**Expensive website, thick interface**

Even though web infrastructures are built in parallel with human needs for language and communication, the interfaces don’t exactly host a variety of user languages and understandings, but rather maintain the hierarchy of language. The CMS software built by Scott was programmed to build the interfaces of presentation websites his clients grew reliant on.

Let me explain. Presentation websites are digital containers to hold the ideas, thoughts, and/or visuals of an artist, cultural institution, brand, or [insert subject]. Their function is to archive information accumulated and maintained throughout the subject’s lifetime and represent the subject as a contained body that has a significant value on the market.[[1]](#footnote-1) To avoid user unfriendly representation of information,[[2]](#footnote-2) it is best if the information is assigned a specific place on the template’s grid.

Building a template is the first important step for interface organization. Most of them have similar structures: header, body, and footer. The header is where the menu usually is, and helps users navigate through the website. The body, usually organized by a grid, contains most of the content. A footer indicates the subject’s copyright data, and behaves like a ‘stamp’ in the digital landscape. My website. My information. My content. *My design*.

Today, there are a lot of open-source templates ready to represent a subject in the best possible way. They come in various structures and styles—the amount there is to choose from accommodates most user’s taste in the current market. While some are hiding in the gutter of the internet, a lot of them are included in the hosting package, ready to be installed. They come with pre-filled information that the user modifies to their businesses:

Where is the institution located? Rodenrijsstraat, 1062 JA Amsterdam.

When was the artist born? 1991.

Where do I buy a ticket? RIGHT HERE.

Such templates are useful for subjects that need to make an online appearance and profit as soon as possible; they can help launch a website within a month, or even two weeks, if an intern or assistant is involved.

There are slower, and more expensive practices for coding templates. Custom templates, for example, are coded more intimately and in close(r) collaboration with the client.[[3]](#footnote-3) The client pays for the web developer’s listening skills until a depiction of their true personality and style appears on the interface. The more defined personality and style are *before* the coding process, the less expensive the service will be.

However they are built, most templates on the internet today follow the reading logic of a printed book. They have preceding and following pages, previous and new articles. When the user clicks through, their information-consuming experience is similar to that of flipping book pages. Except that, in the digital domain, all pages appear within the frame of a screen. Fixed components for navigation (i.e. menu) that appear across the pages are placed close to the corners of the screen so that they don’t disturb the reader’s focus of a single page.

I wondered about this structure translated to the digital world. At the Venice Biennale of Architecture in 2021 (‘How will we live together?’), the Russian pavilion impressed me with an attempt to merge both the physical and digital experiences of the pandemic’s postponement reality. Their project, [pavilionrus.com](http://pavilionrus.com), became a well-formatted container with a function to seize the textual thoughts about various topics that emphasize the country’s position of waiting towards the biennale’s opening. Yes, most countries were in the same position, but not all of them defined their format. The website makes clear how their accumulated content, which represents what the country reflected and learned from being in this ‘unusual’ position, was curated through a classification system: keywords assembling content randomly. This gesture wants to scream the success of breaking free from the traditional classifications (i.e. chronological order) but, in my opinion, only depicts an imprisonment by the same logic, just different terminology.

As expected, the digital was then brought *back* to the physical, in the form of a publication: *Voices (Towards Other Institutions)*. I read the introduction text out loud to myself: ‘Within this multiplicity of angles, the book as a whole works almost as a polyphonic piece in the making, showcasing a web of unexpected connections across thematic clusters. Voices here are gathered into a single volume and according to a seemingly random structure. In balance between a traditional publication and a website, texts are not organized into chapters but associated with a glossary of ten keywords which, just like hashtags, trace connections across various contributions and re-assemble them into a multiplicity of possible indexes.’

How shall I put this delicately… I was not necessarily impressed by the country’s choice to organize the information through keywords (instead of categories) as much as I was by their confidence of doing so. Somehow it sounded as if they’d found the ultimate solution to the issue of the traditional systematic organizations (again, i.e. chronological order) by making the design choices they’d made for both publications. But it also sounded like they hadn’t defined the issue, yet strongly stood behind their solution for it. What is the issue at stake, exactly? Dixit the Russian pavilion: the limited choice of classifying *options* given by the traditional formats. Well. True, but there’s no need to occupy the whole pavilion in order to ‘solve' this. Isn’t it impressive that most structures don’t seem to care about their content not being read?

The pavilion’s idea of randomness brings me to an excerpt from a poem I wrote about the Venice Biennale of Architecture in 2021:

How to make the word strategic sound sexy again?

I know!

It’s in the random.

Surround random with walls that are trained to predict exactly what is it that makes random random and,

what is it that makes the random randomly appear??

Watch.

While simultaneously encouraging the random to do what random does,

accommodate the clues in a place not strictly squared.

In other words,

make random predictable.

For yourself.

It is safe to say that,

we all agree.

Designing the design designs us back.

Solving the solutions makes it clear that we have a lot of problems.

Safe.

Most templates on the internet today, no matter how they were designed, enforce strongly the practices of what Gerald Weinberg in his book *The Psychology of Computer Programming* calls ‘compression’: ‘If the entire program fits onto one page, all relevant parts are obviously on that page.’ Compression is what makes the distance between the form and the content. It is the core discipline of creating linguistic and visual hierarchy, no matter how one chooses to call the buttons that summon the content: categories, hashtags, tags, keywords. Compression kills the random.

What fascinates me about Weinberg’s observation is how he reflects on compression through the programmer’s memorial capacity. For example, he writes that there are two memorial concepts related to the practices of compression: ‘locality’ and ‘linearity’. Locality is similar to the feeling of a space—a deja-vu effect—while linearity corresponds to sequential remembering obtained over time. He argues that a ‘good’ programmer needs to have a strong sense of locality through linearity, that one needs to have a feeling of familiarity in space as much as one needs to have a technical knowledge of a space. I’d like to think that a similar memorial capacity applies to my idea of a good web developer. That without *feeling* the space of the interface he’s building, he cannot fully claim to *know* the space.

Understanding the two different modes of memorial capacity, it seems to me that web developers such as Scott and Larry prioritise memorial linearity more than locality, or both options for remembering. They don’t mess around with feelings. Their entire programs, templates, and structures always fit exactly on one page, where they can control whatever happens within them. They are in full control of their stacks.[[4]](#footnote-4)

Understanding the web developer’s common need for (linearly-oriented) compression when building a website, I can humbly say that presentation websites tell the subject’s story linearly, from beginning to its current end. For example, if I refer back to the architecture of ZipSpace museum’s website:

Homepage,

About page,

News page,

Archive page,

Shop page?

I read the following:

Welcome to our reception where you can:

find out who we are,

see what we’re very busy with,

what our taste is like! and,

support/identify with us.

Welcome to our reception where you can:

see what we’re very busy with,

support/identify with us.

Welcome to our reception where you can:

find out who we are,

Circles—sometimes with different beginnings and ends—but always circles.

When such compression is applied to a website, no matter how ‘content-full’ it is, it depicts how perfectly organized the developer’s idea of randomness is. That’s why the templates of presentation websites don’t change much over time; they host specific categories of information within specific corners of the grid. Through categorization, a hierarchy of information is born, and all incoming information is classified as relevant or irrelevant. The truly random information cannot enter this website unless it matches the defined randomness on the website. Random information is irrelevant to the website’s owner, ZipSpace, and only relevant information gets to build its history.

By now, ZipSpace is the template’s *slave* and must produce only the information its template is programmed to display. In their physical space, the program doesn’t change much over time either. Still cleaning the permanent collection and filling up the Plateform. Events are set up with technical equipment to document what the website allows them to publish.

What I’ve seen so far, is how presentation websites like this not only depict what kind of future information is welcome to mark its further history, but also how their controlled system for incoming information will secure them a continuity in the future that is decided by higher institutional bodies, such as those that fund them. Viewed in this way, the presentation website is a meeting point of different classes.

A small-scale example of this is the artist’s website. *Hi there!* To the left is my bio, to the right a list of projects I’m working on. The one I’m currently busy documenting will soon be shown here in the middle, with a picture on top and a short text underneath. *Send me an e-mail*.

A second small-scale example is another artist’s website: Here is my full CV, with links leading directly to the institutional documentation of my work on their websites and silent retreats, erm, I mean *residencies* I went to. *Send me an e-mail.*

An even smaller-scale example is a graphic designer’s website: Here is my name. *Send me an e-mail*.

In their straightforwardness, presentation websites are designed around the answers, and not the questions they might entail. The answer is yes, the subject is open to work. Because of this aura, it is legit to say that for a successful[[5]](#footnote-5) idea/project/opinion/story/product, a presentation website is the safest choice of web type to develop a digital translation in.

Underneath their interface, presentation websites are actually rather heavy. Most users want control over their information and the information’s curation. For this, they need content managers (such as WordPress, Kirby, WebFlow, SquareSpace, Cargo, etc.) and not just HTML, CSS, and Javascript—basic tools for building any kind of a website or idea.

By now, we have reached the point of facing yet another door to the market: the tech industry. But let’s turn around for a moment, because I want to go back to Scott.

The reason why Scott likes to build presentation websites is that users, accustomed to seeing always the same templates, no longer speculate on the many ways their stories can be told, and yet are still in need of telling them. At this point in his career, Scott’s clients no longer question why cyan is his third color of choice after black and white. It’s just, like, *so him*. And that, they tell themselves, is one of the reasons why he’s expensive.

Another one is his linguistic assertiveness. Scott enjoys evangelizing about how simple products are the hardest to produce. He enjoys informing his clients how long (ten years) it took for him to build a software they’re about to invest in. He enjoys how the clients who don’t even try to tell their stories differently make him feel: trusted. Their docility gives him the headspace he needs to focus on advancing the technically complex structure that lives underneath his websites, making them autonomous. Autonomy is important to him, for he knows that any other dedicated web developer could gain a greater technical capacity to build exactly the software he built.

With Larry by his side, Scott’s year seems easier somehow. Having more time than usual to think about random stuff has made way to a different kind of craving bubbling up inside of him. Technically speaking, what if he built a social network? Technically speaking, would it break his CMS? Technically speaking, how many users are we talking about? Technically speaking, what are they gonna talk about?

And low and behold: as if the tech industry heard his contemplative worships, a commission to build a real-time website for educational purposes lands right in Scott’s lap. Real-time websites are the second kind of a website in the current state of our digital world.

Real-time websites are similar to presentation websites because they are also heavy, need a subject to contain across the grid, and are built through (linearly-oriented)compression. The experiences of the two, however, differ.

Real-time websites are immersive through their interaction design. Not visually immersive, like virtual reality, but just enough to make one forget about the passing of time. Built with the purpose to make the users feel connected, they allow their users to interact with conversation triggering components (i.e. chat, ability to comment, poke, block, report, like, heart, delete, edit, send, undo…). These user interactions are extracted in texts, feedback, rating, opinions, preferences, and so on. They are also known as bits.

Compared to the read-only user experience on presentation websites, on real-time websites, the users are made to believe that they can do a lot more than just *consume* the content. Their serotonin level, heightened by their user abilities to edit, produce and manage the content, makes them feel liberated in their agency. The more interactions they are enabled to perform, the less aware of their constraints they become. The updated version of their universally shared online freedom is more and more standardized. Social media platforms are obvious examples of this, but we can also think of e-com platforms and educational websites, both expected to have a different purpose in their digital existence.[[6]](#footnote-6) Whichever type of the website I land on, none feels like home to me.

1. In case of a product: worth buying; in case of an artist: worth employing; in case of an institution: worth funding. [↑](#footnote-ref-1)
2. Which the users don’t have the attention span for. [↑](#footnote-ref-2)
3. For example exchanging website unrelated information such as book references, music, event recommendations, etc. [↑](#footnote-ref-3)
4. ‘Stacktivism is ambivalent and struggles with totality, the global scale, and the planetary whatever. “Think big but act in small steps,” that’s the motto. We Are Infrastructure. Stacktivism fights against the comfort of ignorance and tries hard to overcome drifting-off- by-design, the tendency to blissfully float above it all. While defining what stacktivism could become, it is good to keep in mind that we’re free to use Bratton’s The Stack as a theory toolbox and not interpret it as a hermetic belief system. Designs can intermingle. In line with Bratton, stacktivism claims to oversee all levels. It grasps the politics of code, algorithms, and AI. It is aware of the behavioral science manipulation of moods through careful interface design choices. It is alert to 5G electronic smog, phishing emails, fake news, and other sleazy suggestions from your “friends.” How good are your bot detection skills? This hyperawareness comes at a high price. Not everyone is a stacktivist. ☹’ — Geert Lovink, *Stuck on Platform* [↑](#footnote-ref-4)
5. Profitable for both web-developer and client; web developer can charge a lot if the client is guaranteed to receive the funding. [↑](#footnote-ref-5)
6. ‘The fundamental difference between mass media and digital media is interactivity. Books, radio, and television are “read only” media. We watch, but only have a choice over how we will react to the media someone else has made. This is why they are so good for storytelling: We are in the storyteller’s world and can’t talk back. Digital media, on the other hand, are “read-write”. Any digital file that is playable is also sharable and changeable. (Files can be locked, at least until hackers figure out how to break the lock, but such protection is ultimately against the bias of the medium. That’s why it so rarely works.) As a result, we are transitioning from a mass media that makes its stories sacred, to an interactive media that makes communication mutable and alive.’ — Douglas Rushkoff, *Program of be Programmed* [↑](#footnote-ref-6)