Journal of Media Critiques [JMC]
doi: 10.17349/jmc117106 P-ISSN: 2056-9785 E-ISSN: 2056 9793
http://www.mediacritiques.net jmc@mediacritiques.net

HTML5, DIGITAL RIGHTS MANAGEMENT (DRM), AND THE RHETORIC OF OPENNESS

MICHAEL S. DAUBS[*]

ABSTRACT

This paper traces linkages between the commoditisation of the Web and "app-centric media", an environment composed of a multitude of concrete-but-connected software applications. Within this environment, multiplatform HTML5 apps are often framed as the antithesis of Apple's iOS and Google/Android "siloed" mobile app platforms, but this rhetoric of openness masks corporate involvement in the development of HTML5 and the commoditisation of the very protocols used to build the Web. To illustrate this process, this paper examines one new element of HTML5 that was hotly debated: the inclusion of digital rights management (DRM) protocols. Proponents of DRM in HTML5 argued it would increase overall interoperability while balancing the rights of content creators, providers and users. This paper argues, however, that it instead essentially legitimises U.S.-centric copyright protections on a global scale and allows the future development of the Web to be dominated by a select group of media institutions.

Keywords: HTML5, digital rights management (DRM), mobile apps, informational capitalism.

INTRODUCTION

HTML5 is the latest version of Hypertext Mark-Up Language used to build Web content that can run on multiple platforms including traditional Web browsers and mobile devices such as smartphones and tablets. As a result, it has been heralded by many in the tech industry and the in the popular press alike as the future of Web and application development and a "cornerstone" of an open Web (World Wide Web Consortium, n.d.). The high expectations that many have for HTML5 are predicated upon the belief that it will enable a new generation of multi-platform applications that return the Web to its open roots, empower independent developers, and thus challenge the dominance of closed, proprietary systems—a "mythos" which stems from idealist rhetoric about the Web's creation and HTML's historical development. What I intend to demonstrate is how this celebratory rhetoric concerning HTML5, which Andrew Schrock (2014, p. 825) calls a "rhetoric of openness", masks corporate

[*] Lecturer, Media Studies, School of English, Theatre, Film and Media Studies, Victoria University of Wellington, New Zealand. michael.daubs@vuw.ac.nz

involvement in the development of HTML5 and the commoditisation of the very code used to build the Web.

The goal of this article is to merge a political economic analysis, platform studies (Bogost & Montfort, 2009; Gillespie, 2010), and infrastructure studies (Schrock, 2014) in order to argue that contemporary conceptualisations of the Web differ wildly those of World Wide Web inventor Tim Berners-Lee, who valued and promoted concepts such as openness, interconnectivity, and accessibility. As both platform studies and infrastructure studies recognise, technologies such as HTML5 do not just involve material (e.g., computers, servers, cables) and immaterial (e.g., software, code) elements, but also the practices and social norms that accompany them (see, for example, Bowker, Baker, Millerand, & Ribes, 2009; Gitelman, 2006). Somewhat contradictorily, those involved in the development of HTML5 actively employ imagery of autonomy, empowerment and independence in a way that mytholigises the Web and ultimately naturalises and legitimises both the commodification of code and the centrality of commercial interests to the contemporary World Wide Web.

Media, Code, Power, and Politics

This mythologising of new technologies and media forms is nothing new, of course. Early cinemas and nickelodeons were celebrated as a form of relatively affordable escapism in the early 1900s for members of the working class, and a potential way for the "lower classes" to experience culture. But in the minds of cultural traditionalists of the early 20th century, cinemas were perceived as a threat to so-called high culture and, often, morality itself (Citrom, 1982, p. 45). Television was subject to the same sort of reification in the late 1940s and early 1950s. For some, television was seen as a cultural threat that disrupted traditional gender roles (Spigel, 1990, p. 87) but for others, television was seen as having the potential to fulfil a utopian promise – a way to combat juvenile delinquency, unify the family unit, and revive domestic life (Spigel, 1992, p. 45). However, the most persistent myth of television might be televisual liveness, i.e., the idea that an event is broadcast when it occurs. While this simultaneity was once a technological necessity, Jane Feuer (2003) asserts that liveness is a myth that masks the fact that "all of television is deliberately constructed, and that much of it is constructed in the service of a commercial mission" (Levine, 2008, p. 394). As such, liveness is as much ideological and political as it is technical.

The reason for this brief overview of historical media narratives and myths is to demonstrate that recent reifications of digital, networked, and mobile media is not a new phenomenon, and that these myths generally serve a political or ideological purpose, specifically one that favours corporate interests. Indeed, many of the negative and, especially, positive outcomes associated with previous media forms have been resurrected and connected to HTML5. The politics behind the framing of this Web technology connects to the concept of "app-centric media" (Daubs & Manzerolle, 2015), an outgrowth of the current media environment composed of three main elements:

- a multitude of concrete-but-connected software applications or 'apps';
- the emergence of HTML5; and,
- the commoditisation of the World Wide Web.

There is a tension at play between these elements. Since the early days of its development, HTML5 has been framed as the antithesis of Apple's iOS and

Google/Android "siloed" mobile app platforms, rhetoric that separates HTML5 from these commercial systems.

Jussi Parikka argues, however, that "We should pursue the question 'if code makes the world, then who makes code, and what sustains such operations?'" (2014, p. 35). Code is indeed an intriguing object. It is in many ways invisible to us, yet is the virtual DNA of the vast majority of systems we interact with in our day-to-day experiences. Scholars such as Steven Johnson (1997), Lev Manovich (2002, 2005) and William J. Mitchell (1995) have noted that code is not neutral but intensely political. Mitchell explicitly claims:

> [C]ontrol of code is power. For citizens of cyberspace, computer code…is the medium in which intentions are enacted and designs are realized, and it is becoming a crucial focus of political contest. Who shall write the software that increasingly structures our daily lives? What shall that software allow and proscribe? Who shall be privileged by it and who marginalized? How shall the writers of the rules be answerable? (p. 112).

The article represents an attempt to address these important questions, particularly in relation to HTML5. Luke Goode (2009, p. 1303) argues that it is often easy to "slide into a discourse of 'democracy'" when discussing digital systems rather than "interrogating what might be termed the 'politics of code.'" I hope to resist that temptation and argue instead that the emergence of app-centric media is having a marked effect on conceptualisations of the Web that differ wildly those of World Wide Web inventor Tim Berners-Lee, who valued and promoted concepts such as openness, interconnectivity, and accessibility. Prevailing rhetoric concerning the development of the app-centric media poses a direct challenge to the concept of an open Web by employing imagery of autonomy, empowerment and independence for both the users and producers of apps in a way that ultimately naturalises and legitimises both the commodification of code and the centrality of commercial interests. In order to demonstrate this development, I will discuss the controversial inclusion of digital rights management [DRM] protocols in HTML5.

In October 2014, the World Wide Web Consortium (W3C) recommended HTML5 as an official standard. That HTML5 spec included an element that would, in effect, integrate DRM, a technological enforcement of copyright protection that prevents certain uses of content or technologies, into the Web (Electronic Frontier Foundation, DRM, 2013). In the months prior to the W3C's HTML5 recommendation, the proposed inclusion of DRM led to significant debate among W3C members, the wider Web development community, and in the tech-related popular press. Critics of the W3C's decision to include a form of DRM in HTML5 argue that it represents a violation of the Web as an open platform. Defenders argue the inclusion of DRM capabilities will balance the rights of content creators, providers, and users and thus allow more individuals and institutions to engage in Web, software and mobile app, and hardware development, freed from the fear of lawsuits.

I argue that the inclusion of DRM is indicative of a reconfiguration of the Web in a manner consonant with informational capitalism (see Arvidsson & Colleoni, 2012; Castells, 2000; Fuchs, 2009), formalizing both media content and the license keys needed to distribute it as commodities. This process privileges large, multinational media institutions and limits, rather than expands, the range institutions engaged in the development of software and hardware used for the dissemination of online media content.

A Short History of the HTML and the (Mobile) Web

In order to understand why HTML5 is positioned as the cornerstone of an open Web, and how the inclusion of DRM reveals that as a rhetorical construct, it is illustrative to begin with a brief comparison of the historical development of the World Wide Web and HTML with the emergence of a mobile Internet moulded by corporate interests. The Web was developed in the late 1980s and early 1990s by Tim Berners-Lee, a software engineer from the United Kingdom. The idea of the Web stemmed from his experiments with hypertext systems in 1980 during a six month consulting position at CERN, a nuclear physics research facility in Geneva, Switzerland (Campbell-Kelly & Aspray, 2004, p. 268). He revisited the system he developed then, which he called Enquire, when he returned to CERN in 1984 and began to push for a "more expansive hypertext program" (Lambert, 2005, p. 16).

The inspiration for this new global hypertext program was to allow physicists all over the world to collaborate and share information without having to worry about interoperability between different networks and computer systems. Thus from the start, Berners-Lee was dedicated to principles such as openness and accessibility. In 1987, he began cooperatively working with his CERN colleague Robert Cailliau, who was experimenting with Apple's hypertext-based Hypercard database software, to further develop this system, demonstrating Berners-Lee's dedication to open collaboration (Lambert, 2005, p. 16). But it was Berners-Lee who envisioned the system as a "marriage of hypertext and the Internet" (Campbell-Kelly & Aspray, 2004, p. 269). Finally, in 1989, the pair made a formal proposal to CERN for what they called "the World Wide Web" (Campbell-Kelly & Aspray, 2004; Lambert, 2005).

Berners-Lee is also the original creator of HTML, which uses a series of tags to standardize the formatting of Web content. Berners-Lee adhered to a philosophy of accessibility when conceiving HTML as well, basing it on existing and familiar systems (to computer developers, at least) such as Standardized General Markup Language (SGML), which was the "international standard for coding computer languages since 1986 (Lambert, 2005, p. 17). HTML is technically a mark-up language rather than a coding language; as such, it is a protocol, i.e., "a set of rules that govern networked relations" and "aid communication on the World Wide Web" (Galloway, 2001, p. 82). Galloway (2001, p. 83) argues that protocol is "how technological control exists" in decentralises networks like the Web. Because of their ability to control and enforce, protocols, including HTML, are also political entities—take, for example, HTML tags which were predominantly based on the English language, institutionalising an English-centric bias similar to the way domain names were limited to unaccented Latin alphabet characters until 2010 (see ICANN, 2010).

HTML was meant to provide people with a relatively accessible tool with which they could build their own products (i.e., webpages) and interlink those documents—and ideas—freely. It is this functionality that leads Berners-Lee (2011) to claim that the Web developed "from the grassroots up", outside the highly-capitalized interests of the existing information communication technologies (ICT) industry. He belief in the grassroots nature of the Web is evident in comments such as this, from a 2014 article in Wired: "Anyone with an interest in the web's future – and that's everyone, everywhere – has a role in ensuring it achieves all it can" (Berners-Lee, 2014). As a point of comparison, he specifically criticizes sites such as Facebook and LinkedIn in a 2011 op-ed for Scientific American, for the way the sites are "walled off" from others on the Web, making each "a central platform—a closed silo of content, and one that does not give you full control over your information in it" (Berners-Lee, 2011).

In contrast to the early Web, the development of the mobile Internet was primarily shaped from the start by corporate interests and "market competition over standards, devices and services" (Daubs & Manzerolle, 2015, p. 4). Despite the attempts of many prominent tech companies (including Apple), the commercialization of the mobile Internet in the 1990s and early 2000s was impeded by the lack of a clear dominant system. For example, while the development of a "wireless application protocol" (WAP) standard in the 1990s was an important step, WAP was limited from the outset by the limited number of mobile devices, a still-developing pre-3G mobile infrastructure, and the "wireless markup language" which was significantly different from conventional HTML coding. As a result, there were only a limited number of developers capable of producing mobile content, which limited the possibility of translating Web content for use on mobile devices.

The result was that the mobile Internet, including the mobile Web, "was shaped primarily by telecommunications companies and disproportionately geared towards the needs of corporations and elite users" (Daubs & Manzerolle, 2015, p. 4). It was in this environment that Research In Motion, now Blackberry, emerged as a prominent mobile hardware and software company. It made its money by focusing on corporate users and was celebrated in one 2002 Merrill Lynch report for having "clearly created and dominated a new category of IT technology – i.e., mobile corporate email devices" (Astle, 2002, p. 2). As the authors of that report eloquently noted: "No longer do we daydream through boring meetings or read the sports section of the paper while we wait for our flights – instead, we are thumb pounding our BlackBerrys as we respond to and direct our business through these little devices" (Astle, 2002, p. 2). Interestingly, that report also noted that while RIM was doing well in the corporate sector, it was not as well positioned in the consumer market where its plans had been "limited" (Astle, 2002, p. 6). This was not seen as a problem, however, since it was believed that interest in handhelds was "weakening" and corporate clients provided greater ARPU (average revenue per unit/user) (Astle, 2002, p. 6).

That all changed with the introduction the introduction of the iPhone in 2007. The popular appeal of Apple's smartphone, the iTunes app and music management software, and Apple-approved mobile apps encouraged the mobile industry to embrace a mass user-base and, at the same time, solidified the "consumer" as the principle user of the mobile Internet. One of the reasons the iPhone and other smartphones to follow were popular was their ability to run apps—small, specialised software applications, each designed for a specific purpose. The emergence of the mobile Internet as a platform for the development and consumption of apps highlights the importance of standardized application programming interfaces or APIs and other user-friendly development tools. APIs are a set of functions that allow developers to incorporate elements of applications in their own app, "[l]ike the gears that can combine different parts of a machine" (Qiu, 2016, p. 2) and are offered by media corporations, often for free, to supposedly increase worker autonomy by reducing the amount of money and sophisticated technical knowledge required for production.

Eventually, as Blackberry's popularity and market share waned, along with that of other major players of the mobile market of the late 1990s and early 2000s such as Nokia, two major players dominated the mobile market: Apple (with its iOS and Apple App Store) and Google (with its Android operating system and apps available via Google Play). Despite significant differences in their development and profit models, both companies developed systems dependent upon the development and sale of native apps. A native app is one that is designed specifically for a given platform. It optimizes the integration of hardware and software, but developers must "pay for access to important development tools and services. Apple charges a base annual rate of $100.00 USD for developers to both access software development tools and to give access to iTunes as a distribution channel" (Daubs & Manzerolle, 2015, p. 7). On top of that, both Apple and Google take a cut of profits from sales or in-app advertising, usually powered by their in-house advertising arms (iAd and AdMob). The existence of these proprietary advertising systems means that there is an additional hindrance to cross-platform compatibility, and users and developers alike are locked into particular corporate ecosystems.

HTML5: Hope for an Open Web?

These "walled gardens" represented by Google's and Apple's app development and distribution systems are an anathema to proponents of an open Web, particularly those eyeing a mobile industry dominated by an Apple/Google duopoly. It was in this environment that HTML5 was developed, along with hopes of a return of the Web—mobile or otherwise—to its open roots (see, for example, Schrock, 2014, p. 823). As evidence of the increasing shift to app-centric media, much of this rhetoric is centred on the ability to develop HTML5 apps that can function with relative uniformity on any device, circumventing perceived issues with the "walled gardens" of native apps.

In stark contrast to earlier versions of HTML, however, HTML5 was initially developed not by the World Wide Web Consortium [W3C], the Berners-Lee led group that oversaw the development of previous versions of HTML, but by a group called the Web Hypertext Application Technology Working Group [WHATWG] which was dominated by technology insiders, worked privately, and operated with a private corporation—Apple—"cheering from the sidelines" (Lawson & Sharp, 2011, p. xi). Eventually The W3C would co-develop HTML5 with WHATWG but, perhaps because of the presence of ICT insiders in WHATWG, many of whom were involved in mobile development, HTML5 apps increasingly function as stand-alone apps that use APIs just like their native counterparts. These APIs "provide services or data to third-party developers, so these developers can remix and remake proprietary data owned by corporations such as Google, Facebook and Twitter into new applications and programmes" (Qiu, 2016, p. 1).

The reliance on APIs is in part due to the complexity of HTML5. The initial working draft of the HTML1 specification, which outlined its elements and proper use, was a mere 40 pages in length (Berners-Lee & Connolly, 1993); in contrast, the HTML5 specification is a "900-page gorilla" (Jordan, Lawson, & Sharp, 2010). According to Gray Norton (2014), one of the developers of an HTML5 application development framework called Enyo, this makes developing applications from scratch an "extremely tall order." APIs allow developers to rely on ready-made libraries of code, user interface [UI] controls, and other "widgets" so that the actual code is "abstracted away" (Schaaf & Norton, 2014).

Valuable APIs

These APIs tend to provide value the institutions that create and provide them—for example, in the form of valuable metadata used to track user behaviours, determine user demographics, or improve profitable targeted advertising. One example of this is Google Maps, which, using an API provided by Google, can be incorporated into others websites in order to provide tailored information such as directions, nearby hotels or restaurants, and more. The data generated by users of these maps, even if that map is included on another website such as the Guggenheim Museum website as shown here, is collected and analysed by Google, and can be linked to other information such as past Google searches, email contents, and more, particularly if that person has a Google account. But most users are unware of this data collection behind the scenes. As Gillespie (2007, p. 93) argues, "These tools are generally designed in a language that is only comprehensible by a very small community. It is much more difficult for an interested citizen, who does not have the necessary programming expertise, to raise a subtle concern about the way a certain application or protocol organizes activity." Even developers have little-to-no control over how the APIs they incorporate into their apps operate. Qiu (2016, p. 2) argues, for example, that institutions such as Google could make APIs more open, but "profit-seeking vendors restrict the openness of Open APIs to reinforce control over programmers in contemporary Internet culture."

That invasive nature has led some to develop crowdsourced, open-source alternatives, such as OpenStreetMap.org, which offers what it calls "Web Map Frameworks" to allow developers to incorporate OpenStreetMap maps into apps and websites. Because of the "indirect" nature of corporate involvement via APIs, however, the commoditization that commercial APIs enable is often overlooked, or simply overshadowed by the democratizing rhetoric often used in conjunction with the Web and HTML5, leaving alternatively like OpenStreetMap ignored.

Digital Rights Management [DRM]

The decision to include DRM in HTML5 made the increased commoditisation of the Web and HTML itself more visible, however, which is the reason it sparked such intense debate within Web development circles. The basic idea of DRM is to prohibit unauthorized uses of content (Burk & Gillespie, 2006, p. 239), but the real purpose of DRM is to allow corporate institutions to exert influence over hardware, software and users rather than directly control content itself. DRM is part of what's called a trusted system, in which encryption is "paired with a second level of control, in which the receiving device no only decodes the encrypted content, but also obeys a series of rules about the content's subsequent reproduction and redistribution of it, ensuring that such actions are made impossible for the average user" (Gillespie, 2007, p. 52). Gillespie thus argues that trusted systems are "built on a fundamental mistrust—a mistrust of technology manufacturers, who must be licensed into submission, and a mistrust of users, who are seen as immoral pirates until they can be technologically compelled to be good consumers" (Gillespie, 2007, p. 17). The emphasis on controlling both technology manufacturers and users is central to understanding the politics of HTML5.

Some of the most visible forms of DRM, for example, are region encoded DVDs and video games or protocols that prevent users from copying a Blu-ray movie to a mobile device, both of which are designed to prevent users from reselling or illegally sharing that movie. The Electronic Frontier Foundation [EFF] claims that the use of DRM tools has increased since the Digital Millennium Copyright Act of 1998 (DMCA) became law in the United States. The DMCA is, as Tarleton Gillespie (2004) argues, a dramatic strengthening of copyright law in the United States in a way that explicitly benefits (media) corporations, and represents one of the most dramatic changes to copyright law in the way it "shifts copyright law's traditional attention to use of the work (a legal abstraction that pointed to something beyond each individual manuscript or recording, to the creation itself) to regulating access to a specific artifact: this DVD, that book, this copy of the software" (pp. 240-241). The emergence of DRM technologies can therefore be directly tied to the protection of corporate media interests and profits, particularly the U.S. music, film and television industry. As Nica Elkin-Koren (2007, pp. 1127-1128) outlines: "The purpose of the DMCA was to tackle piracy and to strengthen the effectiveness of DRMs…The DMCA, which was originally enacted to confront piracy, has turned DRMs into an effective means of governing the use of copyrighted works in digital format. Consumers cannot legally hack technical measures that limit their ability to use copies they have legally purchased."

The DRM Debates

Proponents of the inclusion of DRM in HTML5 position it as necessary to protect the future of the Internet. Tech writer Peter Bright (2013) made the sweeping claim that a refusal to adopt DRM would lead to media providers "abandoning the Web." Even Berners-Lee argued that DRM in HTM5 would "increase overall interoperability" by eliminating the need for proprietary media plugins, such as Adobe's Flash Player or Microsoft's Silverlight, while balancing the rights of content creators, providers and users (qtd. in Meyer, 2013). In a widely distributed 2010 missive that (somewhat ironically) dismissed Flash as a closed system with several technical drawbacks, then Apple CEO Steve Jobs (2010) stated: "New open standards created in the mobile era, such as HTML5, will win on mobile devices (and PCs too)." This reality, however, might also explain Apple's intense involvement in the development of HTML5. Apple has even made attempts to patent elements of HTML5, an act which has drawn the ire of the W3C even as they partnered with Apple on HTML5's development (Jackson, 2011).

Those opposed to DRM argue that its inclusion in HTML5 would represent a violation of the principle of the open Web, creating "serious impediments to interoperability and access for all" (Electronic Frontier Foundation, EFF Makes Formal Objection, 2013) particularly because it will give (commercial) content providers direct control over a "user agent"—the generic term for a Web browser. EFF's International Director Danny O'Brien (2013) argued that the proposed DRM protocols would be the first step to "[a] Web where you cannot cut and paste text; where your browser can't 'Save As...' an image; where the 'allowed' uses of saved files are monitored beyond the browser…and maybe even where we can no longer effectively 'View Source' on some sites." Perhaps not surprisingly, both positions have elements of truth to them but also, at best, oversimplify or, at worst, misrepresent the way the proposed DRM system would work.

DRM, HTML5, and Informational Capitalism

Rather than limiting the interoperability often characteristic of conceptualizations of the "open" Web, HTML5's approach to DRM instead tailors the Web and Web-based content in a manner consistent with informational capitalism, a transnational system "characterised by high concentrations of ownership and highly networked multinational firms who, by controlling IPRs and networks of distribution, reduce their reliance upon labor" (Coombe, Schnoor, & Ahmed, 2006, p. 904). As described by Christian Fuchs (2009, p. 394), networks, organizations, and users in informational capitalism are dynamic and globally distributed so that capital, power, commodities, and information are "processed globally at high speed." These characteristics become evident once you examine how the DRM enacted by the W3C works.

Technically speaking, HTML5 does not contain DRM protocols. Instead, it includes an API called Encrypted Media Extensions [EME] that can be included in Web applications such as browsers or mobile apps. This API is necessary to decode encrypted audio and video content streamed online. In other words, Web application developers (and those that develop operating systems for Web-enabled hardware) must enable this form of DRM—i.e., must allow EME to operate—in order to allow their users to access encrypted streaming media. To make matters more complicated, this API does not even actually perform the decryption but is instead just a communication protocol. The full process (see Figure 1) is as follows:

1. A user attempts to access encrypted media via a Web application.
2. This initiates an EME implementation.
3. The EME interacts with something called a Content Decryption Module [CDM]—a piece of client-side software or hardware responsible for decoding encrypted content.
4. The CDM requests a "key" from a remote license server, called a key server.
5. The key server generates a license key. It is this key that actually enables the decoding of encrypted media. A key also has a lifespan, meaning the key determined how long a user can access the streaming media.
6. This key is passed back to the requesting Web application,
7. which passes it on to the CDM,
8. which then decrypts the media content using the transmitted key and
9. gives the user access to the requested media content.

Unless there is an error, this process is invisible to – and purposefully hidden from – the user.

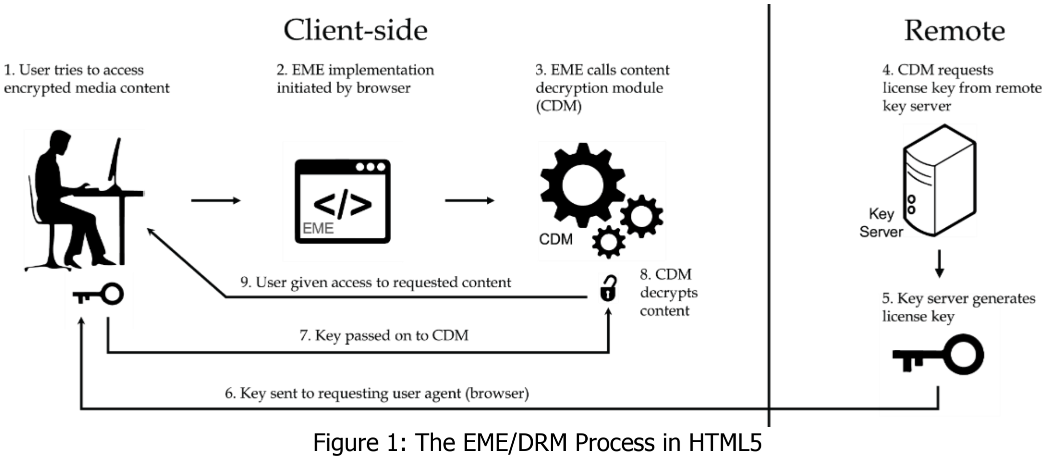
Client-side
Remote
1. User tries to access encrypted media content
2. EME implementation initiated by browser
3. EME calls content decryption module (CDM)
4. CDM requests license key from remote key server
</>
EME
CDM
Key Server
9. User given access to requested content
8. CDM decrypts content
5. Key server generates license key
7. Key passed on to CDM
6. Key sent to requesting user agent (browser)

Figure 1: The EME/DRM Process in HTML5

From Open Web to Corporate Web

DRM systems are not designed to allow corporate institutions control over content per se, but rather to allow them to exert influence over hardware, software, and users or as Boing Boing co-editor Cory Doctorow (2013) succinctly summarizes, "for the sole purpose of adding burdensome conditions" to the access tools people use. In order to be effective (i.e., a "burden"), decryption protocols needs to be protected, thus an EME-based DRM system requires the creation and enforcement of patented content licenses in order to limit end user capabilities—particularly because, most of the process takes place client-side, i.e., on the user's computer or equipment. This license system essentially acts in the same manner as the proprietary plug-in system it supposedly replaces. Qiu (2016, p. 4) even argues that APIs are "usually proprietary software and most of them are designed for marketing purposes." In fact, this system would encourage more license systems by legitimizing them as a valid element of HTML5. It is the ultimate realisation of what Lawrence Lessig (2004, p. 11) calls a "permission culture" in which "creators to create only with the permission of the powerful, or of creators from the past." That control now extends to mere access to texts rather than creation of texts.

This license requirement has several far-reaching effects. One is the creation of a new and highly valuable class of information. While some scholars (e.g., Jenkins, 2006; Lotz, 2007; Murray, 2003) argue that content is the most important commodity in a digital media environment, this complex EME/license process creates a system in which decryption keys also become a valuable commodity. The value added to content licenses in this system would potentially enable license providers, particularly those such as Netflix or Google who offer expansive and popular streaming media options, to charge a fee to CDM developers wishing to access their decoding systems. Doctorow uses DVD systems as a point of comparison, stating "[I]f a patent (or patents) can be included in the decoding system for DVDs, you can threaten manufacturers with patent-violation suits unless they take out a licence." The same idea now applies to manufacturers of Web-enabled devices that wish to offer Netflix or HBO Go as a service; they may be forced to pay for that privilege or risk developing a product with limited consumer appeal.

Such rights could be difficult for less capital-rich institutions to procure. Even modest license fees could discourage some smaller, independent software and hardware developers from creating new products, which would potentially stifle innovation and effectively limit the number of organizations involved in delivering media content. In order to allow their users to access encrypted media streams such as Netflix, "user agents" and Web-enabled hardware such as mobile devices and so-called "smart TVs" must enable EME processes. Otherwise, they risk losing their users to other browser and devices. There is only scant evidence of this happening so far, but the November 2016 announcement that Netflix would be integrated into U.S. cable giant Comcast's set top boxes (Brodkin, 2016) is perhaps an indicative development, and other recent developments suggest that independent developers are feeling some pressure.

Whether or not to enable EME extensions is arguably a greater consideration upon independent developers such as Mozilla, makers of the Firefox browser, who reply upon partnership deals to fund the free downloads of their software. The corporation was recently praised by the EFF (2014) for the "vital work they've done—and must keep doing—for the open Internet." Indeed, Mozilla was quite vocal in their opposition to the DRM/EME system proposed by the W3C. However, Mozilla capitulated in May 2014 and implemented support for EME in their popular Firefox Web browser (Bright, 2014).

Fighting DRM: The Case of Mozilla and Firefox

As it turns out, this is not unfamiliar territory for Mozilla. In 2013, the company introduced Firefox OS, a low-cost, mobile smartphone operating system. Firefox OS was one attempt by Mozilla to follow through on theW3C's commitment to extend the open standards principles central to its mission to mobile devices and platforms. Firefox OS phones were available in approximately 30 countries, and the company had announced plans to expand into Africa (Mozilla, 2014), where the spread of mobile broadband has been quickly outpacing new wireline Internet connections.

The Firefox OS project also involved the development of yet another apps store, reminiscent of Google's and Apple's app stores, called the Firefox Marketplace. However, in contrast to the closed models these two corporations represent, the Firefox Marketplace is open source; through a process called "forking", developers could develop their version of the Marketplace code and create their own app stores. In addition, and unlike the native apps designed for Apple and Android systems, the apps sold there were non-proprietary (Daubs & Manzerolle, 2015, p. 64).

Firefox OS faced several difficulties that limited its ability to become a major player in the mobile market, however, some of which stemmed from Mozilla's resistance to DRM protocols. In the words of David Meyer, despite its low cost, the Firefox OS phone would be "up against cheap Android phones that support DRM, and that means it was up against a platform that can offer users Netflix, Spotify and so on. Not being able to offer this kind of content will hinder Firefox OS's growth, and by extension the proliferation of web apps in general" (Meyer, 2013). In short, Mozilla's Firefox OS was not able to offer users what they wanted simply because they remaining committed to an open, DRM-free platform. This realisation is what perhaps contributed to Mozilla's decision to include enable DRM in new versions of its Firefox Web browser in 2014 (Doctorow, 2014). Beyond this complication, as the above discussion addresses, even a commitment to open source applications did not mean Firefox OS apps could escape the dynamics of capitalism. As Michael Daubs and Vincent Manzerolle (2015, p. 64) summarise, "[T]he dedication to open standards meant that Firefox Marketplace apps were dependent upon frameworks such as HTML5 which…incorporate, rather than subvert, the logic of capital."

A Web Aligned with Information Capitalism

Ultimately, the challenges Firefox OS faced were too great, and the company announced in late 2015 that it would stop producing Firefox OS smartphones (though the Firefox OS may live on in other Internet of Everything [IoE] applications) (Lunden, 2015). These later examples, and the DRM debate concerning HTML5, are representative of the current digital network environment. Felix Stalder (2005, p. 16) provides a succinct summary: "So, this is what is at stake: a profound struggle over the stuff digital, networked culture will be made out of. Will it be a culture of fixed objects, circulating through an infrastructure of control, where everything that is not authorized is prohibited?" HTML5 is at the centre of this complex issue. If coding is power, as Mitchell claims, then the ability to shape coding systems must represent an even greater power, and that process was led by commercial entities.

Lessig (2004, pp. 19-20) argues that "we are less and less a free culture, more and more a permission culture." Qiu (2016, p. 5) echoes this idea, stating: "code can achieve results that follow the purpose of designers." With DRM now a component of HTML5, the technical possibilities of the Web are being shaped by corporate media interests in order to benefit those media interests. Gillespie (2007, p. 127) even links DRM back to a moral argument that is related, perhaps, to the discussions of media and morality heard in relation to television and cinema, stating that DRM would "ensure that we all make the moral choice to be consumers rather than thieves." The choice to be an ethical consumer or producer is stripped away and consumption is naturalised through enforcement.

This is a significant development for both developers and users; systems for capital and control are now built into the fabric of the Web itself. The EME system outlined here in effect represents the legitimization of a patent or license system—a first for HTML. The commodification of a foundational Web protocol makes it difficult to build non-commodified open-source alternatives like OpenStreetMap.org. Thus, while many celebrate the Web's "radically open, egalitarian and decentralised platform structure" (Berners-Lee, 2014), Galloway (2001, p. 83) argues this architecture "is precisely that which makes protocological/imperial control of the network so easy. In fact, the various Internet protocols mandate that control may only be derived from such a distributed architecture." Rather than making the Web more accessible, HTML5 privileges established media giants such as Netflix, Google and Microsoft, all of whom worked on and enthusiastically promoted EME in the lead-up to its inclusion in the HTML5 spec and already dominate online streaming media.

Thanks to their efforts, there are multiple, W3C-sanctioned ways these companies can profit—on the media content itself, on access to CDMs, and by collecting profitable metadata, just to name a few—leading Doctorow (2013) to argue that DRM in HTML5 only serves the interests of "shortsighted media giants who dream of a world where your mouse rings a cash-register with every click[.]" These select "shortsighted" giants now dominate the development of the Web and the technologies used to access it, which severely restricts opportunities for developers through the implementation of increasingly valuable API keys that allow them control over external developers (Qiu, 2016, p. 13).

The EME API also allows third parties—namely corporate media entities—a modicum of control over a user's hardware and software. Furthermore, the enforcement of DRM would require significant embedded, often automated surveillance to track and analyse user behaviour (i.e., monetizable data), and modify it in ways that protect the intellectual property rights of corporations. As Raymond Yee (2008, p. 122) notes:

Through keys, the API provider knows something about the identity of an API user (typically at least the API key holder's email address if nothing else) and monitors the manner in which a user is accessing the API.... Though such tracking, the API provider might also choose to enforce the terms of use for the API—from contacting the user by email to shutting down access by that key…to suing the user in extreme cases!

Indeed, HTML5's approach to DRM essentially legitimizes U.S.-centric copyright protections on a global scale. Thus, the way in which DRM is enacted in HTML5 significantly undermines the Berners-Lee's and the W3C's claims that EME is consistent with an open Web. One might argue that HTML5 renders the web even less transparent than Free (Libre) and Open Source Software [FLOSS] since, as Stalder (2005, p. 19) argues, FLOSS is "unlikely to contain such hidden features" that "reflect overarching agendas of the companies which are unchecked, and cannot be checked, by outside developers or users. Such features are hidden for a good reason: people do not want them."

While the corporatization of the Web is nothing new, what is new here is the W3C's endorsement of this system, all while it continues to employ rhetoric of an "open Web." This rhetoric masks a "significant revision of the original concept of the 'open Web'" in which "the ideal of 'openness' no longer references the free flow of information on the Web, but rather references 'access to the market'" (Daubs & Manzerolle, 2015, p. 54). In the DRM system embedded into HTML5, that access is increasingly restricted to a few privileged institutions who will have increasing influence over the future of an informational capitalism dominated Web.

REFERENCES

Arvidsson, A., & Colleoni, E. (2012). Value in Informational Capitalism and on the Internet. The Information Society, 28(3), 135-150.

Astle, T. B. (2002). Research in Motion Ltd. Rolling out the RIM (pp. 1-18). Toronto: Merrill Lynch.

Berners-Lee, T. (2011, 04 May). Long Live the Web: A Call for Continued Open Standards and Neutrality. Retrieved 08 July, 2013, from http://www.scientificamerican.com/article.cfm?id=long-live-the-web

Berners-Lee, T. (2014, 23 August). Tim Berners-Lee on the Web at 25: the past, present and future. Retrieved 19 December 2016, from http://www.wired.co.uk/article/tim-berners-lee

Berners-Lee, T., & Connolly, D. (1993). Hypertext Markup Language (HTML): A Representation of Textual Information and MetaInformation for Retrieval and Interchange. Fremont, Calfornia: Internet Engineering Task Force. Retrieved from http://tools.ietf.org/pdf/draft-ietf-iiir-html-00.pdf.

Bogost, I., & Montfort, N. (2009). Platform Studies: Frequently Questioned Answers. Paper presented at the Digital Arts and Culture Conference 2009: After Media, Embodiment and Context, Irvine, California.

Bowker, G. C., Baker, K., Millerand, F., & Ribes, D. (2009). Toward information infrastructure studies: Ways of knowing in a networked environment International handbook of internet research (pp. 97-117): Springer.

Bright, P. (2013, 11 May). DRM in HTML5 is a victory for the open Web, not a defeat. Ministry of Information. Retrieved 14 June, 2015, from http://arstechnica.com/business/2013/05/drm-in-html5-is-a-victory-for-the-open-web-not-a-defeat/

Bright, P. (2014, 15 May). Driven by necessity, Mozilla to enable HTML5 DRM in Firefox. Technology Lab. Retrieved 14 June, 2015, from http://arstechnica.com/information-technology/2014/05/driven-by-necessity-mozilla-to-enable-html5-drm-in-firefox/

Brodkin, J. (2016, 07 November). Netflix arrives on Comcast TV boxes, won’t be exempt from data cap. Retrieved 08 November, 2016, from http://arstechnica.com/information-technology/2016/11/netflix-arrives-on-comcast-tv-boxes-wont-be-exempt-from-data-cap/

Burk, D. L., & Gillespie, T. L. (2006). Autonomy and morality in DRM and anti-circumvention law. Triple C: Cognition, Communication, Cooperation, 4(2).

Campbell-Kelly, M., & Aspray, W. (2004). Computer: A History Of The Information Machine (2nd ed.). Boulder, Colorado: Westview Press.

Castells, M. (2000). The Rise of the Network Society. Cambridge, MA: Blackwell.

Citrom, D. J. (1982). Media and the American Mind from Morse to McLuhan. Chapel Hill, North Carolina: The University of North Carolina Press.

Coombe, R. J., Schnoor, S., & Ahmed, M. (2006). Bearing Cultural Distinction: Informational Capitalism and New Expectations for Intellectual Property. UC Davis L. Rev., 40, 891.

Daubs, M. S., & Manzerolle, V. (2015). App-centric Mobile Media and Commoditization: Implications for the Future of the Open Web. Mobile Media and Communication, 4(1), 52-68.

Doctorow, C. (2013, 12 March). What I wish Tim Berners-Lee understood about DRM. Technology Blog. Retrieved 28 August, 2013, from http://www.theguardian.com/technology/blog/2013/mar/12/tim-berners-lee-drm-cory-doctorow?CMP=twt_fd

Doctorow, C. (2014, 14 May). Firefox's adoption of closed-source DRM breaks my heart. Technology. Retrieved 17 May, 2014, from http://www.theguardian.com/technology/2014/may/14/firefox-closed-source-drm-video-browser-cory-doctorow?CMP=twt_gu

Electronic Frontier Foundation. (2013). DRM. Retrieved 13 October, 2013, from https://www.eff.org/issues/drm

Electronic Frontier Foundation. (2013, 29 May). EFF Makes Formal Objection to DRM in HTML5. Retrieved 28 August, 2013, from https://www.eff.org/press/releases/eff-makes-formal-objection-drm-html5

Elkin-Koren, N. (2007). Making Room for Consumers under the DMCA. Berkeley Technology Law Journal, 22(3), 1119-1155.

Feuer, J. (2003). The Concept of Live Television: Ontology as Ideology. In E. A. Kaplan (Ed.), Regarding Television: Critical Approaches: An Anthology (pp. 12-21). Frederick, Maryland: University Publications of America.

Fuchs, C. (2009). A contribution to the critique of the political economy of transnational informational capitalism. Rethinking Marxism, 21(3), 387-402.

Galloway, A. (2001). Protocol, or, how control exists after decentralization. Rethinking Marxism, 13(3-4), 81-88.

Gillespie, T. (2004). Copyright and commerce: The DMCA, trusted systems, and the stabilization of distribution. The Information Society, 20(4), 239-254.

Gillespie, T. (2007). Wired Shut. Cambridge, Massachusetts: MIT Press.

Gillespie, T. (2010). The politics of ‘platforms’. New Media & Society, 12(3), 347-364.

Gitelman, L. (2006). Always Already New. Cambridge, Massachusetts: The MIT Press.

Goode, L. (2009). Social news, citizen journalism and democracy. New Media & Society, 11(8), 1287-1305.

ICANN. (2010, 05 May). First IDN ccTLDs Available. Retrieved 19 December, 2016, from https://www.icann.org/news/announcement-2010-05-05-en

Jackson, J. (2011, 11 July). W3C Chastises Apple on HTML5 Patenting. Retrieved 10 September 2013, from http://www.pcworld.com/article/235395/article.html

Jenkins, H. (2006). Convergence Culture: Where Old and New Media Collide. New York: New York University Press.

Jobs, S. (2010, April). Thoughs on Flash. Retrieved 3 September, 2013, from http://www.apple.com/hotnews/thoughts-on-flash/

Johnson, S. (1997). Interface culture: how new technology transforms the way we create and communicate (Vol. 1st). San Francisco: HarperEdge.

Jordan, M., Lawson, B., & Sharp, R. (2010, 27 September). HTML5: The 900-Page Gorilla with a Wide Ensemble. Retrieved 2 October, 2013, from http://www.peachpit.com/articles/article.aspx?p=1629150

Lambert, L. (2005). The Internet: A Historical Encyclopedia (H. W. Poole Ed.). Santa Barbara, California: MTM Publishing.

Lawson, B., & Sharp, R. (2011). Introducing HTML5. Berkeley, California: New Riders.

Lessig, L. (2004). Free Culture: How Big media Uses Technology and the Law to Lock Down Culture and Control Creativity. New York: Penguin Press.

Levine, E. (2008). Distinguishing television: the changing meanings of television liveness. Media, Culture & Society, 30(3), 393-409.

Lotz, A. D. (2007). The Television Will Be Revolutionized. New York: New York University Press.

Lunden, I. (2015, 08 December). Mozilla Will Stop Developing And Selling Firefox OS Smartphones. Retrieved 09 December, 2015, from https://techcrunch.com/2015/12/08/mozilla-will-stop-developing-and-selling-firefox-os-smartphones/

Manovich, L. (2002). The Language of New Media. Cambridge, MA: The MIT Press.

Manovich, L. (2005). Generation Flash. In G. M. Buurman (Ed.), Total Interaction: Theory and Practice of a New Paradigm for the Design Disciplines (pp. 66-77). Boston: Birkhäuser.

Meyer, D. (2013, 9 October). The DRM dilemma facing the open web. Retrieved 10 October 2013, from http://gigaom.com/2013/10/09/striking-a-balance-or-striking-out-the-drm-dilemma-that-faces-the-open-web/

Mitchell, W. J. (1995). City of bits: space, place, and the infobahn. Cambridge, Mass: MIT Press.

Murray, S. (2003). Media Convergence's Third Wave Content Streaming. Convergence: The International Journal of Research into New Media Technologies, 9(1), 8-18.

O'Brien, D. (2013, 2 October). Lowering Your Standards: DRM and the Future of the W3C. Retrieved 14 Jine, 2015, from https://www.eff.org/deeplinks/2013/10/lowering-your-standards

Parikka, J. (2014). Cultural techniques of cognitive capitalism: Metaprogramming and the labour of code. Cultural Studies Review, 20(1), 30-52.

Qiu, Y. (2016). The openness of Open Application Programming Interfaces. Information, Communication & Society, 1-17.

Schaaf, K., & Norton, G. (2014). Native-quality HTML5 apps - from phones to tablets to TVs Paper presented at the Apps World North America, San Francisco.

Schrock, A. R. (2014). HTML5 and openness in mobile platforms. Continuum, 28(6), 820-834.

Spigel, L. (1990). Television in the Family Circle: The Popular Reception of a New Medium. In P. Mellencamp (Ed.), Logics of Television: Essays in Cultural Criticism (pp. 73-97). Bloomington, Indiana: Indiana University Press.

Spigel, L. (1992). Make room for TV: Television and the Family Ideal in Postwar America. Chicago: University of Chicago Press.

Stalder, F. (2005). Open Cultures and the Nature of Networks. Frankfurt am Main: Revolver - Archiv für aktuelle Kunst.

World Wide Web Consortium. (n.d.). Standards. Retrieved 23 August 2013, from http://www.w3.org/standards/

Yee, R. (2008). Learning Web Services APIs Through Flickr. Pro Web 2.0 Mashups: Remixing Data and Web Services, 121-170.