関わらず、Netscapeは2000年11月リリースのNetscape6でDOM Level 1を実装する。結果的にDOM Level 1はブラウザ戦争を戦う2つのブラウザが共に実装する実効性のある標準仕様となった。

とはいえ、標準仕様のブラウザへの実装は、提案者のMicrosoftですら標準化プロセス終了から1年以上経た後である。また、Netscape (AOL)も結果的に追随して実装している。言い換えれば、Microsoftが「実装を予定した」仕様が、市場競争によって追認されたということとなる。

表 7. DOM Level 1 の標準化動向

日時	事象
1995/12/4	Netscape Navigator 2.0 Beta3 公開。初の JavaScript を実装したブラウザ。
1996/1/20	Microsoft の Charlie Kindel が HTML-ERB メーリングリストに Internet Explorer のスクリプト言語対応仕様“INSERT and SCRIPT”紹介を HTML-ERB メーリングリストに投稿
1996/8/12	Microsoft Internet Explorer 3 公開。初の JScript を実装したブラウザ。
1996/8/19	Netscape が“JavaScript-based Style Sheets”仕様を HTML-ERB に提案
1996/9/9-10	HTML-ERB 会合にて、Microsoft の Scott Isaacs が標準仕様策定を提案
1996/9/15	Microsoft の Scott Isaacs が DOM の前身である仕様草案“HTML Object Model Overview”を HTML-ERB ML に投稿
1996/11/21	Netscape が ECMA International に JavaScript の標準化提案
1997/1/13	Microsoft の Scott Isaacs が HTML Object Model Activity Group の Charter Draft を投稿
1997/3/11	MS の Scott Isaacs が DOM 基本仕様群の draft を提出
1997/3/25-26	第 1 回 DOM WG F2F 会合。この会合で、Level 0/ 1/ 2 のマイルストーンを置くことが決定される。
1997/4/24	Level 0 の Early Draft 提出
1997/4/25	WG の公開メーリングリスト・公開ウェブサイトを設置
1997/6/27	DOM Level 1 Early Draft 提出
1997/10/9	DOM Level 1 FPWD 公開
1998/3/11	Process-ERB から初版の Process Document が外部公開
1998/3/31	Netscape Communicator 5.0 のソースコード公開
1998/8/18	DOM Level 1 PR
1998/10/1	DOM Level 1 Specification Recommendation
1998/11/9	Process Document に Last Call for Comment の段階が設定される
1998/11/24	Netscape、AOL に買収される
1999/3/18	Internet Explorer 5.0 release, DOM level 1 の一部機能を実装
1999/11/11	Process Document に実装主義が明記され、CR が設定される
2000/7/1	IE 5.5 DOM Level 1 のほとんどの機能を実装
2000/11/14	Netscape 6 DOM Level 1 を実装

6.6. 実装主義の導入

6.6.1. Last Call for Comment の導入

1998年11月12日版 Process Document¹⁰⁵からは、WDからPRに移行する前に、ChairによるTeam staffと他WGのChairに向けたLast Call for Comment (LC)が出されるというステップが追加されている。実質的にはLevel 1策定時に導入されていた方針であるが、Level 2からは明文化されたルールとして適用されるようになったのである。

6.6.2. Call for Implementation による実装主義の導入

1999年11月1日版 Process Document¹⁰⁶では、LCとPRの間にCandidate Recommendation (CR)の段階が追加されている。CRに進めるための要件は、以下の通りである。

Requirements for Entrance:

The Director must be satisfied that the Working Draft has successfully completed the Last Call with all comments resolved and that the Working Group has prepared an adequate implementation report.

Associated activities:

The Working Group requests implementation experience and uses this to refine the specification as necessary.

There is no requirement that a Working Draft have two independent and interoperable implementations to become a Candidate Recommendation. Instead, this is the phase at which the Working Group is responsible for formally acquiring that experience or at least defining the expectations of implementation¹⁰⁷.

¹⁰⁵ Jacobs, I. (ed). (1998, November 12). *World Wide Web Consortium Process Document (version November 12, 1998)*. Retrieved June 5, 2014, from <http://www.w3.org/Consortium/Process/Process-19981112>

¹⁰⁶ World Wide Web Consortium. (1999, November 1). *World Wide Web Consortium Process Document (version November 1, 1999)*. Retrieved June 5, 2014, from <http://www.w3.org/Consortium/Process/Process-19991101/>

¹⁰⁷ *ibid.* 6.2.3 Candidate Recommendations (CR). Retrieved June 5, 2014, from <http://www.w3.org/Consortium/Process/Process-19991101/tr#RecsCR>

つまり、CRに進むためにはWGが実装事例を提示しなくてはならず、CRステータス段階でWG非参加の会員から実装ベースでのレビューが求められるようになったのである(図20)。

この、TRのステータスがRecommendationに達する、つまり標準仕様が確定する以前に“Call for Implementation”が呼びかけられるという方針は、W3Cの標準化プロセスがもつ特徴である。多くの標準は先に仕様を確定させ、準拠する製品、サービスを開発、供給するという手順が踏まれてきた。しかしW3Cにおける標準化プロセスでは、ステークホルダーに議論中の仕様の実装が呼びかけられ、実装事例ベースで更に議論が積み重ねられた上で標準仕様が固めていくという手順となっている。この「仕様確定する以前に実装事例が供給される」という標準化プロセス運営ポリシー(図21)は、W3Cにおいて「実装主義(Implementation-oriented)」と呼ばれている¹⁰⁸。

	Proposal	ウェブなどで公開されている仕様提案。個人が公開する場合もあれば、任意で結成されたコミュニティにて策定されたものもある。W3Cをはじめとする標準化団体に持ち込まれていない段階のもの。	
	(Submitted) Draft	W3Cに対して提出された仕様案。ウェブで公開されることも多い。	
W3Cにおける標準化トラック内	Working Draft(s)	First Public Working Draft	WGにおける議論のたたき台として認められた仕様案。WG内で認証され、ChairからDirectorに提出された上で、Directorがアナウンスするという手順を踏む必要がある。
		Working Draft(s)	WG内で議論段階の仕様案。繰り返し改定される。
		Last Call for Comment	WG内での議論の最終段階として、期限を区切り(最低3週間)コメントを求める。寄せられたコメントに対する対応が全て終了した段階で次のステップに進むことができる。
	Candidate Recommendation	Call for Implementationが、内外に広く求められる状態。アナウンスはDirectorによってAdvisory Committee(AC)に対して行われる。つまり、全ての会員企業・組織に対し、仕様案を実装し、コメントするように呼びかけられることになる。	
	Proposed Recommendation	内外の実装事例をもとにブラッシュアップされた状態の仕様案。最低限2つ以上の相互可用性を満たした実装事例が揃った段階で、ACに対しCall for Reviewが出される。	
	Recommendation	ACのReviewを経た、確定標準仕様。	

図 20. Process Document によって定められた標準化プロセスにおける仕様書(Technical Report)のステータス

¹⁰⁸ この表現は、以下の資料などで表記されている。特に日本(W3C/Keio)において、仕様策定プロセスを説明する際に用いられることが多い。

例： Brewer, J. and Horton, S. (Eds.) (2002, October 11). *World Wide Web, Web Accessibility Initiative, Implementation Plan for Web Accessibility*. Retrieved November 24, 2013, from <http://www.w3.org/WAI/impl/expanded.html>

一色正男. (2010, December 1). *W3C (World Wide Web Consortium)へのご案内*. Retrieved November 24, 2013, <http://www.w3.org/2010/Talks/1206-echonet-mi/20101201EchonetForumpre.pdf> 等

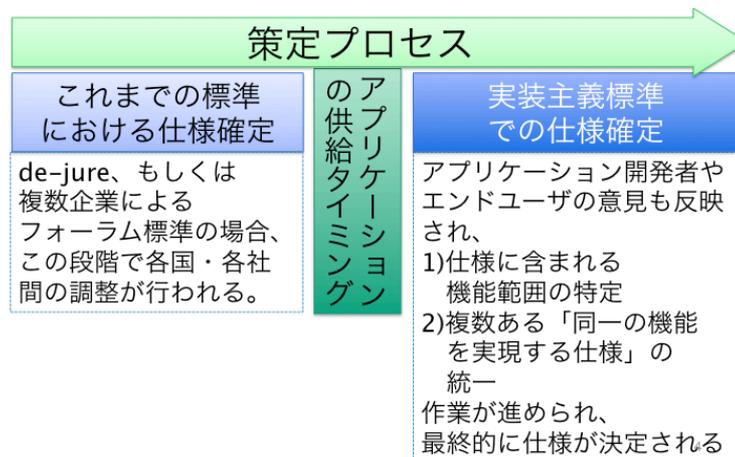


図 21. 実装主義プロセス

実装主義について、Process Document に戻り、もう少し詳細に分析したい。実装主義とは、既に実装事例が存在する仕様案のみが標準として採用される、という仕様策定上の方針である。Process Document において実装主義の論拠となっている条文は以下のものである。

Shown that each feature of the technical report has been implemented. Preferably, the Working Group should be able to demonstrate two interoperable implementations of each feature¹⁰⁹. (下線筆者)

この条文により、W3C で提案仕様が標準として採択されるためには、相互可用性を実現する複数の実装事例の提示が必要とされることになる。これは AC に対して、すなわち全会員企業・団体に対して呼びかけられるものであり、PR 段階で行われる Call for Review で寄せられた指摘や要望には全て応える必要がある。つまり W3C で標準として採択される仕様は、実装を踏まえた検討を経て全ての会員企業・団体から認められていることを意味する。

各 WG は必ずしも、全ての W3C 会員企業・組織が参加しているわけではない。WG は仕様提案を元にして発足するため、参加者は提案仕様に賛同する、もしくは提案仕様が業務・活動に深く関係する企業・団体が中心となる。PR の

¹⁰⁹ Jacobs, I. (ed.). (2005, October 14). 7.4.4 Call for Review of a Proposed Recommendation. *World Wide Web Consortium Process Document 14 October 2005*. Retrieved November 28, 2013, from <http://www.w3.org/2005/10/Process-20051014/tr.html#cfr>

段階で実装が呼びかけられるということは、呼びかけの対象は WG 非参加の企業・団体である。この手順が設定されることにより、より広いステークホルダーが仕様策定プロセスに関与することとなる。

6.6.3. 実装主義導入の意図

実装主義の導入意図について、W3C Chief Operating Officer である Ian Jacobs はインタビューに対し以下のように答えている。

we adopted a policy of requiring implementations before a special occasion would have been a recommendation. My recognition is sometime in 2000, 2004 or 2005. If you'll search you'll probably find it. So we didn't always have that policy that earlier recommendation is quick... become a recommendation before implementation. And what we discovered... There are several reasons why we adopted the change in our policy. The obvious one is like... Sometimes there are errors in a special occasion you don't realize until you actually go to implement. And you know there could be design flows or something... So implementation is testing, it's really about testing. And we are getting much more, I was getting much more regular about testing, our specification. So the candidate recommendation phase that we added our process was inexplicit decisions to make some level of testing, part of our required phase¹¹⁰.

実装主義の導入はテスト実施の効果を狙い、フィードバックを得るためであったと振り返っている。また、元 DOM WG の Chair も務めていた現 Interaction Domain Leader の Philippe Le Hégarret は、

That's 1997. So yes, in those time we only had 3 stage. Working Draft/ Proposed Recommendation, and then we had Recommendation. And then we call Last Call or call Recommendation. One of the major differences

¹¹⁰ 2012 年 8 月 14 日実施、Ralph Swick 氏インタビュー。

between the process that we are using back in the 1990s, and it seems then, there were more emphasis on implementations. So in the early 2000, for example, as we institution of DOM working group at the time, we added kind of a Recommendation phase. Because at the time DOM working group wanted to... they thought we believed the documentation is ready, but we'd like to show you it can be implemented. So we need a medium stage between Working Draft and Proposed Recommendation, where we can get feedback from the implementers. So that's one we added to Candidate Recommendation.¹¹¹.

と答えており、目的はフィードバックを得るためだったと明言している。そして、

Prior to that, getting feedback from implementers ~ was not priority, which of course create problems, because publishing specification no idea where they are going to be implemented on that. So, we added the Candidate Recommendation phase for that.

But in those days, it was very early on, and... For example here we say there was no requirements that Working Drafts as to be independent qualification to become a Proposed Recommendation¹¹².

とも話しており、CR 導入以前はフィードバックを受ける手段が限られていたことを問題していたのである。

6.7. Level 2: 実効性のある標準仕様策定成功

6.7.1. モジュールに分割されて進められた仕様策定

Level 1 が Core(/ XML)/ HTML という大きく 2 つのモジュールに分割されていたのに対し、Level 2 の Recommendation は Core/ Views/ Style/ Events/ Traversal-Range/ HTML の 6 つに分割されている。また、各モジュールは個別に仕様書として公開されている。また HTML は、途中から他の仕様書から完全に

¹¹¹ 2012 年 8 月 22 日実施、Philippe Le Hégarret 氏インタビュー

¹¹² *ibid.*

分割されたプロセスにて仕様策定が進められた。本論文では仕様の分割以降は、HTML5の先行仕様である Level 2 HTML に絞って分析を行う。

Level 2 の策定が本格的に開始されたのは、1998 年 8 月 25～26 日に開催された DOM WG の F2F 会合からである。会合 1 日目に、Level 1 と後方互換性を保つこと、モジュールに分割して策定を進めることが決められ¹¹³、各モジュールに対し Editor が割り当てられた (表 8)。Microsoft と Netscape は、それぞれ 2 つずつのモジュールを担当している。また、Events モジュールは、Microsoft と Netscape から 1 名ずつ Editor が派遣されている。なお、初期段階では HTML モジュールは存在しない。

一方 Level 2 では、モジュールごとに Draft の作成と議論を進め、その結果を WG 全体の電話会議にて報告し、モジュール間の調整を行うという手法で策定が進められた。

6.7.2. 実装主義導入と 2 社の実装

1999 年 11 月 1 日版 Process Document¹¹⁴ において実装主義が導入されたが、DOM WG においては先行して 1993 年 9 月 23 日版 Draft¹¹⁵ に対する Last Call for Comment が出されている。Last Call for Comment は IG¹¹⁶ のみならず、公開メーリングリストに対しても行われている¹¹⁷。その後 11 月 24 日の電話会議において、Last Call for Comment の実施を受けて Draft を CR に移行させるべきかについての投票が行われ、賛成：9、反対：1 という賛成多数で移行する方針が議決された¹¹⁸ものの、仕様策定プロセス上は、仕様書に含まれる全ての機能で実装事例が存在していないことから、CR から PR に即座にプロセスを進めていくことはできなかつた¹¹⁹。

¹¹³ World Wide Web Consortium. (1998, September 18). *DOM WG Minutes 19980825*. Retrieved June 24, 2014, from <https://www.w3.org/DOM/Group/meetings/m19980825.html>

¹¹⁴ World Wide Web Consortium. (1999, November 1). *World Wide Web Consortium Process Document (version November 1, 1999)*. Retrieved June 5, 2014, from <http://www.w3.org/Consortium/Process/Process-19991101/>

¹¹⁵ Wood, L. et al. (eds.) (1999, September 23). *Document Object Model (DOM) Level 2 Specification - Version 1.0 - W3C Working Draft 23 September, 1999*. Retrieved August 2, 2014, from <http://www.w3.org/TR/1999/WD-DOM-Level-2-19990923/>

¹¹⁶ Le Hors, A. (1999, September 23). *DOM Level 2 Last Call WD*. Retrieved 30 July 2014, from w3c-dom-ig@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-ig/1999Sep/0239.html>

¹¹⁷ Le Hors, A. (1999, September 23). *New DOM Level 2 Working Draft*. Retrieved 7 August 2014, from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/1999JulSep/0101.html>

¹¹⁸ Wood, L. (1999, November 24). *minutes from DOM teleconference 19991124*. Retrieved 30 July 2014, from w3c-dom-ig@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-ig/1999Nov/0132.html>

¹¹⁹ Le Hors, A. (2000, January 28). *DOM Level 2 Implementations*. Retrieved August 7, 2014, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/2000JanMar/0032.html>

表 8. DOM Level 2 仕様策定プロセス初期のモジュールと担当 Editor¹²⁰

モジュール	Editors
Range	Peter Sharpe, SoftQuad; & Vidur Apparao, Netscape
CSS	Jared Sorensen, Novell & Chris Wilson, Microsoft
Events	Tom Pixley, Netscape; & Chris Wilson, Microsoft
Queries & Iterators & Filters	Jonathan Robie, Texcel; & Joe Lapp, webMethods

しかし結果的には、Netscape からソースコードを引き継いだ Mozilla が 1999 年 12 月 22 日にリリースした¹²¹M12 build と、Microsoft が 1999 年 12 月 25 日にリリースした Internet Explorer 5.5 Beta に、既に Level 2 の一部機能を実装しており¹²²、主要ブラウザの最新バージョンへの実装が早期に進むという状態になった。

6.7.3. Call for Implementation への外部ステークホルダーの関与を進める動き

Call for Implementation は、Process Document 上は WG に参加していない W3C 会員企業・団体（具体的には代表者である AC Representative¹²³）に対して行われる（レビューを受け付ける）ものだが、WG では公開メーリングリストにも呼びかけて広く実装と、それに基づくレビューを受け付けることとなった¹²⁴。また、実際に実装事例の報告が寄せられている¹²⁵。

実装事例の報告を踏まえ、9 月 13 日には Chair メーリングリストと Director に対し、Draft ステータスの PR へ進める申請がなされる¹²⁶。

6.7.4. テストプログラム開発・供給の担い手における変化

6.7.4.1. 外部開発者コミュニティによるテストサービス

標準仕様の策定には、他のソフトウェアプログラムと同様に他の仕様との衝突などのバグや問題を抱えていないかについてチェックするためのテストプログラムの開発が必要である。標準仕様の場合、仕様そのものの問題を見つけ出

¹²⁰ *ibid.*

¹²¹ Mozilla の FTP サイトディレクトリ (Retrieved August 7, 2014, from <http://ftp.mozilla.org/pub/mozilla/releases/m12/>のファイル/フォルダ更新日より)

¹²² Gould, M. (2000, January 28). Exclusive Hover Buttons With DOM 2 Events. *webreview.com*. Retrieved August 7, 2014, from Internet archive:

<http://wayback.archive.org/web/20000229130836/http://www.webreview.com/pub/2000/01/28/dhtml/index.html>

¹²³ AC は W3C 会員企業・組織の代表者で構成される Advisory Committee の略称であり、運営諮問会議を指す。AC に出席する会員企業・組織の代表者は AC Representative (AC-rep) と呼ばれる。

¹²⁴ Wood, L. (2000, February 4). *Level 2 and moving towards PR*. Retrieved August 7, 2014 from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/2000JanMar/0074.html>

¹²⁵ Ball, S. (2000, April 3). *DOM Level 2 Implementation*. Retrieved August 7, 2014 from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/2000AprJun/0000.html>

¹²⁶ Wood, L. (2000, September 13). *request for publication of DOM Level 2 as Proposed Recommendation*. Retrieved August 7, 2014, from chairs@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/chairs/2000JulSep/0106.html>

すだけではなく、ウェブブラウザが仕様を正しく実装しているかをチェックするためのテストも必要となる。DOM Level 2 策定プロセスにおいては、テストプログラムの開発プロジェクトも、外部のウェブサイト開発者にオープンにされるに至った。

ウェブ標準に対するテストプログラムの提供は、Cascading Style Sheet (CSS) と呼ばれる、ウェブサイトのデザインを記述・制御する仕様において始まった。CSS はウェブサイトのデザインとコンテンツの分離というコンセプト¹²⁷の導入によって導入された仕様だが、新たな技術仕様を全てのブラウザ供給者が素早く実装できるわけではない。結果的に CSS の導入はブラウザ間の互換性実現にとって障壁となったのである。

そのような状況を打開すべく開発、提供されたのが Acid テストと呼ばれるものである。Acid テスト(Acid 1)(図 22)¹²⁸は CSS level 1 策定に関与した Invited expert¹²⁹の Todd Fahrer が開発・提供する、CSS level 1 が正しく表示されるかのテストページである。テストページが存在することで、各ブラウザ供給者がどの程度標準を正しく実装しているかが明らかとなる。テストページは、ブラウザ供給者が新たな標準仕様を迅速に、正しく実装することを促すために作成されたのである。

なお、CSS level 1 が Recommendation になったのは、1996 年 12 月 17 日¹³⁰であり、Acid 1 が公開されたのは 1999 年 1 月 26 日である。つまり標準化プロセス終了後、2 年以上経ても相互可用性が充分保たれている状態に至らない状況であった。

¹²⁷ コンテンツとデザインの分離については、付録 II を参照のこと。

¹²⁸ World Wide Web Consortium. (1999, January 29). *CSS1 Test Suite: 5.5.26 (Acid 1)*. Retrieved June 24, 2014, from <http://acid1.acidtests.org/>

¹²⁹ Fahrer は米国サンフランシスコに所在するウェブサイト製作会社 Studio Verso に勤務するデザイナーであり、W3C CSS WG などに Invited Expert の立場で参加した。(The Web Standard Project. (1999, October 4). Todd Fahrer. *Members of the CSS Samurai*. Retrieved June 24, 2014, from http://archive.webstandards.org/css/members.html#Todd_Fahrner)

¹³⁰ Håkon Wium, L. & Bos, B. (1996, December 17). *Cascading Style Sheets, level 1 - W3C Recommendation 17 Dec 1996*. Retrieved June 25, 2014, from <http://www.w3.org/TR/REC-CSS1-961217>

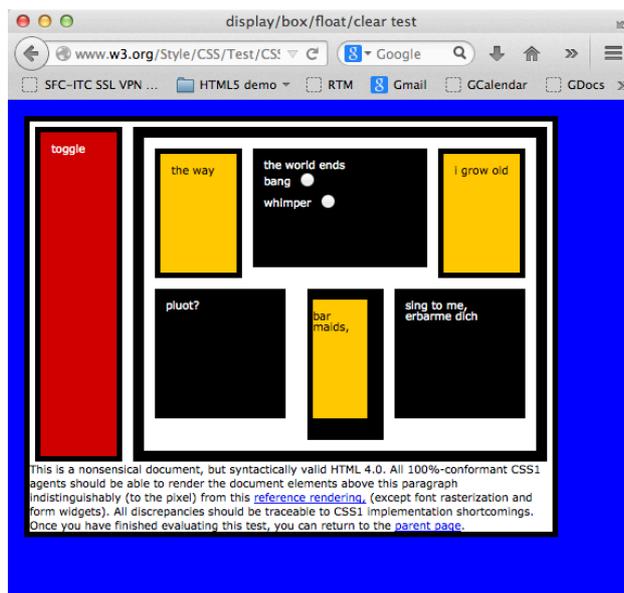


図 22. Acid 1 の結果画面 (仕様が正しく実装されているとこのように表示される)¹³¹

ウェブサイト制作者にとり、標準の実効性欠如に伴うブラウザ間相互可用性の欠如は、ブラウザごとに HTML ソースを書き換えるというコスト負担が生まれるため、忌避すべき事態である。

そのため、ブラウザ供給者に標準への準拠を求める開発者コミュニティ、The Web Standard Project (WaSP) が 1998 年に設立された¹³²。設立には Acid 1 を開発した Todd Fahrer も関与している。

WaSP の Mission Statement には、

Our goal is to support these core standards and encourage browser makers to do the same, thereby ensuring simple, affordable access to Web technologies for all¹³³.

と記載されており、ブラウザベンダ（設立当時は特に Netscape と Microsoft）に W3C 標準への準拠を迫ることを活動内容としている。

¹³¹ World Wide Web Consortium. (1999, January 29). *CSS1 Test Suite: 5.5.26 (Acid 1)*. Retrieved June 24, 2014, from <http://acid1.acidtests.org/>

¹³² The Web Standard Project. (n. d.) *History of the Web Standards Project*. Retrieved June 25, 2014, from <http://www.webstandards.org/about/history/>

¹³³ The Web Standard Project. (1998). *WaSP Baseline Standards Proposal*. Retrieved June 24, 2014, from <http://archive.webstandards.org/mission.html>

Todd Farhner は WaSP においても CSS Action Committee、通称 The CSS Samurai と呼ばれる CSS 仕様の普及と相互可用性実現のための活動グループの結成メンバーともなっている。なお The CSS Samurai のメンバーには、WHATWG 創設の中心人物であり HTML5 の Editor である Ian Hickson や、DOM の公開メーリングリストにおいて積極的にフィードバックを返している¹³⁴David Baron (6.2 項参照) などが含まれている¹³⁵。この 3 人を含むほとんどのメンバーは、当時ブラウザベンダを含む W3C 会員企業・団体に属していなかったが、その後 Ian Hickson は Opera から Google に、David Baron は Mozilla Foundation に就職・転籍するなど、現在でもウェブ標準仕様策定において影響力を発揮し続けている。

WaSP は単に声明を発するだけでなく、元々 W3C が提供していた Acid テストの後継バージョンの提供に乗り出した。それが Acid 2/ Acid 3 である。

Acid 2 は CSS level 2.1 styling と .png ファイル表示を主な対象とし、2005 年 4 月に公開された。Acid 3 は CSS のみならず、ECMAScript や DOM level 2 など、ウェブアプリケーション実行環境のための仕様も対象に含んでいる。

Acid 2/ Acid 3 は W3C の手を離れ、WaSP のクレジットにおいて Ian Hickson¹³⁶ が開発・公開している。

6.7.4.2. W3C による公式プログラムの開始

Acid 2/ Acid 3 は WaSP という、W3C 外部の開発者コミュニティによって開発、運営されてきたが、DOM level 2 の策定プロセスにおいて、W3C 自身によるテストプログラム開発プロジェクトが再開されることとなった。

2000 年になると、1997 年 3 月の F2F 会合から活動が開始された DOM WG も、活動期間が 3 年を超えたということで、WG と IG の Charter、DOM Activity Proposal の改定作業が始められている。改定された Charter には Quality Assurance の節が設けられ、以下の記述が加えられた。

The Working Group must also have a public test suite for the functionalities developed in the DOM Level 3 documents (see also Success

¹³⁴ フィードバックの一例として、以下の投稿を提示する。

Baron, L. (2000, January 6). *Comments on DOM2 (Core, HTML, CSS)*. Retrieved January 12, 2015 from [www-dom@w3.org Mailing list archive: https://lists.w3.org/Archives/Public/www-dom/2000JanMar/0008.html](https://lists.w3.org/Archives/Public/www-dom/2000JanMar/0008.html)

¹³⁵ The Web Standards Project CSS Action Committee. (1999, October 4). *Members of the CSS Samurai*. Retrieved June 25, 2014, from <http://archive.webstandards.org/css/members.html>

¹³⁶ 後に WHATWG を立ち上げ、WHATWG ならびに W3C において HTML5 仕様書の Editor に就任する。

criteria). This test suite must address at least the bindings embedded in the specifications.

External groups (like NIST) have been and are developing test suites for DOM. These external groups will work closely with W3C QA Activity to define common goals. The DOM WG will coordinate with both the QA activity and these groups for issues related to coverage, correctness and release of these suites¹³⁷.

Charter にこの記述が入ったことで、アメリカ国立標準技術研究所（National Institute of Standards and Technology, NIST）と共同運営でテストプログラム開発を行う試みが進められることとなった。そして2000年5月13日に、DOMのテストプログラムを運営していたNISTに所属するMary BradyがDOM IGに加入し¹³⁸、開発が本格的に進められることとなった。

6.7.5. 市場占有率による影響が抑制された状態での仕様の確立と普及

Level 2はMicrosoftとNetscapeという2大ブラウザベンダが協調して仕様策定を進めるかに見えた。

しかし、2001年になってMicrosoftの関与が薄くなり始める。2001年3月に、MicrosoftからDOM WGに参加し、主に発言していたJonathan MarshがWGから脱退する¹³⁹。また、定期的に行われているWGのF2F会合も、2001年4月24-25日の回を最後にMicrosoftからの参加は無くなった。

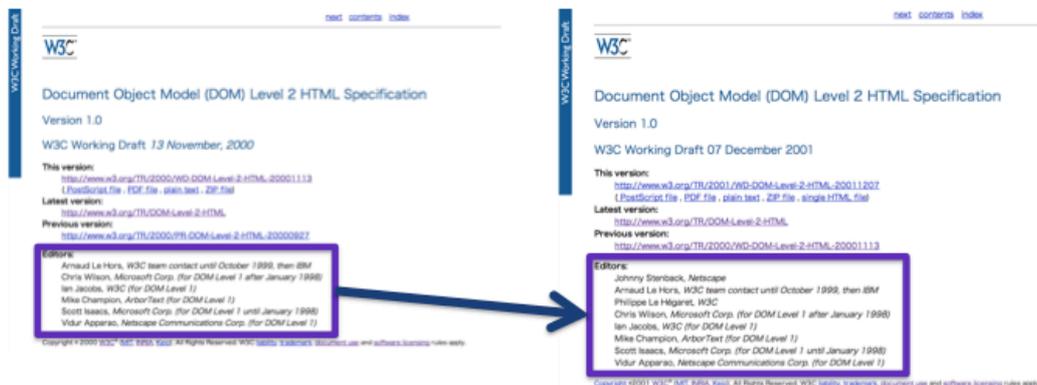
一方Netscapeは、2001年12月7日版のDOM Level 2 HTML (DOM2HTML) Working Draft¹⁴⁰からJohnny StenbackがEditorに加わるなど、積極的に策定プロセスに関与するようになる(図23)。Level 1はMicrosoftがDraft提案し、仕様策定プロセスをリードしてきたのに対し、Level 2はMicrosoftの関与が薄くなり、Netscapeが深く関与するかたちで進められた。

¹³⁷ Le Hégarret, P. (2000, November 20). *Quality Assurance. Document Object Model Working Group Charter - version November 20, 2000*. Retrieved August 7, 2014, from <https://www.w3.org/DOM/Group/wg-charter-2000.html#quality>

¹³⁸ Wood, L. (2000, March 14). *new DOM IG members*. Retrieved August 7, 2014 from w3c-dom-ig@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-ig/2000Mar/0046.html>

¹³⁹ Marsh, J. (2001, March 30). *Change of Microsoft representation*. Retrieved 15 April 2014, from w3c-dom-wg@w3.org mailing list archive, <https://lists.w3.org/Archives/Member/w3c-dom-wg/2001JanMar/0118.html>

¹⁴⁰ Stenback, J. et al. (2001, December 7). *Document Object Model (DOM) Level 2 HTML Specification Version 1.0 W3C Working Draft 07 December 2001*. Retrieved 23 April 2014, from <http://www.w3.org/TR/2001/WD-DOM-Level-2-HTML-20011207/>



2000年11月13日版のEditors欄では、Netscape所属は“for Level 1”と注記があるVidur Apparaoがいるのみだが、2001年12月7日版はJohnny Stenbackが筆頭に記載されている。

図 23. DOM Level 2 仕様策定への Netscape の関与

(左：2000年11月13日版 WD¹⁴¹、右：2001年12月7日版 WD¹⁴²)

DOM2HTML は2003年1月9日に Recommendation¹⁴³となるが、3名に絞られた Editor の一人を Netscape が輩出し、プレスリリース¹⁴⁴の推薦文も提供するなど、DOM2HTML 策定ならびに普及活動を積極的に行う姿勢であった。

DOM2HTML 実装は、Netscape が2000年11月にリリースした Netscape 6¹⁴⁵によって、次いで Microsoft が翌年である2001年8月にリリースされた IE バージョン 6¹⁴⁶において行った。ブラウザ市場において圧倒的なシェアを獲得していた Microsoft が追随して準拠した。つまり市場シェアの小さいブラウザベンダが主に貢献して策定された仕様が実効的な標準となったのである。

21世紀に入ると、Netscape、Microsoft 以外のブラウザベンダも存在感を見せ始める。ノルウェーの電話会社である Telenor の子会社としてスタートした Opera Software は、市場シェアとしては小さいものの、スペックの低いハードウ

¹⁴¹ Le Hors, A. et al. (2000, November 13). *Document Object Model (DOM) Level 2 HTML Specification Version 1.0 W3C Working Draft 13 November 2000*. Retrieved 23 April 2014, from <http://www.w3.org/TR/2000/WD-DOM-Level-2-HTML-20001113/>

¹⁴² Stenback, J. et al. (2001, December 7). *Document Object Model (DOM) Level 2 HTML Specification Version 1.0 W3C Working Draft 07 December 2001*. Retrieved 23 April 2014, from <http://www.w3.org/TR/2001/WD-DOM-Level-2-HTML-20011207/>

¹⁴³ Stenback, J. et al. (2003, January 9). *Document Object Model (DOM) Level 2 HTML Specification Version 1.0 W3C Recommendation 09 January 2003*. Retrieved 15 April 2014, from <http://www.w3.org/TR/2003/REC-DOM-Level-2-HTML-20030109/>

¹⁴⁴ World Wide Web Consortium. (2003, January). *Testimonials for DOM Level 2 HTML*. Retrieved 15 April 2014, from <http://www.w3.org/2003/01/doml2html-testimonial>

¹⁴⁵ Zakas, N. C. (2011). *Professional JavaScript for web developers 2nd edition*. John Wiley & Sons.

¹⁴⁶ Hammond, D. (2015). *DOM Level 2 HTML, Web Browser DOM Support*. Retrieved January 12, 2015, from <http://www.webdevout.net/browser-support-dom#dom2html>

エア上での動作に優れるブラウザを開発したり、ゲーム機等コンピュータ以外の機器向けのブラウザの採用事例を増やしたりといった形で利用を広げてきた。2001年12月にリリースした Opera6 では、DOM2HTML を含む DOM Level 1/2 仕様の一部を実装しており¹⁴⁷、DOM 仕様の普及は着実なものとなっていたのである。

表 9. DOM Level 2 の標準化動向

日時	事象
1998/8/25-26	DOM WG F2F 会合にて、Level 2 策定作業が開始される。また、モジュールごとの担当 Editor が任命される。
1998/11/25	Level 2 Editor's Draft
1998/12/17	DOM Level 2 FPWD
1999/11/11	Process Document に実装主義が明記され、CR が設定される
1999/12/22	Mozilla が DOM level 2 の一部機能を実装した M12 build を公開
1999/12/25	Microsoft が DOM level 2 の一部機能を実装した IE バージョン 5.5 Beta を公開
2000/4/25	DOM Level 2 PR
2000/9/27	DOM Level 2 仕様書を Core/ HTML/ XML に分割。それぞれ PR ステータスは維持。
2000/11/13	Level 2 Core Recommendation
2000/11/13	Level 2 HTML はステータスが WD に差し戻される
2000/11/14	DOM Level 2 を実装した Netscape 6 がリリースされる
2001/3/30	Microsoft の Jonathan Marsh が DOM WG から離脱。
2001/5/24・25	DOM WG F2F この回を最後に、MS の F2F 参加が途絶える。
2001/8/27	DOM Level 2 を実装した Internet Explorer 6.0 がリリースされる
2001/12/18	DOM Level 2 を実装した Opera 6.0 がリリースされる
2002/6/5	Level 2 HTML CR
2002/11/8	Level 2 HTML PR
2003/1/9	Level 2 HTML Recommendation

¹⁴⁷ Opera Software. JavaScript Environment: DOM and DHTML. *Web Specifications Supported in Opera 6*. Retrieved 15 April 2014, from <http://www.opera.com/docs/specs/opera6/#javascript>

第7章. Phase 3: 仕様案の競合を通じた運営方針の確立- HTML5/ XHTML

本章では 2014 年 10 月 28 日に完了した、HTML5 の仕様策定プロセスを分析する¹⁴⁸。これまで分析してきた事例との大きな違いは、W3C 内において競合仕様が先行して策定されていたことである。

HTML3.2/ 4.0 と DOM Level 1/2 は、独自仕様拡張の出現を抑制することが課題であり、競合仕様との競争に打ち勝つことではなかった。一方 HTML5 は、W3C 内において既に競合する仕様の策定と普及活動が進められており、仕様間競争を調整することが課題であった。また、ウェブアプリケーション実行環境としての機能が增大することにより、仕様策定に関与するステークホルダーの数と多様性が高まったことへの対応も必要とされた。

7.1. 2つの次世代 HTML コンセプト

1997 年末に HTML4.0 の標準化プロセスが完了する。その後、マイナーバージョンアップバージョンである HTML4.01 の策定作業が続くが、1997 年末で現行 HTML の策定作業の大勢が決したといえる。W3C 設立時にあったデザインとコンテンツの分離という技術的理想がひとまず達成されたといえる。そして次世代の HTML がどのようなコンセプトになるのかについて議論が始まった。そこで出てきた 2 つの方向性が、HTML ファイルに埋め込まれた情報をアプリケーションで処理できるようにするというもの (XHTML) と、サーバサイドで動作するプログラムのインターフェイスとしての機能を充実させるというものがあった。最終的にこの 2 つの方向性が集約され、HTML5 という仕様が確立することとなる。

7.1.1. XHTML: アプリケーションで処理できるデータを埋め込める HTML

XHTML のコンセプトは図 24 によって示される。図 24 はあるイベントの告知を内容とするウェブサイトであり、開催日時や内容といった情報が含まれる。従来の HTML であれば、スケジューラーに日程を追加する際は文字列をコピー・ペーストするか、目で読み取った情報を転記するよりほかなかった。これ

¹⁴⁸ 3.3 節にあるように、HTML5 の策定プロセスを分析対象としており、HTML5.1 については除外している。

を、アプリケーションがHTMLファイルを読み込めば当該情報が記述された箇所がイベント情報であると判別でき、インポートして正しく処理できるようにしようとするものである。この機能の応用範囲は広く、レストランガイドにある店舗評価や価格帯などの情報が、検索サービスの結果画面に整形されて表示される(図 25)など、多様なプログラムでの活用が可能となる。

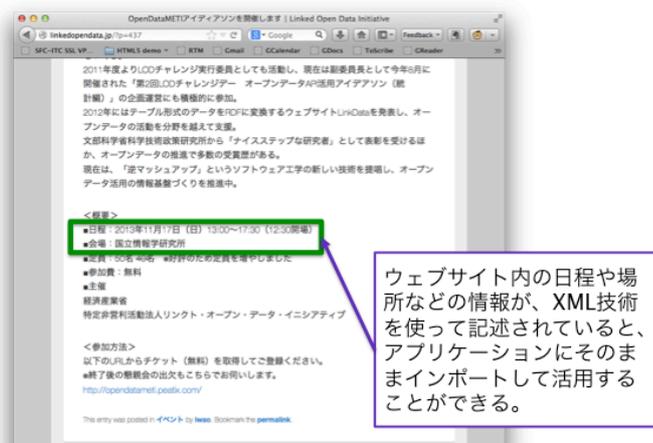


図 24. XML ベースで HTML を再開発するメリット

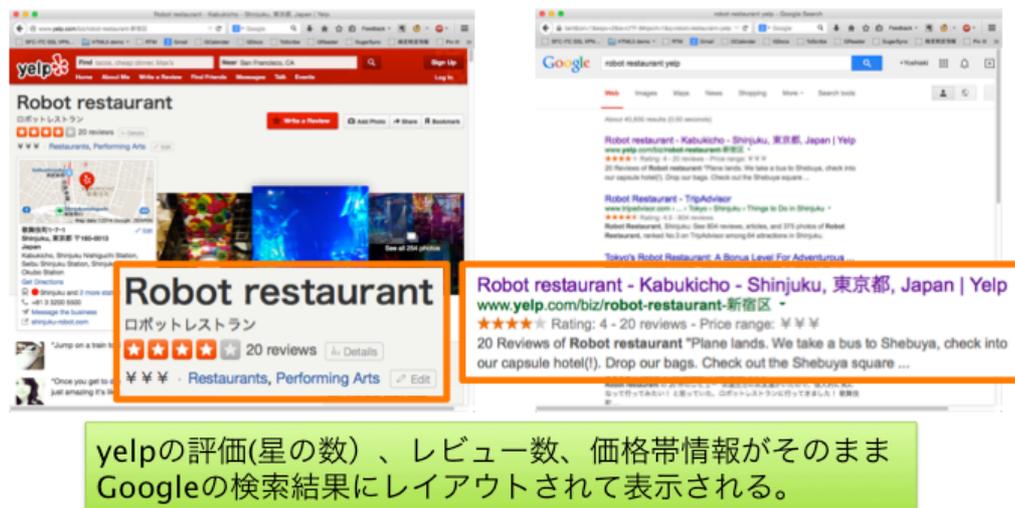


図 25. HTML 中の情報をプログラム処理した例¹⁴⁹

¹⁴⁹ 左：Yerp. (n. d.). *Robot restaurant*. Retrieved November 20, 2014, from <http://www.yelp.com/biz/robot-restaurant-新宿区>

右：Google. (n.d.) Search result of “robot restaurant yelp”. Retrieved November 20, 2014, from <https://www.google.co.jp/search?q=robot+restaurant+yelp>

しかし、HTML の内容をプログラム処理できる形式にするということは、HTML の記述方法を根本的に変更するということである。すなわち、後方互換性を失うこととなる。その上に、プログラム処理を前提とした場合、とにかくブラウザで表示できればよいという前提で HTML ファイルを作成するよりも、記述に厳密性が求められる。XHTML というコンセプトは、ウェブサイト制作者に労力を強いるという側面があったのである。

7.1.2. HTML5: 既存仕様と後方互換性を保ちつつ、アプリケーションとしての機能を充実させることを主眼においた仕様

もう一つの次世代 HTML 仕様である HTML5 は、当初 Web Application 1.0 と Web Forms 2.0 という 2 つの別々の仕様として提唱された。

Web Application 1.0 は、ローカルデバイスが取得したデータを取り込むための API を中心とした仕様である。GPS チップが生成する緯度経度データ、カメラが生成する画像／動画データなど、これまでブラウザからはアクセスできなかったデータを取り込み、サーバに受け渡すことで、ウェブアプリケーションの機能を拡張することができる。

Web Forms 2.0 は、HTML4 以前から存在した Form と呼ばれる機能の拡張版である。Form とは、ウェブサイトの閲覧者がサーバにデータを送信するために HTML 中に設置される要素である (図 26)。HTML という仕様において、Form は早期のバージョンから実装されているが、ウェブアプリケーションにおいては、ユーザがサーバに送信するデータが多様になるため、機能の強化が求められる。

XHTML との違いは、既存の HTML をベースにして機能を増やすため、後方互換性が確保できているという点だ。そのため、ウェブサイト制作者に今まで以上のコストを掛けずに、ウェブアプリケーションという新たな機能を実現することができるようになる。

2 つのアーキテクチャの違いを図 27 にまとめた。XHTML が後方互換性を犠牲にしても W3C 創設時から共有されていた HTML/ウェブの理想像を実現しようとしていたのに対し、HTML5 は後方互換性を保持したままウェブアプリケーションというイノベーションの実現を狙ったものであった。また、サーバとデバイスの役割が異なることからわかるように、アーキテクチャ自体が異なっている。



図 26. ウェブサイト(HTML)内に設置された Form の例

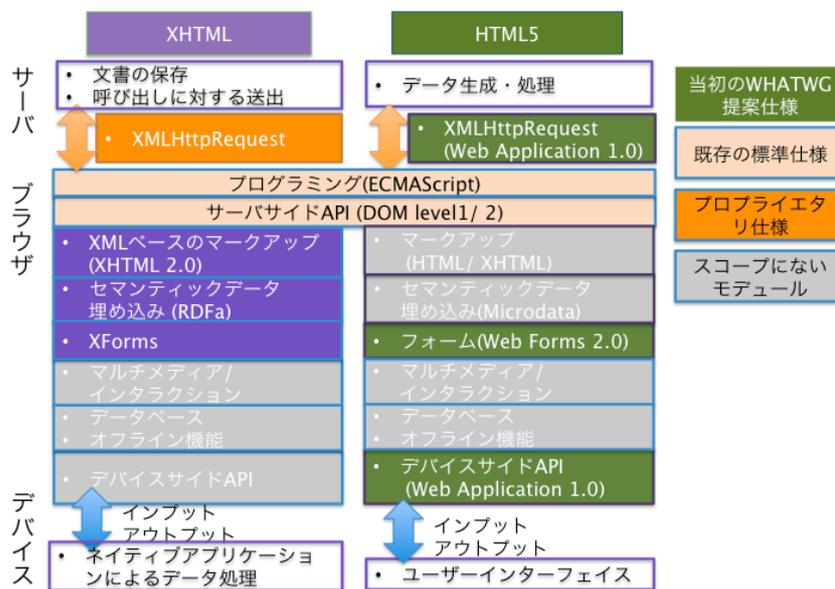


図 27. 当初の XHTML と HTML がスコープに入れていた機能、ならびにアーキテクチャの差異

7.2. クローズドな投票による HTML5 仕様提案の棄却

7.2.1. Workshop 決議による XHTML 開発開始

1998年1月18日、W3C Team staffの Dave Raggett は、“XML extensions to HTML”と題したメモ¹⁵⁰を発表した。このメモに記述されていたコンセプトが、XHTMLとして開発されることとなる。メモのタイトルにある XML は Extensive Markup Language と呼ばれる「プログラムで処理するデータ交換に適した言語」仕様のことである。XML 技術をベースに、ウェブサイト制作のための言語である HTML

¹⁵⁰ Raggett, D. (1998) *XML extensions to HTML*. Retrieved January 24, 2014, from <http://www.w3.org/People/Raggett/xml-in-html.html>

を作り直すことにより、ウェブサイトの中にプログラム処理に適したデータを埋め込むことができるようにしたのである。

XHTML は SGML WG¹⁵¹ で策定された XML 技術を用いることから、WG 間の連携をはかるために XHTML Coordination Group (CG)¹⁵²¹⁵³¹⁵⁴が設置された。

XML1.0 の仕様策定作業が終了した¹⁵⁵翌日である 2 月 10 日に、“XML in HTML” Coordination Meeting という Hypertext CG の F2F 会合が開かれる。

Hypertext CG には HTML WG、XML WG の他に、関係する仕様開発を進める多様な WG の関係者が参加した¹⁵⁶。会議ではそれぞれの仕様策定経験を基にして、複数の手法が提案、検討された¹⁵⁷。

CG での議論を踏まえ、次期 HTML 仕様開発プロセスを始めるために開催されたのが Shaping the Future of HTML と題された Workshop である。当時の Process Document において Workshop は、W3C 会員間で技術やポリシーに関する意見交換を行う、もしくは特定の論点に関する関心を会員間に醸成するために開催される会合と規定されている¹⁵⁸。

この Workshop には当時の主要な W3C 会員企業・団体が揃い、HTML の次期バージョンの開発を中止し、XHTML を次世代の HTML として新規開発する方針を決定したのである¹⁵⁹。

7.2.2. HTML5 の提案と否決

HTML5 が最初に標準化提案されたのは 2003 年 9 月、ノルウェーに拠点を置

¹⁵¹ SGML という仕様、ならびに HTML、XML との関係については付録 II を参照されたい。

¹⁵² XHTML 策定開始当時の Process Document における CG の役割は、WG 間の調整と規定されていたが、後に W3C と外部標準化団体との間の調整が追加される。現在 CG は関係 WG の Chair、外部標準化団体との連絡窓口 (Liaison) を務める外部専門家 (Invited Expert) らで構成される。

¹⁵³ XHTML 策定開始当時の Process Document の該当部分 : Cargill, C. et al. (1998, March 11). 3.3.3 - Coordination Groups. *World Wide Web Consortium Process (version March 11, 1998)*. Retrieved July 19, 2014, from <http://www.w3.org/Consortium/Process/Process-19980311#GroupsCG>

¹⁵⁴ 2014 年 11 月時点の Process Document の該当部分: Jacobs, I. (ed.) (2005, October 14). 6.3 Coordination Groups. *World Wide Web Consortium Process Document - 14 October 2005*. Retrieved July 19, 2014, from <http://www.w3.org/2005/10/Process-20051014/groups#GroupsCG>

¹⁵⁵ Bray, T. et Al. (1998) Extensible Markup Language (XML) 1.0 W3C Recommendation 10-February-1998. Retrieved February 26, 2014, from <http://www.w3.org/TR/1998/REC-xml-19980210>

¹⁵⁶ 参加者リストを含む会議の概要 <https://www.w3.org/MarkUp/CoordGroup/9802/xml-in-html> (2014 年 1 月 23 日参照)

¹⁵⁷ XML in HTML Meeting Report <http://www.w3.org/TR/1998/NOTE-xh-19980511.html> (2014 年 1 月 23 日参照)

¹⁵⁸ Cargill, C. et al. (1998, March 11). 3.4.1 - Workshops. *World Wide Web Consortium Process (version March 11, 1998)*. Retrieved July 19, 2014, from <http://www.w3.org/Consortium/Process/Process-19980311#EventsW>

¹⁵⁹ "Shaping the Future of HTML" ウェブサイト <http://www.w3.org/MarkUp/future/> (2014 年 1 月 23 日参照)

くブラウザベンダである Opera Software (Opera)によってであった。Opera は W3C に対し “XHTML Module: Extensions to Form Controls”¹⁶⁰ と題した仕様を提案する¹⁶¹。この仕様は翌年 2 月には Web Forms 2.0 と名称が変わる¹⁶²。2.0 と名付けられているのは、HTML 4 の Form 機能の発展版という位置付けであるからである。

名称に“XHTML”が含まれていることからわかるように、この提案は XHTML に対抗して行われたものではなかった。ではなぜこのような提案がなされ、それが後に HTML5 と名称が変わって XHTML との間で激しく次世代標準の座を争うこととなってしまったのだろうか。

仕様の提案者であり、後に W3C において HTML5 仕様の Editor を務める Ian Hickson は、インタビューに対し以下の通りに答えている。

*Back in 2003, when XForms was going through its final stages (the "Proposed Recommendation" vote stage), the browser vendors were concerned that it wouldn't take off on the Web without being made a part of HTML*¹⁶³

*out of that big discussion (which unfortunately is mostly hidden behind the W3C's confidentiality walls) came a proof of concept showing that it was possible to take some of XForms' ideas and put them into HTML 4.*¹⁶⁴

Hickson はこの提案を、当時 Opera は既に自社ブラウザに XHTML を実装している¹⁶⁵が、XHTML ならびに XHTML に対応した Form 仕様である XForms が

¹⁶⁰ Hickson, I., *XHTML Module: Extensions to Form Controls Opera Working Draft, September 2003* Retrieved February 1, 2014, from <https://lists.w3.org/Archives/Member/w3c-archive/2003Sep/att-0014/hfp.html>

¹⁶¹ XHTML Module: Extensions to Form Controls from Ian Hickson on 2003-09-05 (w3c-archive@w3.org from September 2003) <https://lists.w3.org/Archives/Member/w3c-archive/2003Sep/0014.html> (2014年2月1日参照)

¹⁶² Hickson, I. *Web Forms 2.0 Working Draft 5 February 2004*. Retrieved February 26, 2014, from <http://www.hixie.ch/specs/html/forms/web-forms-2>

¹⁶³ Lawson, B. (2009, May 13). *Interview with Ian Hickson, editor of the HTML 5 specification*. Retrieved January 29, 2014, from <http://www.webstandards.org/2009/05/13/interview-with-ian-hickson-editor-of-the-html-5-specification/>

¹⁶⁴ *ibid.*

¹⁶⁵ 当時の Opera ブラウザの最新バージョンは Opera 7 であり、このバージョンから XHTML に対応していた。(Opera Software. (2003). *Web Specifications Supported in Opera 7*. Retrieved February 24, 2014, from <http://www.opera.com/docs/specs/opera7/>)

仕様策定の最終段階に差し掛かっているのに関わらず普及の目処がたたないことを懸念しており、既存の HTML 仕様に XForms のコンセプトが適用可能であることを示そうという意図で行った、と語っている。

W3C における Form 機能向上のための試みは、1999 年に XHTML に対応した Forms 仕様が HTML WG に提案されたことにより始まった¹⁶⁶。HTML4.01 以前では HTML 仕様書の中に含まれていた Form を、独立した仕様として開発することとなった。それが XForms である。

2002 年に HTML4.01 を XML 技術ベースに置き換えた XHTML1.0 が、2003 年には XForms1.0 も Recommendation となった。この段階でプロプライエタリな技術も併用すれば、ウェブアプリケーションを構築することはある程度可能な状態となった(図 28)。しかし、肝心の XHTML と XForms が普及する目処が立っていなかった。

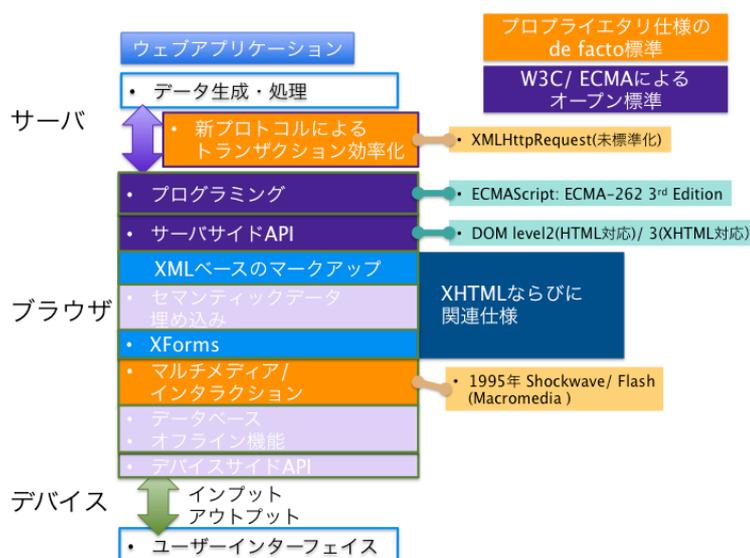


図 28. 2003 年時点におけるウェブアプリケーション実現のための機能開発・標準化動向

¹⁶⁶ w3.org ドメイン内に保存されている最も古い XHTML 対応 Form 仕様の要件定義書として、Schnitzenbaumer, S. et Al. (1999, April 22). *XHTML Forms 1.0 Requirements*. Retrieved March 4, 2014, from <https://www.w3.org/MarkUp/Group/1999/forms-req-19990422/> が存在する。

これら2つの仕様は既存のHTMLに対する後方互換性がなく、ウェブサイト制作者の支持を受けることができなかった。その上当時最大シェアを占め、ほぼ寡占状況にあった Microsoft Internet Explorer (IE)¹⁶⁷ ¹⁶⁸が XHTML を実装できていなかった¹⁶⁹¹⁷⁰。その理由として Microsoft の IE 開発担当の Chris Wilson は IE のセキュリティ対策と CSS 対応を優先していることを挙げている¹⁷¹。IE が XHTML に対応するのはバージョン9からであり、そのリリースは2011年である。

この結果、XHTML1.0が Recommendation となつてから7年後の2007年の時点ですら、全ウェブサイト¹⁷²に占める XHTML の割合は20%¹⁷³に過ぎなかった¹⁷⁴。この20%はブログ等の Contents Management System によって自動生成されるページも多いと推測されるため、多くのウェブサイト制作者は XHTML ではなく HTML でウェブサイトを構築し続けるという状況が続いたのである。

この状況は、静的なコンテンツをウェブサイトとして公開するだけであれば大きな問題とはならない。当時最大の市場シェアをもつ IE バージョン6 (IE6) が実装している HTML4.01 の機能さえあれば、充分だからである。しかし、動的なウェブアプリケーションを提供しようとするならば大きな問題である。なぜならば、IE6 を用いている多くのエンドユーザにとって、ウェブアプリケーションを使おうと思わない限りは HTML4.01 しか実装していない IE6 を用いても大きな問題とはならない。つまり、高度ウェブアプリケーションを実現するための環境(XForms など)がいつまでたっても普及しない状況となつたのである。

¹⁶⁷ 1998年にNetscapeはAmerica Onlineに買収されるとともに、Navigatorのソースコードを無償公開している。この年にMicrosoftとNetscapeとの間のブラウザ戦争は実質的に終了し、その後ブラウザ市場はMicrosoft Internet Explorerが支配する状態が続くこととなる。

¹⁶⁸ Net Applications社の統計によると、HTML5の開発がWHATWGにて開始された2004年第4四半期のMicrosoft Internet Explorerの市場占有率は91.4%に達していた。

¹⁶⁹ 具体的にはapplication/xhtml+xmlを処理する機能が実装されていなかった。

¹⁷⁰ W3Cは2006年3月9日に各ブラウザのXHTML対応をまとめているが、Mozilla Firefox、Opera、Apple Safariの最新バージョンはapplication/xhtml+xml、text/xml、application/xmlの3つのXMLメディアタイプに全て対応しているのに対し、Microsoft Internet Explorerは当時の最新バージョンである7.0 Beta 2も完全には対応できていない。(Ishikawa, Y. (2006). *XHTML media type test - results*. Retrieved February 1, 2014, from <http://www.w3.org/People/mimasa/test/xhtml/media-types/results>)

¹⁷¹ Chris Wilson 「The <?xml> prolog, strict mode, and XHTML in IE」『IEBlog』
<http://blogs.msdn.com/b/ie/archive/2005/09/15/467901.aspx> (2013年8月15日参照)

¹⁷² ここでは、HTMLとXHTMLファイルを指す。

¹⁷³ ここでのXHTMLの定義は、名前空間としてxmlns="http://www.w3.org/1999/xhtml"としてあるサイトである。

¹⁷⁴ HTML5策定作業のコミュニケーションチャネルであるWHATWGのIRCチャネルにおけるWHATWG/W3C HTML WGのEditorであるIan Hicksonの発言(<http://krijnhoetmer.nl/irc-logs/whatwg/20070704#l-230>)から。この発言に関する日本語の解説は矢倉眞隆「統計に見るXHTMLの普及状況」『Web標準ブログ』(<http://standards.mitsue.co.jp/archives/001230.html>)。

一方 W3C の HTML WG¹⁷⁵では、XML ベースで HTML4.01 の機能を実現するというコンセプトで開発が進められた XHTML1.0 の策定作業終了後、XHTML2.0 の開発を開始した。

XHTML2.0 は仕様書中に

*it is not intended to be backward compatible with its earlier versions.*¹⁷⁶

と記述されているなど、後方互換性を犠牲にして XML ベースの機能向上を目指すという方針で開発されている。後方互換性の欠如は、XHTML2.0 の普及の大きな阻害要因となる。

当時のアーキテクチャ(図 28)において、最も大きなボトルネックとなったのは Form であった。ウェブアプリケーションの実行環境はサーバサイドにあるものの、クライアントサイドからのデータ入力機能が充実することが非常に重要である。

2003 年当時、サーバサイド API である DOM は、HTML に対応した Document Object Model (DOM) Level 2 HTML Specification Version 1.0¹⁷⁷ が Recommendation になっていたのに対し、Form 仕様は XML 対応の XForms のみが開発されたていた¹⁷⁸。つまり、ウェブアプリケーションを推進したい層にとり、最もクリティカルに必要とされていたのが既存の HTML に後方互換性がある高機能な Form 仕様であった。

XForms が XHTML に依存していることがウェブアプリケーション普及の障壁になっているという状況を打開するには、2つの方向性が考えられる。1つは XHTML を普及させること。しかしこれまでに述べてきたとおり、この試みは当時成功しているとはいえなかった。もう1つの方向性は既存の HTML と互換性のある Form 仕様を標準化することである。

繰り返し述べている通り、ウェブアプリケーションは複数のモジュールが連携して動作することによってはじめて成立する。そのため、機能実現に必要な

¹⁷⁵ 2007 年に立ち上げられ、HTML5 の策定にあたった現行の HTML WG とは別の WG である。

¹⁷⁶ McCarron, S. et Al. (August, 2002). Introduction. *XHTML™ 2.0 W3C Working Draft 5 August 2002*, Retrieved March 1, 2014, from http://www.w3.org/TR/2002/WD-xhtml2-20020805/introduction.html#s_intro

¹⁷⁷ Stenback, J., et Al. (2003, January 9) *Document Object Model (DOM) Level 2 HTML Specification version 1.0*, Retrieved February 23, 2014, from <http://www.w3.org/TR/DOM-Level-2-HTML/>

¹⁷⁸ 2003 年 10 月 14 日に、XForms 1.0 が Recommendation (Dubinko, M., et Al. 2003, October 14) *XForms 1.0 W3C Recommendation*, Retrieved February 23, 2014, from <http://www.w3.org/TR/2003/REC-xforms-20031014/> となっている。

仕様に準拠しないモジュールが使われている場合、エンドユーザはサービスを利用することができない。つまり、各モジュールを提供するプレイヤーが、皆準拠する標準仕様が必要となる。

具体的には、主要なブラウザ¹⁷⁹全てが実装する標準仕様を策定しなければならない。ウェブ技術の標準化団体として最も大きな影響力をもち、ほぼ全てのステークホルダーが参画している W3C において採択され、主要ブラウザと主要なウェブアプリケーションで共通して実装される、つまり実効性のある標準仕様を創り上げる必要があるのだ。

既に XHTML を実装していることから分かる通り、Opera は XHTML 技術の導入にはどちらかといえば積極的な企業であった。しかし、XHTML の普及は進まず、ウェブアプリケーションの実現には遠い状況が続いていた。このような状況を打破すべく開発・提案されたのが HTML5 であった。

7.2.3. Workshop であらためて XHTML が選ばれる

Web Forms 2.0 は XForms と同様の機能を提供する競合仕様である。この仕様を W3C 内で策定しようとするならば、全体の同意を取り付けなければならない。Web Forms 2.0 推進派がその機会として選んだのが、2004 年 6 月に開催された Workshop on Web Applications and Compound Documents であった。

Process Document における Workshop の規定は、既に外部の専門家や関心のある実務者も含めて技術やポリシーについて意見交換する場に移行しており¹⁸⁰、参加対象者は W3C 非会員も可能となっている¹⁸¹。

Compound Documents とは、XML、MathML¹⁸²、Scalable Vector Graphic (SVG)¹⁸³ など XML ベース言語の複数フォーマットを混在させる環境¹⁸⁴を意味する。つまりこの Workshop のテーマは、XML ベースの多様なプログラム処理に適したデータの流通と、それを処理するウェブアプリケーションのための仕様開発であり、既存の XHTML + XForms を中心とした仕様開発という路線に則った議論で

¹⁷⁹ 2003 年当時は、Microsoft Internet Explorer、Mozilla Firefox、Opera、Apple Safari の 4 つ。

¹⁸⁰ Jacobs, I. (ed.) (2004, February 5). 9 Workshops and Symposia, *World Wide Web Consortium Process Document - 5 February 2004*. Retrieved July 19, 2014, from <http://www.w3.org/2004/02/Process-20040205/events.html#GAEvents>

¹⁸¹ Jackson, D. (2005, January 13). Expected Audience, *W3C Workshop on Web Applications and Compound Documents*. Retrieved July 19, 2014, from <http://www.w3.org/2004/04/webapps-cdf-ws/#Expected>

¹⁸² 数式を表現するためのマークアップ言語

¹⁸³ 図形描画のためのマークアップ言語であり、XML をベースに開発されている。

¹⁸⁴ World Wide Web Consortium (2004). *Compound Document Formats (CDF)*. Retrieved February 26, 2014, from <http://www.w3.org/2004/CDF/>

あった¹⁸⁵。

Opera は、この Workshop に対し Mozilla と共同で Position paper を提出している¹⁸⁶。その内容には以下のポイントが含まれる。

- 1) 後方互換性を確保し、移行への明確なプロセスを提示する
- 2) 洗練されたエラー対処機能を有する
- 3) エンドユーザが直面するウェブアプリケーション・コンテンツ開発者が生み出したエラーの影響は可能な限り少なくする
- 4) オープンプロセスで開発・策定する

これらのポイントは、開発者のみならずエンドユーザにも受け入れられやすいものであり、二面ある市場双方を意識した内容といえる。ただし、Workshop のテーマである Compound Documents とは全く異なるコンセプトの提案に対する投票結果は、賛成 8 対反対 11 に終わり、提案は却下された¹⁸⁷。W3C 内での方針への同意取り付けは失敗に終わったのである。

7.3. 市場動向に後押しされての HTML5 策定プロセス開始

7.3.1. W3C 外部におけるオープンな仕様開発活動

Workshop が開催された 2 日後の 2004 年 6 月 4 日、Web Forms 2.0 仕様書の Editor を務める Ian Hickson が、WHATWG (Web Hypertext Application Technology Working Group) を設立することを発表した。

Hickson が設立当日に公開した、外部のエンジニアに対する開発参加を広く呼びかける文書¹⁸⁸ では、1) Web Forms 2.0、2) Web Apps (Applications) 1.0、3) Web Controls 1.0 という 3 つの仕様を WHATWG では開発するとしている。

2004 年 8 月の Hickson からの WHATWG のメーリングリストに対する投稿では、

¹⁸⁵ Jackson, D. (2004). *The W3C Workshop on Web Applications and Compound Documents*. Retrieved February 26, 2014, from <http://www.w3.org/2004/04/webapps-cdf-ws/>

¹⁸⁶ Hickson, I. (2004). *Position Paper for the W3C Workshop on Web Applications and Compound Documents*. Retrieved February 1, 2014, from <http://www.w3.org/2004/04/webapps-cdf-ws/papers/opera.html>

¹⁸⁷ Jackson, D. (2005, January 13). Summary. *W3C Workshop on Web Applications and Compound Documents*. Retrieved February 1, 2014, from <http://www.w3.org/2004/04/webapps-cdf-ws/summary>

¹⁸⁸ Hickson, I. (2004, June 4). *WHAT open mailing list announcement*. Retrieved February 27, 2014, from <http://www.whatwg.org/news/start>

The intention is that what we have informally been calling "HTML5" is what is specified as HTML4 + Web Forms + Web Apps. So it is intended to be a part of HTML5.¹⁸⁹

と記載されており、初めて“HTML5”という名称が登場する。WHATWG が創設され HTML5 の開発が始まったことにより、2 つの競合する次世代 HTML 仕様が W3C の内外で同時に開発されるという事態が始まったのである。

WHATWG は、以下の様な文言で自らを定義している。

The WHATWG is a growing community of people interested in evolving the Web. It focuses primarily on the development of HTML and APIs needed for Web applications.

The WHATWG was founded by individuals of Apple, the Mozilla Foundation, and Opera Software in 2004, after a W3C workshop. Apple, Mozilla and Opera were becoming increasingly concerned about the W3C's direction with XHTML, lack of interest in HTML and apparent disregard for the needs of real-world authors. So, in response, these organisations set out with a mission to address these concerns and the Web Hypertext Application Technology Working Group was born¹⁹⁰.

この文言の中にあるポイントは2つある。1つは、WHATWG を“a growing community of people”と定義しており、自らを標準化団体ではなく開発者の「コミュニティ」として位置づけている点。もう1つは、“founded by individuals of Apple, the Mozilla Foundation, and Opera Software”と、Apple、Mozilla、Opera の3社の関与があることは示しているものの、あくまでも「個人」の集合体によって形成されたコミュニティであることを強調している点である。なぜわざわざ「個人」の集合体であることを強調したのであろうか。

WHATWG で策定された仕様書の著作権に関する議論の際、仕様書の著作権をパブリックドメインにするという提案に対し、Ian Hickson が以下の通りに回

¹⁸⁹ Hickson, I. (2004, August 3). [whatwg] Web Applications Markup Language 1.0 is an XUIL? Retrieved February 24, 2014, from whatwg@whatwg.org Mailing list archive: <http://lists.w3.org/Archives/Public/public-whatwg-archive/2004Aug/0019.html>

¹⁹⁰ WHATWG. (2008). FAQ: What is the WHATWG? version 22th April 2008. Retrieved March 3, 2012, from http://wiki.whatwg.org/index.php?title=FAQ&oldid=3152#What_is_the_WHATWG.3F

答している。

*I suggested it, but was told that it was better to have a copyright supporting a liberal license than to have it in the public domain. One reason was, if I recall correctly, that if we later wanted to submit this to a standards organisation, and they wanted to own the copyright (as W3C would, e.g.), then we wouldn't be able to if we had assigned it to the public domain.*¹⁹¹

ウェブアプリケーションのアーキテクチャが、サーバ・ブラウザ・デバイスという複数のコンポーネントが連携して動作して始めて成立するコンセプトであり、主要なステークホルダーが参加する標準化団体でオーソライズされる必要がある。また CSS など W3C で策定されている仕様との関係性を調整する作業も不可避である。そのため、WHATWG は策定する仕様書を W3C に持ち込むことを想定して活動していたのである。W3C に仕様案を持ち込むことを目標に据えるならば、WHATWG は W3C に競合する標準化団体として捉えられるのは得策ではない。

また、XHTML2.0 の Editor に Opera の Jonny Axelsson が加わっており¹⁹²、Mozilla においても XHTML2.0¹⁹³ と XForms¹⁹⁴ を Firefox に実装するプロジェクトが開始されている。

このため、「企業・団体によって構成される標準化団体」である W3C とは異なる、「個人によって形成された開発者コミュニティ」という定義を採用したのである。

WHATWG における仕様策定で大きな影響力をもつのは Ian Hickson だが、メーリングリストでは Opera、Mozilla、Apple に在籍しない外部の一般エンジニアからの意見の吸い上げを積極的に行っている(図 29)。

¹⁹¹ [whatwg] Copyright of specifications

<http://lists.whatwg.org/htdig.cgi/whatwg-whatwg.org/2004-August/002071.html>

¹⁹² McCarron, S. et Al. (August, 2002). *XHTML™ 2.0 W3C Working Draft 5 August 2002*, Retrieved March 1, 2014, from <http://www.w3.org/TR/2002/WD-xhtml2-20020805/introduction.html>

¹⁹³ Mozilla Foundation. *Bug 161463 - (xhtml2) XHTML 2.0 tracking*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_bug.cgi?id=161463

¹⁹⁴ Mozilla Foundation. *Bug 97806 - (xforms) Implement W3C XForms in browser and composer*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_bug.cgi?id=97806

つまり、既に標準化プロセスが進んでいる XHTML と XForms 仕様策定を押し進めるグループと調整を進める前提で、提案は受理されたのである。

2005年11月15日にW3Cは、Web Application Formats WGを設立する¹⁹⁸。このWGは、XML-based User-interface Language (XUL) や eXtensible Binding Language (XBL) などのMozillaがMozilla Application Frameworkと呼ばれるプロジェクトを通じて開発した (Boswell, 2002) 仕様を標準化することを目的としていた¹⁹⁹。OperaはXHTML 2.0を策定するHTML WGやXForms WGではなく、このWGにWeb Forms 2.0を持ち込むことにした²⁰⁰²⁰¹。

7.3.3. ブラウザ供給者とコミュニティ主導のコンセンサス形成

W3Cでは、年に一度Team staffと仕様策定者が一同に介して会議を行うW3C Technical Plenary / Advisory Committee Meetings Week (TPAC)と呼ばれる年次全体総会を開催する。その全体会議においてHTML5に関するセッションが開かれた。それは、HTMLファイルの中にアプリケーション処理可能なデータを埋め込むための仕様であるMicroformatsをテーマとしたパネルである²⁰²。Microformatsは、HTML5に含まれるMicrodataの元となった仕様である。

このパネルはGoogleのIan Hickson、OperaのHåkon Wium Lie、Technoratiに所属しながらMicroformatsの仕様策定を進める草の根コミュニティを主催するTantek Çelik、W3C Team StaffのDan Connollyらがパネリストを務めた。

Çelikは元々Microsoftに在籍し、The W3C Workshop on Web Applications and Compound DocumentsにおいてMicrosoftのPosition Paperを執筆した後にXHTML1.0の策定に貢献する²⁰³など、W3Cの標準化プロセスに長期に渡り関与しているエンジニアである。Çelikはこの後Stanford Universityの修士課程に復学。

¹⁹⁸ Lesch, S. (2005, November 15). *Join the Web Application Formats Working Group (Call for Participation)*. Retrieved March 5, 2014, from w3c-ac-members@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-ac-members/2005OctDec/0036.html>

¹⁹⁹ Jackson, D. (2005, November 15). *Scope, Web Application Formats Working Group Charter*. Retrieved March 5, 2014, from <http://www.w3.org/2006/appformats/admin/charter#scope>

²⁰⁰ Operaは初期のWeb Controls 1.0もこのWGに持ち込み、Widget 1.0という名称で標準化プロセスを進めた。(van Kesteren, A. and Caceres, M. (2006, November 9). *Widgets 1.0 - W3C Working Draft 9 November 2006*. Retrieved March 5, 2014, from <http://www.w3.org/TR/2006/WD-widgets-20061109/>)

²⁰¹ Widget 1.0はその後名称を変更したものの、Recommendationとなっている。(Caceres, M. (2011, September 27). *Widget Packaging and XML Configuration - W3C Recommendation 27 September 2011*. Retrieved March 5, 2014, from <http://www.w3.org/TR/2011/REC-widgets-20110927/>)

²⁰² microformats.org (2008, December 20). *Microformats panel at W3C Plenary Day*. Retrieved March 5, 2014, from Microformats Wiki: <http://microformats.org/wiki/events/2006-03-01-w3c-plenary-microformats>

²⁰³ Pemberton, S. et Al. (2002, August 1). Acknowledgements. XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition) - A Reformulation of HTML 4 in XML 1.0 - W3C Recommendation 26 January 2000, revised 1 August 2002. Retrieved March 3, 2014, from <http://www.w3.org/TR/xhtml1/#acks>

microformats.org などウェブ標準仕様開発・普及促進を目的とするコミュニティを運営しながら 2009 年に修士号を取得²⁰⁴。2010 年 5 月からは Mozilla Foundation に所属²⁰⁵するなど、HTML5 の実装・普及活動の中心人物の一人となる。

TPAC 全体総会において HTML5 に関するテーマのセッションが、W3C Team Staff も参加して行われたということは、W3C 内部において大きなインパクトのあることであった。

2006 年 4 月 5 日 van Kesteren は Web Application Formats WG のメーリングリストに、Opera 社内でこの WG に Web Forms 2.0 の仕様書案を持ち込むことについて検討していることを投稿する²⁰⁶。

同年 8 月 16 日 WG Chair の Arthur Barstow は、Chairs Mailing List に、Web Forms 2.0 を Web Application Formats WG の FPWD として公開することを申請し²⁰⁷、即日受理されたのである²⁰⁸。

更に 2006 年 5 月 19 日 W3C Team Staff の Ian Jacobs は、AC mailing list に XForms Activity を HTML Activity に統合するとともに、後方互換性のある XHTML1.0 の後継仕様を策定する XHTML1 WG を設立するという趣旨の提案を投稿する²⁰⁹。

この申請の受理、また XHTML 標準仕様策定 WG の再編案に対しては XHTML2.0 + XForms 開発の中心人物である IBM の John Boyer を中心に激しい反発が生じた²¹⁰²¹¹²¹²。8 月 29 日には、AC のメーリングリスト宛に IBM が正式な反対声明を投稿する²¹³。このような激しい反対が生じることが予測されたにもか

²⁰⁴ Stanford University Computer Science Department (2009). *Masters Alumni*. Retrieved March 3, 2014, from http://web.archive.org/web/20090831131632/http://www-cs.stanford.edu/People/masters_alumni

²⁰⁵ CNET News. (2000, May 26). *Mozilla hires open-standards guru Celik*. Retrieved March 3, 2014, from http://news.cnet.com/8301-13577_3-20005987-36.html

²⁰⁶ van Kesteren, A. (2006, April 5). *Submission: Web Forms 2*. Retrieved March 5, 2014, from member-appformats@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/member-appformats/2006Apr/0001.html>

²⁰⁷ Barstow, A. (2006, August 16). *Transition Request: 1st publication of Web Forms 2.0*. Retrieved from chairs@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/chairs/2006JulSep/0038.html>

²⁰⁸ Bratt S. (2006 August 16). *Re: Transition Request: 1st publication of Web Forms 2.0*. Retrieved March 5, 2014, from chairs@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/chairs/2006JulSep/0040.html>

²⁰⁹ Jacobs, I. B. (2006, May 19). *Draft Charters for Renewal of HTML Activity (Advance Notice)*. Retrieved March 5, 2014, from w3c-ac-members@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-ac-members/2006AprJun/0048.html>

²¹⁰ Boyer, J. (2006 August 16). *Re: Transition Request: 1st publication of Web Forms 2.0*. Retrieved from chairs@w3.org mailing list: <https://lists.w3.org/Archives/Member/chairs/2006JulSep/0041.html>

²¹¹ Boyer, J. (2006 August 31). *XForms, Web Forms 2.0 and the future of XML content on the web Smarter Everyone*. Retrieved March 6, 2014, from IBM developerWorks, Smarter Everything, Smarter Everywhere: <https://www.ibm.com/developerworks/community/blogs/JohnBoyer/date/200608?lang=en>

²¹² 他にも chairs@w3.org mailing list において、2006 年 8 月 16 日から 23 日にかけて、感情的な内容のやりとりが繰り返された。(World Wide Web Consortium. (2006). *chairs@w3.org from July to September 2006: by date*. Retrieved March 6, 2014, from <https://lists.w3.org/Archives/Member/chairs/2006JulSep/>)

²¹³ Boyer, J. (2006 August 29). *IBM Position Statement on XForms and Web Forms 2.0*. Retrieved March 6, 2014, from w3c-ac-forum@w3.org mailing list:

かわらず、なぜ W3C は、一度は否決した HTML5 の仕様提案を受理し、WG の再編を進めたのであろうか。その背景には HTML5 と XHTML、それぞれの実装事例普及状況がある。

7.3.3.1. XHTML の普及状況

既に XHTML1.0 が Recommendation となって 6 年以上、XHTML1.1 が Recommendation となって約 5 年を経過しているのにも関わらず(表 10)、XHTML と XForms の普及は進んでいなかった。その状況が如実に示されたのが、2006 年 3 月に W3C Team の石川雅康によって発表された 2 つのテスト結果である。

表 10. 2006 年 5 月時点における XHTML/ XForms 標準化プロセス進捗状況²¹⁴²¹⁵

仕様名	最新版仕様のステータス	最新版仕様書公開日
XHTML 1.0	Recommendation	26 January 2000 ²¹⁶
XHTML 1.0 (2nd edition)	Recommendation	1 Aug 2002 ²¹⁷
XHTML 1.1	Recommendation	31 May 2001 ²¹⁸
XHTML 2.0	Working Draft	27 May 2005 ²¹⁹
XForms 1.0	Recommendation (2nd edition)	14 Mar 2006 ²²⁰
XForms 1.1	Working Draft	9 December 2005 ²²¹

<https://lists.w3.org/Archives/Member/w3c-ac-forum/2006JulSep/0018.html>

²¹⁴ Michel, T. and Ishikawa, M. (2006, June 16). *XForms Working Group Roadmap*. Retrieved March 5, 2014, from <http://www.w3.org/MarkUp/Forms/Group/xforms-roadmap.html>

²¹⁵ Pemberton, S. and Klotz, L. (2013, April 4). Schedule of Deliverables. *The Forms Working Group Public Website*. Retrieved March 5, 2014, from <http://www.w3.org/MarkUp/Forms/#deliverables>

²¹⁶ Pemberton, S. et Al. (2000, January 26). XHTML™ 1.0 The Extensible HyperText Markup Language - A Reformulation of HTML 4 in XML 1.0 - W3C Recommendation 26 January 2000. Retrieved March 5, 2014, from <http://www.w3.org/TR/2000/REC-xhtml1-20000126/>

²¹⁷ Pemberton, S. et Al. (2002, August 1). XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition) A Reformulation of HTML 4 in XML 1.0 W3C Recommendation 26 January 2000, revised 1 August 2002. Retrieved March 5, 2014, from <http://www.w3.org/TR/2002/REC-xhtml1-20020801/>

²¹⁸ World Wide Web Consortium. (2001, May 31). World Wide Web Consortium Issues XHTML 1.1 and Ruby Annotation as W3C Recommendations - Two New Specifications Deliver Enhanced Modularity and Internationalization Support. Retrieved March 5, from <http://www.w3.org/2001/05/xhtml1-ruby-pressrelease.html>

²¹⁹ Axelsson, J. (2005, May 27). *XHTML™ 2.0 W3C Working Draft 27 May 2005*. Retrieved March 5, 2014, from <http://www.w3.org/TR/2005/WD-xhtml2-20050527/>

²²⁰ Boyer, J. M. et al. (eds.). (2006, March 14). *XForms 1.0 (Second Edition) - W3C Recommendation 14 March 2006*. Retrieved October 29, 2014, from <http://www.w3.org/TR/2006/REC-xforms-20060314/>

²²¹ Boyer, J. M. (ed.). (2005, December 9).. Retrieved October 29, 2014, from

XHTML media type test²²²は、当時流通しているほぼ全てのブラウザに対し XHTML ならびに HTML ファイル(application/xhtml+xml、text/xml、application/xml という3つのメディアタイプを宣言したファイル)を読み込ませ、正しくファイル形式を認識してエンドユーザに提示できるかをテストしたものである。

もう1つのテストは XHTML entities test²²³である。こちらは XHTML 仕様書で定められた“&”、“√”、“ä”のような特殊文字や記号が、ブラウザで正しく表示されるかをテストしたものである。

2つのテストの結果²²⁴²²⁵は、Mozilla Firefox、Opera、Apple Safari の最新バージョン(当時)が application/xhtml+xml、text/xml、application/xml の3つの XML メディアタイプを正しく認識し、記号を表示できるのに対し、Microsoft Internet Explorer は当時の最新バージョンである 7.0 Beta 2 であっても完全には対応できていないというものだった。当時 Internet Explorer は 8 割程度の市場シェアを獲得しており²²⁶、XHTML の機能を利用できるブラウザの普及率は 2 割以下ということになる。

また、当時策定途上である XHTML2.0 については、Firefox²²⁷、Opera²²⁸、Apple²²⁹ がともに実装しない方針を示していた。つまり、XHTML2.0 は普及の見通しが全く立っていない状態であった。

実装主義を採用する W3C において XForms 1.0 が Recommendation となってい

²²² Ishikawa, Y. (2006). *XHTML media type test*. Retrieved February 6, 2014, from <http://www.w3.org/People/mimasa/test/xhtml/media-types/>

²²³ Ishikawa, Y. (2006). *XHTML entities test*. Retrieved February 6, 2014, from <http://www.w3.org/People/mimasa/test/xhtml/entities/>

²²⁴ Ishikawa, Y. (2006). *XHTML media type test – results*. Retrieved February 1, 2014, from <http://www.w3.org/People/mimasa/test/xhtml/media-types/results>

²²⁵ Ishikawa, Y. (2006). *XHTML entities test – results*. Retrieved February 6, 2014, from <http://www.w3.org/People/mimasa/test/xhtml/entities/results>

²²⁶ Wikipedia の *Usage share of web browsers* (Retrieved March 6, 2014, from http://en.wikipedia.org/wiki/Usage_share_of_web_browsers)の項目にまとめられている統計結果によると、*XForms 1.1 - W3C Working Draft 09 December 2005*IE の市場シェアは、84.0% (2006 Q2, Net Applications)、86.3% (2006 Q2, TheCounter.com)、86.6% (June 2006, WebSideStory)。

²²⁷ Bugzilla@Mozilla (プロジェクト管理システム)に XHTML2.0 実装のプロジェクトが追加されているが実装には至らず、2010年3月13日にプロジェクトは終了した。

(参照: Mozilla Foundation. *Bug 161463 - (xhtml2) XHTML 2.0 tracking*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_bug.cgi?id=161463.)

Mozilla Foundation. *History - Changes made to Bug 161463 - (xhtml2) XHTML 2.0 tracking*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_bug.cgi?id=161463)

²²⁸ Opera の CTO である Håkon Wium Lie はインタビューにおいて

“it’s unrealistic to think that all web authors will switch to an XML-based syntax which demands that browsers stop processing the document on the first error.” (Braun, H. (January, 2007) *Interview mit Håkon Wium Lie, English version*. Retrieved from http://www.heise.de/ct/Redaktion/heb/w3c/lie_english.html)と回答し、XHTML2.0 の実装予定がないことを表明している。

²²⁹ Stachowiak, M. (2007, January 17). *HTML Standards Process Returning from the Grave*. Retrieved from Surfin! Safari: <https://www.webkit.org/blog/89/html-standards-process-returning-from-the-grave/>

るのは、XForms はもともと Lotus Forms (現 IBM Forms) と呼ばれる帳票管理ソフトで用いられる技術仕様を標準化したものであり、Lotus Forms には当然実装されていた²³⁰こと、Mozilla の公式アドオン²³¹や FormFaces というほとんどのブラウザに対応したサードパーティ製アドオン²³²も供給されていたという背景がある。しかし、XForm はそもそも XHTML を前提とする技術であるため、ブラウザ本体にはほとんど実装が進まなかったのである。

7.3.3.2. WHATWG HTML5 (Web Forms 2.0 + Web Applications 1.0) 実装状況

一方 Web Forms 2.0 のブラウザへの実装は早速 Web Forms 2.0 を提案した Opera によって実行されている。2006 年 6 月にリリースされた Opera バージョン 9 は、Web Forms 2.0 と Web Application 1.0 を不完全な状態とはいえ実装した。また、Web Application 1.0 に含まれる CANVAS²³³は、もともと Apple が Safari バージョン 1 に実装していた技術であり、Mozilla も Firefox1.5 から CANVAS と client-storage を実装済みである²³⁴。WHATWG を創設した 3 つのブラウザベンダは揃って HTML5 の自社ブラウザへの実装を進めている。

XMLHttpRequest は、元々は Microsoft が Internet Explorer に実装したプロプライエタリな技術であった。しかし、Mozilla が 2001 年に²³⁵、Apple が 2004 年に²³⁶、Opera が 2005 年に²³⁷自社ブラウザに実装し、全ての主要ブラウザに実装されるデファクト標準となった²³⁸。

²³⁰ Boyer, J. (2007, January 29). *Workplace Forms and Next Generation Web Applications*. Retrieved March 6, 2014, from Internet Archive, IBM developerWorks Blogs:

<https://web.archive.org/web/20070219112402/http://www-03.ibm.com/developerworks/blogs/page/JohnBoyer>

²³¹ Mozilla XForms という Firefox 用アドオン(<https://addons.mozilla.org/ja/firefox/addon/mozilla-xforms/>) が提供されている。

²³² 2014 年 3 月時点では、OSS として提供されている。(http://sourceforge.net/projects/formfaces/)

²³³ JavaScript を用いて描画する機能。

²³⁴ Dumbill, E. HTML の将来、パート 1 : WHATWG
<http://www.ibm.com/developerworks/jp/xml/library/x-futhtml1/>

²³⁵ Mozilla Developers Network. (2012). *XML Extras*. Retrieved February 24, 2014, from
https://developer.mozilla.org/en/docs/XML_Extras

²³⁶ Apple (2005). *Dynamic HTML と XML:XMLHttpRequest オブジェクト*, Developer Connection. Retrieved February 28, 2014, from Internet Archive:

<https://web.archive.org/web/20080514053848/http://developer.apple.com/jp/internet/webcontent/xmlhttpreq.html>

²³⁷ Opera8 に実装。(Opera Software, (2012), *Opera バージョン履歴*. Retrieved February 24, 2014, from
<http://www.opera.com/docs/history/>)

²³⁸ その後 W3C に持ち込まれ、2006 年に FPWD が公開(van Kesteren and Jackson. (2006). *The XMLHttpRequest Object W3C Working Draft 05 April 2006* Retrieved February 24, 2014, from
<http://www.w3.org/TR/2006/WD-XMLHttpRequest-20060405/>) されて以来、2014 年 1 月においても標準化プロセスが進められている (2014 年 1 月 30 日に公開された W3C の Working Draft: van Kesteren et Al. (2014) *XMLHttpRequest Level 1 W3C Working Draft 30 January 2014* Retrieved February 24, 2014, from
<http://www.w3.org/TR/2014/WD-XMLHttpRequest-20140130/>).

2006年5月は、XHTML2.0が普及の見通しが立たない一方、HTML4.01と後方互換性のあるXHTML1.0とHTML5の実装が広がるという状況であった。

W3Cは政府・国際団体から独立した民間組織であり、強制力をもって標準仕様を普及させることはできない。策定した仕様が普及しない状態が続くことは、標準化団体としての正統性を既存することにつながる。そのためW3Cは、普及の見込みが立たないXHTML、特にXHTML2.0のみをHTMLの後継仕様として開発し続ける状況を脱し、HTML4.01と後方互換性のある仕様策定を行う体制を構築する方針に傾き始めたのである。

最終的に2006年10月27日、W3C DirectorのTim Berners-Leeは自身のブログでHTMLの新バージョン開発を始める旨を表明し²³⁹、その約1か月後の11月21日にW3CとしてのWG再編案を正式に対外的に発表する²⁴⁰。再編案はBarstowの要望に則した内容となっていた。翌日の2006年11月22日、W3C TeamはACに対して再編案に対するCall for reviewを出した^{241,242}。

7.4. 更なるプロセスのオープン化とロイヤルティ・フリー・ポリシーの確立

7.4.1. 新HTML WGの活動内容と目標設定

再編案に対しては、WHATWGメンバーから別途コメントが出されている。2006年11月21日にIan Hicksonが出したコメントは、

** Regarding technical matters, there shouldn't be a difference between being a working group member as a W3C Member Company, a W3C Invited Expert, or participating as a non-W3C Member. This should be made explicit in the charter; currently the charter implies that non-members are not full members of the working group.*

** There should not be any secret mailing lists, even for administrative purposes.*

²³⁹ Berners-Lee, T. (2006, October 27). *Reinventing HTML*. Retrieved March 6, 2014, from timbl's blog: <http://dig.csail.mit.edu/breadcrumbs/node/166>

²⁴⁰ Lilley, C. (2006, November 21). *HTML and Forms Activity Proposals*. Retrieved March 6, 2014, from <http://www.w3.org/2006/11/HTML-Forms-recharter-background>

²⁴¹ Jacobs, I. B. (2006 November 22). *HTML and Forms Activities Proposals (Call for Review)*. Retrieved March 6, 2014, from w3c-ac-members@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-ac-members/2006OctDec/0054.html>

²⁴² World Wide Web Consortium. (2006 November 22). *Call for Review: HTML and Forms Activities (closed since 19 January 2007)*. Retrieved March 6, 2014, from <https://www.w3.org/2002/09/wbs/33280/2006-HTML-Forms/>

** The public mailing list should be www-html@w3.org, so as to not lend the appearance of attempting to ignore the existing community.*

** There should not be teleconferences or face-to-face meetings, as most participants could not afford to attend such meetings. There could be the option of annual meetings, open to all participants, with some of the less well financed working group members being sponsored by the other members to attend, but such meetings would have to be informal, with no actual decision-making ability, so as to prevent the non-attending members from feeling disenfranchised. (The existing community numbers in the hundreds; teleconferences and face-to-face meetings are unmanageable at such numbers anyway.)²⁴³*

といったものである。標準化プロセスを公開し、W3C 非会員でも、資金力がないエンジニアでも参加できるようにするというオープンプロセスの採用を求めている。これに対し、Charter 案に Chair 候補として名前が記載されている Microsoft 所属の Chris Wilson は、W3C HTML WG と WHATWG が連携して活動する方針であることをブログ²⁴⁴で表明している。

7.4.2. 新 HTML WG 発足と体制固め

2007 年 3 月 7 日に、W3C によって次期 HTML を策定するワーキンググループの立ち上げと参加者募集のプレスリリース²⁴⁵が発表され、新 HTML WG の活動が開始された。そして早速体制固めの活動が始まった。

2007 年 4 月 9 日に、Apple の Maciej Stachowiak が Apple、Mozilla、Opera 三社連名で、活動方針案をメーリングリストに投稿している²⁴⁶。その内容は、1) WHATWG において策定されている HTML5 (つまり、Web Forms 2.0 と Web

²⁴³ Hickson, I. (2006 November 21). *Re: Charters for review*. Retrieved March 6, 2014, from [w3c-html-cg@w3.org mailing list archive:](https://lists.w3.org/Archives/Member/w3c-html-cg/2006OctDec/0171.html)

<https://lists.w3.org/Archives/Member/w3c-html-cg/2006OctDec/0171.html>

²⁴⁴ You, me and the W3C (aka Reinventing HTML) - Albatross! - Site Home - MSDN Blogs

<http://blogs.msdn.com/b/cwilso/archive/2007/01/10/you-me-and-the-w3c-aka-reinventing-html.aspx>

²⁴⁵ W3C プレスリリース「HTML 標準の更新に着手 HTML の今後を方向付ける Web ブラウザ事業者や開発者ら」：<http://www.w3.org/2007/03/html-pressrelease.html.ja> (2013 年 5 月 4 日アクセス)

²⁴⁶ Stachowiak, M. (2007, April 9). *Proposal to Adopt HTML5*. Retrieved May 15, 2013, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/0429.html>

Applications 1.0)を策定する仕様の草案として採用すること²⁴⁷、2) WG で策定する仕様の名称を正式に“HTML5”とすること、3) 仕様の Editor に Ian Hickson を任命すること、である。また、HTML5 設計の基本方針(案) “HTML Design Principles (Proposed)”²⁴⁸ もあわせて提案している。

Ian Hickson は既に 3 月 14 日に、Web Application Format (WAF) WG で行われているのと同様に、WHATWG と W3C WAF WG において同一内容の Web Forms 2.0 仕様を作成・公開し、自らが両者の Editor を務めることに前向きな姿勢を示している²⁴⁹。

この提案に対し Microsoft 所属の Co-Chair である Chris Wilson は、WHATWG 版 HTML5 仕様書 Editor の Ian Hickson が独りで W3C 版仕様書の Editor を担うことに反対し、追加の Co-editor を任命することを要望した²⁵⁰。Co-editor が必要であるかについては議論が生じたが、最終的に Apple に所属する David Hyatt を候補とする²⁵¹ことで決着する。なお、XForms 2.0 仕様の Editor である IBM の John Boyer が Form WG に関係する人間を Co-Chair とするよう要請する投稿をした²⁵²ものの、Wilson によって一笑にふされている²⁵³。

最終的に Wilson による修正案がそのまま投票にかけられた^{254,255}。その結果、W3C HTML WG の仕様策定作業は、WHATWG で策定された仕様書をベースに進めることとなり、また Ian Hickson が WHATWG 版と W3C 版の HTML5 仕様の Editor をともに務めることとなった。

HTML WG のコミュニケーションは、当初より公開のメーリングリスト

²⁴⁷ van Kesteren, A. (2007, April 10). *Re: Proposal to Adopt HTML5*. Retrieved May 15, 2013, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/0472.html>

²⁴⁸ Stachowiak, M. et Al. (2007, April 4). *HTML Design Principles (Proposed) Revision as of 03:56, 4 April 2007*. Retrieved March 7, 2014, from <http://www.w3.org/html/wg/wiki/index.php?title=ProposedDesignPrinciples&oldid=2342>

²⁴⁹ Hickson, I. (2007, March 14). *Re: Preparing to launch the Forms Task Force ...* Retrieved April 17, 2014, from www-forms@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/www-forms/2007Mar/0028.html>

²⁵⁰ Wilson, C. (2007 April 19). *HTML5 proposal response*. Retrieved from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/1240.html>

²⁵¹ Stachowiak, M. (2007, April 21) *Nomination for Co-Editor: Dave Hyatt*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/1280.html>

²⁵² John Boyer, J. (2007, April 27). *Re: Nomination for Co-Editor: Dave Hyatt*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/1611.html>

²⁵³ Wilson, C. (2007, April 27). *RE: Nomination for Co-Editor: Dave Hyatt*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/1615.html>

²⁵⁴ Connolly, D. (2007, May 01). *Decision process: Formal Objections and consensus in the W3C process*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007May/0146.html>

²⁵⁵ World Wide Web Consortium (2007). *Shall we Adopt HTML5 as our specification text for review? (closed since 04 May 2007)*. Retrieved March 7, 2014, from Web-Based Straw-poll and Balloting System: <https://www.w3.org/2002/09/wbs/40318/htmlbg/>

“public-html@w3.org²⁵⁶”で行われていた。また W3C 全体として、IRC も併用されるが、W3C Team Staff で HTML WG の Team Contact²⁵⁷である Michael Smith がメーリングリストにおいて、W3C²⁵⁸だけではなく WHATWG²⁵⁹の IRC チャンネルもあわせて紹介している²⁶⁰。その後 Smith が主に用いるのは WHATWG チャンネルであり²⁶¹、W3C における HTML5 仕様策定プロセスは、オープンかつ WHATWG における活動と密接にリンクしながら進むという構図が完成した。

実際に Ian Hickson が public-html@w3.org メーリングリストに対し、非会員から寄せられたフィードバックを参考にしており(図 30)、WHATWG、W3C という2つのオープンな策定プロセスが連携しながら進むという構造になっている。

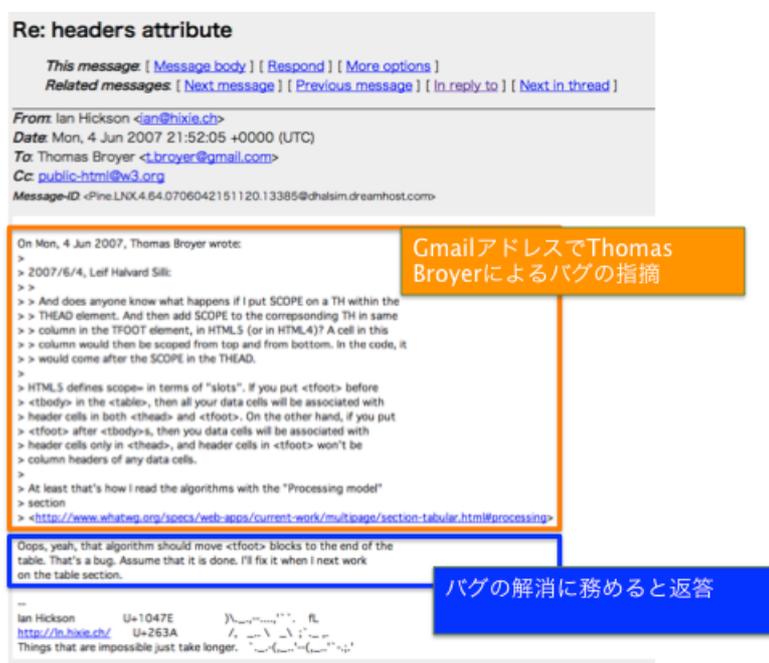


図 30. public-html@w3.org メーリングリストにおける
非会員からのフィードバック²⁶²

²⁵⁶ public-html@w3.org Mailing list Archives. Retrieved March 7, 2014, from <http://lists.w3.org/Archives/Public/public-html/>

²⁵⁷ Team contact は、WG の運営、具体的には会議の日程調整、議事録のまとめ、Process Document に基づいた仕様策定作業に対する助言を行う役割を担う W3C team staff であり、WG ごとに任命される。

²⁵⁸ #html-wg channel on irc.w3.org, port 6665 or port 80

²⁵⁹ #whatwg channel on irc.freenode.net

²⁶⁰ Smith, M. (2009, June 7). *value of IRC conversations [was: Summary of Thursday's IRC conversation about @summary]*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2009Jun/0254.html>

²⁶¹ 筆者による W3C/Keio Office における取材より。

²⁶² Hickson, I. (2007, June 4). *Re: headers attribute*. Retrieved April 16, 2014, from public-html@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-html/2007Jun/0105.html>

7.4.3. ロイヤルティ・フリー・ポリシーの明文化

インターネットのアプリケーションとして開発され、また当初は学術情報共有プラットフォームとして提供されたウェブは、明文化されていないもののロイヤルティ・フリーの原則がステークホルダー間で共有されていた。

この方針を明文化する作業が、1999年10月に Patent Policy WG が設立され、第1回 F2F 会合が開催されることから始められた。

2001年8月に初めて公開された Patent Policy Framework の Working Draft²⁶³において、ロイヤルティ・フリー、つまり標準仕様を実装する際に用いられる全ての技術に対し知的財産使用料を請求しないという原則が記載されている。

2003年5月20日には、ACがロイヤルティ・フリー・ポリシーに同意し²⁶⁴、基本原則とされることとなった。また、同日付で公開された W3C Patent Policy²⁶⁵にて、明文化されることとなった。

7.5. 次世代 HTML 仕様の HTML5 への一本化

7.5.1. XHTML の一部機能を HTML5 に残すための試み

W3C の WG が再編され、新 HTML WG にて HTML5 の仕様策定作業が始まってからも、XML 技術普及のための試み、特に HTML5 仕様が XML データを取り扱うことができるようにするための提言や活動は続いた。

また、HTML5 に XForms の機能を取り込むことを目的として、HTML WG と XForms WG のメンバーが参加する Taskforce の設置について検討された。新 HTML WG の Charter²⁶⁶には、設立予定の Forms Taskforce への依存関係 (Dependency) が盛り込まれ、Taskforce の設置は既定路線となった。

XForms 仕様策定の中心人物であり、Forms WG の Chair に就任した IBM の John Boyer は、新 HTML WG に参加するにあたってのアンケートにおける “What tasks are you here to help with?” という質問に対し、

* *forms taskforce*

²⁶³ Weitzner, D. (ed.) (2001, August 1). *W3C Patent Policy Framework - W3C Working Draft 16 August 2001*. Retrieved July 20, 2014, from <http://www.w3.org/TR/2001/WD-patent-policy-20010816/>

²⁶⁴ Berners-Lee, T. (2003, May 20). *Announcement: W3C Patent Policy Approved*. Retrieved July 20, 2014 from w3c-ac-members@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-ac-members/2003AprJun/0033>

²⁶⁵ Weitzner, D. (ed.) (2003, May 20). *W3C Patent Policy - W3C Policy 20 May 2003*. Retrieved July 20, 2014, from <http://www.w3.org/Consortium/Patent-Policy-20030520.html>

²⁶⁶ Lilley, C. (2007, 7 March). *HTML Working Group Charter*. Retrieved March 3, 2014, from Internet Archive, <http://wayback.archive.org/web/20070309123748/http://www.w3.org/2007/03/HTML-WG-charter.html>

I am interested in working out a forms language for HTML that meets as well as possible the requirements that inspired both XForms and WF2. I am especially interested in an architecture that allows relatively seamless scale up from html4 tag soup forms to xforms, where I view aspects of WF2 as being somewhere in the middle with possibly some pieces not being able to morph, without adjustment, to something more like xforms (which I see ultimately manifesting itself in the lifecycle of forms that grow sufficiently complex over time²⁶⁷).

と回答し、Taskforce の設立と XForms 開発の成果を HTML5 に反映させることに意欲を見せている。

2007 年 8 月に、ようやく各 WG からの Taskforce 参加者が決まり²⁶⁸、10 月 2 日に Charter が Opera の Anne van Kesteren から提案される²⁶⁹。そして 10 月 12 日に Taskforce のウェブサイト²⁷⁰が公開され、また Charter 案も公開された。ところが、この案に対するコメントが 1 つも寄せられず²⁷¹、また 2007 年 11 月 29 日から 4 月 1 日まで Taskforce のメーリングリストに全く投稿がなされない状態が続くこととなる。

この停滞を打破しようとしたのが IBM の John Boyer である。Forms WG において、XForms 仕様を HTML(5)に準拠させることを意図した XForms 1.2 Simplified Syntax Proposal 仕様を提案²⁷²し、Forms Taskforce に対してレビューを求める投稿²⁷³を行う。ところが、Apple の Maciej Stachowiak は以下のとおりに返答し、Taskforce での検討を拒んだのである。

²⁶⁷ World Wide Web Consortium. (2007). *Results of Questionnaire HTML WG tasks - Web-Based Straw-poll and Balloting System*. Retrieved March 3, 2014, from w3c-ac-forum@w3.org mailing list archive, <http://www.w3.org/2002/09/wbs/40318/tasks83/results>

²⁶⁸ Lilley, C. (2007, August 22). *Please construct a task force charter*. Retrieved March 3, 2014, from public-forms-tf@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms-tf/2007Aug/0000.html>

²⁶⁹ van Kesteren, A. (2007, October 22). *Charter Proposal*. Retrieved March 3, 2014, from public-forms-tf@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms-tf/2007Oct/0000.html>

²⁷⁰ <http://www.w3.org/2007/10/forms-tf/>

²⁷¹ van Kesteren, A. (2007, November 29). *charter review over*. Retrieved March 3, 2014, from public-forms-tf@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms-tf/2007Nov/0000.html>

²⁷² Boyer, J. (2008, March 27). *XForms 1.2 Simplified Syntax Proposal*. Retrieved March 3, 2014, from public-forms@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms/2008Mar/0097.html>

²⁷³ Boyer, J. (2008, April 2). *XForms Simplified Forms Syntax Review Needed*. Retrieved March 3, 2014, from public-forms-tf@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms/2008Apr/0009.html>

Our charter scope is to define "a set of guidelines for architectural consistency in form technologies". Our deliverables are a W3C Note outlining these guidelines, and then reviewing the evaluations of the HTML WG and the Forms WG of their technologies with respect to these guidelines.

Unfortunately, we have failed to develop these guidelines so far, due to inactivity of the TF, so we cannot provide review with these guidelines in mind²⁷⁴.

結果として、この提案は Taskforce ではレビューされず、Boyer の試みは頓挫することとなる。

John Boyer は諦めず、2008 年 12 月 19 日に HTML に対応した XForm 仕様である“XForms for HTML”の Working Draft²⁷⁵を公開する。しかし、XForms WG のメーリングリストで議題に上ることもなく、Draft が更新され、標準化プロセスが進むという事態には至らなかった。

7.5.2. XHTML2 WG の機能不全

XForms WG よりも更にひどい状況にあったのが、XHTML2 WG である。2007 年 3 月に WG が設置されて以来、2008 年末までに公開された Technical Report は XHTML 1.1 2nd edition の FPWD²⁷⁶のみであった。

2009 年に入ると HTML5 への追い風と XHTML に対する逆風がさらに強まる。2 月に Adobe が HTML WG のメンバーになるとともに、HTML5 開発へ資金援助することが表明された²⁷⁷。これで Mozilla、Opera、Apple、Google という主要ブラウザベンダの多数に加え、最大の市場シェアを有するウェブサイト・コンテンツ制作ツールベンダが HTML5 支持に回ったこととなる。

W3C では、WG Charter の有効期限は 2 年間と定められている。そのため、2009 年に入ると、HTML WG/ XHTML2 WG/ XForms WG の 3 WG 体制の今後に関する

²⁷⁴ Stachowiak, M. (2008, April 2). *Re: XForms Simplified Forms Syntax Review Needed*. Retrieved March 3, 2014, from public-forms-tf@w3.org mailing list archive, <http://lists.w3.org/Archives/Public/public-forms-tf/2008Apr/0002.html>

²⁷⁵ World Wide Web Consortium (2007, October 12). *Forms Task Force Charter (PROPOSAL) \$Revision: 1.1*. Retrieved March 3, 2014, from <http://www.w3.org/2007/10/forms-tf/charter-proposal>

²⁷⁶ McCarron, C. (2007, February). *XHTML™ 1.1 - Module-based XHTML - Second Edition W3C Working Draft 16 February 2007*. Retrieved April 20, 2014, from <http://www.w3.org/TR/2007/WD-xhtml11-20070216/>

²⁷⁷ World Wide Web Consortium (2009, February 2). *HTML 5 Receives Support for Authoring Materials. Press Release*. Retrieved April 20, 2014, from <http://www.w3.org/News/2009#entry-6709>

る議論が W3C 内で始まることとなった。

7.5.3. 2009 年 3 月 AC Meeting

2009 年 3 月 23、24 日の二日間で行われた AC ミーティングでは XHTML2 のリブランディングと WG 再編成が提起され、HTML WG と XHTML2 WG 両者の Chair 間で統合を検討するという結論がくだされた(図 31)²⁷⁸。

そして 2009 年末にて XHTML2 WG の活動が終了し、XHTML2 の仕様策定活動を終結することが決定した。HTML5 と XHTML という 2 つの競合する仕様が並立する状況が、ようやく解消したのである。



図 31. 2009 年 3 月 AC ミーティングの様子²⁷⁹

²⁷⁸ Le Hégarat, P. et al. (2009. March 24). *W3C AC Day 2, 24 Mar 2009 (minutes)* Retrieved March 3, 2014, from <https://www.w3.org/2009/03/24-ac-minutes>

²⁷⁹ 一色正男. (2010, December 6). *W3C (World Wide Web Consortium)へのご案内*. Retrieved June 5, 2014, from <http://www.w3.org/2010/Talks/1206-echonet-mi/20101201EchonetForumpre.pdf>

7.5.4. Mozilla による XHTML の実装除外

Mozilla は、Firefox の描画エンジンである Gecko で XML を描画するための機能を提供する eXtensible Tag Framework (XTF) の実装²⁸⁰と、XML Events の実装²⁸¹を 2012 年 11 月 15 日に削除している²⁸²。この削除により Firefox は、バージョン 19 以降で XForms 非対応となった²⁸³。これにより、主要ブラウザへの XForms 実装の試みは、全て頓挫することとなったのである。

7.6. 機能開発ならびに普及促進施策の新たな形態

7.6.1. 独自仕様の実装を元にした標準仕様提案：Google のオープン標準戦略

7.6.1.1. ブラウザ参入計画と Mozilla Firefox 開発への貢献

Google は Gmail や Google Maps 等のウェブアプリケーション提供以前から、自社でブラウザを開発する計画を有していた。2001 年 3 月に Google に移籍した²⁸⁴Eric Schmidt 現 Executive chairman は、入社当初に共同創業者の Larry Page と Sergey Brin から自社ブラウザの開発計画を提案されている²⁸⁵。この提案を Schmidt は受け入れなかったが、2005 年 1 月に Mozilla Foundation に所属して Firefox 開発の中心的役割を務めていた Ben Goodger²⁸⁶と Darin Fisher²⁸⁷を雇用する。Google における職務は、引き続き Firefox ならびに Firefox 向けアドオンの開発であった²⁸⁸。

Mozilla による HTML5 が実装されたブラウザの開発を支援し、自らはウェブアプリケーションを開発・供給するというこの施策は、補完財供給者と役割分担前提で連携して競争優位を創出 “Divided technical leadership” (Bresnahan & Greenstein, 1999) 戦略に基づいたものといえるだろう。Microsoft が Inter と組んで、Microsoft Windows OS と Intel X86 アーキテクチャ CPU、いわゆる Wintel と

²⁸⁰ Mozilla Foundation. (n. d.) *History - Changes made to Bug 749448 - Remove XTF*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_activity.cgi?id=749448

²⁸¹ Mozilla Foundation. (n. d.) *History - Changes made to bug 656311 - Remove XML Events, or improve the implementation*. Retrieved March 3, 2014, from https://bugzilla.mozilla.org/show_activity.cgi?id=656311

²⁸² Swisher, J. et Al. XForms. *Mozilla Developers Network*. Retrieved March 3, 2014, from <https://developer.mozilla.org/en/docs/XForms>

²⁸³ Alexander, S. (2012, November 15). Mozilla XForms has gone. Posted to Alex's blog Retrieved March 3, 2014, from <http://asurkov.blogspot.jp/2012/11/mozilla-xforms-has-gone.html>

²⁸⁴ CNET News. (2001, March 26). *Novell's Schmidt joins Google at critical time*. Retrieved 2014, March 15, 2014 from <http://news.cnet.com/2100-1023-254750.html>

²⁸⁵ Levy, S. (2008, September 2). Inside Chrome: The Secret Project to Crush IE and Remake the Web. *Wired*. Retrieved March 15, 2014, from http://www.wired.com/techbiz/it/magazine/16-10/mf_chrome?currentPage=all

²⁸⁶ Wagner, J. (2005, January 25). Goodger Goes to Google. *InternetNews*. Retrieved March 15, 2014, from <http://www.internetnews.com/dev-news/article.php/3463841>

²⁸⁷ CNET News. (2005, January 27). *More Mozilla talent Google's way*. Retrieved March 15, 2014, from http://news.cnet.com/More-Mozilla-talent-Google's-way/2110-1032_3-5553877.html

²⁸⁸ Mozilla Digital Memory Bank. (2006, June 12). *Interview with Darin Fisher*. Retrieved March 15, 2014, from <http://mozillamemory.org/detailview.php?id=946>

いう組み合わせによってパーソナルコンピュータ市場を支配した手法をウェブブラウザとウェブアプリケーションの組み合わせで行おうとしたと分析することができる。

7.6.1.2. アドオンによる独自先行実装と標準化プロセスへの持ち込み

Google の Firefox 開発チームは、複数デバイスのブラウザ間でブックマークや履歴を同期できる Google Browser Sync や、フィッシング等の危険性があるサイトリストに含まれる URL にアクセスする際に警告を出す Google Safe Browsing といった Firefox 向けアドオンを開発している。これらは後に、Google の自社製ブラウザである Chrome に実装されている^{289,290}。

Google は更に、Internet Explorer など他のブラウザにも組み込むことのできるアドオンソフトウェアを提供している。Gears はオフライン機能など、ウェブアプリケーションの機能向上を実現するブラウザ向けアドオンソフトであり、Opera や Mozilla 等外部企業の協力を得て²⁹¹、BSD license 準拠のオープンソースソフトウェアとして開発された。

Google I/O 2008 というイベントにおける Gears を紹介するプレゼンテーションの概要文は、以下のように記述されている。

HTML5 is a new set of proposed extensions to HTML that radically improve the capabilities of web applications. However, without implementations in a majority of browsers, these proposals remain just that, and out of reach for developers.

The Gears mission is to begin implementing these APIs today, across as many browsers as possible, as quickly as possible. In this talk, I'll explain why we are doing this, what our motives are, and show how implementing web standards is good for Google and good for the web²⁹².

²⁸⁹ Google Labs. *Google Browser Sync*. Retrieved March 15, 2014, <http://www.google.com/tools/firefox/browsersync/>

²⁹⁰ Google. *Google Safe Browsing*. Retrieved March 15, 2014, <http://www.google.com/tools/firefox/safebrowsing/>

²⁹¹ Boodman, A. and Erik Arvidsson, E. (2007, May 30). *Going offline with Google Gears*. Retrieved March 8, 2014, from Gears API Blog: <http://gearsblog.blogspot.jp/2007/05/posted-by-aaron-boodman-and-erik.html>

²⁹² Boodman, A. (2008, June 9). *HTML 5, Brought to You by Gears*. Retrieved March 13, 2014, from 2008 Google I/O Session Videos and Slides: <https://sites.google.com/site/io/html5-brought-to-you-by-gears>

また、Google Japan ソフトウェアエンジニアの河内隆仁氏は、インタビューにおいて以下のとおり発言している。

いまある HTML の仕様を基に次のバージョンである HTML 5 の仕様を決めようとする標準化のプロセスは長いものだ。そのプロセスの途中で完成した機能が徐々に Web ブラウザに取り込まれるが、機能が Web ブラウザに取り込まれてリリースされるまでは長い期間を要するものだ。Gears のアプローチとしては、HTML 5 の仕様になりそうな機能、または、いまの HTML に欠けている機能をなるべく早く提供する。そうやって Gears に採用された機能が、もしかしたら HTML 5 の仕様として取り込まれることもあるかもしれない。²⁹³

つまり Gears 開発・提供の目的は、1) 策定途上の HTML5 の機能を先行して実行できる環境を提供する、2) 将来的に標準に取り入れられる可能性がある機能を先行して提供する、3) これを通じて HTML5 の機能拡張と普及促進を目指す、というものである。

Gears はバージョンアップ毎に機能を増やしてきた。2008 年 7 月にリリースされバージョン 0.1 ではウェブアプリケーションのオフライン動作機能を提供している。2008 年 2 月にリリースされたバージョン 0.2 では、サーバとの高度なプロトコル (Workers) を実装し、またウェブサイト制作者向けツールも追加されている²⁹⁴。2008 年 8 月にリリースされたバージョン 0.8 では、デバイスに搭載された GPS から位置情報を取得する Geolocation API²⁹⁵とデバイスからファイルを開くなどの機能を提供する Desktop API を追加した²⁹⁶。Geolocation API 機能の追加に伴い、Google は携帯端末向けに位置情報をベースにした検索サービスの提供を始めている²⁹⁷。

²⁹³ @IT 編集部. (2008, July 14). *HTML 5 をのみ込んで Web ブラウザを拡張する Gears*. Retrieved March 13, 2014, from <http://www.atmarkit.co.jp/fwcr/special/gdd2008/google01.html#03>

²⁹⁴ Boodman, A. and Erik Arvidsson, E. (2008, February 19). *Gears 0.2 Released!*. Retrieved March 8, 2014, from Gears API Blog: <http://gearsblog.blogspot.jp/2008/02/gears-02-released.html>

²⁹⁵ その後 GPS からだけではなく、Wifi アンテナからの位置情報取得機能を追加している。(Block, S. (2008, October 21). *Increased Accuracy for Location in the Browser*. Retrieved March 8, 2014, from Google UK Developer Blog: <http://google-ukdev.blogspot.jp/2008/10/increased-accuracy-for-location-in.html>

²⁹⁶ Popescu, A. (2008, October 22). *Gears 0.4 is here!*. Retrieved March 8, 2014, from Gears API Blog: <http://gearsblog.blogspot.jp/2008/08/gears-04-is-here.html>

²⁹⁷ Genera, P. (2008, September 11). *Introducing Mobile Search with My Location*. Retrieved March 8, 2014, from Official Google Mobile Blog: <http://googlemobile.blogspot.jp/2008/09/introducing-mobile-search-with-my.html>

またサードパーティによる実装事例も増加していた。To Do List 管理の Remember the Milk²⁹⁸、オフィススイートウェブアプリケーションの Zoho²⁹⁹、OSS Contents Management System の Wordpress³⁰⁰など、複数のサードパーティが実装している³⁰¹。

7.6.1.3. 自社製ブラウザ、Chrome の市場投入

Firefox 開発への関与や、アドオンソフトウェアの開発と平行して、Google はブラウザ開発技術を有する企業の買収も進めている。2005 年 8 月に Google は携帯電話向けプラットフォーム開発企業である Android を³⁰²、2005 年 11 月には Vector graphic 技術をもつ Skia を³⁰³、2006 年 1 月には携帯電話向けブラウザ開発企業である Reqwireless を³⁰⁴、2007 年 5 月にはセキュリティ技術企業である GreenBorder を³⁰⁵、それぞれ買収している。Android には、オープンソースウェブサイトレンダリングエンジンである Webkit をベースにしたブラウザが組み込まれており、これら企業の買収によってブラウザ開発のための技術と人員を増強したことになる。

遂に Google は 2008 年 9 月 2 日に自社製ブラウザである Chrome を市場投入する³⁰⁶。Chrome 開発チームは、Firefox/アドオン、Gears 開発に従事したエンジニアと、買収先の人員によって構成された。

Google はウェブアプリケーションと実行環境であるブラウザ、更にそれぞれの設計を規定する標準仕様を全て自ら開発し、提供するという体制を構築することとなった。これにより、自らが開発した技術を迅速に実装し、エンドユーザに提示することが可能となった。

²⁹⁸ Kilani, O. (2007, June 5). *Tasks on a plane*. Retrieved March 11, 2014 from Remember The Milk Blog: <http://blog.rememberthemilk.com/2007/06/tasks-on-a-plane/>

²⁹⁹ Vegesna, R. (2007, November 26). *Zoho Writer Supports Offline Editing*. Retrieved March 11, 2014 from Zoho Blogs: <http://blogs.zoho.com/general/zoho-writer-supports-offline-editing.html>

³⁰⁰ Ozz, A. (2008, July 2). *WordPress Gears*. Retrieved March 11, 2014 from Hot Off the Press, WordPress.com: <http://en.blog.wordpress.com/2008/07/02/gears/>

³⁰¹ Google. (n. d.). *GearsHistory - Timeline of major events*. Retrieved March 11, 2014 from Gears Wiki, <https://code.google.com/p/gears/wiki/GearsHistory>

³⁰² Elgin, B. (2005, August 16). *Google Buys Android for Its Mobile Arsenal*. *Businessweek.com*. Retrieved March 14, 2014 from <http://www.businessweek.com/stories/2005-08-16/google-buys-android-for-its-mobile-arsenal>

³⁰³ Pulse 2.0. (2008, October 4). *A History of Google Acquisitions and Where They Are Today*. Retrieved March 14, 2014 from <http://pulse2.com/2008/10/04/a-history-of-google-acquisitions-and-where-they-are-today/>

³⁰⁴ Mills, E. (2006, January 9). *Google buys Canadian wireless-software company*. *CNET News*. Retrieved March 14, 2014 from http://news.cnet.com/8301-10784_3-6024842-7.html

³⁰⁵ Hines, M. (2007, May 9). *Google buys into security, acquires GreenBorder*. *InfoWorld*. Retrieved March 14, 2014 from <http://www.infoworld.com/d/security-central/google-buys-security-acquires-greenborder-379>

³⁰⁶ Google. (2008, September 2). *Google Chrome: A New Take on the Browser*. Retrieved March 11, 2014 from Google News announcements: http://googlepress.blogspot.jp/2008/09/google-chrome-new-take-on-browser_02.html

なお Chrome は順調に市場シェアを伸ばし、主要ブラウザベンダ 5 社のうち、4 社が HTML5 の実装を推進するという構図が完成することとなる。

7.6.1.4. Gears 開発中止、仕様のブラウザ実装と標準化提案

翌年 2009 年の Google I/O では、基調講演で HTML5 が取り上げられた。TechCrunch に掲載された速記録によると、Eric Schmidt CEO (当時)は、

We have spent 20 years trying to build a programming model that is the right one. Then the Internet arrived. "It's time."

This is the beginning of the real win of cloud computing, of applications (on the web)³⁰⁷.

と述べ、本格的なウェブアプリケーションを前提としたアーキテクチャが主流になる時期が到来したことを強調した。

引き続き Microsoft から移籍した Vice president of engineering の Vic Gundotra は、

"Never underestimate the web." At Microsoft (where he used to work) we thought web apps could never rival desktop apps.

"The web has won." A more powerful web made easier.

New HTML5 standards. A chance to do things differently.

Almost half billion people now using modern open source browsers³⁰⁸.

と述べ、HTML5 が進化したウェブアプリケーションを実現するための鍵となる技術であり、既に 5 億人以上が対応ブラウザを利用していると指摘した。ここでの“modern open source browsers”は、Firefox に加え、WebKit というオープンソース描画エンジンをベースに開発された Apple Safari と Google Chrome を含

³⁰⁷ Siegler, M. G. (2009, May 27). Live From Google I/O 2009. *TechCrunch*. Retrieved March 17, 2014, from <http://techcrunch.com/2009/05/27/live-from-google-io-2009/>

³⁰⁸ *ibid.*

んでいる³⁰⁹。

HTML5 の機能として紹介されたのは、Geo Location、Application Caching and Database、Web Workers といった Gears が提供していた機能が含まれる。特に Geo Location については、iPhone に搭載される Safari ブラウザが対応するとアナウンスされている。

このプレゼンテーションをきっかけに HTML5 は、仕様策定関与者やブラウザ開発従事者以外の層からも注目を集めるようになった。また、既に HTML5 の実装に取り組んでいるブラウザの市場シェアは 30%以上³¹⁰に達した。

このような状況が整ったことにより、Google は Gears の開発を終了させるという意思決定をすることとなる。2009 年 12 月に開発責任者の Linus Upson が PC Magazine のインタビュー³¹¹で

*"Yes, we are not driving forward in any meaningful way [on Gears],"
Upson said in an interview. "*

*"We're very focused on moving HTML 5 forward, and that's where
we're putting all of our energy,"*

と回答し、Gears の開発を終了し、開発資源を HTML5 の仕様策定・実装に振り向けると明言した。その後 2011 年 3 月に、Gears の開発中止が正式に発表される³¹²。

開発中止に伴う取材に対するコメント、インタビュー、ブログ記事の中で、関係者は以下のように回答している。

³⁰⁹ 後に WHATWG をサポートするもう 1 社のブラウザ、Opera も加えられている。(Krazit, T. (2009, May 27). Google: The browser is the computer. *CNET*. Retrieved March 17, 2014, from http://news.cnet.com/8301-17939_109-10250196-2.html)

³¹⁰ Wikipedia の *Usage share of web browsers* (Retrieved March 6, 2014, from http://en.wikipedia.org/wiki/Usage_share_of_web_browsers)の項目にまとめられている統計結果によると、HTML5 実装に取り組んでいるブラウザ、つまり Google Chrome、Mozilla Firefox、Apple Safari、Opera の市場シェア合計は、30.9% (2009 Q2, Net Applications)、39.2% (2009 July, Stat Counter, Desktop)。

³¹¹ Hachman, M. (2009, December 1). Google Gears Is Dead; Long Live HTML 5.0. *PCMag.com*. Retrieved March 17, 2014 from <http://www.pcmag.com/article2/0,2817,2356492,00.asp>

³¹² Boodman, A. (2011, March 11). Stopping the Gears. *Gears API Blog*. Retrieved March 17, 2014, from <http://gearsblog.blogspot.jp/2011/03/stopping-gears.html>

"When we started the Gears project, three years ago, maybe three and a half years ago now, we did it because we couldn't get the browser vendors interested in building offline applications. And so, so we said, okay, we'll build a plugin that could do it. And lo and behold, once we shipped Gears, suddenly the browser vendors got very interested in adding capabilities to build offline applications."

"And so, I think Gears has accomplished its mission very well, in getting these capabilities into HTML 5," Upson added. "In fact, the team that designed Gears was also instrumental in designing the HTML 5 versions of those APIs. You can almost think of what's in HTML 5, with app cache, and database, and those things, as essentially Gears [version] 2, and that's how we view it."

- Linus Upson³¹³

"We are excited that much of the technology in Gears, including offline support and geolocation APIs, are being incorporated into the HTML5 spec as an open standard supported across browsers, and see that as the logical next step for developers looking to include these features in their websites,"

-Anonymus Google Spokesman in e-mail³¹⁴

With all (application caches, IndexedDB API, File API, geolocation, notifications, and web worker APIs) this now available in HTML5, it's finally time to say goodbye to Gears.

Now that these features have all been adopted by browsers and have official W3C specs, they are available to more developers than we could have reached with Gears alone.

³¹³ Hachman, M. (2009, December 1). Google Gears Is Dead; Long Live HTML 5.0. *PCMag.com*. Retrieved March 17, 2014 from <http://www.pcmag.com/article2/0,2817,2356492,00.asp>

³¹⁴ Milian, M. (2009, November 30). What's powering Web apps: Google waving goodbye to Gears, hello to HTML5 [Updated]. *latimes.com*. Retrieved March 17, 2014, from <http://latimesblogs.latimes.com/technology/2009/11/google-gears.html>

-Aaron Boodman, Gears Team³¹⁵

これらの声明から、Google は Gears を通じて高度なウェブアプリケーション開発に必要な技術に対する需要喚起と理解を果たし、HTML5 という標準仕様に取り込ませることに成功したといえよう。

7.6.2. オープン標準仕様に関するディベロッパー・リレーション施策の活用

7.6.2.1. ディベロッパー・リレーション施策とは

HTML5 という言葉がウェブアプリケーション・コンテンツ開発者の間に普及する大きなきっかけとなったのは、前述のとおり 2009 年 5 月に実施された Google I/O のキーノートスピーチである。Google I/O は米国カリフォルニア州・サンフランシスコにて、毎年 5 月に開催される。このイベントはマスメディア向けではなく、Google が提供するサービス、製品に関係する開発者向けのイベントと位置づけられている。

Google は I/O 以外にも世界各地で開催される Google Developer Day などのイベント、ウェブサイトからの情報提供等さまざまなチャネルを活用して開発者に対し情報提供を行うと同時に、自社技術を売り込む。このような開発者向けプロモーション活動は、情報通信産業ではディベロッパー・リレーション施策と呼ばれ、Google 以外にも Microsoft の Microsoft Developers Network (MSDN) など、多数の企業が実施している。

このような活動は、ほとんどが自社製品・サービスに関する技術情報を、自社の従業員を通じて提供するという形態がほとんどである。例えば Apple は Apple Developer³¹⁶というウェブサイトを通じて情報を提供するとともに、開発者向け有料サポート³¹⁷の仕組みを用意している。また年に 1 度 World Wide Developer Conference (WWDC) というイベントを実施している。WWDC は、数多くのメディアに取り上げられる新製品のお披露目会であるが、と同時に開発者への F2F での情報提供・交換の場でもある。なお、開発者向けに情報提供を行う役職は、エバンジェリストと呼ばれることが多い。

³¹⁵ Boodman, A. (2011, March 11). Stopping the Gears. *Gears API Blog*. Retrieved March 17, 2014, from <http://gearsblog.blogspot.jp/2011/03/stopping-gears.html>

³¹⁶ <https://developer.apple.com/>

³¹⁷ 2014 年 4 月 20 日時点で、Mac Developer Program (MacOS アプリケーション開発者向けサポートプログラム)、iOS Developer Program (iOS アプリケーション開発者向けサポートプログラム)の 2 つのプログラムが用意されており、それぞれ年会費は 99US\$。(Apple Developer Programs. *Apple Developer*. Retrieved April 21, 2014, from <https://developer.apple.com/programs/>)

サービス・アプリケーションの API を提供している企業は、その API を用いてプログラムやサービスを構築するコンテストを開催することもある。Google が、2004 年から 2011 年までサービス展開していた Google Desktop の API をテーマとしたコンテストに応募したエンジニアは、感想を以下のように述べている。

僕は Google ファン。はるか昔の話で、5-6 年前、(Windows) Vista がでる直前に Google Desktop 検索が非常におもしろいと思った。カスタムガジェットが作れた。それがおもしろくて、リリース 3 日後に為替のアプリを作った。Yahoo! (Finance) からとりやすいデータだったら、ガジェットを作った。(開発したガジェットの)登録サイトがあつて登録したら、忘れた頃に Google 本社からメールがきた。「おもしろいものを作ってくれてありがとう、お礼に T シャツ送るから住所を教えてよ」と。

(メールは)カリフォルニアから直接英語できた。自動翻訳かけたけど(最初は)SPAM かと思った。忘れたところに UPS が届いて、“Desktop Search”の T シャツと帽子がはいつていた。それにやられた。グッズはやばい(ほど魅力的だった)。

-- HTML5west.jp 主宰 村岡正和³¹⁸

このように、コンテスト開催と報奨によって API やサービス・技術の利用拡大と、ロイヤルティ向上・拡大を実現させるという試みが広く行われている。

7.6.2.2. 外部開発者を通じた普及施策

多くの企業が採用してきた手法は、エバンジェリストを社内技術者が務めるというものであった。しかし、企業がオープンソースコミュニティと協働したり、プロプライエタリな技術を標準化したりするようになる中で、エバンジェリストの役割を社外の開発者に求める企業が出現する。その 1 つが日本サン・マイクロシステムズである。

³¹⁸ 2011 年 9 月 4 日実施、村岡正和氏インタビュー

ひとつは、Sun の技術を良く知っている人間を派遣する。あとは、イベント開催告知のサポート。Sun Developer Connection という会員制プログラムを利用。(イベントの)プロモーション支援としてメールマガジンにイベントを掲載する。また、講演者として日本 Sun の人間もしくは海外法人(所属エンジニア)のコーディネート(も行う)。スポンサーとしてイベントを支えることはしばしば行われ、金銭的補助、会場提供、運営スタッフの提供などが行われた。

--株式会社ボイスリサーチ 西村一彦 取締役・CTO³¹⁹

このような試みが Google の日本法人でも開始される。石原直樹が中心となって始めた社外技術者の情報発信をサポートするプログラム、それが API Expert プログラムである。プログラムは、マーケティング活動の業務委託先企業に所属する西村一彦、横田昌彦と一緒に企画、実行されていた。

API Expert プログラムの内容は、具体的には社外のトップエンジニアに API Expert の称号を付与し、技術情報を提供するかわりに、準公式コミュニティを設立・運営することで、広く Google の技術情報を広めてもらうというものである。また、月一回東京の Google 日本法人本社で全 API Expert を集めたミーティングが開催される。

このプログラムの目的を、西村は以下のとおりに語っている。

エコシステムを創りたいという発想。それを使いこなしてアプリケーションとして提供していくような人を巻き込みたい。

--株式会社ボイスリサーチ 西村一彦 取締役・CTO³²⁰

また西村と共にプログラム開発に取り組んだ横田昌彦氏は、以下のように述べている。

なるべく多くのエンジニアに使ってもらえるような、影響力のある(社外の)エンジニアに働きかける。

³¹⁹ 2011年4月11日実施、西村一彦氏インタビュー

³²⁰ 2011年4月11日実施、西村一彦氏インタビュー

当時 Twitter はあまり（流行してい）なかったので、blog にかいてもらう、それでボトムまでいくのでは。

-- アビダルマ株式会社 横田 昌彦³²¹

つまり、Google が自ら情報発信を全て担うのではなく、社外のエンジニアに情報発信やエンジニア間のコミュニケーションを担ってもらうことにより、継続的な技術情報の普及とロイヤルティの醸成を狙ったのであった。

API Expert に期待された役割は、任命された技術分野の専門家としてアドバイザー的な役割を果たすこと、情報を広めること、コミュニティを運営することであり、一方的に Google 側に利得があるような関係とも思える。しかし、API Expert としても、Google から任命されたということによって知名度が拡大するとともに、コミュニティ運営を通じて人脈も広がることから、タスクを通じて充分メリットを享受することができるようになっている。

API Expert プログラムについて発表した Google Developers Day 2008 (Tokyo) を取材した記事には、以下の様な記述がある。

石原氏は、まだコミュニティが立ち上がったばかりで具体例はないものの、エキスパートの利点として、Google 内部で行われる非公開テストに招待してフィードバックを送ると共にその技術を先行して習得できると述べた。Google Developer Advocate³²² の Chris Schalk 氏も、エキスパート制度の原型となった米国 Google の制度「API Guru」を紹介し、グッズや非公開テストでのフィードバックを Google からの支援に挙げた。安藤³²³氏は、コミュニティによってほかの技術者との出会いが生まれ広がりができたことを感謝し、先の後藤氏のような Google でしかわからない事象に回答が得られる安心感を卒直に語った³²⁴。

また、石原は別の取材で以下のように語っている。

³²¹ 2011年2月18日実施、横田昌彦氏インタビュー

³²² デベロッパー・リレーション活動において、社外エンジニアとのインターフェイスを担う役職。

³²³ Google API Expert (Google Maps API)の安藤幸央氏

³²⁴ ASCII.jp: 世界は日本の Google コミュニティの鐘の音を聞か (2/3) | Google 徹底解剖
<http://ascii.jp/elem/000/000/142/142607/index-2.html>

「気軽に質問することができ、情報をシェアし、サンプルアプリケーションを公開する場となるコミュニティーは、開発者にとってたいへん重要です。Google として開発者の方々が自発的に作って下さっているコミュニティーをきっちりサポートできるようなプランを検討しています³²⁵」

プログラム設計者は、Google 社員とのフェイス・トゥ・フェイスでの交流を通じた情報提供や、社外エンジニア間の交流が API Expert 任命者はメリットとして感じられると考えている。

API エキスパートというオフィシャルな役職を与えること。彼らに金銭的なメリットはない、今は2ヶ月に1回集まってもらって、ボランティア。API エキスパートの認定を使って仕事をしている人はあまりいないが、本を書いた人もいる。

-- アビダルマ株式会社 横田 昌彦³²⁶

実際に API Expert の一人は、以下のように語っている。

API Expert ミーティングとかで、あとまあ各コミュニティ間で何苦勞してる？とかどんなふうに運営してる？とか情報共有をすれば、自分のコミュニティをやるときにプラスに働きますし。

(Win-win 関係になっていることについて) それは間違い無いですね。そういうのが認められないと会社の方にもフィードバックがでない。そういう意味では会社にも貢献できているのかなと思いますけど。

会社に対してある程度のフィードバックを返せるというのは予めある程度考えてはいたのですが、結局うちの会社はあんまり W3C に対して注目していなかった。W3C 的な仕事をしますと

³²⁵ マイナビ. (2008, June). <特別企画>日本人 Web 開発者の支援を強化する Google の狙い. マイナビ転職. Retrieved March 18, 2014 from <https://tenshoku.mynavi.jp/eng/feature/02.cfm>

³²⁶ 2011 年 2 月 18 日実施、横田昌彦氏インタビュー

言うっても、そこのある程度コミュニティの中とかで活動して、その中で個人ベースで成果が出てくると、逆にそのニュースが会社に入ってきて、それがドライブになって会社の仕事として立ち上がるんですよ。で、今そういう意味で言うと現状はコミュニティの仕事もやっていますが、会社の業務としても *WebSocket* を中心として、チームとして立ち上げてもらって業務にしているんですよ。

そんなに狙ったというわけではないですね。自然的にそうなるだろうなとは思ってはいましたが、自然に、ですね。

--NTT コミュニケーションズ 小松 健作³²⁷

7.6.2.3. 中立性を保つことによる訴求効果の拡大

API Expert はもともと Google が提供する API の利用拡大を意図して企画されたプログラムである。しかし当初から Google は自社のプロプライエタリ技術・サービスに限定せず、オープンな標準技術も対象にしていた。2008 年 6 年のプログラム発表時から、Google が主導して策定・普及活動が進められていたオープンな標準規格である *Open Social* を対象とした *Expert* も任命されている。

だがこのやり方を、W3C において標準化が進められている *HTML5* にそのまま持ち込むのは弊害が存在した。提案仕様が標準として普及するためには、特定企業の利害を反映したという認識が広がることはマイナスである。Google が開発に貢献しても、オープンな標準技術として評価、賛同していると捉えられた方が、普及に対して効果的である。西村は以下のように語っている。

Google が語るとあたかも (*HTML5* というオープンな標準が)*Google* の持ち物のように見られてしまう可能性があった。*Google* としてはそれが本望ではなかったはず。あくまでもサポートしていきたいというスタイル。

Google 側としては本当に重要な情報をマーケットから吸い上げたり、逆にマーケットに対して情報を発信する、そのハブとして *API Expert* を位置づけている。

--株式会社ボイスリサーチ 西村一彦 取締役・CTO³²⁸

³²⁷ 2011 年 9 月 7 日実施、小松健作氏インタビュー

コミュニティの創出、拡大のために Google という企業の資源を投じるも、その活動が全面的に Google によって支えられていると認識されることは避けたいというディレンマについて、横田氏は以下のように語っている。

政治的だけど、彼らにコミュニティを作ってもらおうと思ったけれど、準公式コミュニティという微妙、API Expert は公認だったが、これ(API Expert が作って運営しているコミュニティ)は決して Google がオフィシャルとして認めた訳ではない。

-- アビダルマ株式会社 横田 昌彦³²⁹

API Expert によって設立、運営されているコミュニティは「Google の」ではなく、「準公式コミュニティ」という位置付けをされている³³⁰。HTML5 をテーマとするコミュニティ、html5j.org (旧 html5developer.jp)が発足した。

コミュニティの主な活動内容は、月に 1 回のペースでの開催と、年に 1 回「HTML5 Conference」と称する大規模なイベントの開催である。その他開発者向けイベントや地方コミュニティ主催のイベントへの出展、講師派遣なども手がけている。こういったイベントや勉強会を運営するため、月に 1 度のペースでコアメンバーの対面ミーティングが開かれる。

このコミュニティも、他のサービス・仕様と同様に「準公式」コミュニティという体裁で運営されている。実際に Google の競合ブラウザベンダである Opera が勉強会やコアメンバーミーティングの会場を提供したり³³¹、Opera に在籍する Web Evangelist (当時)の Daniel Davis が運営スタッフに参画している。また、2012 年 9 月に実施されたイベント、HTML5 Conference 2012 は、Google のみならず、Microsoft、Mozilla、Opera のブラウザベンダ 4 社と W3C がスポンサーに並んでいる³³²。

運営スタッフは中立性について以下のとおりに語っている。

³²⁸ 2011 年 4 月 11 日実施、西村一彦氏インタビュー

³²⁹ 2011 年 2 月 18 日実施、横田昌彦氏インタビュー

³³⁰ Google. *Google Code Site Group Japan*. Retrieved March 19, 2014, from <https://groups.google.com/forum/#!forum/codesitejp>

³³¹ 小松健作. (2010, June 19). 「第 7 回 HTML5 とか勉強会」活動報告：レポート. *gihyo.jp*. Retrieved March 19, 2014, from <http://gihyo.jp/news/report/2010/06/1801>

³³² html5j.org *HTML5 Conference 2012*. Retrieved March 19, 2014, from <http://events.html5j.org/conference/2012/09/>

結局コミュニティというのは自発的に動くもので、Google は単に支援しているだけです。だからそういう意味では Opera さんとかMS(Microsoft)さんも支援してくれますし、最初に立ち上げる時に協力してくれた Google という立場があるのは確かなので、あそこから準公式っていうのは Google が持つべきだと思っただけですけど、だからといって Google のために何かをやっているという感じはないですね。HTML5 については別に Google の技術というわけではないので。他の API Expert とそこはちょっと違うんですけど。

ある程度 Google なり色々なところからのコミットメントがあると、そのコミュニティの存在価値って高まるじゃないですか。で、その後ろ盾の中でそのコミュニティが発展してくれればいいというふうには僕には見えないんですよ。正直言って。

やっぱりバズワードとしての広まりもあるので、それもいろんな自然発生的な流れで来ているのかなという気もしますが最初の立ち上げのところで、僕は当時関わってなかったのによくわかりませんが、Google さんがある程度支援したということは確かにあると思うんですけど。だからといって白石さんの話を聞いていても、それは色々相談に乗ってもらっていた程度の話で別に…そういう意味ではコミュニティ活動なんてボランティアで、特定企業に関わっちゃいけないとも思っていますし、あんまりそういう色が見えちゃったら逆にコミュニティ、少なくともああいうオープンなコミュニティではないですよ。

--NTT コミュニケーションズ 小松 健作³³³

コミュニティ入って何かをして、みんなでこういうことを勉強していったら面白いんじゃない、というようにやっているときになると、あつちは近い、こつちは遠いといった距離感は無いですね。

私もダニエルさん(当時 Opera Software に在籍)とはコミュニティの勉強会でお会いして知り合ったのですが、Opera とはなんぞや

³³³ 2011年9月7日実施、小松健作氏インタビュー

というそんな感じではないんですけど、そんなに知っているわけではないんですけど、すごい遠いブラウザベンダーとか、そういう感じはコミュニティの中ではないです。

--株式会社産経デジタル 一條 美和子³³⁴

このように競合する他のブラウザベンダや W3C を巻きこむことで、コミュニティの活動に中立色が加えられている。結果として、オープンな標準としての HTML5 という性格に対する理解が広がり、また複数のブラウザベンダが協力して普及活動を展開するという構図を生み出したのである。

7.6.2.4. コミュニティによる自律的な活動へ

html5j.org が主宰する最大のイベントは「HTML5 Conference」である。2011年に初回開催されたこのイベントは、2013年には参加者 2000 人という規模にまで拡大した³³⁵。中立を標榜しているとはいえ、コミュニティの活動が軌道に乗るまでの裏側には、Google の力があつた。横田はこう語る。

正直そこにお客さんを誘引していく必要があつたけど、Google がつくっているということでお客さんがある程度はいつてきてくれ、なんとか立ち上げに至つた。

--アビダルマ株式会社 横田 昌彦³³⁶

また、コミュニティを作るといっても、継続的な活動としていくためには工夫が必要である。そのためにも様々な施策が打たれてきている。

立ち上げるだけでは ML で終わってしまうので、活動を行った。(その一環として) 単純な勉強会。ハッカソンというものを導入してきた。

³³⁴ 2011 年 9 月 8 日実施、一條美和子氏インタビュー

³³⁵ html5j.org (2013). *HTML5 Conference 2013*. Retrieved March 19, 2014, from <http://events.html5j.org/conference/2013/11/>

³³⁶ 2011 年 2 月 18 日実施、横田昌彦氏インタビュー

最初は Google が始めて、その後はコミュニティが主体（となって運営するようになった）。その後も Google は応援する形になって、色んな会場でやっていく。リアルな場、ワーキングスペースがあればやっていく。実際に API の技術を作って動くものがあったから、コミュニティは活性化していった。

当時は日本でハッカソンという言葉はなかった、Google の中で社内研修でやっていた。アメリカのやり方をそのまま日本にもってきただけではうまくいかないだろう、（ということで）日本独自にローカライズしたものを作った。Google のオフィスを使って、Google 主体のハッカソンを半年から一年。多くの方が来てくれて、満足していただいた参加者が自発的にハッカソンをやってくれた。

-- アビダルマ株式会社 横田 昌彦³³⁷

このように、アメリカで生まれたハッカソン文化を、日本の実情に合わせた形にアレンジし、継続的に注目を集め、集客を実現できるようにしたことが、コミュニティの拡大や HTML5 の普及につながったといえよう。

石原直樹らの跡を継いで Google Japan で Developer Advocate を務める北村英治のブログによると、

ご存知の方は多いと思いますが、Microsoft でいうところの MVP³³⁸ のような、優秀な外部の開発者を認定する Google Developers Expert という制度があります。GDE の皆さんとの情報交換も重要なお仕事です。

これは以前 Google API Expert と呼ばれていたもので、元々日本発のプログラムが、昨年から全世界共通のプログラムになりました (Naoki Ishihara と Fumi Yamazaki のおかげです!)³³⁹。

³³⁷ 2011年2月18日実施、横田昌彦氏インタビュー

³³⁸ Most Valuable Professional の略称。Microsoft が世界的に展開している外部開発者表彰・支援プログラム。

³³⁹ 北村英治. (2013, April 11). Google の Developer Advocate とはどんな仕事なのか? . *Tender Surrender*. Retrieved March 19, 2014, from <http://blog.agektmr.com/2013/04/google-developer-advocate.html>

この日本で生まれた API Expert を中心としたデイベロッパー・リレーション・プログラムは、全世界的に実施されることとなったのである。

html5j.org の活動や API Expert など中心メンバーの活動を通じ、日本では技術者の間で HTML5 への理解が広がり、実装に関わる技術者の数も増加した。と同時に Google がいち早く自社サービスや自社ブラウザに実装し、普及を狙った仕様案がいち早く開発者間で認知、支持される要因ともなった。

HTML5 をテーマとした勉強会や、外部開発者、コミュニティとの連携を通じたデイベロッパー・リレーション活動は競合他社を含めて世界中に広がり、W3C 自らが 2012 年にデイベロッパー・リレーション・チームを創設するにまで至っている。

このコミュニティを率いる HTML5 API Expert の一人が白石俊平氏である。白石氏は当初 Open Social の API Expert を任命されていた³⁴⁰が、その後 HTML5 の API Expert に転じる。その経緯を以下のとおりに語っている。

ずっと、*Gears* の本(白石, 2007)を書いていたおかげで、*Gears* で仕事をしていたんです。オフライン RIA (Rich Internet Application)³⁴¹を作るという業務ばかりやっています。

Google Gears のオフライン RIA(仕様)から HTML5 へもそういった要素がいっぱい継承されてるんですけど、(自分が蓄積しているスキルとして)そういうところのノウハウばかりもっていて、ソーシャル技術のほうがかんたん薄くなって行って、これはどうしたものかなあ、と。*Google Gears* っていう技術自体も Google さんに見捨てられているのもよくわかっていて。

明らかに(*Gears* 開発者)メーリングリストの流通量も少ないし。バージョンアップもほとんどされないし。*Google* さんみたいな活発な会社が手がけているプロダクトにしてはあまりにもバージョンアップが(ない)。こんな廃れていきそうな技術のノウハウばかり溜まってただけでどうしよう、と思っていたのが 2009 年 5 月くらいの *Google I/O* で、自社プロダクトの宣伝するのかと思ったら

³⁴⁰ 武藤健志. (2008, June 16). 「Ask The Expert」コミュニティを支援する Google とエキスパートたち. 世界は日本の Google コミュニティの鐘の音を聞くか. *ASCII.jp*. Retrieved March 19, 2014, from <http://ascii.jp/elem/000/000/142/142607/index-2.html>

³⁴¹ デスクトップ上で動作するウェブにアクセスして

HTML5 を取り上げた。調べてみたら Gears をベースにした API がいくつもあるんですよ。

(自分が蓄積してきた)この (Gears の技術をベースにした) ノウハウを殺さないためには、これ(HTML5 の API Expert としての活動)をするしかないと (考えた)。Google さんに頼んで、HTML5 の API Expert にさせてくれ、って頼んで、(そうしたら Google から)「いよ」って。そこからコミュニティを立ち上げてという経緯です。

--html5j.org ファウンダー 白石俊平³⁴²

もともと Gears の技術を用いてビジネスを行っており、本まで出したエンジニアが HTML5 のコミュニティを主宰するという流れは、Gears の開発・供給に対する Google の狙いがあてはまった例といえよう。

7.6.3. マルチステークホルダーによる協働形態の確立

7.6.3.1. Social Web Incubation Group の活動と、外部開発者コミュニティとの連携強化

W3C が公開メーリングリストを設け、外部開発者を策定プロセスに関与させるという試みは DOM Level 1 策定時から行われている(6.2 項参照)。また、テストプログラム開発は WaSP という開発者コミュニティが担っていた時期があり(6.7.4.1 参照)、その後は W3C と NIST との共同事業として進められた。

21 世紀に入ると、Microformats など、仕様そのものを企業や政府・公的機関、IETF や W3C のような標準化機関が関与しない、開発者が草の根で形成するコミュニティが主導して開発・提案する事例が増加していた。

このような状況に対応するため、2009 年 4 月 6 日に Social Web Incubator Group(SWXG)が設立された。Incubator Group (XG)は、ウェブの萌芽的な技術やユースケースに対し、今後標準仕様策定が必要となるか、また既存の標準仕様策定作業との関係性をどうすべきかについて検討する時限的な活動部会である。活動期間は 1 年以内を基本とし、具体的な施策提言を含む報告書の提出が最終的なアウトプットとして想定されている³⁴³³⁴⁴。

³⁴² 2011 年 12 月 14 日実施、白石俊平氏インタビュー

³⁴³ World Wide Web Consortium. (2012, April 19). *Incubator Activity – version April 19, 2012*. Retrieved June 14, 2014 from <http://www.w3.org/2005/Incubator/#XGs>

³⁴⁴ World Wide Web Consortium. (2010, September 23). *About XGs – version September 23, 2010*. Retrieved

SWXG は、Twitter や Facebook 等のソーシャルネットワークサービスの普及に対する W3C の関与と、策定が望まれるソーシャルサービスの開発に適用される標準仕様のフィージビリティを分析することを目的として設置された³⁴⁵。

SWXG の活動は、週に 1 度の電話会議とメーリングリスト・IRC 上での議論によって進められた。電話会議には、Identity/ Profiles/ Privacy といったソーシャルウェブの基幹技術や制度、メタデータ付与機能であるタギング³⁴⁶などソーシャルサービスで広く実装されている機能などについて、開発ならびに普及活動を行っている個人・企業・コミュニティを招いてヒアリングを行い、現状把握をするとともに、W3C との関係性や W3C による標準化の可能性について検討し、その結果が最終報告書³⁴⁷にまとめられた。

ヒアリングは、DiSo³⁴⁸ Project というコミュニティ活動を主催し、activity stream³⁴⁹と呼ばれるライフログデータに付与するメタデータ仕様の開発・提唱を行っている Chris Messina³⁵⁰ や Creative Commons ライセンスメタデータの整備・普及に携わる MIT の Oshani Seneviratne³⁵¹ などを招待して行われている。SWXG の活動は、新たな標準仕様策定活動の開始にはつながらなかったものの、仕様策定の議論により広いステークホルダーの意見を反映させるため、関連する Interest Group (IG)、Business Group (BG)、Community Group (CG)³⁵²といった新たな種類の作業部会の設置につながった(表 11)。このうち IG と CG は、参加が非会員にもオープンとなっている部会であり、BG も正規の会員になるよりも低い費用で参加可能である。IG、BG や CG の設立により、より多くのステークホルダーが仕様策定に関与する体制が整えられたのである。

June 14, 2014 from <http://www.w3.org/2005/Incubator/about.html>

³⁴⁵ World Wide Web Consortium. (2009, April 6). W3C Launches Social Web Incubator Group. *W3C News*. Retrieved June 25, 2014, from <http://www.w3.org/News/2009#item47>

³⁴⁶ ウェブ上でメタデータを付与し、ユーザ間で共有するサービスはソーシャルタギングと呼ばれている。ソーシャルタギングについては、深見&國領(2007)を参照のこと。

³⁴⁷ World Wide Web Consortium. (2010, December 6). *A Standards-based, Open and Privacy-aware Social Web - W3C Incubator Group Report 6th December 2010*. Retrieved June 24, 2014, from <http://www.w3.org/2005/Incubator/socialweb/XGR-socialweb-20101206/>

³⁴⁸ Distributed Social Networking applications のアクリニムで、「分散的に実装される社会ネットワークアプリケーション」の意。

³⁴⁹ Activity Streams 公式ウェブサイト:<http://activitystrea.ms/> (retrived on June 24, 2014)

³⁵⁰ World Wide Web Consortium. (2009, December 2). - *DRAFT - Social Web Incubator Group Teleconference - 02 Dec 2009*. Retrieved June 24, 2014, from <http://www.w3.org/2009/12/02-swxg-minutes.html>

³⁵¹ World Wide Web Consortium. (2010, April 14). - *DRAFT - Social Web Incubator Group Teleconference - 14 Apr 2010*. Retrieved June 24, 2014, from <http://www.w3.org/2010/04/14-swxg-minutes.html>

³⁵² Business Group、Community Group については、こちらのサイトを参照のこと。World Wide Web Consortium. (n. d.). *About W3C Community and Business Groups*. Retrieved December 3, 2013, from <http://www.w3.org/community/about/>

表 11. W3C に設置される作業部会(Group)

種別	機能
Working Group (WG)	標準仕様策定のために設立される作業部会。
Interest Group (IG)	WG における仕様策定の議論に対し、仕様のユーザの立場から提案を行ったり、新たな仕様に関する情報を共有したりすることを目的として設立される部会。基本的には、非会員にも参加資格は開放されており、標準仕様普及のためのマーケティング施策の一環として開設される事例もある。
Community Group (CG)	仕様策定への関与者を拡大するために設けられた部会。参加資格はオープンであり、知的財産権に関する縛りも緩い。議論の内容は仕様策定（つまり、WG）にフィードバックされる。想定する参加者は個人であり、これまで個人やコミュニティ主導で行われてきた仕様開発、普及活動を W3C に取り込むことを念頭に置いている。
Business Group (BG)	技術や仕様ではなく、特定の事業分野、ビジネスモデルをテーマとして活動する部会。非会員でも少額の会費を支払えば参加可能。参加者間の情報共有・交換とともに、仕様策定へのフィードバックを目的として運営される。
Coordination Group (CG)	WG 間や、他標準化団体との間で調整が必要となった際に設けられる部会。

7.6.3.2. オープンなテストプログラム開発プロジェクト

7.6.3.2.1. スポンサーから公式活動へ

XHTML の仕様策定時においても、W3C Team が各ブラウザの実装状況をテストしている。が、HTML5 や関連仕様のカバーする範囲は大変広がっており、追いつかない。そのため、ユーザを巻き込んでテストを行うという試みが提唱された。

Google の Chrome チームに所属する Developer Advocate³⁵³である Paul Irish と Adobe の Web Opener³⁵⁴である Divya Manian が Move the Web Forward というプロジェクトを立ち上げる³⁵⁵。Move the Web Forward のウェブサイトには、以下のよう目標が記述されている。

Our goal is to make it easy for anyone to get started contributing to the platform, whether that's learning more about how it works, teaching others, or writing specs³⁵⁶.

つまり、広く開発者が仕様策定ならびに普及に関与・貢献するという形態の実現が目指されているのである。

そして Move the Web Forward のコミュニティは、Divya Manian が所属する Adobe のサポートを得て 2012 年 6 月にサンフランシスコにてテスト開発をテーマとするハッカソンイベントを開催する³⁵⁷。このイベントには Adobe 社員のみならず、W3C team staff、Google/ Mozilla/ Microsoft という 3 つのブラウザ供給者からも講師やハンズオンスタッフが派遣されている³⁵⁸³⁵⁹。このハッカソンは Test the Web Forward という名称で開催され、以降中国、フランス、オーストラリア、日本などで定期的開催されるようになった (図 32)。

³⁵³ Google の役職名。ディベロッパー・リレーション活動を担当。

³⁵⁴ Adobe 社の役職名。オープンなウェブ標準仕様の開発・普及に貢献する役割。ディベロッパー・リレーション活動担当。

Irish, P. (2011, December 3). Moving The Web Forward. *Paul Irish – I make the www fun*. Retrieved June 25, 2014, from <http://www.paulirish.com/2011/moving-the-web-forward/>

³⁵⁶ Adobe Systems. (2012) *Move The Web Forward*. Retrieved June 25, 2014, from <http://movethewebforward.org/>

³⁵⁷ Test The Web Forward <http://rhauck.github.io/testtwf-2013/#/10>

³⁵⁸ World Wide Web Consortium. (2012). Speakers. *Test the Web Forward San Francisco*. Retrieved June 25, 2014, from <http://testthewebforward.org/events/2012/sanfrancisco.html#speakers>

³⁵⁹ World Wide Web Consortium. (2012). Experts. *Test the Web Forward San Francisco*. Retrieved June 25, 2014, from <http://testthewebforward.org/events/2012/sanfrancisco.html#experts>

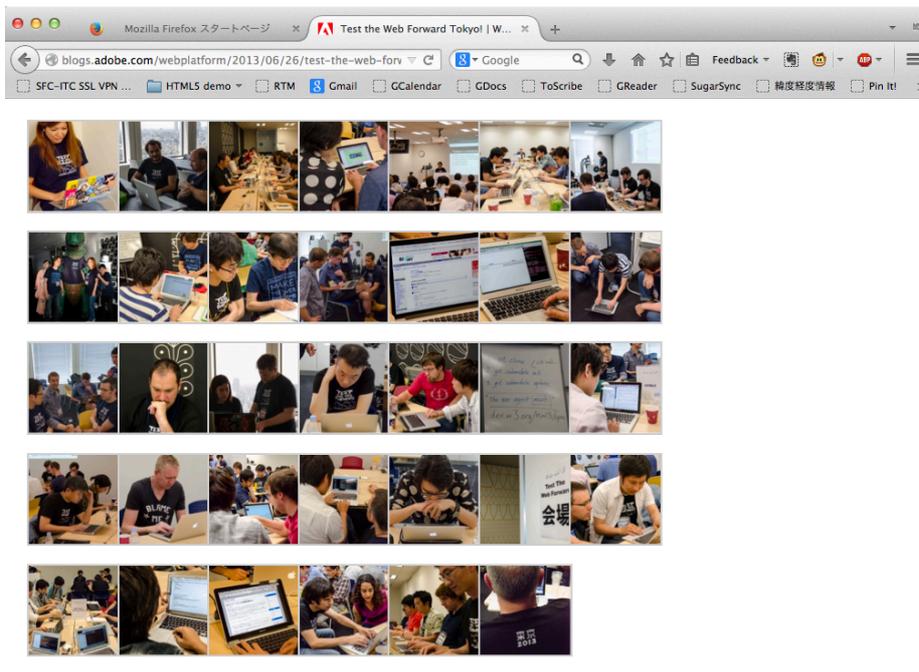


図 32. Test the Web Forward Tokyo 2013 の様子³⁶⁰

2012年10月からは、W3Cがこのイベントをスポンサードするようになる³⁶¹。
2013年10月、Test the Web Forward の活動は W3C の公式活動となった³⁶²³⁶³。

7.6.3.2.2. OSS 開発ツールの積極的活用

Test the Web Forward は、元々OSS 開発者としても活躍していた Paul Irish と Divya Manian が中心になって始められた活動を起源に持つ。それだけに、テストの開発、実施報告には OSS 開発で用いられるツールが活用される。その1つが GitHub³⁶⁴である。

GitHub は、Linus Torvalds らが Linux カーネルのソースコードのバージョン管理するためのリポジトリとして開発された git をウェブサービスとして提供する

³⁶⁰ Hauck, R. (2013, June 26). Test the Web Forward Tokyo!. *Web Platform Team Blog*. Retrieved June 25, 2014, from <http://blogs.adobe.com/webplatform/2013/06/26/test-the-web-forward-tokyo/>

³⁶¹ Berjon, R. (2012, October 1). Test The Web Forward. *W3C Blog*. Retrieved June 25, 2014, from <http://www.w3.org/blog/2012/10/test-the-web-forward/>

³⁶² Langel, T. (2013, October 30). Welcoming Test the Web Forward to W3C. *W3C Blog*. Retrieved June 25, 2014, from <http://www.w3.org/blog/2013/10/welcoming-testtwf-to-w3c/>

³⁶³ Hauck, R. (2013, October 30). Test the Web Forward: Now Part of the W3C. *Web Platform Team Blog*. Retrieved June 25, 2014, from <http://blogs.adobe.com/webplatform/2013/10/30/test-the-web-forward-now-part-of-the-w3c/>

³⁶⁴ <https://github.com/>

サービスとして誕生した。現在では、Linux カーネルのみならず、他のソフトウェアのコードや、HTML や Word ファイルなど、あらゆるファイルのバージョン管理を行うことが可能なプラットフォームとして機能している。なお、GitHub は主に OSS 開発に用いられているが、クローズドな開発プロジェクトにも活用可能である。また、運営は GitHub inc. という営利企業によって行われている。

Test the Web Forward で制作されたテストは GitHub の w3c/web-platform-tests リポジトリに登録され、管理される³⁶⁵。また、テストのコードのみならず、テストコード作成のための教材も GitHub にて公開されている³⁶⁶。

元々 W3C では、Mozilla Foundation が開発したバグ管理システムである Bugzilla を仕様策定プロセスにおいて利用している³⁶⁷。仕様策定の議論が公開の IRC やメーリングリストで進められ、バグの管理や解消、ならびにテストコードの開発・共有も OSS に由来するオープンな環境で行われることで、仕様策定参加者の多様性が向上するとともに、標準仕様のプロモーションとしても機能している。

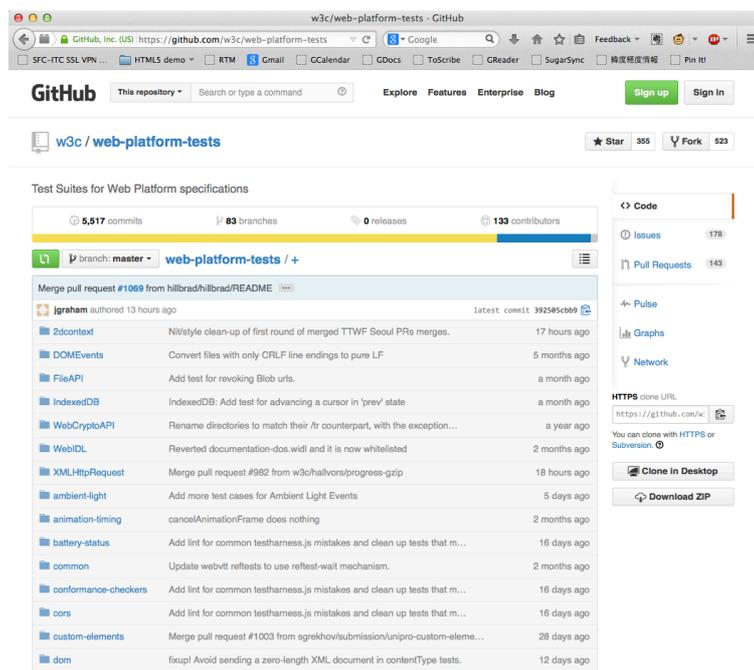


図 33. GitHub の Test the Web Forward 等、W3C 標準仕様テストのリポジトリ³⁶⁸

³⁶⁵ w3c/web-platform-tests · GitHub. Retrieved June 25, 2014, from <https://github.com/w3c/web-platform-tests>

³⁶⁶ Rebecca Hauck, R. (n. d.) Retrieved June 25, 2014, from <http://adobe.github.io/web-platform/presentations/test-the-web-forward-tutorial/index.html#/>

³⁶⁷ W3C が運用する Bugzilla の Main Page. Retrieved June 25, 2014, from <https://www.w3.org/Bugs/Public/>

³⁶⁸ w3c/web-platform-tests · GitHub. Retrieved June 25, 2014, from <https://github.com/w3c/web-platform-tests>

7.6.3.3. webplatform.org を通じた普及活動における協働

2012年10月20日、W3CはGoogle、Microsoft、Mozilla、Operaといったブラウザ供給者やAdobe、Facebookなど³⁶⁹と共同で、webplatform.org³⁷⁰というサイトを開設した³⁷¹。

HTML5の普及には仕様提案・開発に関与する各企業の自発的な宣伝やDeveloper Relation活動が大きな影響を及ぼしてきた。この状態についてW3C Interaction Domain LeaderのRalph Swickは、

But, you know, after all, for the past 15 years, every single browser vendors, they have their own documentation of what HTML is. If you go to the Microsoft website, you are going to have documentation for what is inside Microsoft, Internet Explorer. That's their definition of what HTML is, from their point of view. All of here is to put allow those people on the same table, and get them to work together to agree on a subset of that. So why are they introducing new HTML. Which is going to be adopted on that, we don't know. If they get adopted, we'll put them standards, is don't get about it, that will not³⁷².

とインタビューに回答している。仕様策定に関与する各社は自身の提案を、まず自ら実装し、それを自らのサイトで宣伝する。実際に他者による実装事例が増えていけば、その段階で標準になるといったスキームであったのである。

webplatform.orgは、策定中のものも含め、各者がバラバラに、分散して提供してきた技術情報を集約するために設置された。これにより、開発者がウェブアプリケーションやウェブサイトを作成・開発する際にかかる情報探索コストが軽減される。情報発信を共同で行うことにより、新たな仕様の普及を促進できる効果が狙われている。

Test the Web Forward や webplatform.org など、HTML5の仕様策定中に導入さ

³⁶⁹ その他の参加企業・団体はこちらを参照のこと：<http://www.webplatform.org/stewards/> Retrieved on November 5, 2014.

³⁷⁰ WebPlatform.org Retrieved November 5, 2014, from <http://www.webplatform.org/>

³⁷¹ World Wide Web Consortium. (2012, October 8). *Announcing Web Platform Docs: Alpha release of community resource for developers and designers who use open web standards*. Retrieved November 5, 2014, from <http://www.w3.org/2012/10/webplatform.html>

³⁷² 2012年8月22日 Philippe Le Hégarret氏インタビュー

れた試みは、標準化団体である W3C と仕様策定に関与する会員企業・団体、そしてウェブアプリケーション・コンテンツ開発者を含めた協働形態を強化するものであった。それが仕様の迅速な普及と分裂の回避、つまり新たな仕様の実効性確立に寄与してきたと考えられる。

7.7. ステークホルダーの協調体制確立

7.7.1. Microsoft が HTML5 を実装できない事情

WHATWG によって開発が進められてきた HTML5 が W3C の標準化プロセスに持ち込まれるにあたって、初代 HTML WG Co-Chair であり、Microsoft に所属する Chris Wilson は重要な役割を果たしてきた。

WHATWG 陣営は W3C で HTML WG が再発足する以前から、Microsoft をいかに巻き込むかに心を砕いていた。Microsoft に所属する Chris Wilson に WHATWG 参加を打診³⁷³していた。WHATWG への参加は断られたが、その理由として、

I said something to the effect of “I agree with your goals, all except standardizing error-correction”. However, I also said “you have no patent policy, and that makes it impossible for me to join³⁷⁴.”*

と回答し、WHATWG にパテントポリシーの規定がないことが問題であり、W3C が「ラバースタンプを押す」役割を負う必要があると述べている。結果として Chris Wilson は、新 HTML WG の初代 Co-Chair に就任している。

しかし、HTML5 仕様策定に深く関与する社員が存在することが、そのまま自社製品に仕様を実装することにはつながらない。W3C HTML WG 設立当初の方針決定をめぐる議論において、Wilson は IE によって HTML5 で作成されたサイトを表示するために、モードを切り替える機能を実装する方針であることを示している³⁷⁵。

このような方針が示された背景には、IE が他社のブラウザと比較して、後方互換性を確保するのに多大なりソースが必要という状況があった。IE は、1990

³⁷³ Wilson, C. (2007, January 10). You, me and the W3C (aka Reinventing HTML. *Albatross!*, MSDN Blogs . Retrieved March 7, 2014, from <http://blogs.msdn.com/b/cwilso/archive/2007/01/10/you-me-and-the-w3c-aka-reinventing-html.aspx>

³⁷⁴ *ibid.*

³⁷⁵ Wilson, C. (2007, April 12). *Versioning and html[5]*. Retrieved March 7, 2014, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2007Apr/0612.html>

年代後半には市場シェアが 90%を超える時期があるほど、広く普及していた。また、Netscape Navigator との間で繰り広げられたブラウザ戦争の結果、W3C 標準に含まれない独自の拡張仕様が実装されたり、アドオンソフトが供給され、動作にそういった仕様が必須となるシステムやアプリケーションが開発、供給されたりしてきた。このようなシステムやアプリケーションを動かし続けたいというニーズに応えながら、HTML5 仕様に含まれる機能を実装するのは難しい作業である。

それだけではない。Microsoft はプロプライエタリなプラグインである Silverlight の開発・普及を進めることでウェブのマルチメディア化、インタラクティブ化に対応しようとしてきた。この機能は HTML5 仕様に追加された `<video>` タグや、関連仕様である様々な API と直接競合するものである。

Silverlight は 2007 年 4 月 30 日にベータバージョンが公開³⁷⁶、9 月 6 日に正式版(バージョン 1.0)が公開された³⁷⁷。こうして Microsoft は W3C において HTML5 策定プロセスに関与しつつも、独自にプロプライエタリなアドオンの開発を進めるといった戦略をとり続けた。

Wilson は、議事進行と他の WG 等で策定されている仕様とのコーディネートといった Chair の役割³⁷⁸を果たすものの、HTML5 の仕様開発に参加し、実装するという方針を Microsoft 全社に採用させるまでは至っていなかったのである。

7.7.2. 最新の(しかし策定が終了していない)標準に対応しない Microsoft への反発

Microsoft が Internet Explorer になかなか実装しない状況は、ウェブサイト開発者間の反発を生んでいた。HTML5 や Cascading Style Sheet Level 3 (CSS3) といった新たに策定が進められている仕様群は、Chrome、Firefox、Opera などでは実装されている一方、IE での実装は進まなかった。そのため、HTML5 や CSS3 が提供する機能を用いてウェブサイトを開発する場合、利用者が多い IE バージョン 6 が対応していないことに対応するため、Progressive Enhancement³⁷⁹、

³⁷⁶ Microsoft. (2007, April 30). *Microsoft Lights Way for Next-Generation Web Development and Design at MIX07*. Retrieved May 4, 2013, from <http://www.microsoft.com/en-us/news/press/2007/apr07/04-30MIX07PR.aspx>

³⁷⁷ Sneath, T. (2007, September 6). Silverlight 1.0 - The Release. *MSDN Blogs*. Retrieved May 4, 2013, from <http://blogs.msdn.com/b/tims/archive/2007/09/06/silverlight-1-0-the-release.aspx>

³⁷⁸ Lilley, C. & Brewer, J. (2008, January 23). Role of the Group Chair. *The Art of Consensus*. Retrieved May 4, 2013, from <http://www.w3.org/Guide/chair-roles>

³⁷⁹ HTML5 や CSS3 が実装済みのブラウザで表示される際には、それら仕様の機能を活かした表示・動作をする一方、実装されていないブラウザでも問題のない形でウェブサイトを構築するという考え方。

Graceful Degradation³⁸⁰、Polyfill³⁸¹といった考え方でウェブサイトを開発するという場合がある。これらは、エンドユーザにとってはどのブラウザでも同様のユーザ体験ができるものの、開発者はそのためにブラウザ別に動作する実装を作りこまなくてはならず、余計なリソースが必要となる。

このような観点から、開発者の間で IE バージョン 6、そして Microsoft に対する反発が広がることとなった。その状況は、ウェブサイトテーマとするメディアのみならず、CNN が記事として取り上げる³⁸²ほどとなる。

ネガティブキャンペーンが行われるようになった。その 1 つが Twitter や Facebook のアイコンに、IE バージョン 6 の利用に反対の意志を示すアイコンを付与するという Twibbon の IE6 Must Die という活動³⁸³である(図 34)。開発者の反発は、IE に対するネガティブキャンペーンであり、Microsoft は対応を迫られることとなった。

7.7.3. Microsoft の方針転換

IE に HTML5 を積極的に実装しないという方針は、ようやく 2010 年 10 月に転換された。10 月 28 日に行われた Microsoft のイベント、Professional Developers Conference において Steve Ballmer CEO が行ったキーノートスピーチで、HTML5 をクロスプラットフォームの基盤とする戦略を発表する

これにより、Microsoft がプロプライエタリなアドオンの普及から、HTML5 に対する注力へと戦略を明確に変更したことが明らかとなった。

この戦略変更の前から Microsoft の HTML5 に対する注力度合いは高まっていた。2009 年 8 月 7 日の IE Program Manager である Adrian Bateman が W3C HTML WG の公開メーリングリストに投稿した当時の仕様案に対するフィードバック³⁸⁴について Google の Mark Pilgrim が評価する³⁸⁵など、Microsoft が既に HTML5 開発に対し、実効的な貢献をしていることが明らかとなった。

³⁸⁰ HTML5 や CSS3 の機能を活かした形でコンテンツを制作するが、非実装ブラウザでも最低限の機能は果たせる設計とするという考え方。

³⁸¹ HTML5 や CSS3 が実装されていないブラウザ、特に Internet Explorer6 では、そのブラウザに独自実装された機能を用いて、他のブラウザと同様の機能・表現を実現するという考え方。

³⁸² Sutter, J. D. (2009, August 6). Web citizens trying to kill Internet Explorer 6. *CNN.com*. Retrieved April 24, 2014, from <http://edition.cnn.com/2009/TECH/08/06/internet.explorer.six/>

³⁸³ IE6 Must Die - Support Campaign. *Twibbon*. Retrieved April 24, 2014, from <http://twibbon.com/support/IE6-Must-Die/>

³⁸⁴ Bateman, A. (2009, August 7). *Feedback on the current editor's draft*. Retrieved May 4, 2013, from public-html@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html/2009Aug/0389.html>

³⁸⁵ Pilgrim, M. (2009, September 16). This Week in HTML5 – Episode 35. *The WHATWG Blog*. Retrieved May 5, 2013, from <http://blog.whatwg.org/this-week-in-html5-episode-35>

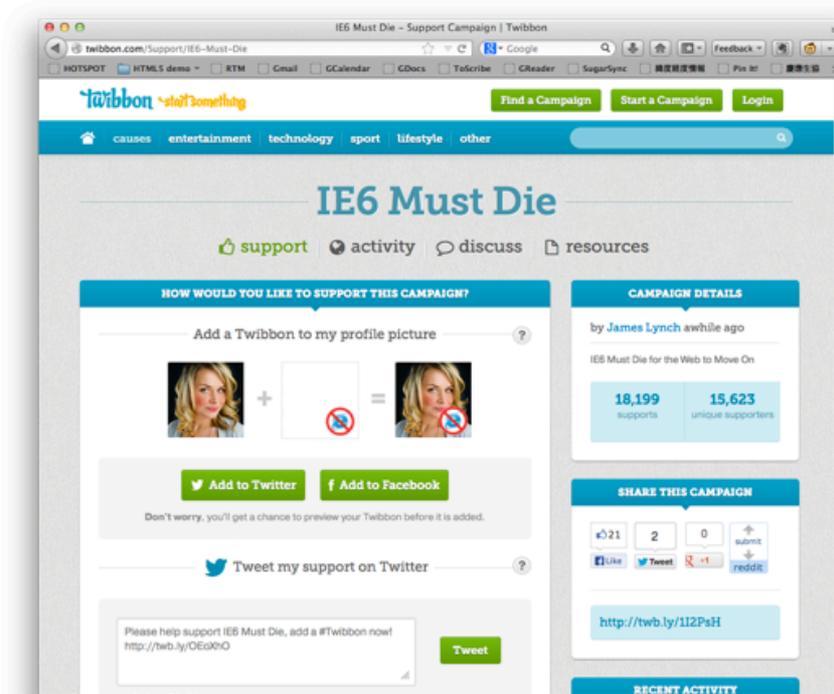


図 34. Twibbon IE6 Must Die³⁸⁶

また、2010年1月にはHTML5の関連仕様である Scalable Vector Graphic (SVG) のWGに加入。2010年6月に公開された SVG 1.1 (Second edition)の Working Draft には、Microsoftの社員名が Editor として記載されており、SVGの仕様策定に明示的に関与しているのがわかる。

仕様開発への関与だけではなく、自社ブラウザ製品であるIEへの実装も進み始めた。2010年3月9日付で公開されたIE開発者ブログの投稿³⁸⁷では、IEの次期バージョンであるIEバージョン9開発におけるHTML5の実装の取り組みや、W3Cの開発プロセスへの関与についてアピールしている。また、2010年5月5日に公開されたW3C HTML WG co-chairであるPaul Cottonのインタビュー³⁸⁸においても、WGに対する取り組みや、IEに対する実装の努力がアピールされている。

2011年9月14日、IEバージョン9の正式版公開時にMicrosoftは、IEより

³⁸⁶ ibid.

³⁸⁷ Bateman, A. (2010, March 9). Working with the HTML5 Community. *IE Blog*. Retrieved May 4, 2013, from <http://blogs.msdn.com/b/ie/archive/2010/03/09/working-with-the-html5-community.aspx>

³⁸⁸ le Phigaret, P. (2010, May 5). Interview: Paul Cotton on Microsoft Participation in the W3C HTML Working Group. *W3C Blog*. Retrieved May 4, 2013, from http://www.w3.org/QA/2010/03/interview_paul_cotton_on_micro.html

も Chrome や Firefox の利用者層が多くを占める³⁸⁹ South by Southwest Interactive Festival (SXSWi) というイベントでリリース記念パーティーを実施し、HTML5 対応機能のデモを実施し³⁹⁰、HTML5 対応が進んでいることをアピールした。IE バージョン 9 のリリースを通じ、Microsoft は HTML5 への本格的な対応を完了し、全ての主要ブラウザ事業者が HTML5 の実装を推し進めることとなり、HTML5 が実効的な標準仕様の地位を確立したのである。なお、2013 年 5 月時点で W3C では HTML5 は最終草案の段階にあり、少なくとも勧告となる 2 年以上前から全主要ブラウザが HTML5 に対応したこととなる。

また Microsoft は古いバージョンの(特に数の多いバージョン 6 を中心とする)ユーザに対し、最新バージョンへの移行を促すキャンペーンを積極的に行うようになる³⁹¹。中でも有名なのは、ウェブサイト開発者が主催して行われた IE バージョン 6 の葬式というイベントに対し、Microsoft が社として献花するというものであろう (図 35)。



図 35. Internet Explorer 6 の葬式というイベントの公式ウェブサイト³⁹²

³⁸⁹ McCarthy, C. (2011, March 9). Microsoft to launch Internet Explorer 9 at SXSWi. *CNET*. Retrieved May 4, 2013, from http://news.cnet.com/8301-13577_3-20041126-36.html

³⁹⁰ McCarthy, C. (2011, March 14). At SXSW, IE9 envisions a happy,appy Web. *CNET*. Retrieved May 4, 2013, from http://www.cnet.com/8301-14013_1-20043132-284.html

³⁹¹ Naraine, R. (2011, March 4). For security reasons alone, you should ditch IE6. *ZDNet*. Retrieved April 24, 2014, from <http://www.zdnet.com/blog/security/for-security-reasons-alone-you-should-ditch-ie6/8294>

³⁹² *IE 6 Funeral*. Retrieved August 13, 2013, from <http://www.ie6countdown.com/>

仕様プロセスが公開され、外部開発者がテストプログラムに参加することによって HTML5 が提供する機能と、当該技術を利用する際に生じるコストに対する理解が広がることによって HTML5 の受容が広がるとともに、まだ仕様策定プロセスが完了していないにもかかわらず、ブラウザ供給者に対する実装圧力が高まった。これにより、全てのブラウザ供給者が自ら標準仕様開発に参加するとともに、競い合うように仕様を実装するようになったのである。

³⁹³ 現在は Microsoft が自ら運営する IE 6 Countdown というサイト (Microsoft. *IE 6 Countdown*. Retrieved April 24, 2014, from <http://www.modern.ie/en-us/ie6countdown>) にリダイレクトされている。このサイトは地域別 Internet Explorer 6 のシェアを表示し、最新バージョンへの移行を促す内容となっている。

第8章. 仮説導出

8.1. 各仕様の実効性獲得状況

ここまで W3C が創設されてから現代に至るまでの仕様策定プロセス運営方針の変化と、HTML、ならびに DOM の仕様策定状況について分析してきた。各仕様の策定期間と、主要ブラウザの実装状況をまとめたのが図 36 である。

これを見ると、W3C 創設初期の HTML3.2、HTML4 が短期間で策定プロセスを完了したものの、ブラウザベンダの独自実装を抑制することはできず、結果的に実効性のある標準仕様の確立には失敗したといえる。

DOM Level 1 は、当時の 2 大主要ブラウザである Microsoft IE と Netscape Navigator が共に実装し、実効性のある標準仕様となった。ただし、仕様策定が進められていた時期は、両社間で激しく争われたブラウザ戦争の終盤であり、Netscape が AOL に吸収合併され、Navigator のソースコードがオープンソース化される時期であった。Microsoft の提案を基に開発された DOM Level 1 の仕様が、市場競争の結果として Netscape に受け入れられたと見ることも可能であり、W3C の標準化プロセス運営が標準仕様の実効性獲得にどの程度貢献したかを推し量るのは難しい。

一方 DOM Level 2 は、Netscape が先行して実装し、それを Microsoft や Opera が追うという形で実装が広がった。つまり市場競争の結果ではなく、W3C における標準化作業が機能したと考えられる。また、各社が実装した時期は、策定プロセスが終了するよりも早いタイミングである。

HTML5 は XHTML + XForms という競合仕様が存在したことにより、当初 W3C 外部での開発作業となった上に、W3C での策定作業開始後も WG 間の対立関係が継続した。しかし、W3C 内での一本化に成功し、全ての主要ブラウザベンダが開発に参加し、自社ブラウザに実装している。HTML5 の仕様策定作業は、厳しい利害調整と普及活動に成功し、標準仕様の実効性獲得に成功した事例と捉えることができる。

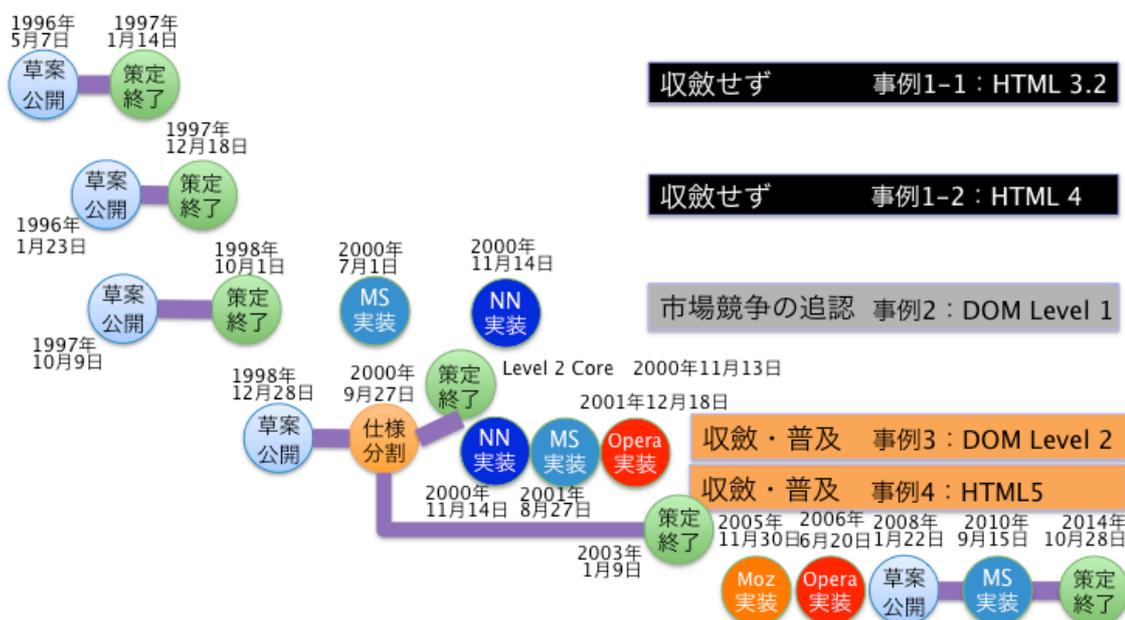


図 36. 各バージョンの策定期間と主要ブラウザの実装時期

8.2. 仕様策定プロセス運営方針の変化と仮説導出

では、各仕様はどのような運営方針の元で策定されたのであろうか。図 37 は、W3C の仕様策定プロセス運営方針において生じた変化をまとめたものである。HTML3.2 から HTML5 策定完了までの 20 年ほどの間に導入された施策は、

- 1) 仕様がカバーするスコープを「機能」で定義する
- 2) 仕様策定プロセスにおける実装主義の導入
- 3) 仕様策定プロセスのオープン化

の 3 つに集約される。図 37 を見る限り、これら 3 つの施策が導入されるに連れ、策定された標準の実効性が増している。そのため、この 3 つの施策が協調型イノベーション成立要因の仮説として導出できる。

8.2.1. 仕様がカバーするスコープを「機能」で定義する

DOM WG の活動開始時に、Level 0/ 1/ 2 という機能をベースにしたスコープ設定が行われた。これにより、仕様がカバーするスコープを「機能」定義するという手法が W3C において定着することとなった。

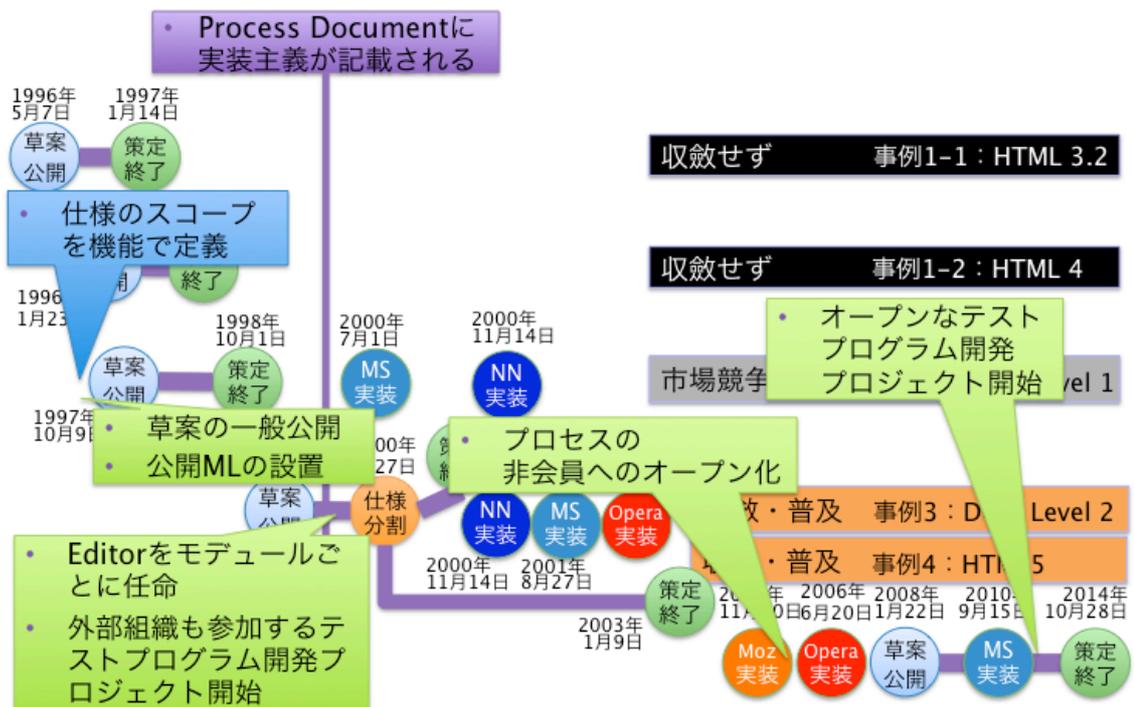


図 37. W3C における仕様策定プロセス運営ルールの変化

Lawrence Lessig は標準化の手法を Coordinating と Regulation に分類している (Lessig, 1999-b)。多くの標準化団体で行われていることは、会員によって提出された仕様(書)を“認証”する形で標準仕様が策定される。例えば ISO においては、OpenOffice.org のファイルフォーマット形式である Open Document Format for Office Applications (ODF) と、Microsoft Office のファイルフォーマットである Office Open XML (OOXML) という2つの全く異なる仕様を標準として採択している。公的機関によって民間企業やコミュニティが開発・提案した仕様を“認証”し、調達要件などに指定することによって普及をコントロールするといった Regulation 標準では、このように同一機能を提供する複数の標準仕様が併存する状態が成立し得る。

一方 Coordinating 標準は、複数の主体・機器間のやりとりをコーディネートすることによってそれまでなかったものを生み出す (Lessig, 1999-b, pp. 759) 機能を果たすものである。存在しなかったものを生み出す、ということはイノベーションの創出にほかならない。この論文において Lessig は IETF によって策定された TCP/IP を Coordinating 標準の例として挙げ、インターネットそのものが Coordinating 標準によって生み出されたものであるとしている (Lessig, 1999-b,

pp. 760)。

当初 W3C は、HTML の標準化を Netscape と Microsoft が既に各自が実装している独自拡張仕様を統一する形で実現しようとした。しかし、その手法は対立する 2 つの企業に対し、製品設計を変更させようとするものであり、regulation 標準を創りだそうとする試みであったといえるだろう。

一方 DOM WG 発足時に導入された手法は、Charter において策定する仕様のスコープを「機能」で定義するというものであった。この手法により、機能を実現する仕様を WG メンバー間の協働によって創りだすこととなった。WG における策定プロセスに参画した各主体の利害を調整して、新たな機能を実現するという手法で生み出された仕様は、まさしく Coordinating 標準である。

この手法が導入された DOM Level 1 の策定プロセスにおいて生じた事象について 0 節に掲載したデータを再掲したい。

Netscape reps had to leave early. They will read level zero ready for discussion next week. We have missed our schedule of first public draft out by April 30.

Action NS - give estimate of when level zero part 1 could be done by

-1997 年 4 月 30 日電話会議の速記録(下線は筆者による)³⁹⁴

Action NS to let us know what items should be changed, by this week's phone call I hope.

-1997 年 5 月 9 日電話会議の速記録(下線は筆者による)³⁹⁵

これらの速記録から、Netscape は電話会議への出席に熱心でなく、アクションリストの実行を急かされている状況であったことがわかる。DOM Level 1 は Microsoft が主導権をもって策定されていた仕様であったものの、ブラウザ戦争で激しく競合していた Netscape による対案の提出などの目立った動きは見られ

³⁹⁴ Wood, L. (1997, April 30). *minutes from teleconference*. Retrieved June 1, 2014, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0049.html>

³⁹⁵ Wood, L. (1997, May 13). *minutes from teleconference 19970509*. Retrieved June 1, 2014, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0061.html>

ず、最終的に標準仕様を実装している。こうしたことから、Netscape が取り得た行動は議事進行やアクションリストの消化に対するサボタージュくらいとなった。また、

Since we won't be doing the window model in level one, maybe it shouldn't be in the requirements.

-1997年5月9日電話会議の速記録⁹⁶

といった発言がされているとおり、スコープから外れた仕様提案は基本的に却下されるという運用がされている。これにより、やみくもな独自仕様提案の発生の抑制されるようになったと考えられる。

スコープを「機能」で定義するという手法が導入されたことによって、同一の機能を実現する仕様は必然的に同一の WG で議論され、複数の仕様が提案された場合は統一された仕様のみが標準として採択される(図 38)こととなり、それが、仕様の収斂に寄与することとなったのである。

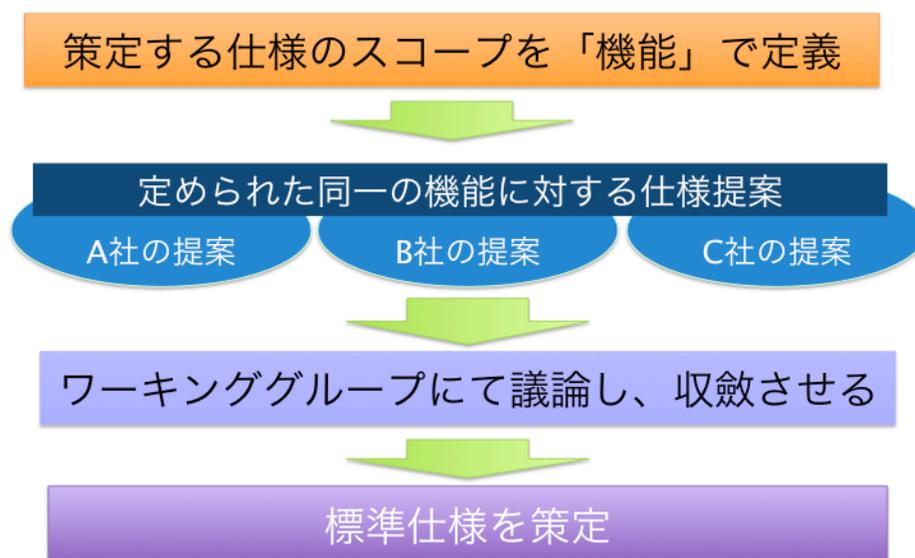


図 38. 仕様のスコープを「機能」で定義した場合のプロセス

³⁹⁶ Wood, L. (1997, May 13). *mintues from teleconference 19970509*. Retrieved June 1, 2014, from w3c-dom-wg@w3.org mailing list archive: <https://lists.w3.org/Archives/Member/w3c-dom-wg/1997AprJun/0061.html>

8.2.2. 仕様策定プロセスにおける実装主義の導入

実装主義は、「独立した複数の実装事例の存在が標準として認証されたために必要とされる」というルールであり、2005年10月より導入されたものである。Process Document 上の根拠は、以下の箇所である。

Shown that each feature of the technical report has been implemented. Preferably, the Working Group should be able to demonstrate two interoperable implementations of each feature³⁹⁷. (下線筆者)

実装主義は6.6.3項に記載したW3CのHead of CommunicationのIan Jacobsや元DOM WGのChairも務めていた現Interaction Domain LeaderのPhilippe Le Hégarretのインタビューにおいて述べられているとおり、仕様に対する開発者からのフィードバックを得るという目的のために導入された。

このルールが導入されたことにより提案仕様は、提案者以外のステークホルダーによる実装事例が創出されることが必要となる。これにより仕様提案者は、自ら実装事例を公開するとともに、他のステークホルダーが実装事例を作り出すことを促すようになった。その過程で生じた現象が7.6.2項で分析したステークホルダーによるディベロッパー・リレーション活動である。仕様提案者、ならびに提案仕様に賛同する企業・団体が、社外の開発者に対して自発的に情報提供や技術教育等を行い、社外の第三者による実装事例を増やそうと試みるようになった。

多くの標準仕様は、仕様の中身が確定した後に、準拠する製品・サービスが開発・供給されるという流れで策定・普及していく。実装主義の導入によって、仕様確定の遙か以前の段階で実装事例が公開され、フィードバックが寄せられてくる。フィードバックを基に仕様がブラッシュアップされ、改善された実装事例が公開されるというサイクルが繰り返され、開発者のニーズに即した形で仕様が確立していったものが標準になる(図39)。フィードバックを受けながら洗練させるということはSanchez and Sudharshan (1993)が指摘するように、市場調査を行いながら開発を進める手法を採用している事にもなる。

³⁹⁷ Jacobs, I. (ed.). (2005, October 14). 7.4.4 Call for Review of a Proposed Recommendation. *World Wide Web Consortium Process Document 14 October 2005*. Retrieved November 28, 2013, from <http://www.w3.org/2005/10/Process-20051014/tr.html#cftr>

- これまでの標準仕様策定プロセス



- W3Cにおける実装主義採用プロセス

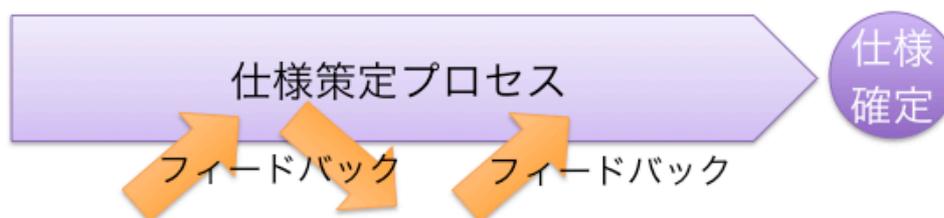


図 39. 実装主義導入による仕様策定プロセスの変化

早期の段階で実装事例が増加し、かつ実装事例をもとにした議論によって仕様は洗練、集約されることから、ネットワーク外部性効果により集約された仕様がそのまま普及する可能性が高くなる。

W3C Head of Communication の Ian Jacobs は、インタビューにおいて以下の通りに語っている。

So one of the challenges is though, part of the expectation is giving broad support. And if you show up and you're the only person who does something even if you have implemented it perfectly, it would be hard to call that a standard, because no one else is using it. So you may, if you show up and 50 companies already supported, and you have a community already, then you satisfied the expectations of broad review. Now, you may want to get more review, but if you do your homework in a community group for example, get your community together, get lots of support, and you show up on the recommendation track, and you've done all your work,

you should be able to go quickly thorough your recommendation track process³⁹⁸.

つまり、完璧な仕様が策定されていても、他者からの支持がなければ標準とは呼べない。多数の支持を獲得できれば、標準化プロセスは迅速に進むというわけである。重要なのはその仕様が実現する機能に需要があるか、そして仕様がより広く用いられるかである。実装主義は実装事例の数とフィードバックにより標準が選定されるプロセスである。Jacobs の発言は、実装主義の本質を示している。

実装主義は、提案仕様に関する情報の迅速な拡散、実装事例の増加、フィードバックの発生とブラッシュアップの実現を促すという効果をもつと考えられる。つまりネットワーク外部性の特徴を活かして、実効性のある標準を早期に確立する手法だといえよう。

更には 7.6.1.2 で分析したとおり、標準仕様に組み込むことを意図した機能をあらかじめオープンソースのプラグインとして開発・提供し、利用が広がるとともにユーザが機能の有用性を認識した段階で、当該機能を実現する仕様を標準化提案するという事例も登場した。標準仕様提案にイノベーションを実現する機能が提案され、かつ仕様の確定を待たずして普及することとなったのである。

第三者による実装事例の増加は、提案者ではない事業者に対して採用を促す効果がある。仕様策定に直接関与するブラウザ供給者などの企業に属さない開発者にとって、ニーズのある機能を実現する仕様が動作しないブラウザが存在することはコスト要因となる。そのため 7.7.2 項で分析したとおり、ウェブサイト開発者によるブラウザ供給者に対する標準採用への圧力が醸成される。つまり実装主義は、独自仕様の開発・導入を抑制するとともに、標準仕様の普及をもたらすのである。

紐付けやプラットフォーム包囲戦略は、基本的にはプロプライエタリな独自仕様の開発を通じて実行される。実際に Google は、自社独自のブラウザである Chrome を市場投入している。従来の紐付け戦略であれば、Chrome にウェブアプリケーションの機能を実現する独自仕様を実装することで競合ブラウザとの差別化をはかったであろう。しかし実際に Google がとった行動は、7.6.1.2 目で

³⁹⁸ 2012 年 9 月 12 日 Ian Jacobs 氏インタビュー

分析したとおり自社で開発した技術をオープンソースとして公開・供給し、その後そのまま標準化提案している。ブラウザが無償で供給されている状況下においては、自社ブラウザの市場占有率を高めること自体が収益を高めることには繋がらない。

そのため、既存の独自実装(Gears)の仕様にこだわる必要性は少なく、仕様は開発者のニーズに即してブラッシュアップされる方が望ましい。また、Microsoft など競合企業によって採用されることが重要であり、パテント収入がもたらされる必要はない。仕様の中身をコントロールしたり、技術からのパテント収入を期待したりしない限り、実装主義は仕様の普及とフィードバックによる洗練化を促進するため、望ましいこととなる。

それよりも、自らが隣接レイヤにおいて差別化を図りやすい技術を標準に組み込んでおけば、実質的に紐付け戦略を成立させることができる。レイヤード・モデューラー・アーキテクチャにおいて有力な戦略、紐付けとプラットフォーム包囲が採用される状況となった場合、実装主義はドミナントデザインの移行をもたらすイノベーションを実現する技術を標準化提案するインセンティブとして機能しているといえよう。

ただし、実装主義の導入は必ずしも標準化プロセスの短縮にはつながらない。図 36 に各仕様の WD が初めて一般公開されてから、Recommendation に到達するまでの期間を記載している。各仕様の所要期間は、HTML3.2 で 8 ヶ月、HTML4.0 で 10 ヶ月、DOM1 で 12 ヶ月に対し、DOM2 で 4 年、HTML5 は 6 年 10 ヶ月も要している。もちろん仕様が公開される以前から策定作業は介されているわけであるので、作業開始時期を起点にして改めて策定期間を算出すると、HTML3.2 で約 1 年半³⁹⁹、HTML4 は約 2 年⁴⁰⁰で標準化プロセスが終了している。一方 HTML4 の次期バージョンの開発には、XHTML の開発が提唱されたタイミング⁴⁰¹を起点とすると 16 年 10 ヶ月をかかっている。DOM も Level 1 が約 2 年⁴⁰²でプロセスを完了しているのに対し、Level 2 は DOM Level 2 HTML が Recommendation となるまでには 4 年以上⁴⁰³かかっている。結局後になって策定

³⁹⁹ 1995 年 8 月の HTML3.0 Internet Draft の効力喪失から、1997 年 1 月に HTML3.2 が Recommendation となるまでの期間として算出。

⁴⁰⁰ 1996 年 1 月の Dave Ragget による提案から、1997 年 12 月の HTML4 が Recommendation になるまでの期間として算出。

⁴⁰¹ 1998 年 1 月 28 日に Dave Ragget から XHTML の原案が提示されたのを起点とする。

⁴⁰² 1996 年 9 月に行われた Scott Issacs による提案から、1998 年 10 月の Recommendation までを期間として算出。

⁴⁰³ 1998 年 10 月に行われた Microsoft による HTML Components の提案から、2003 年 1 月 9 日の DOM

された、実装主義が導入されて以降の仕様の方が、策定プロセスが長期化しているのである。

多くの技術仕様は、バージョンが上がるに連れて機能が増加する傾向にあるため、その開発・策定に時間がかかる。このことを考慮に入れたとしても、実装主義導入は仕様策定プロセスを長期化させる傾向にあるという指摘を否定することはできない。

しかし、これをもって実装主義の導入は策定プロセスの非効率化につながっているという結論とはならない。なぜなら、実装主義プロセスにおいては、準拠するサイトやアプリケーションの供給が仕様確定の遥か以前から始まるため、新たな仕様の普及は迅速に進むからである。

2012年12月にHTML5はCRのステータスに達したが、その際W3Cから出されたプレスリリースには、以下の様な記載がある。

.....
*The World Wide Web Consortium (W3C) published today the complete
.....
definition of the HTML5 and Canvas 2D specifications. Though not yet W3C
.....
standards, these specifications are now feature complete, meaning businesses
and developers have a stable target for implementation and planning⁴⁰⁴.*
(傍点筆者)

つまり、CRとなった時点で、「仕様策定完了」「機能的に完成した」⁴⁰⁵状態となるのである。

なぜこのようなことが言えるのか。それは、CR以降のプロセスはWGを離れ、WG非参加の企業・団体に向けた実装の呼びかけと、フィードバック対応が主となるからである。もともと当該仕様に興味関心がある、もしくは当該仕様の内容がビジネスに大きく影響する企業・団体は、通常WGに参加している。そのため主なステークホルダー間の合意は、Last Call for Commentによって寄せられたフィードバックに対応できれば、達成されたとみなすことができる。

Level 2 HTMLがRecommendationとなるまでの期間として算出。

⁴⁰⁴ World Wide Web Consortium. (2012, December 12). *HTML5 Definition Complete, W3C Moves to Interoperability Testing and Performance*. Retrieved December 3, 2013, from <http://www.w3.org/2012/12/html5-cr.html.en>

⁴⁰⁵ 上記プレスリリースの日本語版表記の抜粋: W3C プレスリリース
World Wide Web Consortium. (2012, December 12). *W3CがHTML5仕様策定完了を発表、相互運用性テストおよびパフォーマンス向上へ*. Retrieved December 3, 2013, from <http://www.w3.org/2012/12/html5-cr.html.ja>

また、仕様書のステータスが CR に到達した後、WG 外部の企業・団体からのフィードバックによって仕様内容を変更することになれば、それまでの実装事例を作り直すというコストが発生する。そのコストを避けるためにも、できるだけ早く広く開発者に受容されるように、仕様の内容を洗練させるというインセンティブが働く。

また提案者は、提案者以外の事業者や、広く一般に対して提案仕様の実装を促す傾向にある。というのも実装事例が増えた場合、その仕様の普及可能性が実証されたとも解釈可能であるとともに、既に存在する実装を変更させることに対するコストを忌避するインセンティブが働くためである。

また、W3C の意思決定は多数決によらないため、全てのステークホルダーが少なくとも反論しない仕様にする必要がある。自分が一方的に利得を得られるような仕様に固執する場合、W3C を通じて標準化するという選択肢は選ばれない。

このような構造があるために、W3C における標準化プロセスは長期間にわたるものの、早期に仕様が収斂し、ブラッシュアップがなされる傾向にある。また競合する独自仕様を、プロセスが進んだ後のタイミングで提案したり、独自に実装したりするという行動が抑制されるのである。

8.2.3. プロセスのオープン化

仕様を用いてウェブアプリケーションやコンテンツを作り出す外部開発者の間で、提案・議論されている仕様案に対する理解が広がる要因となったのは仕様策定プロセスのオープン化である。

プロセスがオープンになったことにより、策定途上の仕様案が広く公開されることとなった。DOM Level 1 の策定プロセス以来、W3C の標準化プロセスは継続的にオープン化されてきたが、実装主義が導入された DOM Level 2 の策定プロセスでは、Call for Implementation に対して公開メーリングリスト経由で実装事例の報告が寄せられるに至った⁴⁰⁶。また、DOM Level 2 が CR ステータスに進む際に Chair から公開メーリングリストに対して以下の様な報告がなされている。

We had quite a lot of feedback on the DOM Level 2 CR and made a

⁴⁰⁶ Ball, S. (2000, April 3). *DOM Level 2 Implementation*. Retrieved August 7, 2014 from www-dom@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/www-dom/2000AprJun/0000.html>

lot of clarifications to the spec. (Still in the process, actually).

Some of the feedback we received convinced us that the internal subset string that we added in the CR draft (after feedback on the Last Call draft) was more trouble than it was worth, particularly since we're starting work on Level 3, which will address the issue properly. So we're taking out the internalSubset parameter on createDocumentType⁴⁰⁷.

このように、プロセスのオープン化は、外部開発者による実装事例の創出を促進し、フィードバックを元にした変更が行われ、仕様の受容性拡大、つまり標準の普及につながっているのである。

また実装事例創出のみならず、Test the Web Forward 等を通じ、策定プロセスへ「参加」する外部開発者が拡大されてきた。W3C において HTML5 の策定を管轄する Interaction Domain Leader である Philippe Le Hégarret は、インタビューにおいて以下のように述べ、

But that was major change that we do, so if you compare specific on HTML4, we never tested HTML. We didn't have a testbed. But we didn't have any implementations, report for HTML4. Because that was really in the late, ah, 1990s. You know. Nowadays we're trying to ship HTML5, and we need other test, which improves test for the feature that comes the differentiation from the HTML4, which we didn't other test at the top⁴⁰⁸.

策定プロセスにおける HTML5 と HTML4 の差異の中でも、テスト開発が重要なポイントの1つであったと指摘している。

新システムの普及には、ユーザ間で実用性と利用する際の簡便性が認識され

⁴⁰⁷ Wood, L. (2000, February 4). *Level 2 and moving towards PR*. Retrieved August 7, 2014 from [www-dom@w3.org mailing list archive: http://lists.w3.org/Archives/Public/www-dom/2000JanMar/0074.html](http://lists.w3.org/Archives/Public/www-dom/2000JanMar/0074.html)

⁴⁰⁸ 2012年8月22日 Philippe Le Hégarret 氏インタビュー

ることが必要であることが先行研究でも指摘されている(Davis, 1989; Gefen and Keil, 1998)。また、開発プロセスへの参加が完成したシステムの満足度向上につながるという研究結果も存在する (Ives & Olson, 1984)。

とはいえ、満足度の向上には単に開発に参加するかどうかだけではなく、割り当てられるタスクの複雑さの調整、システム設計自体の複雑さの調整、ユーザのもつ影響力、ユーザとの間のコミュニケーションが必要である (McKeen et al., 1994)。つまり、システム設計時にはユーザのニーズを把握することが極めて重要であり、供給者・ユーザ間 (Habermeier, 1990)/ 開発者・ユーザ間 (Leonard-Barton & Sinha, 1993) インタラク션을成立させることが必要ということになる。言い換えれば、ユーザも加わったイノベーション・コミュニティ (Lynn et al., 1996) の形成が必要である。そのため、単にプロセスをオープンにするだけではなく、外部開発者の関与を促進する施策を講じる必要がある。

W3CのHead of CommunicationであるIan Jacobsはインタビューにおいて以下のように回答し、仕様書の作成・公開だけではなく、外部開発者に対する情報提供の重要性を指摘している。

Well, I think the number of things typically happen, and one is there are good implementations, there are several implementations, there's developer support, there's good documentation, test suites help people write good implementations, and so there's a whole eco-system of materials outside the standard, that are important to making the standard a success. So traditionally, W3C has focused on the standard, and less on the other materials. So we have had tests, since for a long time, but I think we have more testing now, we recognize how important it is, especially, it's the web platform, spreads to other devices, we need more testing. So, W3C has an organization, is valuable to people, when doing useful things so if the members see W3C has good place to bring all of their tests, to a neutral forum that has a good history of being trustworthy, then we become a valuable organization that it is also repository of tests. So that makes us valuable to the members. If we're the place where people bring documentations, and say "ah, we want a neutral forum for managing web

documentation” that makes us a valuable central place for people to come that can trust us, and that adds values to the members because they see W3C as being a place where developers go, where testers go, where standards development people go, and so all these things that we do well, add value to the organization⁴⁰⁹.

当初開発者へのアプローチは各社・団体が個別に行っていたが、W3C と会員企業・団体が協働して情報提供する webplatform.org の活動が開始された。また、Test the Web Forward などテストプログラム開発プロジェクトが開始された。これらの施策により、より効率的に開発者に対するアプローチが行われるようになり、また開発者間の仕様に対する理解が促進した。多様な形態での情報提供や、仕様策定プロセスへの参加を通じて外部開発者は策定仕様への理解を深め、それが普及につながっているといえよう。

外部開発者は策定された仕様を使ってコンテンツやアプリケーションを作り出す役割を担う。外部開発者が実装を通じてイノベーションに参加しているということは、本論文で分析してきた事例はユーザーイノベーション(von Hippel, 1976)としての性格を有していることとなる。

しかし、Microsoft、Google、Apple、IBM 等大企業のトップエンジニアらによって提案、ブラッシュアップされた仕様のテスト開発や、彼らに対するフィードバックを行うにあたっての心理的、技術的障壁は低くない。

一般的なエンジニアがハイレベルな開発プロジェクトへ参加するには高い障壁が存在する。Linux のコアカーネルなど OSS 開発コミュニティへの参加も、やはり万人にとって容易なものではない。von Krogh et al. (2003) は、オープンソース・ソフトウェアの開発に参加するためには、既に記述されたソースを読み込み、理解を深めた上で自身の専門性を高めてアピールしなければならないなどといったハードルが存在することを指摘している。オープン標準は、OSS と同じ問題を抱えているといえる。単に策定プロセスをオープンにするだけでは、積極的に関与するユーザを増加させるには不十分なのである。

7.6.2 項で分析したとおり、実装主義のもとでプロセスがオープン化されたことによって Google 等ブラウザ供給者を中心としたステークホルダーが自主的に外部開発者＝ユーザがイノベーションへ参加することを促進する施策を講じる

⁴⁰⁹ 2012 年 9 月 12 日 Ian Jacobs 氏インタビュー

ようになった。更に W3C 自身もテスト開発のための情報交換と共同作業のツールとして GitHub などの OSS 開発で普及するツールを活用する。そして、積極的な技術情報や自社開発ツールの提供によって技術的障壁を下げるといった対処を進めてきたのである。

Evans and Wolf (2005)は、企業と外部エンジニアコミュニティとの協働を成立させるためには、サーバ等技術基盤の提供、貢献の透明性確保、信頼の醸成、プログラムのモジュール化、一体感の醸成といった施策が必要と指摘している。W3C は Test the web forward において GitHub 等、OSS 開発で実績あるツールを採用することにより、この要件を満たしている。

OSS 開発プロジェクトへの参加動機には、ソースコードオープン化という方針への賛同 (Lakhani & Wolf, 2005) という信条のみならず、自身の技術者としての評判形成やコミュニティ内でのステータス獲得 (Roberts, et al., 2006; Hars & Ou, 2002)、ビジネスにおける必要性や自身の技術向上・成長 (Hars & Ou, 2002; Lakhani & Wolf, 2005) 、人脈形成 (Hars & Ou, 2002) 、自身が利用するソフトウェアの機能向上 (Hertel et al., 2003) といった利己的な理由があることが先行研究によって指摘されているが、W3C や会員企業・団体が講じてきた施策は、これらの先行研究によって指摘されてきた動機付けのポイントがそのまま活かされているといえよう。

また Google の API エキスパートなど、ブラウザ供給者がそれぞれ展開したディベロッパー・リレーション活動における表彰制度は、ユーザの中でも先進的な立場にあるリードユーザー(von Hippel, 1986)の発掘と彼らの貢献を促進することに寄与してきた。プロセスのオープン化は、単にユーザーイノベーションを成立させるだけでなく、ステークホルダーによる自発的なリードユーザーの発掘と活動支援施策の実行を促す効果も発揮したのである。

第9章. イノベーションの源泉拡張と、組織間の垣根を超えた協働

9.1. 同一企業内での方針不統一

ここまで、HTML5 ならびに先行仕様の実効性獲得状況と W3C の標準化運営方針の変化を分析することにより、協調型イノベーションを成立させる要因の仮説を導出してきた。

仮説として導出された 3 つの要因、1) 仕様がカバーするスコープを「機能」で定義する、2) 仕様策定プロセスにおける実装主義の導入、3) 仕様策定プロセスのオープン化が、単にステークホルダー間の利害調整を通じた仕様の収斂や普及に寄与するのみならず、より広くイノベーションを誘発し、組織を超えた協働が成立させるメカニズムとして機能していると分析できる事象を観察することができた。

本章では、HTML5 仕様案に含まれる一部機能の対案提出という事例を通じ、イノベーションの源泉拡張と、より柔軟な組織を超えた協働が成立する要因の仮説を導出する。

9.1.1. 焦点となった機能と競合した 2 つの仕様

9.1.1.1. ウェブアプリケーション間データ引き渡しプロトコル

ウェブアプリケーションによって提供する機能を高度化するにあたり、複数のアプリケーション間の連携、特にデータの引き渡しを実現する機能が必要とされる。手元のデバイスにインストールするネイティブアプリケーションにおいても、画像処理ソフトで加工したファイルをプレゼンテーションに貼り付けたり、表計算ソフトで作成した表組みをワードプロセッサに貼り付けたりといったように、複数のアプリケーション間でデータを受け渡しするというユースケースは頻繁に生じている。この場合、データの引き渡しという処理を担当するのは OS である(図 40)。

一方ウェブアプリケーションではどうなるか。ウェブアプリケーションは、それぞれのサーバにおいてプログラムが動作し、データが生成される。各アプリケーション間を連携させるのは、ウェブというプラットフォームそのものが担わなくてはならない。そのため、対応するウェブ標準仕様が必要となってくる。標準仕様のスコープに含まれるユースケースとしては、片方向のデータ引き渡し(図 41)、往復でのアプリケーション連携(図 42)という 2 種類が想定され

る。

現行バージョンの HTML、つまり HTML4.01 には、データの引き渡しの仕様は含まれていない。1998 年 4 月に公開された HTML4.0 改訂版⁴¹⁰ では、“mailto:” リンクで電子メールクライアントを起動させ、メールアドレスを引きわたす機能が含まれていたが、HTML4.01 へマイナーアップグレードした際に、削除されている⁴¹¹。現在、主要なブラウザで “mailto” リンクが機能しているのは、各ブラウザベンダが慣習にのっとり、自主的に実装から外していないだけである。この仕様は、ユースケース 1 の片方向のデータ引き渡しを実現する機能である。

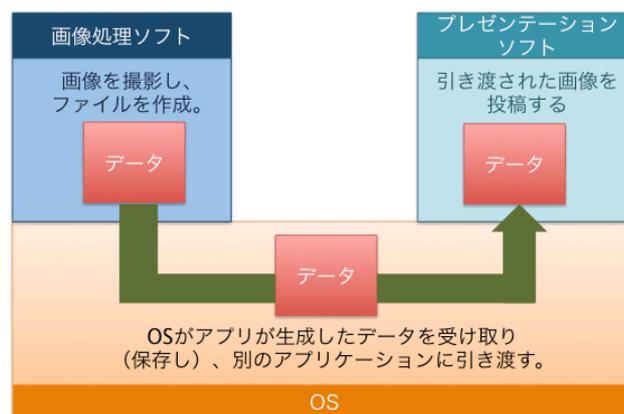


図 40. ネイティブ OS におけるアプリケーション間データ引き渡しのプロセス

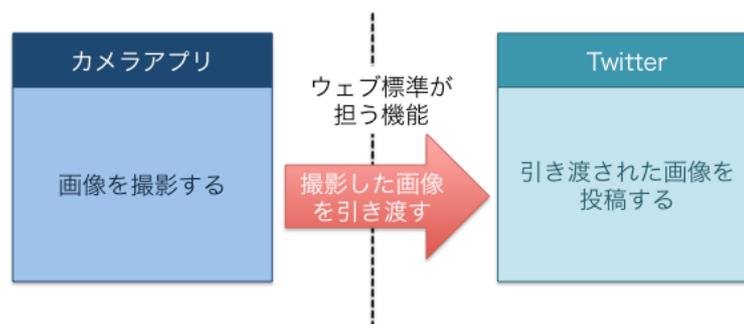


図 41. ユースケース 1: 片方向のデータ引き渡し

⁴¹⁰ Ragget, D. et al. (eds.) (1998, April 24). *HTML 4.0 Specification - W3C Recommendation, revised on 24-Apr-1998*. Retrieved November 14, 2013, from <http://www.w3.org/TR/1998/REC-html40-19980424/>

⁴¹¹ Raggett, D. et al. (1999, December 24). A.1 Changes between 24 April 1998 HTML 4.0 and 24 December 1999 HTML 4.01 versions. *HTML 4.01 Specification: W3C Recommendation, 24 December 1999*. Retrieved November 14, 2013, from <http://www.w3.org/TR/REC-html40/appendix/changes.html#h-A.1.1.12>

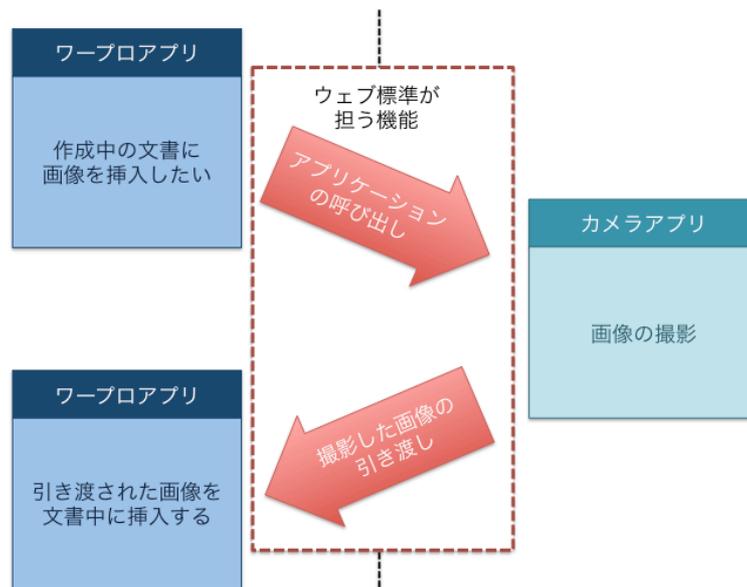


図 42.ユースケース 2: 往復でのアプリケーション連携

9.1.1.2. 片方向のデータ引き渡しのみに対応した HTML5 に含まれる仕様: registerProtocolHandler()、registerContentHandler()

HTML5 の前身である WHATWG Web Applications 1.0 には 2006 年 4 月 13 日版 Working Draft⁴¹²から、W3C 版 HTML5 には 2008 年 1 月 22 日公開された FPWD⁴¹³より、データ引き渡しの仕様として、registerProtocolHandler() と registerContentHandler() という 2 つの仕様が含まれている。

registerProtocolHandler()は、“mailto” リンク同様に特定のアプリケーションやサービスを起動させるプロトコルを設定できる仕様であり、“mailto” 以外にも“sms” など、様々な種類のプロトコルを整備することが想定されている。

もう 1 つの registerContentHandler()は、MIME タイプ(Multi-Purpose Internet Mail Extensions; ファイル形式)別に起動させる特定のアプリケーションやサービスを設定できる仕様である。両者とも、ユースケース 1 の片方向のデータ引き渡しを実現する仕様である。

⁴¹² Hickson, I (2006, April 13). 4.10.2. [SCS] Custom protocol and content handlers. *Web Applications 1.0 Working Draft – 13 April 2006*. Retrieved April 24, 2014, from Internet Archive: <http://web.archive.org/web/20060413213821/http://whatwg.org/specs/web-apps/current-work/#scs-custom>

⁴¹³ Hickson, I. and Hyatt, D. (2008, January 22). 4.5.1. Custom protocol and content handlers. *HTML 5 :A vocabulary and associated APIs for HTML and XHTML, W3C Working Draft 22 January 2008*. Retrieved April 24, 2014, from <http://www.w3.org/TR/2008/WD-html5-20080122/#custom-handlers>

9.1.1.3. 往復のデータ引き渡しに対応した対案: Web Intents

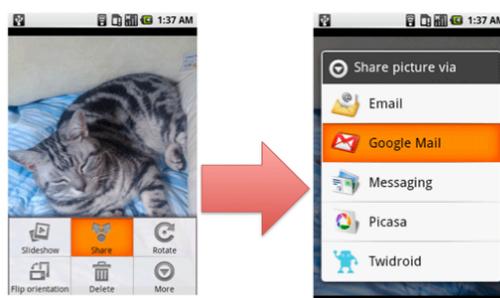
Web Intents は、Google の Paul Kinlan が開発・提唱した、ウェブアプリケーション間の連携を実現する技術仕様である。データ引き渡しのパターンのうち、片方向と往復の両方のパターンを機能として提供する。

Web Intents は Google のスマートフォン OS である Android に実装されているアプリケーション間連携機能である Intent をウェブアプリケーションに適用するというコンセプトで開発された。

初期のスマートフォン OS はマルチタスク機能がなく、またユーザインターフェイス(UI)が画面をタッチ (タップ) するというものである。そのため、Windows や MacOS のように、コピー・アンド・ペーストしてアプリケーション間でデータを引き渡すという操作を行うことが難しい。

そこで Android では、操作対象のデータ (の表示) をホールド(長押し)し続けると、連携先のアプリケーションの選択肢が表示され、それをタップすることでデータを引き渡すという機能、ならびに UI を実装した(図 43)。

Web Intents は、この機能をウェブアプリケーションに応用したものである。Web Intents は片方向、往復という両方のユースケースに対応している。Web Intents では、操作対象のデータを選択すると図 44 のような引き渡し先アプリケーション選択ウィンドウが立ち上がり、アプリケーションを呼び出すことができる。



画像をホールド(長押し)すると、処理内容が選択肢として表示される。

“share”を選んだ場合、その処理が可能なアプリケーションが選択肢として表示される。選択肢をタップすると、データがそのアプリケーションに引き渡される。

図 43. Android OS における Intent のしくみ (画像は Version 1.6 のもの)⁴¹⁴

⁴¹⁴ 画像出典: 杉本礼彦. (2009, April 22). Android の重要な機能、インテント. *Think IT*. Retrieved April 24, 2014, from <http://thinkit.co.jp/article/921/1>

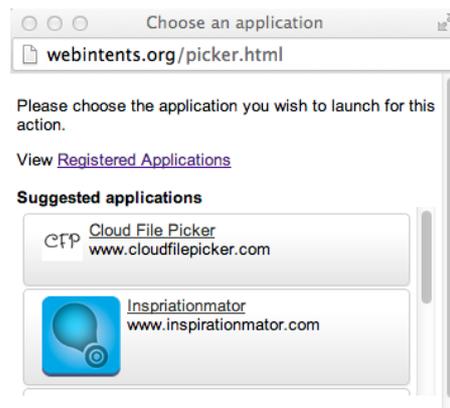


図 44 Web Intents における引き渡し先アプリケーション選択ウィンドウ⁴¹⁵

9.1.2. ソーシャルメディアと OSS ツールを用いた仕様提案

Web Intents を提案した Paul Kinlan は、Google に所属しているものの社内のコンセンサスを得た上 W3C に提案したわけではない。Kinlan はもともと Developer Relation 担当⁴¹⁶の社員であり、ブラウザ (Chrome) の開発に従事していたわけでも、標準化を担当していたわけでもない。Web Intents は Kinlan が個人的に進めていた開発プロジェクトに過ぎない。Kinlan は 2010 年 12 月に、Google App Engine を用いて仕様を公開⁴¹⁷し、Twitter⁴¹⁸を通じて告知された。また、2011 年 7 月には、仕様を GitHub に公開している^{419,420}。

Web Intents を標準にするには、1) Google Chrome 開発チームを説得して、Web Intents を実装させること、2) W3C に持ち込み、標準化プロセスを進めること、の 2 つを実現しなくてはならない。このうち前者の課題は、Chrome 開発チームが 2011 年 8 月に、Web Intents を実装することをブログで表明⁴²¹し、実装に向けた取り組みが始まっている。

⁴¹⁵ 出展：webintents.org 内のデモページ。Retrieved April 24, 2014, from <http://webintents.org/>

⁴¹⁶ 役職名は Developer Advocate

⁴¹⁷ 当初、プロジェクトのウェブサイト Google App Engine を利用して作成されていた (<http://webintents.appspot.com/>)。現在は移転しているが、サイト (<http://webintents.org/>) は Kinlan 個人が運営している。

⁴¹⁸ Kinlan, P. (2010, December 23). Twitter 投稿 Retrieved November 14, from https://twitter.com/Paul_Kinlan/status/17998349231722497

⁴¹⁹ Kinlan, P. (2011, July 1). Web Intents: A fresh look. *Tales of a Developer Advocate*. Retrieved November 17, 2013, from <http://paul.kinlan.me/web-intents-a-fresh-look/>

⁴²⁰ Paul Kinlan が開設した GitHub の Web Intents 仕様公開ページ: Retrieved January 12, 2015 from <https://github.com/PaulKinlan/WebIntents/commits/master?page=21>

⁴²¹ Hawkins, J. (2011, August 4). Connecting Web Apps with Web Intents. *Chromium Blog*. Retrieved November 14, from <http://blog.chromium.org/2011/08/connecting-web-apps-with-web-intents.html>

9.1.3. W3C 内での支持拡大と、企業の垣根を超えた機能開発

新たな仕様を W3C へ仕様提案する場合、通常では Workshop を開催して当該機能がウェブ標準として必要であるか、必要である場合は新たに Working Group を設置するか、もしくは既存 Working Group で議論するかといった策定プロセスの進め方についてもある程度コンセンサスを固めた上で、草案が提出される。しかし Web Intents はこのようなプロセスを経ずに仕様策定プロセスが開始された。それには 2 つの理由がある。

1 つは、Sony/ Sony Mobile を始めとするデバイスベンダが、ウェブとホームネットワークとの連携を実現する標準仕様の開発を継続しており、Web Intents の仕様を拡張することでその機能を実現できると判断したこと。もう 1 つは、既に同じ Google のエンジニアが提案する類似仕様である Web Introducer (旧 Powerbox) が、Device APIs Working Group において既に議論されていたことである。

Web Intents の仕様策定を受け持つにふさわしい Working Group は 2 つ存在した。1 つが Web Applications Working Group (WebApps) である。WebApps は、ウェブアプリケーション向けの API 仕様等を策定することを主目的として設置されている Working Group である。Web Intents はウェブアプリケーション間でデータを引き渡すための仕様、つまり API 仕様であるため、本来であればこの WG に提案されるのが最も適当である。

しかし WebApps は、元々 HTML5 の Editor である Ian Hickson が中心となって策定していた Web Forms 2.0、XML Binding Language (XBL) の仕様策定を担っていた WG であり、WHATWG で検討されていない仕様について議論する雰囲気ではなかった。そのため、これまでの HTML5 開発の文脈に乗っていない Web Intents などの仕様提案を持ち込むのは困難だったのである。

もう 1 つが Device APIs Working Group (DAP) である。DAP は、W3C 内の Vodafone が主催する Security for Access to Device APIs from the Web という Workshop を経て創設された Device APIs and Policy Working Group を前進とする WG である。

Workshop の Chair は、W3C チームスタッフと、Vodafone が中心となって発足した携帯通信キャリアが中心となり⁴²²立ち上げたコンソーシアムである Open

⁴²² OMTP は Vodafone、Orange、AT&T、Telenor といった携帯電話キャリアが中心となって設立された団体だが、その後 Samsung、Sony Ericsson、Nokia などの端末メーカーや、Opera、Access などブラウザベンダを含むソフトウェアベンダも Sponsor/ Advisor として参加している。

Mobile Terminal Platform (OMTP) 所属の人間が務めていた。この会議で OMTP が開発を進めていた BONDI⁴²³という携帯電話端末の共通 API 仕様をベースに、ウェブアプリケーション向け API 標準を策定することが決定され、DAP⁴²⁴で策定作業を担うこととされた。DAP はもともと携帯電話端末、すなわちデバイスの共通 API 仕様を W3C 標準化することを目的として発足した Working Group である。

OMTP ならびに、その活動を引き継いだ Wholesale Applications Community とともに、Apple と Google というスマートフォン OS ベンダにアプリストアを独占されている状況を転換するために、オープンな API 仕様を開発、普及させることを目的として活動している。つまり、DAP は携帯通信キャリア、携帯端末ベンダ、Opera や Mozilla といったスマートフォン OS とアプリストアを有しないソフトウェアベンダの利害を代弁する性格をもっている。

なお、OMTP は W3C 内で DAP の活動が開始された後も、BONDI の開発作業を進め、2010 年 2 月には、BONDI 1.1 がリリースされている。

Battery Status API は、ウェブアプリケーションがスマートフォンの電池残量のデータを読み出すことのできる API である。ユースケースとしては、動画共有サービス利用時に電池残量が少ない時には画質を低くして再生するなど電力消費量が少なく済む動作を自動的に行うことができる、などが想定される。ここから機能を発展させていくと、画像投稿サイト利用時にサービスからスマートフォンのカメラを呼び出し、撮影した画像・動画ファイルを受け取るといったユースケースが想定できる⁴²⁵。その先には、ウェブアプリケーションで取り扱うデータリソースを、インターネットに接続されたあらゆるデバイスに保存されたファイルに拡張したり、ウェブアプリケーションで生成したファイルを引き渡したりというユースケースが想定される。

ウェブアプリケーションは、デバイス上でプログラムが動作し、データが生成・処理されていたものを、ウェブ上に移転させるという試みであった。これに対し DAP の試みは、デバイス側にデータ処理プロセスの一部を引き戻すものであった。

⁴²³ 2010 年 10 月 1 日付けで OMTP は解散。BONDI プロジェクトは携帯通信キャリアが中心となって立ち上げた Wholesale Applications Community(WAC)によって引き継がれる。WAC は 2012 年 7 月に GSM アソシエーションと統合する。

⁴²⁴ 当初の名称は、Device APIs and Policy Working Group であった。

⁴²⁵ HTML Media Capture, Media Capture and Streams, MediaStream Recording, MediaStream Image Capture などの仕様が、こうした機能を提供するために策定されている。

こうした試みに、全く異なる思惑から同調したのが Mozilla であった。Mozilla は 2011 年 8 月に Firefox OS (当時 Boot to Gecko) 開発プロジェクトを公表する。Firefox OS は、HTML5 等でアプリケーションの開発が可能であるスマートフォン向け OS である。2011 年当時、スマートフォンの OS は Apple の iOS と Microsoft の Windows Phone という 2 つのプロプライエタリ OS と、OSS でありながら Google が開発の主導権を完全に握っている Android の 3 つの寡占状態となりつつある状況であり、Mozilla としては、スマートフォン向けのオープンなプラットフォームを確立させることを目標としていた。

Firefox OS は、その名の通りブラウザ上で動作するプログラム=ウェブ標準である HTML5 等をベースとしてアプリケーションを動作させるというコンセプトを採用しており、HTML5 のデバイス操作を実現するための機能拡張は Firefox OS の機能を充実させるための必須条件であった。

DAP では、Battery Status API や Vibration API などデバイスの個別機能毎の API 仕様や、画像・音声といった特定のファイル形式に関する API 仕様を標準化してきたが、ウェブアプリケーションとデバイスの連携を実現する包括的な仕様の開発を望んでいた。その実現に対する大きな障害が、Google と Apple が DAP に参加していないことであった。

スマートフォン OS で大きな影響力をもつ 2 社が策定に関与しないということは、流通する大半のスマートフォンで仕様が実装されないということの意味する。つまり、この 2 社をとりこまないと標準化のプロセスは先に進まないのだ。

ただし、DAP がもともと Apple の iTunes Store と Google Play に対向するために携帯通信キャリア、端末ベンダらが立ち上げたプロジェクトである BONDI をベースとしている以上、Apple と Google 側から歩み寄ってくることは考えにくい。実際、2011 年 11 月に開かれた Technical Plenary / Advisory Committee Meetings Week (TPAC:技術全体会合)における Web Apps 分科会では、

As a point of information

... Apple is unlikely to ever join DAP

... because of IPR concerns and others

... we are somewhat interested in Web Intents

... and would try to comment if it were in Web Apps or joint in Web Apps

... we would not if it were solely in DAP

Maciej Stachowiak (Apple) ⁴²⁶

Darin, Google

... we work together with Apple

... it's fairly important that we be in the group with Apple talking about Web Intents

Darin Fisher (Google) ⁴²⁷

と Apple、Google の社員がそれぞれ発言している。また Web Intents の標準化作業を DAP が単独で進める場合 Apple、そして Apple と共同歩調をとるとした Google は、そこに関与しないと明言しているのである。

この発言がなされる前に、WHATWG 立ち上げに関与したブラウザベンダで唯一 OMTP にも参加する Opera の Charles McCathieNevile は、以下のように述べて Apple と Google に Web Intents の標準仕様策定に参加するよう促している。

We would lean towards

*... DAP *was* a pretty dysfunctional, pointless, stupid thing, 2 years ago*

... it is no longer

*Charles McCathieNevile*⁴²⁸

この発言から、(2011 年の 2 年前である) 1999 年当時、Google と Apple は、DAP を機能しない、無意味な、馬鹿げた活動だと評していたことがわかる。つまり、Apple・Google と DAP メンバー単に知的財産の問題で合意ができなかっただけでなく、感情面も含め相当な距離感があったと判断できる。

携帯通信キャリア、端末ベンダはまだしも、Opera ですら Apple と Google を共同作業者に引きこむことは非常に困難であった。しかし DAP メンバーは、どうしてもその障壁を超える必要があったのである。

このような背景があることから、DAP メンバーにとっての Web Intents は、Google 社員の方から自分たちが実現したい機能を実現する仕様を提案したと評

⁴²⁶ World Wide Web Consortium. (2011, October 31). - *DRAFT - WebApps f2f Meeting 31 Oct 2011* (TPAC 2011 WebApps f2f Meeting Minutes). Retrieved 2013, November 17, from <http://www.w3.org/2011/10/31-webapps-minutes.html>

⁴²⁷ *ibid.*

⁴²⁸ *ibid.*

価されたのである。

Paul Kinlan がウェブ上で Web Intents を発表した 2010 年 12 月時点で、DAP では既に別の Google 社員である Tyler Close、Mark Seaborn らが Powerbox の名で同様の機能を有する仕様を DAP に提案していた。

DAP のメーリングリストでは、AT&T の Bryan Sullivan が、この提案は Web Apps WG にすべきと返答している⁴²⁹ものの、Sony Ericsson (当時)と Mozilla のサポートを経て仕様のブラッシュアップにとりかかる⁴³⁰。Powerbox は Web Introducer と名前を変えて、Google、Sony Ericsson、Mozilla の 3 者による提案として、改めて DAP に提案されることとなる。

Powerbox/ Web Introducer が既に DAP で仕様策定プロセスを進めているのを知った Kinlan は、Web Intents を自身の単独提案として標準化を進めるのではなく、Web Introducer の策定に加わることとした⁴³¹。

Web Introducer は、その後 DAP での標準化プロセスを先に進めることができず、2011 年 7 月 1 日に、Kinlan は Web Intents のプロジェクトを復活させる⁴³²。その 1 週間後、7 月 7 日には Mozilla の Ben Adida が Web Activities という Web Intents の代替仕様案をブログで提案する⁴³³。Web Activities は Web Intents を置き換えることを念頭において提案されている⁴³⁴が、基本的には同じ機能をもつ仕様案をそれぞれで提案しつつ、策定プロセス上で協力してブラッシュアップしていくという前提であり、両者が分裂して仕様提案しているわけではない。

そして翌月の年 8 月に、Google の Chromium⁴³⁵開発チームが Web Intents をブログで提案した。この段階で Web Intents は Kinlan の個人的なプロジェクトから Google Chrome 開発チームのプロジェクトへと変化することとなる。また、デス

⁴²⁹ Sullivan, B. (2010, February 24). RE: [Powerbox] A RESTful proposal for Web enabling devices. Retrieved November 17, 2013, from public-device-apis@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-device-apis/2010Feb/0154.html>

⁴³⁰ Close, T. (2010, May 26). [Powerbox] New draft based on further collaboration and prototyping. Retrieved November 17, 2013, from public-device-apis@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-device-apis/2010May/0133.html>

⁴³¹ Kinlan, P. (2011, March 3). So what is happening with Web Intents?. *Tales of a Developer Advocate*. Retrieved November 17, 2013, from <http://paul.kinlan.me/so-what-is-happening-with-web-intents/>

⁴³² Kinlan, P. (2011, July 1). Web Intents: A fresh look. *Tales of a Developer Advocate*. Retrieved November 17, 2013, from <http://paul.kinlan.me/web-intents-a-fresh-look/>

⁴³³ Adida, B. (2011, July 7). Web Apps Update – experiments in Web Activities, App Discovery. *Mozilla Labs Blog*. Retrieved November 17, 2013, from <http://blog.mozilla.org/labs/2011/07/web-apps-update-experiments-in-web-activities-app-discovery/>

⁴³⁴ Adida, B. (2012, February 9). A simpler, webber approach to Web Intents (or Activities). *Benlog - software, security, policy*. Retrieved October 27, 2013 from <http://benlog.com/2012/02/09/a-simpler-webber-approach-to-web-intents-or-activities/>

⁴³⁵ Google Chrome のベースとなるオープンソースブラウザ。Chromium のソースコードに Google のプロプライエタリなコードを加える事で Chrome となる。

クトップコンピュータ、スマートフォンともにシェアをもつブラウザに Web Intents が実装される道が開かれたこととなった。そして、DAP は Working Group 憲章を改定し、Web Intent が機能として提供するウェブアプリケーションとデバイスの連携機能をスコープに含めることとなった⁴³⁶。

そして 2011 年 W3C TPAC において、DAP と WebApps の 2WG によって Web Intents Task Force (WITF) を設立することが議決されたのである⁴³⁷。

9.1.4. Google 社内で支持拡大するも、HTML5 仕様への組み込みは失敗に

翌年の 2012 年 5 月 15 日、とうとう Google Chrome に Web Intents が実装される⁴³⁸。そして 6 月 26 日には、Web Intents の FPWD⁴³⁹が公開される。この Draft の Editor を務めたのは Paul Kinlan、Google の Chrome/ Chromium 開発チームの James Hawkins と James Hawkins の 3 人である。

また、FPWD の公開にあわせ、複数の拡張仕様が DAP メンバーから提案されている。翌日の 6 月 27 日、Sony Mobile (旧 Sony Ericsson) は Sony と共同で Web Intents の補遺 (Addendum)として Local Services という、UPnP Discovery (ならびに UpnP に準拠した DLNA) や mDNS をウェブに適用し⁴⁴⁰、ホームネットワークでのデバイス連携を可能とする拡張仕様を提案している⁴⁴¹。

また、これまで DAP において BONDI API ベースで策定されてきた Gallery API⁴⁴²と Contacts API⁴⁴³が、それぞれ Pick Media Intent⁴⁴⁴と Pick Contacts Intent⁴⁴⁵の

⁴³⁶ Hazael-Massieux, D. (2011, August 22). *New charter for Device APIs Working Group*. Retrieved October 27, 2013, from public-device-apis@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-device-apis/2011Aug/0042.html>

⁴³⁷ World Wide Web Consortium. (2011, November 3). 21. Introducing Web Intents. *Device APIs Working Group Teleconference: 03 Nov 2011*. (TPAC 2011 の F2F 会合の議事録だが、議事録を登録するシステムの都合上、Teleconference と表記されている。) Retrieved November 18, 2013, from <http://lists.w3.org/Archives/Public/public-device-apis/2011Nov/att-0177/minutes-2011-11-03.html#item21>

⁴³⁸ Hawkins, J. (2012, March 15). Connect with Web Intents. *The Chromium Blog*. Retrieved November 20, 2013, from <http://blog.chromium.org/2012/05/connect-with-web-intents.html>

⁴³⁹ Billock, G. et al. (eds.). (2012, June 26). *Web Intents - W3C Working Draft 26 June 2012*. Retrieved January 12, 2015, from <http://www.w3.org/TR/2012/WD-web-intents-20120626/>

⁴⁴⁰ Nilsson, C. (2012, May 29). *W3C Web Intents - Local Network Service Discovery Proposal May 29 2012*. Retrieved November 20, 2013, from http://www.w3.org/wiki/images/2/2e/V4_W3C_Web_Intents_-_Local_UPnP_Service_Discovery.pdf

⁴⁴¹ Nilsson, C. (2012, June 27). *Web Intents Addendum for local Services uploaded*. Retrieved November 18, 2013, from public-device-apis@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-device-apis/2012Jun/0099.html>

⁴⁴² 端末上の画像、動画といったメディアファイルを取り扱う API 仕様

⁴⁴³ 端末上の連絡先リストへのアクセス API 仕様

⁴⁴⁴ Jungkee, S. (2012, June 27). *[gallery] ED update - Pick Media Intent*. Retrieved November 20, 2013, from public-webintents@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-web-intents/2012Jun/0111.html>

⁴⁴⁵ Robin, B. (2012, June 22). *Pick Contacts Intent*. Retrieved November 18, 2013, from public-webintents@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-web-intents/2012Jun/0105.html>

ように Web Intents の派生仕様という位置づけに再構築して提案された。

Opera、Mozilla といったブラウザベンダを引き込み、Addendum Local Services を提案する Sony や Pick Media Intent を提案する Samsung といったデバイスベンダも味方につけた Kinlan だが、まだ乗り越える壁は残っていた。同じ Google に属する HTML5 の Editor である Ian Hickson である。

2011 年 9 月 19 日、Google の Chrome 開発チームで Web Intents 実装担当の James Hawkins が WebApps メーリングリストに Web Intents の提案を送付する⁴⁴⁶。しかし Ian Hickson は即日反論の返信を投稿する⁴⁴⁷。更に翌日の 9 月 20 日、Hickson は以下の文面を WebApps メーリングリストに投稿し、はっきりと HTML5 に含まれる `registerContentHandler()` と `registerProtocolHandler()` に競合する仕様提案に対し不快感を示している。

*Why not just improve both navigator.registerContentHandler and navigator.registerProtocolHandler?*⁴⁴⁸

WebIntents の普及活動のために公開されているウェブサイトは FAQ ページ⁴⁴⁹ を設け、なぜ `registerContentHandler()` と `registerProtocolHandler()` では不十分であり、新たな仕様を提案するのかについてアピールしている。

しかし結果的に Ian Hickson は最後まで Web Intents に賛意を示すことはなかった。また、HTML5 の草案仕様の策定を担う WHATWG においても、Web Intents を HTML5 仕様の中に採用することは否定された⁴⁵⁰。

そして 2012 年 12 月、Web Intents は Chrome の開発者向けバージョン(canary) から削除された。その理由として、ウェブ利用時に突然ポップアップが登場し、連携先アプリケーションを選択するというユーザーインターフェイスが利用者に受け入れられない可能性が高いことと、タブブラウザにおいて複数タブ間で

⁴⁴⁶ Hawkins J. (2011, September 19). *Adding Web Intents to the Webapps WG deliverables*. Retrieved November 17, 2013, from public-webapps@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-webapps/2011JulSep/1477.html>

⁴⁴⁷ Hickson, I. (2011, September 20). *Re: Adding Web Intents to the Webapps WG deliverables*. Retrieved November 17, 2013, from public-webapps@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-webapps/2011JulSep/1475.html>

⁴⁴⁸ Hickson, I. (2011, September 20). *Re: Adding Web Intents to the Webapps WG deliverables*. Retrieved November 17, 2013, from public-webapps@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-webapps/2011JulSep/1475.html>

⁴⁴⁹ webintents.org. (n. d.). *Web Intents FAQ*. Retrieved April 24, 2014, from <http://webintents.org/faq.html>

⁴⁵⁰ Billock, G. (2012, September 4). [whatwg] Web Intents: Proposed modifications to the spec regarding registration. Retrieved November 3, 2013, from whatwg@whatwg.org mailing list archive: <http://lists.whatwg.org/htdig.cgi/whatwg-whatwg.org/2012-September/037152.html>

の連携に問題が残っていることが挙げられている⁴⁵¹。

その結果、実装事例がなくなってしまったため、実装主義に基づく W3C の標準化プロセスを進めることが不可能となり、2013 年 3 月に Web Intents⁴⁵²を Working Group Note として公開して Web Intents Task Force は活動を終了させた。

9.1.5. 企業間の垣根を超えた協働の継続

W3C 内での標準化活動は一度終了したが、Google の Web Intents 開発チームと、Firefox OS の開発を続ける Mozilla の Web Activity 開発チームとの間でディスカッションが継続されている⁴⁵³。今後、Firefox OS 開発の進展に伴い実装事例が現れた後に、改めて標準化提案される可能性がある。

また、ホームネットワークに接続されたデバイスとの連携仕様に関しては、Web and TV Interest Group で Requirement⁴⁵⁴ が策定され、これに応じる形で DAP では Opera と CableLabs によって提案されてきた Network Service Discovery (NSD) という仕様⁴⁵⁵の標準化プロセスが進められている。Sony が UPnP 対応を NSD ベースで行うことを検討していたり⁴⁵⁶、Chrome のレンダリングエンジンである Blink における実装が検討されたり⁴⁵⁷している。

⁴⁵¹ World Wide Web Consortium. (2012, November 11). 10. Web Intents/Web Activities. - *DRAFT - Device APIs Working Group Teleconference: 01 Nov 2012*. (TPAC 2012 の F2F 会合の議事録だが、議事録を登録するシステムの都合上、サイトでは Teleconference と表記されている。) Retrieved November 3, 2013, from <http://www.w3.org/2012/11/01-dap-minutes.html#item10>

⁴⁵² Billock, G. et al. (eds.) (2013, March 21). *Web Intents: W3C Working Group Note 23 May 2013*. Retrieved November 3, 2013, from <http://www.w3.org/TR/2013/NOTE-web-intents-20130523/>

⁴⁵³ Billock, G. (2013, March 21). *Web Intents/Web Activities F2F get-together report*. Retrieved November 3, 2013 from public-web-intents@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-web-intents/2013Mar/0000.html>

⁴⁵⁴ Pascale, G. (ed.) (2011, December 1). *Requirements for Home Networking Scenarios: W3C Interest Group Note 01 December 2011*. Retrieved November 18, 2013, from <http://www.w3.org/TR/2011/NOTE-hnreq-20111201/>

⁴⁵⁵ Tibbett, R. (ed.) (2014, February 20). *Network Service Discovery: W3C Working Draft 20 February 2014*. Retrieved November 18, 2013, from <http://www.w3.org/TR/discovery-api/>

⁴⁵⁶ Nilsson, C. (2012, April 25). *Status of my Actions from Shenzhen meeting*. Retrieved November 18, 2013, from public-device-apis@w3.org Mailing List: <http://lists.w3.org/Archives/Public/public-device-apis/2012Apr/0045.html>

⁴⁵⁷ blink-dev Google Group Archive. (2013, August). *Intent to Implement: Network Service Discovery (グループのスレッド)*. Retrieved November 18, 2013, from <https://groups.google.com/a/chromium.org/forum/#!searchin/blink-dev/network%20service%20discovery/blink-dev/HT0KZKuTLxM/S3w-SdvjZfUJ>

表 12. ウェブアプリケーション間データ受け渡し
 プロトコル仕様の収斂に至る経緯

日時	事象
2005/7/1	Mobile Web Initiative 設立
2006/4/12	WHATWG 版 Web Applications 1.0 の Working Draft に registerProtocolHandler() と registerContentHandler() が追加される
2008/1/22	W3C 版 HTML5 Working Draft 第一版リリース。registerProtocolHandler()、registerContentHandler() が含まれる。
2008/12/10-11	Mobile Web Initiative 主導で、Security for Access to Device APIs from the Web が開催される。BOIDI ベースの API 仕様のウェブ標準化を目指す方針が打ち出される。
2009/2/16	The first BONDI widget が MWC2009 にてリリース
2009/8/25	DAP の Issue に、Contacts API と Gallery API の要件定義が追加される
2010/2/19	Powerbox の Editors Draft が公開、DAP WG に提案される。
2010/2/24	AT&T の Bryan Sullivan が、Powerbox の仕様提案は Web Apps WG にすべきと返答。
2010/10/1	OMTP 解散。BONDI プロジェクトは WAC が引き継ぐ
2010/12	Paul Kinlan が WebIntents プロジェクトをスタート
2011/1/5	Powerbox を Web Introducer に改称。Draft を改定
2011/3/3	Paul Kinkan が WebIntents プロジェクトを Web Introducer に統合させると発表
2011/4/11	DAP WG Re-charter で Scope に Web Introducer の機能を含めることを検討
2011/5/20	Web Introducer の Editors Draft が公開
2011/7/1	Paul Kinlan が WebIntents のプロトタイプコードを GitHub で公開=Web Intents プロジェクトの復活
2011/7/7	Mozilla の Ben Adida が Web Activities の提案をブログで公開する
2011/7/25	Boot to Gecko (Firefox OS) プロジェクトの発表
2011/8/4	Google Chromium Team が WebIntents の実装に向けた試みを進めていることを発表
2011/8/15	Opera の Rich Tibbett が Network Service Discovery の仕様提案を DAP の ML に行う
2011/8/22	Re-charter Web Introducer/ Intents 系 API がスコープに入る。
2011/9/19	Google James Hawkins が WebApps WG のメーリングリストで Web Intents の標準化提案のポストを投稿する。ここから、Google/ Mozilla の WebIntents/ Web Activities team と Ian Hickson との間の議論が始まる。
2011/9/28	Sony Ericsson の Claes Nilsson が Web Intents など、サービスディスカバリー仕様の比較表を TF の Wiki で公開
2011/10/31	TPAC で DAP/ WebAPPs のジョイントセッションが開かれ、WebApps WG の Charter のスコープに WebIntents が追加される方向で決着
2011/11/18	W3C Web Intents TF の ML が作成される
2011/11/18	W3C Web Intents TF の ML が作成される
2011/12/1	Requirements for Home Networking Scenarios が Web and TV IG の Note として公開
2011/12/6	James Hakins が WHATWG に W3C Web Intents TF の活動紹介をし、intents tag を whatwg に提案する
2012/2/1	The Gallery API の FPED が公開

日時	事象
2012/2/15	James Hawkins が registerProtocolHandler/registerContentHandler via Web Intents を WHATWG に提案する
2012/5/15	Chrome に WebIntents が実装される
2012/3/9	Web and TV IG で Requirements for Home Networking Scenarios の Editor を務めた Ericsson の Jan Lindquist が NSD の Editor 二人に、プロセスを進めることを助言するポストを DAP ML に投稿
2012/6/26	WebIntents First Public Working Draft が公開
2012/6/27	Web Intents Addendum for local Services uploaded の Editor's Draft が公開される。
2012/6/27	The Gallery API が Pick Media Intent に名称を変更して Editor's Draft を公開
2012/7/12	Contacts API を Pick Contact Intent に名前を変えて Editor's Draft を公開
2012/8/7	Networked Service Discovery and Messaging の Working Draft が公開
2012/10/4	Web Intents Addendum - Local Services の FPWD が公開
2012/10/23	Opera が Network Service Discovery を実装
2012/11/1	Chrome から WebIntents の実装が取り除かれると発表
2012/7/17	WAC が解散。活動は GSM Association が引き継ぐ。
2012/12/18	Chrome の canary から WebIntents の実装が取り除かれる
2013/1/24	Robert Nyman が hacks.mozilla.com で Web Activities を紹介する。
2013/3/21	Google と Mozilla のチームがディスカッションし、現状報告と開発継続の旨を webintents TF の ML に投稿
2013/5/23	Web Intents WG note 公開
2013/5/23	Web Intents Addendum - Local Services の最終版 Editors Draft を公開

9.2. ロイヤルティ・フリー・ポリシーの明文化による効果

WebIntents は、Twitter や GitHub といったウェブ上のオープンなコミュニケーション基盤やツールを用いて提案された 1 つの仕様が、W3C での議論を通じてより多くのイノベーションに繋がる仕様提案を生み出し、組織を超えた協働を成立させた事例である。こういった現象が生じた原因として、機能ベースでのスコープ設定、実装主義、プロセスのオープン化に加えて、7.4.3 節で取り上げた

ロイヤルティ・フリー・ポリシーの明文化
を仮説として提示したい。

9.2.1. 組織の垣根を超える協働の促進

Web Intents の仕様提案は、W3C に持ち込まれる以前からオープンな環境で取り組まれてきた。当初 Kinlan 自身のウェブサイトで行われ、その後 GitHub で公開された。

2011 年 7 月 1 日に Kinlan が WebIntents の Draft を GitHub に公開してから、1 週間も経たないうちに、Mozilla の Ben Adida が Web Activity をブログで提案し、両社の協働が開始された。また 2011 年 9 月 19 日に WebIntents が WebApps WG のメーリングリストにおいて標準化提案されてから、3 ヶ月を経ないうちに Requirements for Home Networking Scenarios が Web and TV IG の Note として公開されている。これは、Sony などによる機能拡張が提案直後から進められてきたことを示すものである。

WebIntents の提案は、それまで断絶していた WebApps と DAP による共同での Task Force 設立につながり、Google が Mozilla やデバイスベンダと連携して仕様の開発と普及を推し進めるという構図を創りだした。また、Web Intents の仕様策定プロセスに積極的に関わった Sony が NSD の策定プロセスに関与したり、Chromium/ Blink 開発チームが実装を検討したりするなど、その後の類似機能を提供するなどしている。

もし W3C でロイヤルティ・フリー・ポリシーが明文化されていない場合、提案者である Kinlan の所属する Google が将来にわたって知的財産権を主張しないという保証を得ることはできず、他の企業による機能拡張が迅速にはじまることはなかったであろう。また、Mozilla との協働が成立・継続することもなかったと考えられる。ロイヤルティ・フリー・ポリシーが明文化されていることにより、企業の垣根を超えた協働が迅速に成立し、継続するに至ったと仮説立

ることができる。

また、ウェブ標準の策定に関わるトップエンジニアの労働市場は極めて流動性が高い。例えば現在 Google に所属する Ian Hickson は、最初に W3C に HTML4.01 の次期バージョン開発を提案した際には Opera 所属であったし、更にその前は Mozilla/ Netscape に所属していた。所属組織は変わっても、同一の仕様の策定を推し進め続けているのである。

Hickson は大学を卒業してから最初に Mozilla Project に参画し⁴⁵⁸、更に Web Stanard Project において Acid 2/3 テストプログラムの開発を担当する⁴⁵⁹など、オープンソース、ならびにオープン標準コミュニティにおいて活躍してきた。

また Hickson はインタビューにおいて仕様に追加する機能の優先順位をどう決めるかという問いに対して、

Is the feature compatible with the Web philosophy? If the request is for a way to force the user to read something the way the author wants, it's likely to be a non-starter. If it's something that enables the user to control the page better, that's more likely to be a winner⁴⁶⁰.

と回答しており、自身の勤務先を含む全ての企業の利害ではなく、the Web Philosophy (ウェブの思想)というユーザを含む全てのステークホルダーが共有する価値観の下に標準化活動に取り組んでいることがわかる。

このようなエンジニアは Hickson だけに限らない。彼らは企業の利益を単に代弁するのではなく、企業とウェブ標準コミュニティとの仲介役を務めているのである。

新 HTML WG の初代 Co-Chair である Chris Wilson は、NCSA の Mosaic チームから Microsoft に入って長らく IE の開発に従事してきたが、2010 年 9 月に Google へと移籍している⁴⁶¹。Wilson は自身の LinkedIn ページに

⁴⁵⁸ Hickson, I. (n. d.). Resume. Retrieved February 2, 2015, from <http://ian.hixie.ch/career/resume.html>

⁴⁵⁹ The Web Standard Project. (n. d.). *メンバー紹介ページ(Ian Hickson)*. Retrieved February 2, 2015, from <http://www.webstandards.org/about/members/hixie/>

⁴⁶⁰ Lawson, B. (2013, January 8). Interview with Ian Hickson, *HTML editor*. Retrieved February 1, 2015, from <http://html5doctor.com/interview-with-ian-hickson-html-editor/>

⁴⁶¹ Wilson, C. (2010, September 21). On to a New Stage, *in progress (Chris Wilson の個人ブログ)*. Retrieved February 1, 2015, from <http://cwilso.com/2010/09/21/on-to-a-new-stage/>

**NOTE*: Recruiters: I've got a dream job -- making the web platform better -- with a team of all-stars at one of the few places that I can legitimately pursue that goal. I'm not interested in arbitrary tech jobs; I'm interested in making the web platform better for everyone⁴⁶².*

といった文章を掲出している。また、自身のブログにおいても、

*The Web platform's competition here *IS* all proprietary stacks – WPF, Silverlight, Flash, iOS, Windows Phone, Windows, Metro, MacOS, and yes, Android. That doesn't mean I hate all those stacks, or that I think they're evil – I just want the Web to be a viable competitor to them. When some developer somewhere starts building the NEXT Angry Birds, I hope it can be in the Web platform to begin with, and won't need to be ported to those other platforms⁴⁶³.*

といった文章を公開している。つまり、自身が所属する企業のプロプライエタリなプラットフォームよりも、オープン標準ベースのウェブプラットフォームの普及を優先させるという姿勢を明確に打ち出しているのである。Wilson は Acid 1 テストを開発し、オープンなウェブ標準の普及に取り組むコミュニティである Web Standard Project に 2000～2013 年の間所属し続けている⁴⁶⁴ことから、この姿勢が一貫したものであることがわかる。

極度に人材の流動性が高い技術分野において、標準化プロセスをクローズドにして参加資格を所属企業に紐付けることは、理にかなわないものとなっているのである。また、パテントポリシーがロイヤルティ・フリーでない場合、主要な開発者が移籍した時点で、仕様のブラッシュアップを継続するのは難しい。

⁴⁶² Wilson, C. (n. d.). *Chris Wilson の LinkedIn プロフィールページ*. Retrieved February 1, 2015, from <https://www.linkedin.com/in/cwilso>

⁴⁶³ Wilson, C. (2011, October 3). Promoting the Open Web, and Platform Competition, *in progress* (Chris Wilson の個人ブログ). Retrieved February 1, 2015, <http://cwilso.com/2011/10/03/promoting-the-open-web-and-platform-competition/>

⁴⁶⁴ Wilson, C. (n. d.). *Chris Wilson の LinkedIn プロフィールページ*. Retrieved February 1, 2015, from <https://www.linkedin.com/in/cwilso>

開発者が元々所属していた企業・団体と、移籍先との間で技術を共有することができないため、開発の継続そのものが難しくなる。もし開発が継続したとしても、両者が権利を有するロイヤルティの相殺や、ロイヤルティの分配ルールの規定が必要となる。ロイヤルティ・フリーの原則が徹底されていることにより、人材の流動性が高いウェブ技術領域での標準仕様策定が、潤滑に進んでいるといえよう。

9.2.2. 法人内で集約されていれば日の目を見なかった提案の取り込み

Web Intents はもともと Kinlan の個人的なプロジェクトとして始められ、Web Introducer の後継仕様として DAP に持ち込まれた。Kinlan は W3C 会員企業の Google に所属しているものの、役職は Developer Advocate であり、標準仕様開発・提案に関わる部署には所属していない。そのため、W3C がオープンプロセスを採用していなければ、ウェブアプリケーション間のデータ引き渡しという機能が標準化のアジェンダとなっており、提案されている関連仕様がいくつもあり、それがどこの WG で議論されており、議論のポイントが何なのか、を知ることができない。

そうなれば Kinlan はまず、Web Intents が法人として提案する仕様であるというステータスを得るために Google の社内調整を行う必要がある。Web Intents が Google 社の提案としてオーソライズされ、かつ Kinlan が Google の代表者として W3C の標準化プロセスに関与できる身分を得てはじめて、W3C 内での議論・調整を開始することができる。

プロセスがクローズドである場合、社内から賛同者を集めなければならない。Web Intents は、Kinlan の同僚であり、HTML5 仕様策定の中心人物である Ian Hickson の支持をとりつけることが最後までできなかった。プロセスがクローズドであれば、この段階で Web Intents は世に出なかった可能性が高い。

オープンプロセスであるからこそ、ウェブに仕様案を公開することで社内外から広く、迅速に支持者を集めることができ、標準化プロセスを開始することができたのである。

この事例では、個人がブログで発表した仕様が、標準化プロセスの遡上に乗れり、デモ等の形で多数の実装事例が公開され、それが以降の仕様策定に活かされている。これはオープンプロセスが採用されたことにより、会員企業内調整によって陽の目を見ない可能性のあった仕様提案が、実装するに値するとみなされれば仕様策定のプロセスに乗るということを意味する。オープンプロセ

スは、新規的な機能を実現する仕様提案を促す効果を有しているといえよう。

第10章. 考察と今後の課題

10.1. 思想としての W3C 標準化プロセス

W3C が HTML5 という標準仕様策定を通じて、ウェブアプリケーションというイノベーションを成立させ、かつネイティブアプリケーションからのドミナントデザイン移行を実現させることができた要因として、

H1: 仕様がカバーするスコープを「機能」で定義する

H2: 仕様策定プロセスにおける実装主義の導入

H3: 仕様策定プロセスのオープン化

の3つの標準化プロセス運営方針を仮説として導出した。また、アプリケーション間のデータ引き渡しプロトコル仕様の競合状態発生と解消という事象から、イノベーションの源泉を拡張し、組織間の垣根を超えた協働を促すために有効な要因として、

H4: ロイヤルティ・フリー・ポリシーの明文化

という仮説を導出してきた。

こういったポリシーは、W3C における 20 年にもわたる試行錯誤から生み出されたノウハウが明文化されたのである。W3C は方向性を定めずにただ経験則から標準化プロセス運営ポリシーを確立してきたわけではない。9.2.1 項で引用したインタビューにおいて Ian Hickson が使っていた言葉、the Web Philosophy (ウェブの思想)がステークホルダー間で共有されていたことは、仮説として導出された4つの手法が確立するために大きな影響を与えたと考えられる。

10.1.1. オープンなプロセス、オープンな仕様

ウェブの思想について、W3C は公式文書を公開していない⁴⁶⁵が、ウェブの発明者で W3C の Director である Tim Berners-Lee らによる著作 (Berners-Lee & Fischetti, 1999)の記述から類推することができる。主な箇所をいくつか引用して示す。

⁴⁶⁵ 2012 年に Philosophy of the Web Community Group(<http://www.w3.org/community/philoweb/>)が設置されたものの、メーリングリストへの投稿が 2015 年 1 月時点で合計 2 通、うち 1 通が Team Staff による開設の通知であるため、実質的な活動はなされていないと判断できる。

I wanted the consortium to run on an open process like the IETF's, but one that was quicker and more efficient (Berners-Lee & Fischetti, 1999, p. 92),

まずこの記述から、当初からオープンなプロセスが志向されていたことがわかる。また、

There was honor and pride at stake here, but also the future direction of the Web. The conference was the way to tell everyone that no one should control it, and the consortium could help parties agree on how to work together while also actually withstanding any effort by any institution or company to “control” things (Berners-Lee & Fischetti, 1999, p. 89).

特定の政府や企業に管理・コントロールされず、中立を保つことの重要性も指摘されている。実際に HTML5 の Editor である Ian Hickson は、インタビューにおいて所属する Google から、自社のビジネスに沿った形で標準仕様を策定するよう要求されたり、仕様策定時の決定において自社のビジネスが影響したりするかと質問され、以下のとおり回答している。

No, quite the opposite. When I started, I was given very explicit instructions to the effect that I should put the Web's long-term interests ahead of any of Google's short-term interests.

Having said that, of course, one of the reasons I wanted to work for Google is the unique perspective one can get from working here and from having access to the data Google has. There's no question that that has influenced my decisions⁴⁶⁶.

⁴⁶⁶ Lawson, B. (2013, January 8). Interview with Ian Hickson, *HTML editor*. Retrieved February 1, 2015, from <http://html5doctor.com/interview-with-ian-hickson-html-editor/>

ウェブアプリケーションの普及によって新たな収益源が確立できる Google は、雇用する Editor に対して利益追求のために介入することが差し控えていることがわかる。また、Microsoft も Web Standard Project で活動する Chris Wilson を IE の開発チームとして雇用し続けていたことから、Tim Berners-Lee の思想は HTML5 の仕様策定プロセスにおいて浸透していたといえるだろう。

Berners-Lee らの書籍 (Berners-Lee & Fischetti, 1999) には、以下の様な記述もある。

Written into the consortium's constitution is the stipulation that all the software it produces in support of its work be available to the public. This is a way of promoting recommendations, discussion, and experimentation. It allows anyone to join in the testing of new protocols, and allows new companies to rapidly get into the swing of Web software creation. All anyone has to do is go to the consortium(s) site, www.w3.org, and download these tools for themselves (Berners-Lee & Fischetti, 1999, p. 121).

ここから、仕様案、策定が完了した仕様などを含め、あらゆるものを公開する方針が予め示されていたのがわかる。

W3C の Head Quarter が設置されている MIT Laboratory for Computer Science (現 Computer Science and Artificial Intelligence Laboratory) は、もともと UNIX の X Window System を開発するハードウェアベンダを束ねる中立的な団体である X consortium を運営していた部門である。W3C の設立には X consortium の運営に関わったスタッフが深く関与しており (Berners-Lee & Fischetti, 1999)、2015 年 1 月時点での CEO である Ralph Swick は、元 X consortium のスタッフである。W3C は公共財としてのソフトウェア資産、つまり OSS を、企業を含んだ多数の主体間による協働から創出する Boundary organization (O'Mahony & Bechky, 2008) としての側面をもつ組織であり続けているといえる。

OSS と標準仕様は補完財/アプリケーションが共通して利用するプラットフォームとして機能する (Garud Et. Al., 2002) という点では共通であり、West (2003) は、OSS は実装が万人にフリー(無償で、自由に)供給されるという点で、オープンな標準の究極的な形態であるとする指摘もなされている。

個人、または企業間による協働でのソフトウェア開発プロジェクトは、参加者各自が自己の利益を追求しつつも協調し、公共財を生み出すという Private-collective model(von Hippel and von Krogh, 2003)の形態をとることがある。

それが、W3C のプロセス運営方針に強く反映されており、協調型イノベーションを成立させることのできるプロセス運営方針の確立に寄与したものと考えられる。

ウェブアプリケーションは、標準仕様のバージョンアップを通じた機能追加によって実現、普及した技術である。つまりオープンな場での多数の主体による協働によって実現したイノベーション、つまり Open-collaborative Innovation (Baldwin & von Hippel, 2011) であるといえる。

von Hippel (2005) は、OSS はユーザーイノベーションネットワークの結果形成された財であると指摘しており、協調型イノベーションの成立に OSS コミュニティの活動形態を取り入れることは自然な発想であるといえよう。

O'Mahony は、OSS である Devian 開発コミュニティの運営手法を分析し、OSS を中立的、かつオープンな資源としての性格を保持していくためには、

- (1) 私的所有を制限する形態の配布に関するライセンスを付与する
- (2) 規範的、法的な制裁措置を通じてライセンス条項遵守を促進する
- (3) 法人格を取得して資産を保持し、貢献者を不利益から護る
- (4) 個人の所有権を集団運営される非営利組織へ移管する
- (5) 資産を表すブランドやロゴを商標として登録する
- (6) 組織に登録商標を保有させる
- (7) 能動的にプロジェクトのブランドを保護する

という 7 つの戦略が採用されていると指摘している (O'Mahony, 2003, pp. 1183)。W3C は全ての戦略を採用しており⁴⁶⁷、OSS 開発コミュニティがこれまで採用し、成果を上げてきた戦略をうまくアレンジしながら適用しているといえる。

10.1.2. 学術コミュニティを基盤とした協働プラットフォーム

Berners-Lee らの書籍 (Berners-Lee & Fischetti, 1999) では、

⁴⁶⁷ (3)については、W3C 自体は法人格を有しないが、会員企業・団体は各 Host と契約し、会費を支払う。また仕様等の著作権は Host による協働保有の形態を採っている。

*I wanted the consortium to run on an open process like the IETF's
(Berners-Lee & Fischetti, 1999, p. 92)*

*Like the IETF, W3C would develop open technical specifications
(Berners-Lee & Fischetti, 1999, p. 94)*

のように、W3C の運営方針が IETF を参考にして構築していった旨が述べられている。インターネットはもともとアメリカ国防総省高等研究計画局の支援を受けていたものの、MIT、UC Berkeley といった大学や研究機関が中心となって開発され、仕様策定も研究者間における学術論文の査読をベースにしたプロセスで行われてきた。そしてその手法が現在まで IETF に引き継がれている (Mowery and Simcoe, 2002)。

Ian Jacobs は、インタビューにおいて以下のように回答しており、W3C がプロセスのオープン化をここまで成功させてきた要因として、ウェブ技術の開発が学術研究と密接に関連していることを挙げている。

We have not, traditionally, have a very elaborate developer relation program, but I think that's starting to change. We did our first conference, and there were again, we went to the members and said we need you to talk about this W3C conference you know where your employees are speaking, and get developers in the neighborhood to come to the conference. So I think that the members provide resources to W3C and engineers to W3C work in groups in lots of ways and clearly their developer departments are open to ideas to working with us to help promote the technologies they're working on the W3C. So that's a good trend to get more developers in member networks but also generally involved in W3C and where of W3C work⁴⁶⁸.

Ralph が触れている World Wide Web Conference は、ウェブ技術の領域における最高水準の学術会議であり、W3C 設立以前にはこの会議がブラウザ供給者を

⁴⁶⁸ 2012 年 9 月 12 日 Ian Jacobs 氏インタビュー

含むステークホルダーが一同に介して意見交換を行う場であった。この歴史が、オープンな議論に対する受容につながっていると考えられる。

なお、企業の標準化団体への参加動機に、情報交換、自社の評判形成、将来のアライアンス先候補の発見というものがある (Rosenkopf et al., 2001)。また、von Hippel は製鉄業界を分析し、20 世紀半ばには、同業のライバル企業間のノウハウ交換が行われ、業界全体の生産性向上につながってきたことを指摘している (von Hippel, 1987)。

そこから Collective invention (Allen, 1983)が生じており、企業間の協働によって Inter-organizational innovation (Van de Ven, 1993; Rosenkopf Et. Al., 2001) が成立していることを報告する先行研究が存在する。また、von Hippel (1987)は競合する企業間で協力してイノベーションを実現する事例を分析している。

情報通信技術の発達は、組織間コミュニケーションのコストを下げると同時に、情報漏洩のリスクへの対処などを通じてインフォーマルなコミュニケーションを難しくするという影響も生じさせている。このような状況下において、Process Document という明文化されたルールのもとで組織の垣根を超えてオープンに技術提案や議論がなされる場が維持されている W3C の仕様策定プロセス形態は、協調型イノベーション成立に大きく寄与していると考えられよう。

W3C は、新たな企業間コミュニケーションのためのフォーラムとしての機能を果たしている。この機能が果たしている一つの要因として、学術コミュニティをバックボーンとした W3C の出自があると考えることができる。

10.1.3. “Rough consensus, running code”の制度化としての実装主義 同書には、以下の様な記述もある。

We wrestled over terms — whether the consortium should actually set a “standard” or stop just short of that by issuing a formal “recommendation.” We chose the latter to indicate that getting “rough consensus and running code” — the Internet maxim for agreeing on a workable program and getting it out there to be tried — was the level at which we would work (Berners-Lee & Fischetti, 1999, p. 98).

この文章で触れられている “Rough consensus and running code”は、IETF にて

採用されている標準化プロセス運営方針である。Rough consensus は、厳密な合意形成を追求するのではなく緩やかな合意形成の確認（具体的には拍手によって確認される）によって方針を決めるという手法であり、running code は提案仕様が標準化プロセスに進むには、事前に複数の動作し、相互可用性のある実装事例が提示されていることが必要とされるというルール (Russell, 2006, pp. 55) である。

4.1 節で分析したとおり、Tim Berners-Lee が生み出したウェブの主要な技術仕様である Uniform Resource Locator、Hyper Text Transfer Protocol、そして HTML はまず IETF において標準が取り組まれ、そこで頓挫した HTML の標準化を進めるために W3C が設立されたという経緯がある。

仕様が確定する前に実装事例が供給されるという実装主義は、IETF のルールである Rough consensus and running code を明文化したものだといえるだろう。Tim Berners-Lee によって創られた、OSS 開発コミュニティが育て上げてきた文化と、インターネット開発の現場で培われてきた手法をベースにしたウェブの思想は、W3C の Process Document という形態で明文化され、協調型イノベーションを成立させる原動力となったのである。

10.2. プラットフォーム・ガバナンスと競争政策に関する議論

10.2.1. 垂直統合に対する政府の介入は必要か

8.2.1 項で Lawrence Lessig が標準化の手法を Coordinating と Regulation に分類し (Lessig, 1999-b)、W3C は coordinating 標準にあたること、イノベーションを実現するのは政府や国際機関による Regulation ではなく、Coordinating 標準の形態であることを述べた。では、標準仕様策定を含むインターネットならびにウェブという情報通信プラットフォームのガバナンスはどの主体が、どのような形態で担うべきなのであろうか。ここからは、これまで研究されてきた、互換性標準を策定する主体に関する議論、並びに政府・公共機関による関与についての先行研究をまとめるとともに、インターネットガバナンスやウェブガバナンスがどうあるべきかについて論じる。

Eisenmann et al. (2011) では、プラットフォームは「単一、もしくは複数の媒介者によって開発、運営されるものであり、利用者間の相互作用において用いられるコンポーネントやルール」と定義されている。また、プラットフォームの Tying 戦略、もしくは補完財のバンドルは、競合に対する参入障壁として機能する (Nalebuff, 2004)。そのため公的機関が仕様策定を担わない場合、この役

割を単一の事業者が専有することによる危険性を排除するために、独占の排除を必要とする指摘もなされている (Farrell & Weiser, 2003)。そのため、プラットフォームの補完財市場への参入という垂直統合を禁止すべきだという議論 (Choi & Stefanadis, 2001; Choi, 2004) もなされている。その一方、垂直統合が効率性を高めるという指摘もある (Yoo, 2002)。そのため垂直統合そのものを帰省すべきではないという意見は古くから存在する (Schumpeter, 1950; Hart et al., 1990)

Farrell and Katz (2000) や Farrell (2003) では、第三者に補完財を供給させることにはメリットがあるため、垂直統合がそのままプラットフォームの経済合理性につながるということではない。とはいえ、プラットフォームが補完財市場を第三者に開放した後に、事後的に競合する補完財を市場投入し、実質的に垂直統合を進めることは問題がある。そのため、独占禁止法によって単純に垂直統合を禁止することは有効でないという指摘がなされている。また Weiser (2003) は、規制はプラットフォーム間競争によって寡占状態が出現した際に行えばよいと提起している。

10.2.2. 互換性標準策定を担う主体は誰が担うべきか

これまでの議論を、プラットフォームとしての性質を有する互換性標準に当てはめれば、競争政策が機能するのであれば、あとはコーディネーションさえうまくいけば政府・公的機関が自ら標準仕様を策定する必要はないということになる。この意見は Lessig (1999-b)のみならず Farrell et al. (1998) も、アーキテクチャがオープンになっていれば、各コンポーネントにおけるイノベーションは自然発生し、市場参入も促され、低価格で消費者に供給されると結論づけている。

その結果、仕様のオープン化が進めば、政府が介入する必要性は少なくなり (Lessig, 1999-b)、民間によるオープン標準の方が効率的 Farrell (1989)であるという指摘もある。実際に TCP/IP を始めとするインターネット標準という成功事例も存在する (Mowery & Simcoe, 2002; Leiner et al., 2009)。そもそも、互換性標準は公共財である (Kindleberger, 1983) ため、オープンな標準を共有財とすべきという主張もなされている (Lessig, 1999-a, 2002)。Internet of Things や Web of Things など、多様なコンポーネントがインターネット/ウェブを介して通信することで初めて成立する機能・サービスの開発は、今後ますます重要になっていく。つまり、標準仕様の策定・バージョンアップを通じた協調型イノベーショ

ンが適用される分野は増大していくと考えられる。そのため、民間が標準化団体の機能を担い、Coordinating 標準を策定するという形態が広く採用されるべきであるとする。

10.2.3. 民間標準化団体の経済的自立を成立させる施策の必要性

W3C のように、標準化団体の運営を民間の研究機関が担い、オープンな場でステークホルダーの利害を調整しながら標準を作り上げるという運営方針は、協調型イノベーションの成立に寄与する要因である。しかし、民間の研究機関が標準化団体の運営を担うということは、標準化団体が経済的に自立できる状態であることが必要である。

仕様策定プロセスのオープン化は、仕様の収斂と迅速な普及を実現させる一方、W3C という標準化団体の運営にあたってはデメリットも生じることとなった。Ian Jacobs はこのように語っている。

So I think from the beginning, there was a recognition that public accountability and public availability of specification was important. But there was a desire to balance the member value with public accountability. So a lot of organizations were more comfortable working not in member, not in public space, and I just think over time some groups themselves pushed the, pushed the culture in the direction of working more in public, people saw there were, (that) it wasn't harmful, they saw benefits that's easier to share information, people were more interested, people have more trust in the process when it's more transparent, and so we seem to have been able to move in the direction of more public information and discussion without removing the value of membership. So the balance was one way early on, it shifted, because the culture, the people, who were participating, removed it, into the direction more public. So, that's what happened⁴⁶⁹.

W3C は会員が支払う会費によって運営されている。非会員が仕様策定プロセスにおいて発揮する影響力が大きくなればなるほど、会員のメリットは薄れて

⁴⁶⁹ 2012年9月12日 Ian Jacobs 氏インタビュー

いく。そのため、W3C を退会する会員が出現する場合も想定される。会員の減少は W3C の財政基盤を直撃する。標準化プロセスを運営する組織の基盤が損なわれるのは本末転倒である。オープンな標準仕様策定プロセスの経済的基盤、つまり会費収入を守りながら、関与するステークホルダーを拡大させるためのバランスの取り方は W3C にとって大きな課題である。

10.3. 実装主義の適用範囲についての検討

10.3.1. ソフトウェアの特徴と相性がいい実装主義

最後に、本論文で行った事例分析から導き出した仮説の適用範囲と、仮説を立証していくための課題についてまとめておきたい。

ウェブ標準仕様は、データ仕様の規定や API 設計を含むものであり、ソフトウェア開発の一種と位置づけられる。しかし、これまでほとんどのソフトウェア互換性標準仕様は先に仕様が確定し、実装された製品・サービスは後から供給されるというモデルであった(図 45)。

この、開発の後に実装という直線的なモデルは、ソフトウェア開発手法においてはウォーターフォールモデル(Royce, 1970)と呼ばれる。ウォーターフォールモデルとは、図 46 の通り予め構築するシステムの要件定義を行い、それに従って実装、テスト、手戻りなく手順を進めていく形の開発手法を指す。しかし、システムが大規模かつ複雑になると、当初の計画通りに開発が進むとは限らず、途中でどうしても手戻り (Iteration) が発生する。

また、ソフトウェアの規模が巨大化する中でモジュールが導入されると、各モジュールを開発するプロジェクト同士のすり合わせが必要となり、これを原因とした手戻りも生じることとなった。

これに対し、一度の実装で完璧なものをリリースすることを想定せず、実装に対して分析・評価を加え、ブラッシュアップしていくという反復型開発モデル (Larman and Basili, 2003) が提唱された。特に分析・評価からプロトタイプの実装を繰り返してシステムの完成度を高めていく手法は、スパイラル型開発モデル (Boehm, 1988) と呼ばれている。

ウェブを介して提供されるサービスやアプリケーションは、サーバ上で動作するプログラムと、HTML・CSS・JavaScript 等クライアントのブラウザに送り込まれて動作するによってプログラムによって構成されるソフトウェアにほかならない。そのため、ウェブアプリケーションは、サービスがリリースされてからも、継続的なソースコードの改善や機能追加が行われる場合が多い。



図 45. これまでの一般的なソフトウェア互換性標準のライフサイクルモデル

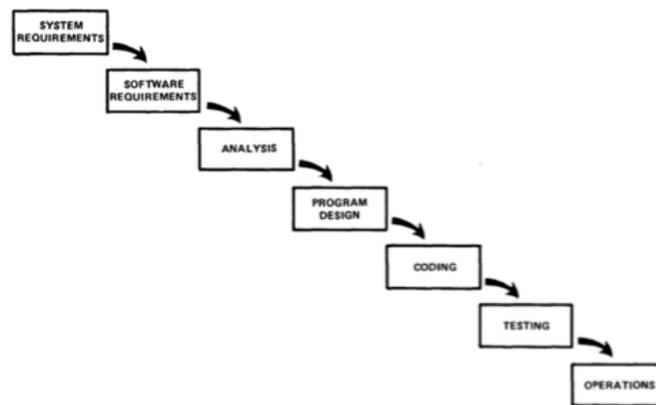


図 46. ウォーターフォール型開発モデルのステップ(Royce, 1970; pp. 329)

Google が 2000 年代に供給が始めたサービス⁴⁷⁰は、当時まずベータ版としてリリースされ、その後もバージョンコードが頻繁に変更された。完成度を高めてからサービスを開始するのではなく、早期にサービスを開始し、ユーザのニーズを踏まえて継続的に改善するという手法を採用する企業は、Google 以外にも多い (O'reilly, 2007)。このような手法が一般化したため、Google は 2009 年 7 月に全てのサービスからベータ版の表記を削除している⁴⁷¹。

実装主義は、実装事例を元に仕様をブラッシュアップするというプロセス運営方針である。フィードバックを元に仕様がブラッシュアップされたならば、既存の実装もそれに合わせて修正する必要がある。また、フィードバックの内

⁴⁷⁰ Google の Gmail など

⁴⁷¹ Glotzbach, M. (2009, July 7). Google Apps is out of beta (yes, really). *Google Official Blog*. Retrieved November 24, 2013, from <http://googleblog.blogspot.jp/2009/07/google-apps-is-out-of-beta-yes-really.html>

容によっては、広範囲な仕様変更が求められる場合もある。

本論文で分析した事例においても、DOM2 HTML や HTML5 で仕様書のステータスが初期に戻されている。また、HTML5 も 2012 年 12 月 17 日に CR となった⁴⁷²ものの、翌年 6 月 17 日に LCWD に差し戻されている⁴⁷³。この差し戻しは仕様の内容が安定しないものを切り出して、策定プロセスを継続するマイナーアップデートバージョンの HTML5.1 に移行し、残された部分についてコメントを求めて内容を固めるために実施されたものであった。このように柔軟に仕様のスコープに含む範囲を変更したり、仕様の中身を改変したりできるのはソフトウェアの特徴を前提としているといえよう。

Ian Jacobs はインタビューにおいて、

A little bit. I would say, we used to be more conservative when we talked to people about using the specifications. So we use to say a long time ago, wait until recommendation. And so now, because things are being deployed faster and because the browser development cycle, for example are much faster, it no longer is useful to tell people to wait. They're using stuff in practice and we sort of just said you know what, we actually also ask people for feedback, and we also ask them to implement in candidate recommendation, so we were sort of giving mixed messages, "wait but please implement", "wait but..." you know, so now we just say you should absolutely try it out, we love the feedback, and you may find that there's not the inter-operability yet that you need, because it's still evolving but as long as you understand the status of the work, then you should feel comfortable experimenting. So that has changed about how we talk about our specifications. I want to do a time check really fast. I have to get back to the advisory board very soon⁴⁷⁴.

と回答し、W3C の仕様策定プロセスの柔軟性はブラウザのバージョンアップ

⁴⁷² World Wide Web Consortium. (2012, December 17). HTML5 Definition Complete, W3C Moves to Interoperability Testing and Performance: First Draft of HTML 5.1 Offers Glimpse at Next Round of Standardization. Retrieved November 6, 2014, from <http://www.w3.org/2012/12/html5-cr.html>

⁴⁷³ World Wide Web Consortium. (2014, June 17). Last Call: HTML5. *W3C News*. Retrieved November 6, 2014, from <http://www.w3.org/blog/news/archives/3908>

⁴⁷⁴ 2012 年 9 月 12 日 Ian Jacobs 氏インタビュー

グレードが頻繁に行われていることに依拠しているという分析をしている。仕様そのものと、仕様に基づいて設計されるブラウザがソフトウェアであることが、実装主義が機能するための要件として重要だと考えられる。

なお、HTML5 の Last call for comment 期間はその後約一ヶ月後で終了し⁴⁷⁵、7 月 31 日には再び CR⁴⁷⁶、9 月 16 日には PR⁴⁷⁷、前述のとおり 10 月 28 日に Recommendation となって策定プロセスは完了している。このようなことから、実装主義は、ソフトウェアベースの互換性標準に対して有効であるといえる。

10.3.2. 財のサービス化による適用範囲拡大の可能性

一方、ハードウェアは一般的に金型や部品などの設計や、製造ライン構築にコストがかかることから、短期間に設計を変更することが難しい。そのため先に実装事例を創出し、フィードバックを受けて仕様を継続的に改変するという実装主義プロセスを適用するのは難しいと考えられる。

しかし、W3C Chief Operation Officer の Ralph Swick 氏はインタビューにおいて以下のように回答している。

the revision time for hardware, the rate which you can change things when you back close to hardware is very much different from the rate that we are accustom to the web technologies before that, so, for the first decade of W3C's work is the lifecycle of the time to revise the specification. It was only limited by how often Microsoft and Mozilla were willing to release a new product; they could release a new product very frequently compared to how often in the hardware product can be released, how often firmware can be updated~. So, I think it's interesting as a consumer. I have most of the interesting pieces of electronics I have now have upgradable firmware⁴⁷⁸.

ハードウェアの設計が固定化するという制約は、テレビ等のハードウェア製

⁴⁷⁵ Cotton, P. (2014, July 24). *WG Decision: Transition of HTML5 to Candidate Recommendation*. Retrieved November 6, 2014, from public-html-admin@w3.org mailing list archive: <http://lists.w3.org/Archives/Public/public-html-admin/2014Jul/0042.html>

⁴⁷⁶ World Wide Web Consortium. (2014, July 31). *W3C Invites Implementations of HTML5*. *W3C News*. Retrieved November 6, 2014, from <http://www.w3.org/blog/news/archives/3988>

⁴⁷⁷ World Wide Web Consortium. (2014, September 16). *Call for Review: HTML5 Proposed Recommendation Published*. *W3C News*. Retrieved November 6, 2014, from <http://www.w3.org/blog/news/archives/4074>

⁴⁷⁸ 2012 年 8 月 14 日 Ralph Swick 氏インタビュー

品がインターネットに接続し、内部で動作するソフトウェアが頻繁にアップデートされるようになったことで小さくなりつつあると指摘している。

W3Cでは、テレビコンテンツとウェブコンテンツの融合や、テレビ受像機や Set top box 等放送受信のための機器でウェブコンテンツを取り扱うための標準仕様を検討・開発する Web and TV IG が 2011 年 2 月 7 日より活動を開始している⁴⁷⁹。IG で展開された議論を通じて上記のような観点は実感をもたれているのである。

テレビのみならず、冷蔵庫や掃除機といった白物家電、自動車等の輸送機器においても、設計要素に占めるソフトウェアの重要性が増してきている。そして、Internet of Thing や Web of Things といったコンセプトに基づき、より多くの機器がインターネットに接続されるようになろうとしている。そのため、内部で動作するファームウェアがアップデート可能となりつつある。そのため、実装主義の適用が可能と考えられる財の領域が拡大していくと考えられる。

Michael Cusumano は、これまで製品 (Product) として捉えられてきた財の領域において、サービス化・プラットフォーム化が進展しており、他者を巻き込んだエコシステムの形成が競争優位に必要であると主張している。エコシステムの形成・発展には、構成する各主体間が協働して実現する目標・活動領域の規定 (scope)、役割分担やアウトプットの柔軟な変更などが必要である (Cusumano, 2010)。同書では、エコシステム戦略の成功事例としてトヨタ、Cisco Systems 等を挙げており、ハードウェア/物理財においても適用範囲が広がっていくと考えられる。

10.4. 本論文の貢献と今後の課題

本論文では、W3C における標準仕様策定プロセス確立の経緯を分析することにより、「協調型イノベーション」成立要因の仮説導出を行った。仕様がカバーするスコープを「機能」で定義することにより、競合する独自仕様の出現が抑制されること、実装主義と呼ばれる「標準仕様確立には実装事例が必要というルールを定め、早期の実装事例供給を促すとともに、フィードバックを受けて継続的に仕様を洗練させる」といったプロセス運営手法が仕様の収斂に有効であること、プロセスのオープン化が「イノベーションの源泉を広く求めるとと

⁴⁷⁹ Funahashi, Y. et al. (eds.). (2011). Progress Made Thus Far. *Web and TV Interest Group Report*. Retrieved November 7, 2014, from http://www.w3.org/2011/webtv/wiki/Web_and_TV_Interest_Group_Report#Progress_Made_Thus_Far

もに、開発にできるだけ多くの外部開発者を参加させることで受容性の拡大を高める」ことで、イノベーションの創出と普及に寄与すること、そしてロイヤルティ・フリー・ポリシーの明文化がイノベーションの源泉拡張と、組織間の垣根を超えた協働を創出するという仮説を導出した。

本論文の理論的貢献は、標準化という機能をイノベーション実現の手法として捉え直し、協調型イノベーションという概念を導出したこと、協調型イノベーションという概念の導出によってオープンイノベーション(Chesbrough, 2003)、Open-collaborative Innovation (Baldwin & von Hippel, 2011)の概念を拡大したこと、市場競争でもなく公的機関による強制にもよらない新たな標準の確立形態を可視化し、Co-opetition (Nalebuff & Brandenburger, 1997)が継続的に行われる場を構築するための要因に関する仮説を抽出したことである。

実務的な貢献は、ソフトウェアプラットフォームにおける互換性標準策定におけるガバナンス形態の効率化につながる事例分析を提供したことである。

しかしこの仮説は、W3C という単独事例の分析に依拠したものである。仮説を実証するために、他のソフトウェア標準/プラットフォームの事例分析を実施することが必要とされる。

類似の条件をもつ事例として、Linux OS 全体の開発・バージョンアッププロセスを分析することを検討している。Linux OS 開発プロセスに関する分析は多数存在するが⁴⁸⁰、その多くはコアカーネルと呼ばれる 1 コンポーネントの開発者コミュニティについて分析したものである。実際の Linux OS は、ユーザーインターフェイスやドライバなど多数のコンポーネントによって構成されており、各コンポーネントの開発は異なる開発者コミュニティによって担われている。そのため、各コンポーネントが協調して動作するためのコーディネーションが必要である。実装主義はソフトウェア開発プロセスの標準仕様策定に対する適用であるが、利害を共有しない（場合がある）多様なステークホルダーによる調整を通じた開発という観点から Linux というプラットフォームの開発過程を捉え直して行きたいと考える。

現在政府や自治体といった公的機関を中心に、保有データをプログラムで処理できる形態で公開するというオープンデータの試みが進められている。オープンデータの成功には、単に保有データを公開するだけでなく、各主体が公開したデータを同様に処理できるための標準仕様が必要となる。

⁴⁸⁰ Raymond (1999)、Hertel et al. (2003)、Lakhani and Panetta (2007)など多数。

オープンデータで公開されるデータには、地理情報や施設情報、気象情報なども含まれており、自動車や空調家電などのハードウェア製品の制御にも活用されることが見込まれている。このような、ハードウェア製品の制御に用いられるデータ仕様開発プロセスを分析することにより、実装主義がウェブアプリケーションのような純粋なソフトウェアプラットフォームのみに適用可能であるのかについて検討していきたい。

更には、IPアドレスやドメインネームといったインターネット資源管理を担う Internet Corporation for Assigned Names and Numbers (ICANN)の移管問題などインターネット/ウェブガバナンスの中でも、互換性標準策定以外の機能において、適した管理主体や手法を導出するため、多様なインターネットガバナンスを担う団体・機関の事例分析を行うことで、本論文の適用範囲を拡大していきたいと考える。

謝辞

修士論文 (深見, 2007)ならびにその基となった論文(深見 & 國領, 2007)は、当時流行していたソーシャルサービスの分析を通じて「意図せざる協働」の構造を抽出したものの、取り上げたサービスのほとんどはその後下火になってしまった。この状況を前にし、次に取り組むべき課題は、協働のスケラビリティを向上させるための要因を発見することとなった。

ウェブのスケラビリティを実現させている最大の要因、それはウェブ標準であると気づいた時、それまでほとんど意識することのなかった SFC にある組織、W3C の存在に気がついたのである。

まず、調査課題導出の試行錯誤から、W3C/Keio にインターンとして飛び込むまで、自分の直感のみを信じて突き進んだ私をサポートしてくださり続けた主査の國領二郎先生に心より感謝申し上げたい。

副査を務めていただいた神戸大学大学院の小川進先生には、W3C の Headquarter が設置されている MIT 滞在中に、Sloan School of Management に在籍する研究者と引きあわせて下さり、Open User Innovation 研究コミュニティに参加するきっかけをつくっていただくなど、研究の幅を広げてくださった。深く感謝申し上げたい。

同じく副査を務めていただいた村井純先生には、W3C/Keio でのフィールドワークをサポートして下さり、また日本のオープンデータに向けた取り組みへの参加を通じて標準化の当事者として活動する機会を作っていただいた。改めて感謝申し上げたい。2013年4月にロンドンで開催された W3C・英国 Open Data Institute・Open Knowledge Foundation 共催の Workshop である Open Data on the Web に Program Committee として参加できたことにより、より多面的に W3C における標準化プロセスを分析することができた。この経験なしには本論文は書き上げられなかったであろう。

副査の飯盛義徳先生には、修士過程からずっと見守っていただいた。飯盛先生の温かい指導、激励なしに、研究を継続することはできなかったであろう。本当にありがとうございました。

MIT Sloan School of Management の Michael Cusumano 教授には、MIT 滞在中にディスカッションさせていただき、貴重なアドバイスをいただいた。深く感

謝申し上げたい。

夏野剛先生には、NTT ドコモ在籍時の経験や、W3C Advisory Board への参加を踏まえたアドバイスを頂戴しただけでなく、W3C でのフィールドワークではきめ細かくサポートいただいた。共著(深見・夏野, 2011)を執筆させていただいた経験は、事例分析を進めるのに非常に役立った。改めて感謝申し上げたい。

W3C/Keio でのインターンを引き受けて下さった一色正男先生には、会議への参加のみならず、7.6.2 項で分析した開発者コミュニティとの接点を作っていたいただいた。その経験が、本論文におけるオープンプロセスによるマルチステークホルダー関与の重要性を認識する大きなきっかけとなった。深く感謝申し上げます。

萩野達也先生には、慶應義塾大学の AC Representative として Social Web Incubator Group などの分科会への参加登録など、折にふれて W3C での活動・フィールドワークをサポートしていただいた。改めて感謝申し上げます。

中村修先生には、W3C/Keio や次世代 Web 応用技術・ラボなど様々な機会でお世話になった。ご一緒に米国オーランドで開催された IETF に参加させていただいたことが、W3C という事例を相対的に分析することができる大きなきっかけとなった。深く感謝申し上げます。

法政大学経営学部の西川英彦先生には、Open User Innovation Workshop や日本マーケティング学会など様々な機会にアドバイスをいただいた。改めて感謝申し上げます。

ウェブ標準という世界に最初に触れるきっかけを作って下さり、様々な確度から研究生活を支えてくださった国立情報学研究所の武田英明先生、大向一輝先生にも深く感謝申し上げます。

W3C/Keio では、Michael (tm) Smith 氏、芦村和幸氏、石倉直子氏、吉澤直美氏、中島博敬氏にも大変お世話になった。W3C/MIT では、受入責任者を引き受けてくださった伊藤穰一 MIT Media Lab 所長、W3C/MIT office で暖かく迎えていただいた Tim Berners-Lee 氏、Jeffrey Jaffe CEO、Susan Westhaver 氏、Amy van der Hie 氏、Maria Auday 氏をはじめとする全ての W3C/MIT Team の皆様に改めて感謝申し上げます。

W3C/MIT では、Ralph Swick COO、Philippe Le Hégarret 氏、Ian Jacobs 氏、Douglas Alan Schepers 氏にはインタビューに協力いただいたのみならず、W3C/MIT での生活においても非常にお世話になった。深く感謝申し上げます。

W3C/MIT のシステムチームの Ted Guild 氏、Alexandre Bertails 氏、Denis Ah-Kang 氏と一緒に MIT キャンパスだけでなくマサチューセッツの様々な場所に一緒に赴いた経験は、W3C 文化に親しむことのできる貴重な機会だった。感謝したい。

当時フェローとして滞在されていた日本電信電話株式会社の石井晋司氏や中島博敬氏との時間があつたお陰でリラックスしてフィールドワークを進めることができた。感謝申し上げたい。Cambridge に時を同じくして滞在し、帰国後もオープンデータの仕様普及活動を共に進めた清水たくみ氏と一緒に過ごした日々が懐かしく思う。感謝申し上げたい。モンリオールのパブで一緒に飲めるのを楽しみにしている。

日本においてインタビューに協力してくださった西村和彦氏、横田昌彦氏、html5j.org のメンバーである白石俊平氏、小松健作氏、一條美和子氏、そして HTML5west.jp のみならず、様々な開発者コミュニティ活動やオープンデータなどの分野においてもご一緒している村岡正和氏に深くお礼申し上げたい。

本論文の執筆にあたっては、他にも数限りない人との出会いがあつた。技術標準をテーマとしているだけに、技術に対する理解にも大変てこずった。技術や背景を丁寧に教示くださり、論文執筆作業を進めることができるようにしてくださった多くのエンジニアに感謝申し上げたい。

博士課程に進学し、W3C/Keio office の扉を叩いた当時、HTML5 の標準化プロセスが完結するのはまだまだ先だと思われていた。現在進行形で仕様が開発され、実装事例が普及し、開発者コミュニティが立ち上がって成長するという現象をリアルタイムで目撃することとなった。

あまりに膨大なデータ、多岐にわたるアクティビティに何度もめげそうになりながらも、どうにか分析・執筆を進めていくうちに、気がつけば HTML5 の策定プロセスは完結してしまっていた。結果的にこの論文は HTML5 が Recommendation となり、World Wide Web が 25 周年、W3C が設立 20 周年という記念すべき年度に完成することとなった。萌芽的な事例ながら、波及力が大きいと直観し、分析を始めた HTML5 仕様策定という事象が結実するタイミングで、ウェブがその機能を拡大させる 2nd step に踏み出したそのタイミングで、ここまでの歴史を記述した博士論文を書き上げることができたのは、本当に幸せなことである。ウェブ標準、そして W3C との出会いを与えてくれた SFC というキャンパスに改めて感謝申し上げたい。

本論文がウェブと、社会がよりよい環境となるのに少しでも貢献できるならば、望外の喜びである。

付録 I. W3C の組織形態

l.i. 会員制の産業コンソーシアム

HTML3.2以降、HTMLの仕様策定を担っている標準化団体がWorld Wide Web Consortium (W3C)である。W3Cは1994年に創設された民間の任意団体⁴⁸¹であり、政府や国際機関から独立した標準化機関である。Teamと呼ばれる専任のスタッフを抱えるが、スタッフはHostと呼ばれる米国マサチューセッツ工科大学(MIT)、欧州情報処理数学研究コンソーシアム(ERCIM)⁴⁸²、日本の慶應義塾大学SFC研究所、中国の北京航空航天大学⁴⁸³という4研究機関によって雇用されている。また、Hostはオフィスも提供する。W3Cは会員制を採用しており、会員企業・団体が納入する年会費によって活動が賄われている。

l.ii. 組織形態

W3Cの組織形態は、図47のとおりに表示することができる。W3Cにおいて仕様策定を行うのはWorking Group (WG)である。WGは、全体のとりまとめをするChair、議論を元に仕様書案を作成、修正するEditor、議論に参加するMemberによって構成される。各WGには最低1名のW3C職員が、会議のセッティングやインフラ(メーリングリストや電話会議等)の管理を担うTeam Contactを担当する。

WGは4つの技術領域(Technology Domain)に分類され、W3C職員のDomain Leadが領域全体を統括する。

⁴⁸¹ W3Cは法人格を持たず、会員組織や参加者個人は、会費納入や著作権・知的財産権などの契約をホストの研究機関との間で取り交わしている。

⁴⁸² 設立当初は、フランス国立情報学自動制御研究所(INRIA)が欧州ホストを担っていた。

⁴⁸³ 2013年1月に、新たにホストを担うこととなった。(W3Cのプレスリリース「W3C Invites Chinese Web Developers, Industry, Academia to Assume Greater Role in Global Web Innovation」(<http://www.w3.org/2013/01/china-host.html.en>)より。(2013年8月10日アクセス))

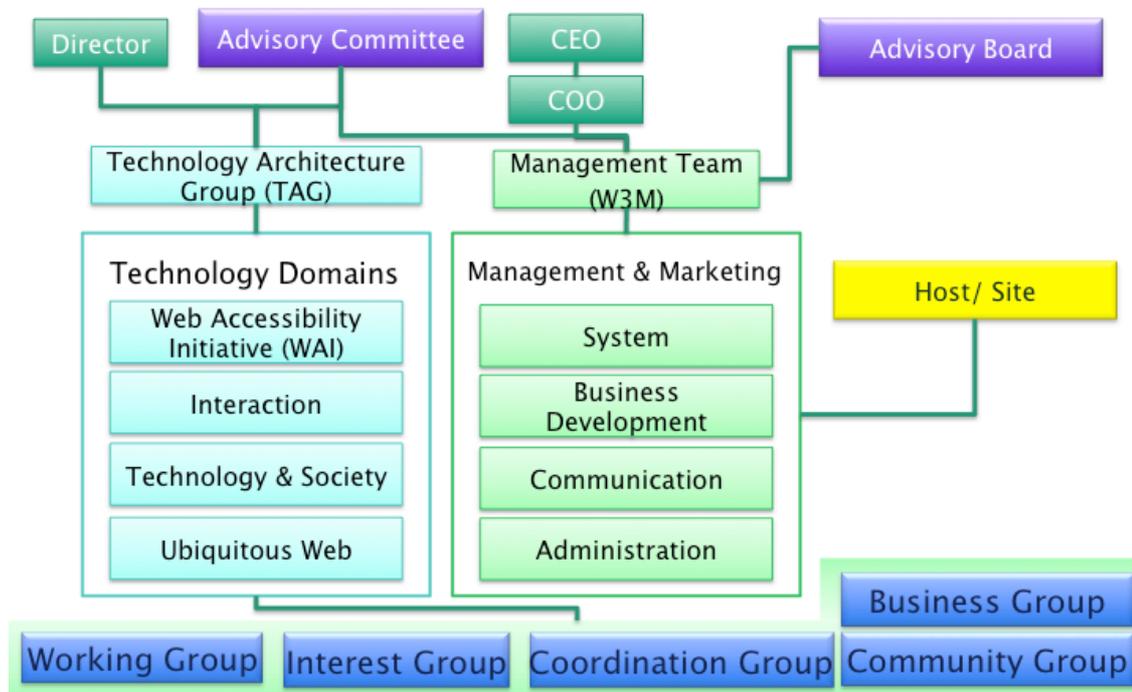


図 47. W3C の組織形態図

I.iii. Technology Architecture Group と Director

技術領域全体を統括し、ウェブ標準＝ウェブの進化の方向性を議論するのが Technology Architecture Group (TAG)である。TAG は、W3C 職員から 1 名と、会員内の選挙で選任された者、合計 8 名で構成される。TAG の議長は慣例的に⁴⁸⁴W3C 職員の代表が任命される。

Director はウェブを発明し、W3C を創設した Tim Berners-Lee が務める。Director の権限は大きく、仕様策定プロセスにおいては仕様提案(Draft)の受理や、標準化プロセスのステータス変更などにおいて Director 判断が求められる。

I.iv. Advisory Committee と Advisory Board

W3C は会員企業・団体によって納入される会費によって運営されている。Advisory Committee は、会員企業・団体の意見を W3C の組織運営や、仕様策定に反映させるために設置される機関であり、会員企業・団体の代表者によって構成される。

Advisory Board は、標準化プロセス運用ルールや、知的財産の取り扱いなど、

⁴⁸⁴ Process Document には、“The Team appoints the Chair of the TAG, who is generally the Director.” と記載されている。

標準化プロセスに関連する問題について討議する機関である。メンバーは、会員企業・団体間の互選によって選任される。Chair は、W3C CEO が務めることが慣例となっている。

l.v. Management Team

W3C は voluntary 標準であるため、策定仕様の普及促進のためのマーケティング施策が必要である。また会費で運営されているため、会員企業の新規勧誘ならびに既存会員の流出を防ぐための施策も必要である。このようなマーケティング施策や、標準化活動で必要とするインフラ整備、事務手続き等を担う部門が Management Team 部門である。なお W3C では、Technical Domain の Domain Lead、Activity Lead を含む全ての職員を Team Staff と呼称している。

l.vi. de Jure 標準との関係

W3C は 2010 年に国際標準化機構 (ISO) と国際電気標準会議 (IEC) により ISO/IEC JTC1 PAS Submitter に認定されており⁴⁸⁵、W3C 標準 (Recommendation) は、初期段階をスキップした形で ISO/IEC の標準化プロセスにおいて検討が開始される。つまり、de jure 標準化機関から認められており、W3C 標準は公的な性格を有するといえる。なお、HTML5 から一世代前のバージョンである HTML4.01 は、ISO/IEC 15445:2000 ならびに JIS X 4156:2000 という公的機関による標準仕様となっている。

⁴⁸⁵ World Wide Web Consortium. (2010, November 3). Global Adoption of W3C Standards Boosted by ISO/IEC Official Recognition. *W3C News Archive*. Retrieved October 19, 2014, from <http://www.w3.org/News/2010.html#entry-8950>

付録 II. HTML というウェブの基幹技術

II.i. ウェブ登場時のコンセプト

ウェブは、現在 W3C の Director を務める Tim Berners-Lee が、欧州原子核研究機構 (CERN) に勤務時に開発したツールが基盤となっている。開発時に提出された企画書⁴⁸⁶によると、開発の目的は “The system must allow any sort of information to be entered. Another person must be able to find the information, sometimes without knowing what he is looking for.”であり、1) Development Project Documentation、2) Document retrieval、3) The "Personal Skills Inventory" という 3 つの用途に活用できるとされている。

また Berners-Lee, et al. (1992) は、ウェブのデータモデルを以下のように表現している。

•Information need only be represented once, as a reference may be made instead of making a copy.

•Links allow the topology of the information to evolve, so modeling the state of human knowledge at any time is without constraint.

•The web stretches seamlessly from small personal notes on the local workstation to large databases on other continents.

•Indexes are documents, and so may themselves be found by searches and/or following links. An index is represented to the user by a "cover page" that describes the data indexed and the properties of the search engine.

•The documents in the web do not have to exist as files; they can be "virtual" documents generated by a server in response to a query or document name. They can therefore represent views of databases, or snapshots of changing data (such as the weather forecasts, financial information, etc.).

⁴⁸⁶ Berners-Lee, T. (2000). *Information management: A proposal (1989)(updated in 2000)*. Retrieved April 12, 2014, from <http://www.w3.org/History/1989/proposal.html>

このように、ウェブは研究者間の情報共有を目的とし、静的な文書を発表、交換、探索するためのインフラ(図 48)として開発されたのである。

なお Berners-Lee は、ウェブというインフラを構成する主要な仕様である Hyper Text Transfer Protocol (HTTP)、Hyper Text Markup Language (HTML)、HTML を保存する場所を示す識別子である Uniform Resource Locators (URL) というウェブを構成する基本要素に加え、HTTP によって送受信される文書を作成・表示するためのソフト⁴⁸⁷である“World Wide Web (ブラウザ)”を開発、提供している。

II.ii. 構造化文書としての HTML

HTML はウェブの発明時に独自の技術として開発されたものではなく、Standard Generalized Markup Language (SGML) をベースに開発されている。SGML は、IBM によって開発された Generalized Markup Language (GML) を改良して開発された構造化言語仕様である。なお SGML は、国際標準化機構 (International Organization for Standardization: ISO) によって標準化されている。

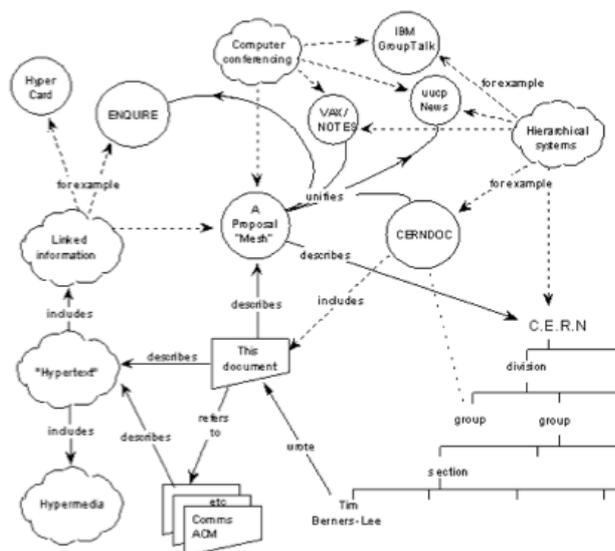


図 48. Tim Berners-Lee が CERN に提出したウェブの概念図⁴⁸⁸

⁴⁸⁷ Berners-Lee, T. (n. d.). *The worldwideweb browser*. Retrieved April 12, 2014, from <http://www.w3.org/People/Berners-Lee/WorldWideWeb.html>

⁴⁸⁸ Berners-Lee, T. (1990, May). Information management: A proposal (updated in May 1990). Retrieved April 12, 2014, from <http://www.w3.org/History/1989/proposal.html>

GML は、元々弁護士として法律事務所で働いていた Charles F. Goldfarb が、文書作成 ～ レビュー ～ 修正といった一連の作業を効率化するための方法として文書の構造化というアイデアに至ったところから開発が始まった。

HTML が人間同士で文書ファイルを容易に共有するために開発されているのに対し、SGML はシステム上で高度な処理が可能となるデータを記述することを目的として開発されている。人間にとって記述しやすく読み取りやすいデータ表現とシステム、つまりプログラム上で処理されやすいデータ表現は大きく異なる。

WYSIWYG (What You See Is What You Get) という言葉に代表されるように、人間がデータをハンドリングするためには、視覚的にわかりやすく複雑な規則が廃されていることが望ましい。一方プログラムで処理される場合は、ルールに従ってさえいけば、複雑な機能であっても処理可能である。そのかわり、1文字のスペルミス、記入漏れが致命的となる場合も多い。つまり、人間が取扱うのに適した仕様設計とプログラム処理に適した仕様設計は相反するものである。

GML、SGML は行政府や法律事務所、軍需産業、重工業、製薬産業など大量かつ複雑な文書を製作し、運用する業態のニーズを反映して開発され、そういった業界で普及が進んだ。そのために仕様は専門のエンジニアによって記述されることを想定し、高機能ではあるが複雑で、かつ厳格な規則が課せられるものとなった。

一方 HTML は、電子文書の専門家以外の幅広い層が自ら文書を製作し、情報発信・交換できることを目的として開発されている。そのため、SGML と比較すると機能が簡略化されているとともに、規則の厳格性を下げている (Berbers-Lee et al., 1999)。

Berners-Lee の狙いはあたり、ウェブは急速に普及することとなった。しかし機能的な面だけから見ると、HTML は SGML からは退化していると評価されることとなる。そして次世代 HTML のコンセプトとして、SGML から取り去った機能を復活・発展させるというものが出てきた。これが XHTML である。

W3C Team staff の Dave Raggett は 1998 年 1 月 18 日に、“XML extensions to HTML” と題したメモ⁴⁸⁹を公表している。このメモに書かれたコンセプトが、XHTML に発展していく。

XHTML の “X” は、XML という W3C の仕様に由来する。XML、正式名称

⁴⁸⁹ Raggett, D. (1998) *XML extensions to HTML*. Retrieved January 24, 2014, from <http://www.w3.org/People/Raggett/xml-in-html.html>

Extensive Markup Language は、「アプリケーションで処理するデータ交換に適した言語」として開発された。HTML は人間が読み取る文書を記述するために、タイトルや見出しといったレイアウトの規定に機能が絞られている。

一方、世の中に流通するデータは多種多様な構造をもっている。商品データベースに蓄積されるデータであれば、品番、商品カテゴリー、価格、原材料情報などの項目が必要である。同じ商品データベースであっても、生鮮食品とトイレットリーでは必要な項目が異なるはずである。XML はデータ構造をユーザが規定することができる。これにより、多様なニーズに対応したデータを生成することが可能になるのである。

HTML はウェブの発明時に独自の技術として開発されたものではなく、Standard Generalized Markup Language (SGML) をベースに開発されている。SGML は、IBM によって開発された Generalized Markup Language (GML) を改良して開発された構造化言語仕様である。なお SGML は、国際標準化機構 (International Organization for Standardization: ISO) によって標準化されている⁴⁹⁰。

GML は、元々弁護士として法律事務所で働いていた Charles F. Goldfarb が、文書作成 ～ レビュー ～ 修正といった一連の作業を効率化するための方法として文書の構造化というアイデアに至ったところから開発が始まった⁴⁹¹。

構造化された文書とは、文書を構成する内容が役割によって分割・整理され、各部分間の関係性が表されているという状態である。本論文は章節項目に分割されて階層構造をもっている。この章節項目への分割と階層構造（上位概念・下位概念の整理）、を行われている文書が構造化文書である。

構造化の利点は、同一のコンテンツをユーザの状況にあわせ最適化して提示することができるという点と、文書の修正や更新が容易になるという点である。文書中に含まれる文字列に対し、「章/節/項/目の見出しである」と定義すること（構造化）と、章の見出しに「“ヒラギノ明朝 Pro W3”の“20 ポイントサイズ”」というフォントを指定する（デザイン）のは別の行為である。構造化とデザインを分割して行えるようにすることで、「全ての章見出しを、明朝からゴシックに変える」といったことが可能となる(図 49)。

⁴⁹⁰ SGML は ISO 標準、ISO 8879:1986 である。また日本標準規格、JIS X 4151:1992 にもなっている。

⁴⁹¹ Goldfarb, C. F. (1996). *The roots of SGML -- a personal recollection*. Retrieved Jan 9, 2013, from <http://www.sgmlsource.com/history/roots.htm>.

第6章 事例 1：HTML 3.2/ HTML 4.0 ～
W3C 初期段階の標準化事例

- 6.1. 2バージョンの並行開発が行われた背景
W3Cで最初に策定された標準仕様はHTML3.2であったが、W3C内の開発ロードマップには、HTML3.2(コードネーム: Wilbur)とHTML4(コードネーム: Cougar)とが同時に構想されていた。各仕様のコンセプトは、当時の主要ブラウザが独自実装していた機能をできるだけ標準化することで相互可用性を確保する(HTML3.2)、HTMLがもつ元々の開発思想に基づいた標準仕様を開発する(HTML4)と異なるものであった。
- 6.1.1. 構造とデザインの分離
HTMLがもつ元々の開発思想とは、「構造化文書を作成するための言語」と



6.事例 1：HTML 3.2/ HTML 4.0 ～
W3C 初期段階の標準化事例

- 6.1. 2バージョンの並行開発が行われた背景
W3Cで最初に策定された標準仕様はHTML3.2であったが、W3C内の開発ロードマップには、HTML3.2(コードネーム: Wilbur)とHTML4(コードネーム: Cougar)とが同時に構想されていた。各仕様のコンセプトは、当時の主要ブラウザが独自実装していた機能をできるだけ標準化することで相互可用性を確保する(HTML3.2)、HTMLがもつ元々の開発思想に基づいた標準仕様を開発する(HTML4)と異なるものであった。
- 6.1.1. 構造とデザインの分離
HTMLがもつ元々の開発思想とは、「構造化文書を作成するための言語」と

文書を構造化して章節項目の見出しを指定しておき、デザインと分離することで、全ての見出しのフォントを一度に変更することが可能になる。

図 49. 文書を構造化する利点の一例

HTML では、タイトルを以下のように表す。

```
<title>The Effect of Implementation-oriented Policy</title>
```

これにより、“The Effect of Implementation-oriented Policy” がタイトルであるとブラウザ側が認識し、それに適した表現を行うことができる。あるブラウザでは太字で表示するであろうし、視覚障害者向けブラウザであれば「この文書のタイトルは“The Effect of Implementation-oriented Policy”です。」と音声で読み上げるであろう⁴⁹²。最近では同じ文書であっても、デスクトップ/ラップトップマシンとスマートフォンとで異なる表示となるサイトも多い(図 50)。デザインとコンテンツを分離し、コンテンツを構造化することにより用いられているデバイスの特性や、利用者の特性、利用時の状況などに応じて最適な形態で提供することが可能となる。文書の構造化は、多様な環境からのアクセスを前提とする際に必要な機能である。

また文書の順序を入れ替えても章番号や箇条書き記号、脚注や引用番号も自動的に変更することが可能である。またデザインの変更時も、構造化しておけば見出しレベルごとに一括してフォーマットを変更することが可能であり、各行ごとにフォントや色の変更する必要はなくなる。

⁴⁹² 文字列などをどのように表示するかについて記述する文書として、Cascading Style Sheets (CSS) という標準仕様が別途策定されている。

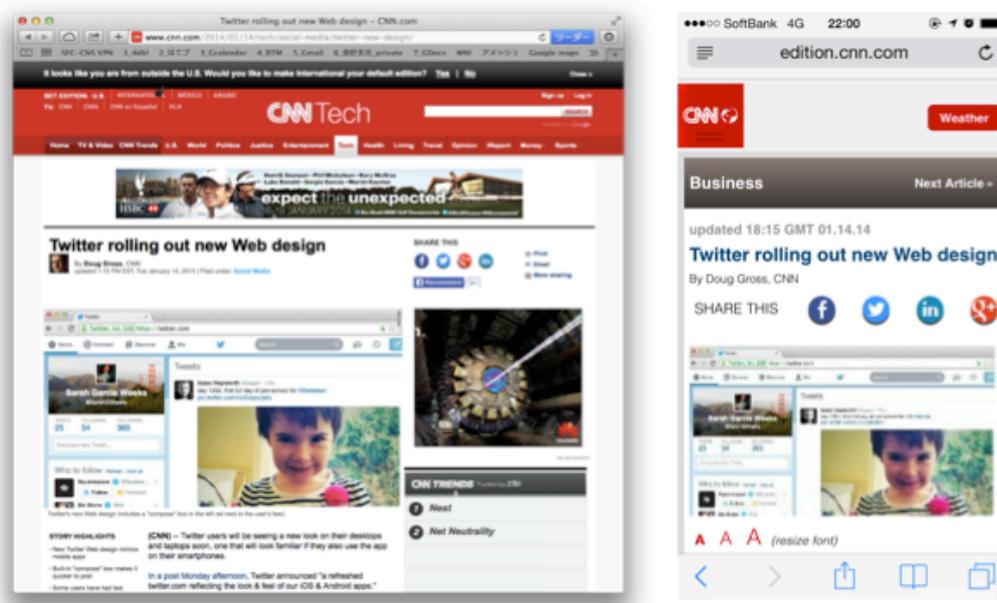


図 50. 同一サイト⁴⁹³を同一ベンダのブラウザ(Apple Safari)を用い、ラップトップマシン(左)とスマートフォン(右)で表示させた例

先に示した構造化文書の例において< >で囲まれた部分はタグ (tag) と呼ばれ、タグで囲んで用いる語彙は要素 (element) と呼ばれる。要素の種類を増やせば、それだけ多様な処理をコンピュータ側で行うことができる反面、文書作成の難易度は上昇する。

HTML の機能はタグ以外にも様々なものがあるが、機能が増えるほど文書記述は複雑となる。また、高度な計算機処理を可能にすればするほど、記述に厳密性が求められる。

II.iii. 構造化の機能に特化した XML の登場

HTML が人間同士で文書ファイルを容易に共有するために開発されているのに対し、SGML はシステム上で高度な処理が可能となるデータを記述することを目的として開発されている。人間にとって記述しやすく読み取りやすいデータ表現とシステム、つまりプログラム上で処理されやすいデータ表現は大きく異なる。

WYSIWYG (What You See Is What You Get) という言葉に代表されるように、

⁴⁹³ Gross, D. (2014, January 14). Twitter rolling out new Web design, *CNN.com*. Retrieved January 14, 2014, from <http://edition.cnn.com/2014/01/14/tech/social-media/twitter-new-design/>

人間がデータをハンドリングするためには、視覚的にわかりやすく複雑な規則が廃されていることが望ましい。一方プログラムで処理される場合は、ルールに従ってさえいけば、複雑な機能であっても処理可能である。そのかわり、1文字のスペルミス、記入漏れが致命的となる場合も多い。つまり、人間が取扱うのに適した仕様設計とプログラム処理に適した仕様設計は相反するものである。

GML、SGML は行政府や法律事務所、軍需産業、重工業、製薬産業など大量かつ複雑な文書を製作し、運用する業態のニーズを反映して開発され、そういった業界で普及が進んだ。そのために仕様は専門のエンジニアによって記述されることを想定し、高機能ではあるが複雑で、かつ厳格な規則が課せられるものとなった。

一方 HTML は、電子文書の専門家以外の幅広い層が自ら文書を製作し、情報発信・交換できることを目的として開発されている。そのため、SGML と比較すると機能が簡略化されているとともに、規則の厳格性を下げている (Berbers-Lee et al., 1999)。

Berners-Lee の狙いはあたり、ウェブは急速に普及することとなった。しかし機能的な面だけから見ると、HTML は SGML からは退化していると評価されることとなる。

W3C 創設後、HTML 開発にあたり SGML から取り去った機能を復活・発展させた仕様を開発する動きが登場する。この動きは、後に W3C 外に競合する仕様策定の活動が生まれるという自体を引き起こす。2000 年代半ば以降、シンプルな HTML のコンセプトをできるだけ踏襲したままニーズに合わせて機能を拡張仕様という路線 (HTML5) と、SGML のコンセプトに立ち返って HTML を作り変えようという路線 (XHTML) との間で激しい争いが生じることになったのである。

参考文献

- Abernathy, W. J., & Clark, K. B. (1985). Innovation: Mapping the winds of creative destruction. *Research policy*, 14(1), 3-22.
- Abernathy, W. J., & Utterback, J. M. (1978). Patterns of industrial innovation. *Technology review*, 80(7): 40-47.
- Allen, R.C. (1983). Collective invention. *Journal of Economic Behavior and Organization*, 4(1), 1-24.
- Almirall, E., & Casadesus-Masanell, R. (2010). Open versus closed innovation: A model of discovery and divergence. *Academy of Management Review*, 35(1), 27-47.
- Anvaari, M., & Jansen, S. (2013). Architectural openness: comparing five mobile platform architectures. In Jansen, S., Cusumano, M. A., & Brinkkemper, S. (Eds.). *Software ecosystems: analyzing and managing business networks in the software industry*, (pp. 138-158). Edward Elgar Publishing.
- Arthur, W. B. (1989). Competing technologies, increasing returns, and lock-in by historical events. *The Economic Journal*, 99(394), 116-131.
- Baldwin, C. Y., & Clark, K. B. (2000). *Design rules: The power of modularity*. MIT Press.
- Baldwin, C., & von Hippel, E. (2011). Modeling a paradigm shift: From producer innovation to user and open collaborative innovation. *Organization Science*, 22(6), 1399-1417.
- Baldwin, C. Y., & Woodard, C. J. (2009). The architecture of platforms: A unified view. In Gawer, A. (Ed.). *Platforms, markets and innovation*. (pp. 19-44) Edward Elgar Publishing.
- Berners-Lee, T., Cailliau, R., Groff, J. F., & Pollermann, B. (1992). World Wide Web: The information universe. *Electronic Networking: Research, Applications and Policy*, 2(1), 52-58.
- Berners-Lee, T., & Fischetti, M. (1999). Weaving the web: the original design and ultimate destiny of the World Wide Web by its inventor. Harper Collins Publishers Inc.
- Besen, S. M. (1992). AM versus FM: The Battle of the Bands. *Industrial and Corporate Change*, 1(2), 375-396.

- Boehm, B. W. (1988). A spiral model of software development and enhancement. *IEEE Computer*, 21(5), 61-72.
- Boswell, D. (Ed.). (2002). *Creating applications with Mozilla*. O'Reilly Media.
- Bogers, M. (2011). The open innovation paradox: knowledge sharing and protection in R&D collaborations. *European Journal of Innovation Management*, 14(1), 93-117.
- Bresnahan, T. F. (1999). New modes of competition. In Eisenach, J. A., & Lenard, T. M. (Eds.), *Competition, innovation and the Microsoft monopoly: Antitrust in the digital marketplace*, (pp. 155-208). Kluwer Academic Publisher.
- Bresnahan, T. F., & Greenstein, S. (1999). Technological competition and the structure of the computer industry. *The Journal of Industrial Economics*, 47(1), 1-40.
- Burgelman, R. A., & Andrew, S. (2001). *Strategy is destiny: How strategy-making shapes a company's future*. The Free Press.
- Carlton, D. W., & Waldman, M. (2002). The strategic use of tying to preserve and create market power in evolving industries. *RAND Journal of Economics*, 33 (2), 194-220.
- Casadesus-Masanell, R., & Yoffie, D. B. (2007). Wintel: Cooperation and conflict. *Management Science*, 53(4), 584-598.
- Chesbrough, H.W. (2003). *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business School Press.
- Choi, J. P. (2004). Tying and innovation: A dynamic analysis of tying arrangements. *The Economic Journal*, 114(492), 83-101.
- Choi, J. P., & Stefanadis, C. (2001). Tying, investment, and the dynamic leverage theory. *RAND Journal of Economics*, 32(1), 52-71.
- Christensen, C. (1997). *The innovator's dilemma: When new technologies cause great firms to fail*. Harvard Business School Press.
- Church, J., & Gandal, N. (1992). Network effects, software provision, and standardization. *The Journal of Industrial Economics*, 40(1), 85-103.
- Cusumano, M. A. (2010). *Staying power: Six enduring principles for managing strategy and innovation in an uncertain world (Lessons from Microsoft, Apple, Intel, Google, Toyota and more)*. Oxford University Press.
- Cusumano, M. A., Mylonadis, Y., & Rosenbloom, R. S. (1992). Strategic maneuvering and mass-market dynamics: The triumph of VHS over Beta. *Business History Review*, 66(1), 51-94.

- David, P. A. (1987). Some new standards for the economics of standardization in the information age. In Dasgupta, P., & Stoneman, P. (Eds.) *Economic policy and technological performance*, (pp. 206-239). Cambridge University Press.
- David, P. A., & Greenstein, S. (1990). The economics of compatibility standards: An introduction to recent research. *Economics of innovation and new technology*, 1(1-2), 3-41.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319-340.
- Eisenmann, T. Parker G. and Van Alstyne, M. (2006). Strategies for two-sided markets, *Harvard Business Review*. 84(19): 92-101.
- Eisenmann, T., Parker, G., & Van Alstyne, M. (2011). Platform envelopment. *Strategic Management Journal*, 32(12), 1270-1285.
- Evans, D. S., & Schmalensee, R. (2002). Some economic aspects of antitrust analysis in dynamically competitive industries. In A. B. Jaffe, J. Lerner & S. Stern (Eds.) *Innovation policy and the economy, Volume 2* (pp. 1-50). MIT Press.
- Evans, D. S., & Schmalensee, R. (2007). *Catalyst code: The strategies behind the world's most dynamic companies*. Harvard Business School Press.
- Evans, P., & Wolf, B. (2005). Collaboration rules. *Harvard Business Review*, 83(7), 96-104.
- Farrell, J. (1989). Standardization and intellectual property. *Jurimetrics*, 30(1), 35-50.
- Farrell, J. (2003). Integration and independent innovation on a network. *American Economic Review*, 93(2), 420-424.
- Farrell, J., & Katz, M. L. (2000). Innovation, rent extraction, and integration in systems markets. *The Journal of Industrial Economics*, 48(4), 413-432.
- Farrell, J., Monroe, H. K., & Saloner, G. (1998). The vertical organization of industry: Systems competition versus component competition. *Journal of Economics & Management Strategy*, 7(2), 143-182.
- Farrell, J., & Saloner, G. (1985). Standardization, compatibility, and innovation. *The RAND Journal of Economics*, 16(1), 70-83.
- Farrell, J., & Saloner, G. (1986). Installed base and compatibility: Innovation, product preannouncements, and predation. *The American Economic Review*, 76(5), 940-955.
- Farrell, J., & Shapiro, C. (1989). Optimal contracts with lock-in. *The American*

- Economic Review*, 79(1), 51-68.
- Farrell, J., & Simcoe, T. (2012). Choosing the rules for consensus standardization. *The RAND Journal of Economics*, 43(2), 235-252.
- Farrell, J., & Weiser, P. J. (2003). Modularity, Vertical Integration, and Open Access Policies: Towards A Convergence of Antitrust and Regulation in The Internet Age. *Harvard Journal of Law & Technology*, 17(1), 85-134.
- Gao, L. S., & Iyer, B. (2006). Analyzing complementarities using software stacks for software industry acquisitions. *Journal of Management Information Systems*, 23(2), 119-147.
- Garud, R., Kumaraswamy, A. and Langlois, R.N. (2002). *Managing in the modular age: Architectures, Networks, and Organizations*. Blackwell Publishing.
- Gawer, A., & Cusumano, M. A. (2002). *Platform leadership*. Harvard Business School Press.
- Gawer, A., Cusumano, M. A., & Strategy, D. S. (2008). How companies become platform leaders. *MIT Sloan Management Review*, 49(2), 28-35.
- Gefen, D., & Keil, M. (1998). The impact of developer responsiveness on perceptions of usefulness and ease of use: an extension of the technology acceptance model. *ACM SIGMIS Database*, 29(2), 35-49.
- Grindley, P. (1995). *Standards, strategy, and policy: cases and stories*. Oxford University Press.
- Habermeier, K. F. (1990). Product use and product improvement. *Research Policy*, 19(3), 271-283.
- Hars, A., & Ou, S. (2002). Working for Free? Motivations for Participating in Open-Source Projects. *International Journal of Electronic Commerce*, 6(3), 25-39.
- Hart, O., Tirole, J., Carlton, D. W., & Williamson, O. E. (1990). Vertical integration and market foreclosure. *Brookings papers on economic activity. Microeconomics*, 1990, 205-286.
- Henderson, R. M., & Clark, K. B. (1990). Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. *Administrative Science Quarterly*, 35(1), 9-30.
- Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel. *Research policy*, 32(7), 1159-1177.

- Iansiti, M., & Levien, R. (2004). Strategy as ecology. *Harvard Business Review*, 82(3), 68-81.
- Ives, B., & Olson, M. H. (1984). User Involvement and MIS Success: A Review of Research. *Management Science*, 30(5), 586-603.
- Jansen, S., & Cusumano, M. A. (2013). Defining software ecosystems: a survey of software platforms and business network governance. In Jansen, S., Cusumano, M. A., & Brinkkemper, S. (Eds.), *Software ecosystems: Analyzing and managing business networks in the software industry*. (pp. 13-28). Edward Elgar Publishing.
- Jansen, S., Brinkkemper, S., Souer, J., & Luinenburg, L. (2012). Shades of gray: Opening up a software producing organization with the open software enterprise model. *Journal of Systems and Software*, 85(7), 1495-1510.
- Katz, M. L., & Shapiro, C. (1985). Network externalities, competition, and compatibility. *The American Economic Review*, 75(3), 424-440.
- Katz, M. L., & Shapiro, C. (1986). Technology adoption in the presence of network externalities. *The Journal of Political Economy*, 94(4), 822—841.
- Katz, M. L., & Shapiro, C. (1992). Product Introduction with Network Externalities. *Journal of Industrial Economics*, 40(1), 55-83.
- Kindleberger, C. P. (1983). Standards as public, collective and private goods. *Kyklos*, 36(3), 377-396.
- Klemperer, P. (1987). Markets with Consumer Switching Costs. *The Quarterly Journal of Economics*, 102(2), 375-94.
- Krugman, P. (1991). History versus expectations. *The Quarterly Journal of Economics*, 106(2), 651-667
- Lakhani, K. R., & Panetta, J. A. (2007). The principles of distributed innovation. *Innovations*, 2(3), 97-112.
- Lakhani, K. R., & Wolf, R. G. (2005). Why hackers do what they do: Understanding motivation and effort in free/open source software projects. In Feller, J., Fitzgerald, B., Hissam, S. A., & Lakhani, K. R. (Eds.). *Perspectives on free and open source software*. (pp. 3-22) MIT Press.
- Langlois, R. N. (2002). Modularity in technology and organization. *Journal of Economic Behavior & Organization*, 49(1), 19-37.
- Langlois, R. N., & Robertson, P. L. (1992). Networks and innovation in a modular system: Lessons from the microcomputer and stereo component

- industries. *Research Policy*, 21(4), 297-313.
- Larman, C., & Basili, V. R. (2003). Iterative and incremental developments: a brief history. *IEEE Computer*, 36(6), 47-56.
- Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., Postel, J. P., Roberts, L. G. & Wolff, S. (2009). A brief history of the Internet. *ACM SIGCOMM Computer Communication Review*, 39(5), 22-31.
- Leonard-Barton, D., & Sinha, D. K. (1993). Developer-user interaction and user satisfaction in internal technology transfer. *Academy of Management Journal*, 36(5), 1125-1139.
- Lessig, L. (1999-a). *Code and other laws of cyberspace*. Basic books.
- Lessig, L. (1999-b). The Limits in Open Code: Regulatory Standards and the Future of the Net. *Berkeley Technology Law Journal*, 14, 759-1117.
- Lessig, L. (2002). *The future of ideas: The fate of the commons in a connected world*. Random House.
- Lieberman, M. B., & Montgomery, D. B. (1988). First-mover advantages. *Strategic Management Journal*, 9, 41-58.
- Link, A. N. (1983). Market structure and voluntary product standards. *Applied Economics*, 15(3), 393-401.
- Lynn, L. H., Mohan Reddy, N., & Aram, J. D. (1996). Linking technology and institutions: the innovation community framework. *Research policy*, 25(1), 91-106.
- McKeen, J. D., Guimaraes, T., & Wetherbe, J. C. (1994). The relationship between user participation and user satisfaction: an investigation of four contingency factors. *MIS Quarterly*, 18(4), 427-451.
- Meyer, M. H., & Lehnerd, A. P. (2011). *The power of product platforms*. Free Press.
- Morris, C. R., & Ferguson, C. H. (1993). How architecture wins technology wars. *Harvard Business Review*, 71(2), 86-96.
- Mowery, D. C., & Simcoe, T. (2002). Is the Internet a US invention? - an economic and technological history of computer networking. *Research Policy*, 31(8), 1369-1387.
- Nalebuff, B. J. (2004). Bundling as an Entry Barrier. *The Quarterly Journal of Economics*, 119(1), 159-187.
- Nalebuff, B. J., & Brandenburger, A. M. (1997). Co-opetition: Competitive and cooperative business strategies for the digital economy. *Strategy & Leadership*, 25(6), 28-35.

- North, D. C. (1990). *Institutions, institutional change and economic performance*. Cambridge university press.
- O'Mahony, S. (2003). Guarding the commons: how community managed software projects protect their work. *Research Policy*, 32(7), 1179-1198.
- O'Mahony, S., & Bechky, B. A. (2008). Boundary organizations: Enabling collaboration among unexpected allies. *Administrative Science Quarterly*, 53(3), 422-459.
- O'reilly, T. (2007). What is Web 2.0: Design patterns and business models for the next generation of software. *Communications & Strategies*, 65(1), 17-37.
- Parker, G. G., & Van Alstyne, M. W. (2005). Two-sided network effects: A theory of information product design. *Management Science*, 51(10), 1494-1504.
- Quittner, J., & Slatalla, M. (1998). *Speeding the net: The inside story of Netscape, How it challenged Microsoft and changed the world*. Grove/Atlantic, Inc.
- Raggett, D., Lam, J., Alexander, I., & Kmiec, M. (1998). *Raggett on HTML 4*. Addison-Wesley Longman Publishing.
- Raymond, E. S. (1999). *The Cathedral and the bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly.
- Roberts, J. A., Hann, I. H., & Slaughter, S. A. (2006). Understanding the motivations, participation, and performance of open source software developers: A longitudinal study of the Apache projects. *Management Science*, 52(7), 984-999.
- Rochet, J.C. & Tirole, J. (2003). Platform competition in two-sided markets. *Journal of the European Economic Association*: 1(4): 990-1029.
- Rosenbloom, R. S., & Cusumano, M. A. (1987). Technological pioneering and competitive advantage: The birth of the VCR industry. *California Management Review*, 29(4), 51-76.
- Rosenkopf, L., Metiu, A., & George, V. P. (2001). From the bottom up? Technical committee activity and alliance formation. *Administrative Science Quarterly*, 46(4), 748-772.
- Royce, W. W. (1970). Managing the development of large software systems. *In Proceedings of IEEE WESCON*, 1-9.
- Russell, A. L. (2006). 'Rough Consensus and Running Code' and the Internet-OSI Standards War. *IEEE Annals of the History of Computing*, 28(3), 48-61.
- Sanchez, R., & Mahoney, J. T. (1996). Modularity, flexibility, and knowledge

- management in product and organization design. *Strategic Management Journal*, 17, 63-76.
- Sanchez, R., & Sudharshan, D. (1993). Real-time market research. *Marketing Intelligence & Planning*, 11(7), 29-38.
- Schumpeter, J. A. (1950). *Capitalism, socialism, and democracy*, 3rd Edition. Harper and Brothers.
- Shapiro, C., & Varian, H. R. (1999). The art of standards wars. In Kumaraswamy, A., & Langlois, R. (Eds.), *Managing in the Modular Age: architectures, networks, and organizations* (pp. 247-272). John Wiley & Sons.
- Simcoe, T. (2007). Delay and de jure standardization: Exploring the slowdown in Internet standards development. In S. Greenstein and V. Stango (Eds.), *Standards and Public Policy*. (pp. 260-295). Cambridge University Press.
- Simon, H. A. (1962). The architecture of complexity. In *Proceedings of the American Philosophical Society*. 106(6), 467-482.
- Teece, D. J. (1986). Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Research Policy*, 15(6), 285-305.
- Tushman, M., & Murmann, J.P. (2002). Dominant designs, technology cycles, and organizational outcomes. In Garud, R. and Kumaraswamy, A. and Langlois, R. N. (Eds.) *Managing in the modular age: architectures, networks, and organizations*: (pp. 316-348). Blackwell Publishing.
- Ulrich, K. (1995). The role of product architecture in the manufacturing firm. *Research Policy*, 24(3), 419-440.
- Utterback, J.M. (1996). *Mastering the dynamics of innovation*. Harvard Business Press.
- Utterback, J. M., & Meyer, M. H. (1993). The product family and the dynamics of core capability. *Sloan Management Review*, 34(3), 29-47.
- Van de Ven, A. H. (1993). A community perspective on the emergence of innovations. *Journal of Engineering and Technology Management*, 10(1), 23-51.
- von Hippel, E. (1976). The dominant role of users in the scientific instrument innovation process. *Research Policy*. 5(3), 212-239.
- von Hippel, E. (1986). Lead users: a source of novel product concepts. *Management science*, 32(7), 791-805.
- von Hippel, E. (1987). Cooperation between rivals: informal know-how

- trading. *Research policy*, 16(6), 291-302.
- von Hippel, E. (2005). Open source software projects as user innovation networks. In Feller, J., Fitzgerald, B., Hissam, S. A., & Lakhani, K. R. (Eds.). *Perspectives on free and open source software*. (pp. 267-278) MIT Press.
- von Hippel, E., & Katz, R. (2002). Shifting innovation to users via toolkits. *Management Science*, 48(7), 821-833.
- von Hippel, E., & von Krogh, G. (2003). Open source software and the “private-collective” innovation model: Issues for organization science. *Organization Science*, 14(2), 209-223.
- von Krogh, G., Spaeth, S., & Lakhani, K. R. (2003). Community, joining, and specialization in open source software innovation: a case study. *Research Policy*, 32(7), 1217-1241.
- Weiser, P. J. (2003). The Internet, innovation, and intellectual property policy. *Columbia Law Review*, 103(3), 534-613.
- West, J. (2003). How open is open enough?: Melding proprietary and open source platform strategies. *Research Policy*, 32(7), 1259-1285.
- Whinston, M. D. (1989). Tying, foreclosure, and exclusion. *NBER Working Paper 2995*. Retrieved September 12, 2014, from <http://www.nber.org/papers/w2995>
- Williamson, O. E. (1975). *Markets and hierarchies: Antitrust analysis and implications*. Free Press.
- World Wide Web Consortium. (1997). HTML background setting standards W3C's ERB reflects on progress, process, and HTML 3.2. *World Wide Web Journal*, 2(1), 5-11.
- Yin, R. K. (2008). *Case study research: Design and methods, Fourth edition*. SAGE Publications.
- Yoo, C. S. (2002). Vertical integration and media regulation in the new economy. *Yale Journal on Regulation*, 19, 171-541.
- Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research*, 21(4), 724-735.
- 白石俊平.(2007). *Google Gears スタートガイド*. 技術評論社.

- 深見嘉明. (2007). ソーシャルタギング -意図せざる協働が生み出す新たな情報流通形態-. 慶應義塾大学大学院政策・メディア研究科 2007 年度修士論文.
- 深見嘉明 & 國領二郎. (2007). 意図せざる協働ーソーシャルブックマークにおけるボトムアップメタデータ生成による情報共有ー. *情報社会学会誌*, 2(2), 6-19.
- 深見嘉明 & 夏野剛. (2011). エコシステム形成のプラットフォーム:標準化活動の行動分析, 國領二郎 & プラットフォームデザイン・ラボ (eds.) *創発経営のプラットフォーム*. 119-154. 日本経済新聞出版社.

