

I Am:

Remix Your Web Identity

I Am:

Remix Your Web Identity

By

Paolo Sordi

Translation by Catherine Bolton

Cambridge
Scholars
Publishing



I Am: Remix Your Web Identity

By Paolo Sordi

This book first published 2015

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

Copyright © 2015 by Paolo Sordi

All rights for this book reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

ISBN (10): 1-4438-7156-7

ISBN (13): 978-1-4438-7156-3

TABLE OF CONTENTS

List of Illustrations	vii
Introduction	1
Chapter One..... The Digital Identity	3
Chapter Two..... My Site: From Geocities to WordPress	17
Chapter Three..... WordPress as an Identity Management System	33
Chapter Four..... Develop Your Identity Theme	55
Chapter Five..... Style Your Web Identity	105
Bibliography.....	117
Notes.....	121
Index.....	131

LIST OF ILLUSTRATIONS

Fig. 1-1: “Life on the Web,” in the browser by Jack Hudson.....	9
Fig. 3-1: Hello World! Welcome to WordPress	41
Fig. 3-2: WordPress plugins in Administration Panels > Plugins.....	43
Fig. 3-3: Syndicated sites settings in Administration Panels > FeedWordPress	44
Fig. 3-4: Feed and update settings in Administration Panels > FeedWordPress	45
Fig. 3-5: Syndicated Posts & Links settings in Administration Panels > FeedWordPress	46
Fig. 3-6: Categories & Tags in Administration Panels > FeedWord Press.....	47
Fig. 3-7: Flickr RSS feed for user.....	47
Fig. 3-8: Goodreads RSS feed for user.....	49
Fig. 3-9: Custom fields for Syndicated Posts & Links settings in Administration Panels > FeedWordPress	51
Fig. 3-10: Last.fm RSS feed for loved tracks by user.....	52
Fig. 3-11: “I Am” website with WordPress default theme	54
Fig. 4-1: “I Am” theme folder, with Foundation	64
Fig. 4-2: “I Am” homepage with ‘I Am’ theme, without full customization	103
Fig. 5-1: “I Am” single page with ‘I Am’ theme, without full customization	105
Fig. 5-2: “I Am” header, with full customization	108
Fig. 5-3: “I Am” single page, with full customization.....	110
Fig. 5-4: “I Am” archives, with full customization	111
Fig. 5-5: “I Am” homepage, before the last touches.....	112
Fig. 5-6: The final “I Am” homepage: a manifesto	115

INTRODUCTION

When Greg Knauss's wife came across Art Braaten's business card in an old envelope tucked away in a long-forgotten drawer, she stumbled across not only a name but a series of memories sparked by this rediscovery. The year 1997. The trip to Canada, just after the couple's first wedding anniversary. And during the trip: the speeding ticket, the biting cold, the disappointment over not getting to glimpse a single whale. But above all, the memory of that vacation was illuminated once more by a walk through the parks and park benches of Victoria, and by the vision and ensuing purchase of a charming birdcage, chosen among at least a dozen displayed on the wooden fence of a small house. Twelve years later, that cage no longer existed, hopelessly ruined by the climate of Southern California, and the name of the man who had built and sold it to the Knausses resurfaced from the past on a little card that had traveled in the box with the cage.

Today Greg Knauss is a Web professional, a developer of websites and applications, but also a writer. To do what he decided to do after his wife found the souvenir of that trip twelve years earlier, he didn't need to be a web designer or know how to write a book. In 2009, Knauss did what many of us would have done. Inspired by the emotions of a happy memory, he went to his computer, opened Google, and typed in "Art Braaten" to discover that the man with whom he and his wife had not merely done business but had enjoyed a pleasant part of a memorable vacation had died two years after their encounter. Braaten had talked to them about his colon cancer and about the forty years he had spent with his wife, but once Greg returned to his house in California, to a life over twelve hundred miles from Victoria, Canada, he would never learn anything further. The web page of an obituary that resurfaced—like the business card—from an archive constantly waiting to be consulted now told him about the life and death of that man, his love for his wife Grace, whom he had married in 1957, their two children, his passion for fishing, trips to Las Vegas, retirement and birdcages, his illness, his treatment, and his final days with his loved ones, ever a "gentle man with a good sense of humour."

Quoting Chris Anderson,¹ we could talk about the "Long Tail" of memory. After all, Amazon and, generally, all other successful digital

marketing models, from iTunes to Netflix and on to Rhapsody, generate a significant part of their profits from secondary titles, books, songs, and overlooked or forgotten films that no brick-and-mortar store could afford to keep on its shelves, as they would physically occupy room that can instead be devoted to the best-sellers accounting for most of their sales. When the store made of atoms suddenly ends its sales curves with the last greatest hits, the shop composed of bits keeps it going, with numbers that are obviously far lower but constant over time. In fact, more than half of the book sales on Amazon are not best-sellers, which at a brick-and-mortar business like Barnes & Noble represent a market of 130,000 titles. Regarding the “Long Tail” of sales, Anderson comments:

You can find everything out there on the Long Tail. There’s the back catalog, older albums still fondly remembered by longtime fans or rediscovered by new ones. There are live tracks, B-sides, remixes, even (gasp) covers. There are niches by the thousands, genre within genre within genre: Imagine an entire Tower Records devoted to ’80s hair bands or ambient dub. There are foreign bands, once priced out of reach in the Import aisle, and obscure bands on even more obscure labels, many of which don’t have the distribution clout to get into Tower at all.²

In exactly the same way, for the past seventeen years the Internet has been enriched with websites, blogs, pages, notes, documents, and articles that generate the traffic of a few navigators with little influence (from a commercial standpoint), but that together come to be deposited in a store of memories left on standby—asleep—until a serendipitous search that starts with a dusty scrap of paper sets it in motion, assigning it a meaning in the recomposition of a memory and the reconstruction of a personal story interwoven with other personal stories.

CHAPTER ONE

THE DIGITAL IDENTITY

1.1 From the Web to the Digital Life

If we read about its history, we learn that the birth of the World Wide Web is closely tied to the problem of memory and loss. In the microcosm of CERN, Tim Berners-Lee imagined a knowledge-management system that could survive time and generational changeover, which in the professional life of the Geneva-based organization meant a staff turnover every two years: a life expectancy so short that the wealth of studies and projects developed over time became impossible for later researchers to recover, forcing them to embark on veritable “detective investigations” to try to trace (unsuccessfully in some cases) documentation fundamental for their work.

Berners-Lee’s theoretical solution to the “company” problem—a solution then put into practice with the URI, HTTP, and HTML protocols—came from hypertext, a word and a concept introduced in 1965 by Ted Nelson, in turn inspired by an article published by Vannevar Bush. While both were dealing to some extent with the problem of information retrieval, they did not have any corporate responsibility on a pragmatic level, yet they reflected on the mind and human memory. Bush’s article is titled “As We May Think” and it was starting with how one thinks that the American scientist imagined a device called Memex “in which an individual stores all his books, records, and communications.”³ In other words, it was “an enlarged intimate supplement to his memory” that could keep up with the prodigious pace of the sum of an individual’s experiences and research. While anchored to a world that was still analogical, the description of the way Memex would work shows close parallels with today’s navigation experiences on the Web through browser panels, perhaps integrated with a Diigo bookmarklet:⁴

A special button transfers him immediately to the first page of the index. . . . As he has several projection positions, he can leave one item in

position while he calls up another. He can add marginal notes and comments.⁵

While legend would have it that Ted Nelson turned to the prefiguration of Xanadu to solve his memory deficits, the work in which the American philosopher tried to imagine the structuring of complexity, mutability, and indeterminateness instead anticipated a change in cultural perspective: thinking of the computer not as a tool for corporate operations or scientific calculations, but as a friend permitting a new medium for education and entertainment that could be customized according to the reader's tastes.⁶

John Sculley, CEO of Apple Computer from 1983 to 1993, recalled that Apple founder Steve Jobs imagined the computer as a "bicycle for the mind"⁷ and it was probably with the iMac, introduced in 1998, that the computer finally became "personal" and shed its bureaucratic image, which the prevailing metaphor of the desk nevertheless continues to evoke. But the all-in-one translucent structure of the iMac and its psychedelic colors assigned a new and highly symbolic position to the very hardware of the PC, one that incorporated the computer into everyday life to make it a central object of domestic furnishings, and no longer a necessary and unfathomable work encumbrance.

Stigmatizing the deceptive charge of transparency, Marcel O'Gorman wondered if the transformation of the computer into a "fashionable friend"—to use Nelsonian terms—was leading users to hand over their identity unconditionally to the logic of the masters of technology:

If this trend continues unchecked, human identity will one day be determined by hardware and software aesthetics, and information will be controlled by the corporate fashion machines through which it is filtered. The computer fashion scene is the site of disempowerment, programmed ignorance, and packaged identity formation.⁸

This is the type of criticism that, in other words and in a new context, Jerome Lanier leveled against the Web 2.0 culture. Nevertheless, the "i" prefix in the name iMac already acknowledged the fundamental nature of the Internet,⁹ in the name of which the Apple computer sacrificed traditional physical memory supports such as the floppy disk, portending its obsolescence. The Web had started to be a global hypertext, "a body of written or pictorial material interconnected in such a complex way that it could not conveniently be presented or represented on paper".¹⁰ From the introduction of the iMac on, the immanence of the Web was interwoven into all of Apple's nomenclature for hardware, from the iPod to the iPhone and the iPad, but also software, from iTunes to iWork and iPhoto. From

the latter viewpoint, iTunes, iPhoto, and iMovie—the latter two currently combined in a package naturally dubbed iLife—have represented the digital transposition of an individual increasingly dependent on a machine to store not only his professional life, but above all his personal one. This distancing from the bureaucratic dimension of what was defined as a “processor” is clear: as Jobs announced, “iLife does for our digital lifestyle what Microsoft Office did for office productivity.”¹¹

Songs, photos, and videos are accumulated in a mountain of bits of memories and emotions that sketch out an identity expressed inside the computer, in an unbreakable bond between hardware and software that models the representation of our memory:

The hardware and software that we use have the power to shape our relationship to information. And if this relationship is controlled by corporate interests, then we must consider the ramifications. In a worst case scenario, “one writes—the ‘under’ says it already—as a subject or underling of the Microsoft [or Apple] Corporation.”¹²

According to O’Gorman, it is our very identity and freedom of expression that are modeled according to the paradigms of a device that only secondarily obeys expressive needs, as it must primarily consider financial interests on a multinational scale. From this standpoint, Apple’s strategy, which started with iTunes and the iPod and was followed by the iPhone, the iPad, and the App Store, is emblematic of the control the Cupertino-based company wields over the entire production chain of contents through its devices and software. The Internet is exploited as a vector of data, the acquisition, distribution, and publication of which are filtered by none other than Apple, which does not limit itself to offering venues, but reserves the right to decide who can enter and who cannot, because of the indissolubility of the relationship between hardware (manufactured by Apple) and software (produced by Apple, but also other companies). The explosion of so-called apps—applications for the mobile devices of the latest generation that exploit the Internet (note: the Internet, not the Web)—is creating a closed network that runs the risk of killing the Web, the application responsible for the success of the Internet, thanks to its universal accessibility, interoperability, and, last but not least, the fact that it is free. Instead, in many cases there is a fee for apps (in effect, a micropayment, modeled after the cost of a track from the iTunes Store) and the discovery of an Eldorado on the Internet but outside the Web seems to have convinced Chris Anderson to join the bandwagon of the new market philosophy applied to the Internet, declaring the death of the World Wide Web:

Openness is a wonderful thing in the non-monetary economy of peer production.

But eventually our tolerance for the delirious chaos of infinite competition finds its limits. Much as we love freedom and choice, we also love things that just work, reliably and seamlessly. . . .

It is only the future of the commercial content side of the digital economy. E-commerce continues to thrive on the web, and no company is going to shut its website as an information resource. More important, the great virtue of today's web is that so much of it is noncommercial. The wide-open web of peer production, the so-called generative web where everyone is free to create what they want, continues to thrive, driven by the non-monetary incentives of expression, attention, reputation, and the like. But the notion of the web as the ultimate marketplace for digital delivery is now in doubt.¹³

In this capitalistically driven scenario, Apple is the true media maker of the twenty-first century, dangerously resembling those of the previous century. Today the company founded by Jobs has built its recent fortunes on the basis of contents and traditional media such as music (iTunes) and cinema (Pixar), and it controls the look and feel of the experience through mediation of the device. And this will be the case until the computer finds a way to share its memory with other memories: in other words, as long as the “wide-open Web” doesn't enter into the personal realm.

1.2 From the Digital Life to Web 2.0

The memory of the Microsoft researcher Gordon Bell takes up 350 gigabytes, if we exclude what Bell listens to and watches. For years, Bell has been digitalizing his conversations, where he has been, his travels, his experiences, his prescription medication, his restaurant bills: in a word, his life. Inspired by Vannevar Bush's Memex, Bell translated—and is translating—into bits every single detailed aspect of his life, in a research project and software (actually a suite that unites numerous specialized applications), which he naturally named “MyLifeBits.” Bell's obsession, like Bush's, is the construction of “an enlarged intimate supplement” to an individual's memory, and his systematic activity of “lifelogging” is aimed at strictly personal use, but the Web has expanded the gray area between what can be considered private and what can be made public, because every datum on the Web is ontologically shared and potentially publishable: every application of the newest generation, including those that Bell mentions as the direct commercial developments of his projects (such as Evernote),¹⁴ now envisions a sharing function for content produced on other online publication and sharing platforms.

Commenting on one of Michel Foucault's texts in which the French philosopher stated that it was impossible to define a work in the face of the millions of traces a person left after his death, Domenico Fiormonte emphasized that the issue is no longer about defining "the work" but measuring its extension:

the individual's written presentation-codification goes well beyond his material self. It crosses over the body and inserts itself into the eternal flow of digital data that precedes, surrounds, and envelops us. Therefore, [the questions is] no longer (not only) what is the work, but what is the individual?¹⁵

If we attempt to answer Fiormonte's question, at least in part, and decide to accept as true Lev Manovich's assertion that we "live in a software culture—that is, a culture where the production, distribution, and reception of most contents and, increasingly, experiences is mediated by software"¹⁶ then we can say that today the "definition of the individual" increasingly goes through the mediation of a browser, the program we use to navigate Web 2.0.

Although there are few technical elements about it, the label of Web 2.0 coined by Tim O'Reilly truly marks a watershed in the history of the conception of applications, both before and after the pervasive explosion of the World Wide Web, a watershed lucidly described by O'Reilly himself in his comparison between Netscape and Google.¹⁷ In the mid-Nineties, Netscape had already perceived the potential of the Web as a platform, and it was in that realm that it challenged Microsoft's monopoly, but on the same ground: the desktop. The difference was that Netscape's desktop, with Netscape's technologies, languages, and software, would have revolved around Navigator, the most widely used browser at the time. Instead, Google set up its empire based on a service rather than a package of applications, and when it started to develop or purchase applications (Gmail, Google Calendar, Blogger, GoogleDocs, etc.), those apps did not rotate *around* the browser but were *inside it*. As O'Reilly commented, "None of the trappings of the old software industry are present. No scheduled software releases, just continuous improvement." These improvements are not released through a plethora of updates to be downloaded and installed, but are used instantly upon opening the program, which with Web 2.0 is the engine of the entire software, the "rendering engine" for a user's life online.

The activities mediated by the computer are driven by the closed surface of our desks—real or virtual—and they have started to converge onto the Web. Sharing, once entrusted to magnetic or optical storage

media, the hardware prosthesis of our old personal computer, is ontologically tied to the cloud on the Web. This is because using the Web means using a browser, and using a browser ultimately means using HyperText Markup Language (HTML), effectively a standard for the production and distribution of contents, and now also of applications. Another historic change that is taking place behind the browser window is the transition to a new version of HTML, namely HTML5, promoted first and foremost by Apple, Mozilla, and Opera in order to exploit HTML from an application standpoint. Based on a universally interoperable language, HTML5 confirms the status of the World Wide Web as a platform for the development of applications that can operate directly in the latest browsers released by Apple, Mozilla, Opera, Google, and Microsoft. They are not designed exclusively for personal computers, as most apps installed on smartphones and tablets are merely web data and content browsers that reproduce by exploiting the web pages of Safari and Chrome. To offer a glaring example, Google released versions of Gmail, Calendar, and Docs based on HTML5 to give users the chance to utilize them offline as well,¹⁸ in an intriguing but telling reversal of the times when an app's killer feature could instead have been that of working online.

It is no accident that the only desktop application Google felt it urgent to develop was a browser, Chrome, and that this browser is the foundation of its operating system, Chrome OS. In fact, in the transition to the Web platform, the operating system is a connection to the Internet and life unfolds inside the software that browses HTML documents. For the illustration celebrating Chrome's second birthday, the artist Jack Hudson depicted a man who, after accessing the Web, communicates with other users through Skype, organizes his vacations, reads the paper on a public park bench, listens to music as he checks recipes for supper, plans trips based on weather reports, and gets information about an e-learning site (fig. 1-1). This is what we do every day, without ever leaving the window of our favorite browser.

never been a better moment than the present to decide to become a content producer. He listed the reasons one by one, noting that now we have:

1. A worldwide network of computers that allow almost anyone to access almost anything immediately, complete with video, sound, and images.
2. Ad networks that let us earn income with the insertion of a single line of code.
3. Social networking tools (Facebook, Twitter, Flickr) where you can grow a following that easily outnumbers the readership of a small newspaper.
4. Print on demand tools like MagCloud that allow you to make printed media at no cost to the publisher.
5. Online buying and selling tools like PayPal that allow you to charge for things at minimal cost, and an online audience comfortable with buying online.
6. A new generation of readers and writers. There have never been more literate people on this planet than there are right now.¹⁹

In other words, the platform switch also corresponds to a handover of control of the means of production and distribution of contents. Amateurs and semi-professionals have entered into a game where the Web and its applications change the rules, incorporate production and entertainment spaces typically offline, and make them reading/writing tools ready to be reused—mashed up—in new context, shareable by nature and publishable by extension: through an application such as Last.fm,²⁰ a solipsistic experience such as listening to a playlist on your iPod can become the interactive content of a personal website. Or, to go back to Powazek's points, photos that had an audience limited to one's circle of family and friends when printed from a roll of film have not only gained a boundless public through digitalization and Flickr,²¹ but have also become contributions publishable on any other web project—perhaps after being touched up on the online photo editor.

Everything is just like Hudson's drawing, without ever closing the windows in your browser. Just a decade ago an activity like photoshopping would have meant downloading a program (or using a CD) that was not only very heavy but also inordinately expensive, selected based on the user's operating system, with preliminary considerations and evaluations regarding the functions that a specific version did or did not have. Furthermore, the availability of the application and the documents that were produced would be tied to the computer on which that application had been installed and where those documents were memorized: sharing had to go through floppy disks, CDs, or ad hoc uploads.

Working on the Web not only ensures the mobility and universal accessibility of data and contents—catered to by hardware for writing and reading media (laptops, smartphones, tablets, cameras, video cameras) that have become cheaper and more portable—but it also modularizes them in a number of independent blocks that, like Legos, can be used to make countless constructions. But the crucial difference here is that, as opposed to a child using Legos, the web author can use the same little bricks simultaneously for different constructions.

In his 2008 examination of the history of the Dewey Decimal System, David Weinberger emphasized that the spatialization of knowledge in the twentieth century, which still impacts more than 200,000 libraries around the world, was closely tied to the materiality of the book and the library, where a book can be located only on that specific shelf of that specific bookcase. Here we find the physical correspondence to a taxonomic position whose modifications, inexorably dictated by the spirit of the era and the disciplines that the times erase or bring forward, produce real movements in the real world of the buildings where books are kept and classified:

Tens of thousands of librarians around the world pick up their razor blades and scrape the white numbers off the spines of millions of books, muttering under their breath about those damn editors who don't understand that every little change means that librarians inhale toxic white dust. Entire card catalogs get discarded, so to speak, and millions of new cards printed up. Books are piled up, moved from this shelf to that.²²

To return to our Lego analogy, each brick that belongs to the “red” category can only be found in the room of red bricks if someone wants to find it. But if it were to become an anonymous sequence of “0” and “1” stored in a computer memory, the brick would be dematerialized and lose its physical nature, gaining potentially infinite positions on the digital shelf, based on all taxonomic associations and not what that object can build. A book transformed into an ePub file, a record into an MP3, or a newspaper into a PDF do not need a position but access routes that can be reproduced and multiplied, just as digital documents can effortlessly be copied and pasted. If in the physical world access to stored memory has a mandatory path, in the computer world access is random—literally—and the explosion of the hypertext spider web has revealed the obsolescence of the metaphor of the desk and folders, since it makes no sense to tie sequences of bits to a space that, in turn, can continually be reconstructed based on different modes with each log. For example, Gmail has entirely abandoned the use of folders to store email messages, instead relying on

tags that can transversally mark communications, without requiring them to reside in a specific and exclusive place. The same idea is behind Flickr, where the user does not assign his or her photographs to categories determined by the application, but can label them freely with key words that, along with other meta-information such as the location or date of the picture, configure horizontal explorations without hierarchies, where the user is an author at least two times over: first of all, as the individual creator of the specific content; secondly, as the collective designer of a series of navigation routes that the software platform builds and updates dynamically, aggregating the explicit and implicit metadata of the uploaded content.²³

1.4 Identity Remix

The aggregation of content belonging to different positions and collections works thanks to another crucial factor of the Web of the newest generation: the definitive separation of the content unit from the primary production context. As Walter Benjamin noted, while technological reproducibility and the serialization of printing deprived the work of art of the aura tied to the uniqueness of the performance, at the same time this maintained a powerful bond—also economic—with the material sources of publication and reproduction, which preserved the indissoluble unity of the product's form and substance.

In the first part of the history of the Web, at least until the late Nineties, the legacy and experience of the typesetting industry, as well as the immaturity of this budding technology, had such an enormous influence on the technique of building Internet pages that this re-created an unbreakable bond between publication context (the site) and content, beckoning designers and authors to incorporate the representation of the form of the document into HTML, needed to label the text structure.

If we read the code of the HTML pages from that period, resembling the tags that Berners-Lee had synthetically deduced from Standard Generalized Markup Language (SGML) and that were needed to describe titles, paragraphs, lists, and connections to a page, we find tags that, as opposed to what is indicated on his birth certificate,²⁴ gave HTML control over font, point size, colors, background, and everything that could be used to sketch out the page appearance and thus its appurtenance to a specific representation context:

```

<table border=0 width=440 CELLPADDING="20">
<tr>
<td width=60><IMG SRC="http://zeldman.com/dot_clear.gif" WIDTH=1
HSPACE=30 HEIGHT=1 ALT=""></td>
<td width=340 BGCOLOR="#0F99C0">
<BR>
<HR SIZE=7>
<BR><IMG vspace=1 SRC="/http://zeldman.com/dot_clear.gif" ALT=" "
width=1 height=1><BR>
<h1><A HREF="http://zeldman.com/toc.html"
onMouseOver="window.status='Core page. More choices.'; return true"
target="_top"></A><FONT
FACE="GEORGIA, TIMES, TIMES NEW ROMAN" size=6>How I
<BR>Lost My <BR>Sponsors<BR></font></h1>
<BR CLEAR=ALL>
<IMG alt="dot_clear" vspace="1" SRC="/http://zeldman.com/dot_clear.gif">
<HR SIZE=7>
<BR><IMG vspace=2 SRC="/http://zeldman.com/dot_clear.gif" ALT=" "
width=1 height=1><BR>
<FONT FACE="GEORGIA, TIMES, TIMES NEW ROMAN"
SIZE="5"><B>Disclaimer</B></FONT>
<BR><IMG vspace=1 SRC="/http://zeldman.com/dot_clear.gif" ALT=" "
width=1 height=1><BR>
<FONT FACE="TIMES, TIMES NEW ROMAN" SIZE="4">Within two
weeks of writing and posting <A HREF="/http://zeldman.com/sponsor1a.html"
target="_top"><B>How I Lost My Sponsors,</B></a> the ad banners
returned, thanks to some unsolicited <A
HREF="http://zeldman.com/sponsor2.html"><B>letters of support</B></a>
and the <A
HREF="http://web.archive.org/web/19970330114449/http://zeldman.com/spon
sor1c.html"><B>basic decency</B></a> of the organization which places
advertising on my site. <BR><IMG vspace=8 hspace=8 ALT=" " width=1
height=1 SRC="/http://zeldman.com/dot_clear.gif">
...
For this reason I still publish and invite you to read <A
HREF="http://zeldman.com/sponsor1a.html" target="_top"><B>How I Lost
My Sponsors.</B></a>
</FONT><BR>
<IMG alt="dot_clear" vspace="4" SRC="http://zeldman.com/dot_clear.gif">
<BR></TD></TR>
</table>

```

Tables, separators, background, transparent images to simulate margins, the selection of font and point size: the application of these labels *inside* the content weakened the digital immateriality of the web pages. In

fact, it could only be reproduced by “manually” cleaning the code and then recontextualizing it in a new form, as if it were the re-edition of a book. The portion of the code shown here with those colors, that page layout, and those fonts would make sense as a reproduction only within the space for which those colors, that layout, and those fonts were designed, unless one has both a visual and content patchwork in mind.

When thanks to the Cascading Style Sheet (CSS) the formal aspects of a web page instead emerged from its structure, content began to be truly independent of the operating platform, the support, and the site itself and its layout:²⁵

```
<article>
<h1>How I Lost My Sponsors</h1>
<strong>Disclaimer</strong>
<p>Within two weeks of writing and posting <A
HREF="/http://zeldman.com/sponsor1a.html" target="_top"><B>How I Lost
My Sponsors,</B></a> the ad banners returned, thanks to some unsolicited <A
HREF="http://zeldman.com/sponsor2.html"><B>letters of support</B></a>
and the <A
HREF="http://web.archive.org/web/19970330114449/http://zeldman.com/spon
sor1c.html"><B>basic decency</B></a> of the organization which places
advertising on my site.</p>
...
<p>For this reason I still publish and invite you to read <A
HREF="http://zeldman.com/sponsor1a.html" target="_top">How I Lost My
Sponsors.</a></p>
</article>
```

With the stylistic rules applied to another file separate from the structure, content such as blog posts or newspaper articles can be distributed from one site to another, one application to another, and a site to an application, without retaining any formal bonds with a page or site, and putting the very recognizability of the font and author at risk.²⁶

After all, the semantic evolution of the code of web pages accompanies another level of functional separation that governs current web content and can be traced to what Manovich defined as “database logic,” so that not the pages so much as every text component (by text here we mean an enlarged notion that includes hypertext or HTML) of every web page is memorized in records or a database that will respond—reconstructing relations with other records and dynamically recomposing the fragments of a real HTML page—only if and when requested by a navigator who, at any time, can become an author. Moreover, this navigator can become an author not only because he or she is able to participate in the construction

of an independent reading path, as intended by hypertext literature from Landow on, but because this person can construct new writing spaces.

In fact, assembling records from originally unconnected databases will form HTML structures of infinitely reproducible content, ready to be repropounded in new and potentially infinite contexts of representation and use: a collage that goes well beyond the cut-and-paste function to which we have become accustomed since the first word-processing programs. The practice of copying and pasting portions of text operates within the conditions set by the application that copies and the application that pastes. Accepting this paradox, it is a static relationship that ends with the encounter confirmed by the keyboard command “cmd+V” (“ctrl+V” for Windows users). After that action, it is up to the author to readapt the pasted content to the new application environment or to paste new content. Uncoincidentally, “copy and paste” is the zero-grade tool offered by social networks to redistribute content on other social platforms, starting with blogs: the purpose of the “share” button under YouTube videos and over Flickr photographs is to paste the fragment of an HTML code containing the reference to the video or photo content onto another HTML page. The operation embeds objects, and this stabilizes the rewriting potential of the content: what will be integrated into the site is that specific video with that specific resolution. A modification, an update of the content of the site that included it, will necessarily go through another copy-and-paste operation performed by the author.

To take up the words of Eduardo Navas, who examined the musical etymology of the term “remix,” the term “copy-and-paste” is a regressive form of remix.²⁷ In the technique of taking music from here and there, introduced and popularized by the hip-hop movement, the sampled track remains recognizable and accessible in its static and individual nature as an object captured on a support. Similarly, in the technique of incorporating HTML code from one site to another, which continues to represent the most popular and immediate Web 2.0 mashup form (read “user friendly”), the pasted code replicates the content of a publication that is merely the static and fixed reflection of another publication.

The 2.0 social platforms have led above all to a “regenerating” remix where sampling is a dynamic technique in which the source does not serve up published content, but is fed by a constantly updated publication process. In this scenario, the author’s intervention does not connect two contents through the output code, but instead interfaces two applications through programming. As Navas noted:

In software mashups, the actual code of the applications is left intact, which means that such mashups are usually combinations of preexisting

sources that are brought together with some type of “binding” technology.²⁸

When YouTube and Flickr publicly provide their Application Programming Interface (API), they offer authors a framework through which they can create a new site, a new application, a new plugin that dynamically negotiates the contents of the source platform, which acts as a data and service base to use in relation to purposes that, in turn, can be renegotiated according to the behavior of the author or user of the regenerative mashup:

Did you ever want to download all your photos on Flickr? Or see all images that have your favorite shade of pink? Or find new ways to upload your content? You can do that and much much more. We’d like to think of all the cool bits on Flickr as seeds for beautiful and inspirational interactions when given the right care from our awesome developer community.²⁹

The mashup galleries that can be found on Flickr³⁰ and YouTube³¹ along with Yahoo Pipes³² show a sort of workshop where one can “rewrite the Web,” combining RSS feeds, geotags, photos, and search engines in original forms of output and aggregations,³³ where the texts, photos, videos, and web images are not so much reproducible as they are simultaneously reprogrammable. If reproduction implies a before and an after, an author and a reproducer, or in terms of remix, a sampler and a sampled, then real-time rewriting only presumes a product for which the original author proposes one of various possible programmings. For example, Fluidr³⁴ is a complete reprogramming of Flickr: navigation, interface, and the function of the contents of the original application have been reformulated by Sidath Senanayake (the author of the reprogramming) in a personal style that gives the product a wholly original configuration because the interactions generated by the mashup are original, starting with the visualization filters applicable to exploration of photos users’ uploads to Flickr.

As Christian Heilmann noted,³⁵ the true Content Management System (CMS) is the entire Web. Starting from this consideration—which may initially seem trite—if we merely consider the multitude of accounts and platforms through which we manage the fragments of our everyday work and social life, then our presence on the Web takes on a completely different perspective. The very existence of a website as the static place of an identity and personal representation thus changes profoundly.

CHAPTER TWO

MY SITE: FROM GEOCITIES TO WORDPRESS

2.1 Once Upon a Time: The Static Website

Perhaps we still do not have an official date marking the end of Web 1.0 and the transition to Web 2.0, but October 26, 2009 could be a highly representative day. That is when Yahoo closed Geocities,³⁶ the website hosting service it acquired in 1999, four years after it was established. Geocities offered the first generation of navigators not only free advertising space, but above all a system for the publication of web pages, which naturally could not be done without knowing HTML. As elementary and relatively easy as it was to learn, even now HTML continues to be a hurdle for when authors who do not feel comfortable with information technology want to write a page. In fact, all we need to do is identify with the user in front of a monitor with a black-and-white notepad and strange elements like *h1* or *ul*, not to mention *div*, all of which enclosed in curious angle brackets, to understand and justify him or her and even offer our solidarity.

But the reasons behind the farewell to Geocities, which in 1998 was the third most visited site on the Web after AOL and Yahoo,³⁷ have nothing to do with the HTML learning curve: the millions of pages written by subscribers over the years (a total of thirty-eight, as it turns out)³⁸ demonstrate that enthusiasm and the fact that it was free pushed millions of navigators to peruse guides on how to color a text green or right-align a photograph, without having any of the visual aids that the WYSIWYG desktop office had popularized for at least a decade. In any case, starting in late 1998 the service also started to offer GeoBuilder, an online visual HTML editor to simplify the creation of pages.³⁹

The real problem with Geocities, at the dawn of the twenty-first century, was its logic. The document management system “thought” in pages: static pages. In other words, for an aspiring author every act of presence on the World Wide Web meant creating an HTML document (we

could almost call it a *physical* document) that had to be planned and built in all its structural aspects (header, navigation, content, footer) and then associated with all the other documents—already created or to be created—in order to compose the website.

Now the first thing that those of us who are old enough to remember the pages of Geocities sites or, if you are one of the fortunate members of the younger generation, those who explore dedicated archives like Geocities.ws⁴⁰ will notice is the graphical and typographical infancy of the Web: flashing text, animated GIFs, psychedelic colors, and a glut of Comic Sans characters.

However, a slightly more in-depth analysis promptly reveals what is missing: an architecture of content, both visual and informative. Writing and publishing a web page—regardless of one’s level of HTML expertise, with or without images, colored or gray—could and can be simple. Organizing numerous web pages on a site, meaning a coherent communication project, a recognizable and distinct voice, required an effort on the user’s part for which the Geocities software offered no assistance whatsoever.

The overwhelming sensation of amateurism that the view of those sites evokes today thus stems from the limitations of authors, whom we certainly should not expect to be information architects, as well as the limitations of the CMS, which did not contemplate any solution to the problem.

If by way of contrast we were to turn to the 1997 and 1998 archives to retrieve the site of a web professional like Jeffrey Zeldman, we would not only—and predictably—notice greater attention to graphics, but also that: a) the blog has existed for as long as the Web has; b) that the site management concept was the same as that of Geocities: in other words, static pages. This means that Zeldman, like nearly all professional authors during those years, “handcrafted” every page, from the home page to archives and from the autobiographical presentation to useful links, organizing, shifting, and connecting the documents one by one—by hand, as it were—when they grew in number as content increased. To grasp how burdensome this work was, we need simply go back in time to the site archives to see that beyond a certain level of sustainability, not even a professional like Zeldman had the time or energy to maintain graphical and stylistic consistency over the years. This work became even more time-consuming as the size of the site—and thus the number of static pages—grew. So if someone like Jeffrey Zeldman couldn’t manage, how was someone like Alexis Yorkis Trincado Frutos supposed to?⁴¹

If we want to draw a parallel, traveling forward in time, the limitations of Geocities are the same as those that Apple surprisingly replicated in 2006 with iWeb, touted as “the easiest way to create professional-looking web pages, online photo albums, blogs and Podcasts in minutes.”⁴² To a certain extent, the Press Office in Cupertino was actually right. There had probably never been such an attractive and fun program to use to create a web page without knowing a single HTML tag. In pure Apple style, and based on predefined themes (objectively elegant if we consider the purpose and consumer target), the totally graphical interface of iWeb makes it possible to handle texts, images, photos, videos, and audio files like objects to drag and drop into the document window, then positioning and placing them in lieu of the “placeholders” envisioned by the theme, in a transparent integration with all the other digital-life applications envisioned by Apple (iPhoto, iMovie, GarageBand, and even iTunes).

The problem with iWeb is that the freedom it offers users to play around is reflected in a multiplication of pages, which in turn multiplies the problems managing them and inevitably jeopardizes the existence of a coherent and uniquely identifiable project. As much as software can find intelligent solutions in the creation of a persistent navigation or an immediate way to put the pages in order (in the case of iWeb, dragging the icon of a page above or below the others in the left sidebar of the application automatically changes the position of the pages on the site menu), the entire burden of the project architecture lies squarely on the user’s shoulders, as does the writing of most of the structural components of the page, which must be repeated, document after created document: site name, page title, footer, sidebar.

Added to this is the work of dealing with FTP configurations⁴³ to transfer the files physically from the computer in which the static pages were created and modified to the server hosting them, making it easy to understand why iWeb was born “old” and is falling into oblivion, to everyone’s general indifference: iWeb was not included in the update of iLife ’11 and has stopped being developed.⁴⁴

In an era in which the file system itself goes to the cloud,⁴⁵ who needs to work on a “page”?⁴⁶

That said, however, anyone who thinks that the issue only affects amateurs is wrong. The crisis of the static page, of the document to create, save, and then share on the Web—using more or less fluid procedures—also affects professional web applications like Adobe Dreamweaver, unquestionably the professional software most widely used by web designers and the first to combine the rigor of control over code with the practicality of the WYSIWYG interface. What separates it from iWeb is a

gap as wide as its reference target (and relative end cost) and yet, like iWeb, Dreamweaver belongs to the family of desktop applications: packages that can be installed on (and are made for) personal computers and that transfer the results from that station. In this case, it publishes them on the Web as the pages of a site. But between 1997, the year Macromedia (acquired by Adobe in 2006) released the first version of the program, and today what seems increasingly evident is the paradox of processing outside the web pages that are born for and thrive on the Web, in an ontological sharing from the very phases of conception.

The outside world has changed for amateurs and professionals alike. For both, the Web is no longer a space where you transfer files, but a platform where you create and distribute content.

This isn't a matter of bells and whistles, it's absolutely fundamental. Ultimately a web site is all about content—posting it and making it findable—and Dreamweaver and the other static HTML editors have proven fundamentally flawed when it comes to these two core tasks (and features such as Dreamweaver's libraries and templates are patches not solutions).⁴⁷

It is incredible to think that Dreamweaver's features are out of proportion with respect to the current practices for developing, planning, and managing a site, just as Photoshop (one of Adobe's other historic programs for retouching and more) has become inadequate for developing and designing the graphics of a web project. Historically, with Photoshop (and comparable programs like Fireworks, also owned by Adobe)⁴⁸ web designers prepare graphical templates of the main pages of a site in order to give customers (or themselves, in the case of web designers working on a personal site) a visual preview of the site. However, while Photoshop contemplates vector elements, it produces pixels, meaning bitmap graphics that, tested on the browser when the page goes live, will exactly and predictably match the settings of the navigator screen and its use of the web application window. This unpredictability becomes even more dramatic if we think of new devices and the new ways in which visitors use web contents: smartphones, tablets, and apps—along with traditional desktop computers and laptops, and the browser itself—have redesigned the hardware and software environments in which texts, images, and videos are published and read, readapting them to the context in which they are used. Underscoring the need to abandon the presentation of web prototypes through static images, Andy Clarke suggested designing directly in HTML and CSS for browsers,⁴⁹ for what Ethan Marcotte called “responsive web design.”⁵⁰

Similarly, developing sites and pages in a static and “unconnected” system such as a desktop application like Dreamweaver will never render the “lives” that the contents of those pages will experience on the Web, redistributed through RSS feeds to other sites and other readers, categorized and labeled in dynamic archives, indexed by search engines, mashed up for new applications, and incorporated into other pages.

What an author⁵¹ needs is not a system to produce pages, but a system to produce, publish, and manage the contents that are born—and thrive—on the Web.

2.2 Blog: Title, Body Text, Publish (and Distribute Everywhere)

The home page of Pitas.com was divided into two simple columns. On the left was the invitation to register with the service, touted as being fast and fun. On the right, before the invitation was repeated, there were two paragraphs of text. The first sufficed to mark the beginning a new era of online self-presentation:

Hey chief, tired of updating your “news” page for your website and want a page you can update easily, quickly and from any web browser? Or maybe you’d like to let some of your friends add news or links to your page, without [*sic*] giving them access to your normal web page? Or you’re sick of editing html pages all the time and then ftping them to your site? Well you’re in luck!⁵²

The blog—more precisely, the weblog—has existed since the advent of the World Wide Web, or at least since the designers, developers, and authors of the new media decided to establish a personal presence on the Web. At the beginning of 1999 Brigitte Eaton started to compile a list of all the weblogs she was familiar with, adopting a simple inclusion criterion: the presence of dated posts arranged in descending chronological order. The sites were veritable logbooks⁵³ in which the authors kept a public trace, documented with the date of publication and their Internet navigations, mainly reporting other pages and other contents of interest found on the Web.

In those years, the territory of weblogs was the privilege of web geeks, people who, for the above-mentioned reasons tied to their profession, knew how to work with HTML code, design and build a page, upload it to an FTP server, and connect it hypertextually to pages on their site or on other websites. The process of creating a post and, above all, its maintenance in the interaction with an overall corpus of the articles—in a

word, the site—was a complex and artisanal process. As a rule, the weblog author put all of his or her latest posts on the home page or a dedicated page called “news” and, when scrolling was about to reach the danger point, he or she would periodically move them to a chronological archive page, then rewriting all the internal connections, which changed as a result of moving the posts.

Pitas promised authors that it would automate all publication operations, naturally with the exception of writing the post itself. In August 1999, a month after Pitas, Blogger—the phenomenon that made blogging popular—made the same promise: it was Blogger that put the last level of customization into practice. Peter Merholz had already anticipated this practice when he put the word *weblog* in the first person plural, namely *wee-blog*, which was instantly shortened to *blog*.⁵⁴

Rebecca Blood⁵⁵ tells us that before Blogger, weblog posts were almost exclusively link-driven, meaning that they listed articles, sites, and news, accompanied by a short author’s note. In the pre-Google era, and before feed readers, this served the function of filtering and selecting, and acted as a watchdog of the accuracy of content published on the Web, which had tangible value, especially if we consider that the “navigator-guardians” were respected and well-known figures in the fledgling world of dot-coms. In its publication interface, Metafilter, another blogging platform born during this period, incorporated a specific field for links to be signaled in the post. However, Edit This Page, Groksoup, LiveJournal, and Pitas itself (which then evolved into Diaryland) inherited this setup, derived from the practice of the first bloggers. Instead,

in the post-Blogger explosion increasing numbers of weblogs eschewed this focus on the web-at-large in favor of a sort of short-form journal. These blogs, often updated several times a day, were instead a record of the blogger’s thoughts: something noticed on the way to work, notes about the weekend, a quick reflection on some subject or another. Links took the reader to the site of another blogger with whom the first was having a public conversation or had met the previous evening, or to the site of a band he had seen the night before. Full-blown conversations were carried on between three or five blogs, each referencing the other in their agreement or rebuttal of the other’s positions. Cults of personality sprung up as new blogs appeared, certain names appearing over and over in daily entries or listed in the obligatory sidebar of “other weblogs” (a holdover from Cam’s original list). It was, and is, fascinating to see new bloggers position themselves in this community, referencing and reacting to those blogs they read most, their sidebar an affirmation of the tribe to which they wish to belong.⁵⁶