



Figure 3.1

Diagram by Metahaven of the six layers of The Stack.

15. The Layers of The Stack

How does this work? What is the model exactly? The Stack is divided into six layers, moving from the global to the local, from geochemical up to the phenomenological: *Earth*, *Cloud*, *City*, *Address*, *Interface*, and *User*. For The Stack, multiple layers co-occupy the same terrestrial location (horizontally) but gather and subdivide their processes vertically into discrete machinic “jurisdictions.” That subdivisional geography is bound within the mechanism of the platform, but at the same time, The Stack platform itself disturbs existing models of jurisdiction and projects its own out into the world. Each layer configures and coheres its own specific forms of governance over what it touches,

and so the difference between one layer and another is also a difference between those forms and processes. That is, the *Cloud* layer is defined by certain spatial and temporal demands that include what we conventionally recognize as *Cloud* infrastructure (e.g., data centers, fiber optics, and in-browser applications), but those demands may press on rather unlike things as well (parking spaces, medical equipment, food supply statistics). The six chapters that follow discuss each one of these layers in sequence, detailing especially how each produces its own governing logics of scale, physicality, textuality, embodiment, force, motion, stability, and, perhaps most important, how each of these logics provides productive accidents that may direct us toward unexpected design outcomes.

The diagram of The Stack in figure 3.1 shows a vertical-sectional relationship between possible positions occupying all six layers at once. It demonstrates that while positions on layers are held simultaneously, each layer governs that position semiautonomously. In the practice of real communication, any one instance of such a stack works only in combination with another, and in this pairing, the real social form of active stacks comes into focus. That structure of connection and communication is similar to OSI or TCP/IP. Connections tunnel down, across, and back up stacks, creating temporary circuit-like connections in a U-shaped trajectory. *User* connects to *User* by initiating a “message” that tunnels all the way down through the other layers to the bottom and then back up again, and so direct communication between *Users* activates all six layers down The Stack and then all six layers again back up The Stack. The whole of the system is invoked and activated by any one connection; the whole is “folded” into each single instance of activation, bending the universal and the particular into one another. We define any one such path taken down and up The Stack in a U-shaped trajectory as a *column*. For one “session,” however fleeting or stable, columns fix one *User* to another by linking layers to those above and below into a whole, but they allow anyone or anything that is a *User* to initiate (or be initiated by) as many other simultaneous columns as needed. That is, as any given column tracks up and down, there is no final instance of vertical or horizontal integration that would truly and ultimately resolve a *User* down to the *Earth* layer or *Cloud* layer for good. Any one *User* will initiate millions of different columns at different moments over time, thereby executing different combinations of nested positions. This slipperiness does not correspond well with modern logics of last-instance sovereignty and citizenship. It’s not that there is no “there there” but rather that moment to moment, there are too many “theres” for any one sovereign geography to finally decide all the others.

At the top of any column, a *User* (animal, vegetable, or mineral) would occupy its own unique position and from there activate an *Interface* to manipulate things with particular *Addresses*, which are embedded in the land, sea, and air of urban surfaces on the *City* layer, all of which can process, store, and deliver data according to the computational capacity and legal dictates of a *Cloud* platform, which itself drinks from

is an odd multiple of p_n such that $mult[n] < j + 2p_n$. If $mult[n] < j$, we can increase $mult[n]$ by $2p_n$ and the same conditions will hold. On the other hand if $mult[n] \geq j$, the conditions imply that j is divisible by p_n if and only if $j = mult[n]$.

```
(If  $p[n]$  is a factor of  $j$ , set  $j\_prime \leftarrow false$  26)  $\equiv$ 
  while  $mult[n] < j$  do
     $mult[n] \leftarrow mult[n] + p[n] + p[n]$ ;
  if  $mult[n] = j$  then  $j\_prime \leftarrow false$ 
```

This code is used in section 22.

27. Index. Every identifier used in this program is shown here together with a list of the section numbers where that identifier appears. The section number is underlined if the identifier was defined in that section. However, one-letter identifiers are indexed only at their point of definition, since such identifiers tend to appear almost everywhere. [[An index like this is prepared automatically by the WEB software, and it is appended to the final section of the program. However, underlining of section numbers is not automatic; the user is supposed to mark identifiers at their point of definition in the WEB source file.]]

This index also refers to some of the places where key elements of the program are treated. For example, the entries for ‘Output format’ and ‘Page headings’ indicate where details of the output format are discussed. Several other topics that appear in the documentation (e.g., ‘Bertrand’s postulate’) have also been indexed. [[Special instructions within a WEB source file can be used to insert essentially anything into the index.]]

Bertrand, Joseph, postulate: 21.

boolean: 15.

c: 7.

cc: 5, 7, 8, 10.

Dijkstra, Edsger: 1, 15.

Eratosthenes, sieve of: 24.

false: 13, 26.

integer: 4, 7, 12, 17, 24.

j: 12.

j_prime: 13, 14, 15, 22, 26.

k: 12.

Knuth, Donald E.: 15.

m: 2.

mult: 24, 25, 26.

n: 23.

new_line: 6, 9, 10.

new_page: 6, 9.

ord: 17, 18, 19, 20, 21, 22, 23, 24, 25.

ord_max: 17, 19, 23, 24.

output: 2, 6.

output format: 5, 9.

p: 4.

page: 6.

page headings: 9.

page_number: 7, 8, 9.

page_offset: 7, 8, 9.

prime number, definition of: 13.

print_entry: 6, 10.

print_integer: 6, 9.

print_primes: 2.

print_string: 6, 9.

row_offset: 7, 9, 10.

rr: 5, 8, 9, 10.

square: 17, 18, 20, 21.

true: 4, 13, 22.

WEB: 1.

write: 6.

write_ln: 6.

ww: 5, 6.

(Fill table p with the first m prime numbers 11)

Used in 3.

(Give to j_prime the meaning: j is a prime number 22)

Used in 14.

(If $p[n]$ is a factor of j , set $j_prime \leftarrow false$ 26)

Used in 22.

(Increase j until it is the next prime number 14)

Used in 11.

(Initialize the data structures 16, 18) Used in 11.

(Other constants of the program 5, 19) Used in 2.

(Output a line of answers 10) Used in 9.

(Output a page of answers 9) Used in 8.

(Print table p 8) Used in 3.

(Print the first m prime numbers 3) Used in 2.

(Program to print the first thousand prime numbers 2)

Used in 1.

(Update variables that depend on j 20) Used in 14.

(Update variables that depend on ord 21, 25)

Used in 20.

(Variables of the program 4, 7, 12, 15, 17, 23, 24)

Used in 2.

D. HOW THE EXAMPLE WAS SPECIFIED

Everything reproduced above, from the table of contents preceding the program to the indexes of identifiers and section names at the end, was generated by applying the program WEAVE to a source file PRIMES.WEB written in the WEB language. Let us now look at that file PRIMES.WEB, in order to get an idea of what a WEB user actually types.

There’s no need to show very much of PRIMES.WEB, however, because that file is reflected quite faithfully by the formatted output. Figure 2 contains enough of the WEB source to indicate the general flavor; a reader who is familiar with the rudiments of T_EX will be able to reconstruct all of PRIMES.WEB by looking only at the formatted output and Figure 2.

Figure 2a starts with T_EX commands (not shown in full) that make it convenient to typeset double brackets [[...]] and to give special typographic treatment to names like ‘WEB’ and ‘PASCAL’. A WEB user generally begins by declaring such special aspects of the document format; for example, if nonstandard fonts of type are needed, they are usually stated first. It may also be necessary to specify the correct hyphenation of non-English words that appear in the document.

Then comes ‘@*’, which starts the program proper. WEB uses the symbol ‘@’ as an escape character for special instructions to the WEAVE and TANGLE processors. Everything between such special commands is either expressed in T_EX language or in PASCAL language, depending on the context.

This last step renders the world dead by conflating memory—which is traditionally and initially regenerative and degenerative—with other more stable forms of media such as paper storage, a comparison that is still with us today at the level of both memory (files) and interface (pages and documents). This conflation both relied on and extended neurophysiological notions of memory as a trace or inscription, like the grooves of a gramophone record. McCulloch, for instance, in 1951, in response to objections posed by von Neumann over memory as reverberatory circuits, outlined a hierarchical memory system that resonated with von Neumann’s schema. There are first temporary reverberations, and second, nervous nets that alter with use (central to conditioned behaviors). The third type of memory, which he sees as an informational bottleneck, however, leaves him unhappily stumped; he is at a loss to describe its location and its operation:

I don’t see how we can tell where we have to look as yet, because in many of the experiments in which there are lesions made in brains, we have had large amounts of territory removed. However, usually we fail to destroy most fixed memories: therefore, we cannot today locate the filing cabinets. I think that sooner or later answers to the question of those filing cabinets, or whatever it is on which is printed “photographic records” and what not, will have to be found.⁸⁰

The term *filing cabinet* is drawn from von Neumann’s own terminology. In his response to McCulloch’s paper, von Neumann, perhaps informed by psychoanalytical arguments that memories never die (one of von Neumann’s uncles introduced psychoanalysis to Hungary and von Neumann apparently loved to analyze jokes) or by his personal experience (he allegedly had a photographic memory and could recall conversations word for word), presents the following “negative” and not entirely “cogent” argument against memory as residing in the neurons:

There is a good deal of evidence that memory is static, unerasable, resulting from an irreversible change. (This is of course the very opposite of a “reverberating,” dynamic, erasable memory.) Isn’t there some physical evidence for this? If this is correct, then no memory, once acquired, can be truly forgotten. Once a memory-storage place is occupied, it is occupied forever, the memory capacity that it represents is lost; it will never be possible to store anything else there. What appears as forgetting is then not true forgetting, but merely the removal of that particular memory-storage region from a condition of rapid and easy availability to one of lower availability. It is not like the destruction of a system of files, but rather like the removal of a filing cabinet into the cellar. Indeed, this process in many cases seems to be reversible. Various situations may bring the “filing cabinet” up from the “cellar” and make it rapidly and easily available again.⁸¹

Von Neumann’s “negative argument” relies on files and the human mind as the owner/manipulator—or, to return to Cornelia Vismann’s argument outlined in chapter 2, chancellor—of files. It also depicts the human brain as surprisingly nonplastic: easily used up and unerasable, hence once more the need for great storage. It also moves away from memory as based on erasable “regenerative” traces toward fantasies of traces

that do not fade: immortality within the mortal machine.⁸² This is a far cry from Vannevar Bush's description of the human mind in chapter 2 as fundamentally ephemeral and prone to forgetting. The digital paradoxically produces memory as storage, in part because logical algorithms need to read and write values. An entire process can fail if one variable is erased.

Memory as storage also allows von Neumann to describe genes as a form of human memory. In *The Computer and the Brain*, he writes, "another form of memory, which is obviously present, is the genetic part of the body; the chromosomes and their constituent genes are clearly memory elements which by their state affect, and to a certain extent determine, the functioning of the entire system."⁸³ With this move toward genes as memory—necessary for his theory of self-reproducing formula—neurons would not stand in for words (true or false propositions), but words (instructions) would come to stand in for neurons.

Descriptions that Can

The deed is everything, the Glory naught.

—*Faust*, Part II

According to William Poundstone, the last anecdote of von Neumann's "total recall" concerns his last days, when he lay dying of cancer at Walter Reed Hospital, a cancer caused by his work on nuclear weapons (the drive for nuclear weapons also powered the development of digital electronic computers; American computers and neoliberalism are both reactions to Nazism).⁸⁴ His brother Michael read *Faust* in the original German to von Neumann and, "as Michael would pause to turn the page, von Neumann would rattle off the next few lines from memory."⁸⁵ Converting to Catholicism before his death, von Neumann was deeply influenced by the work of Goethe, *Faust* in particular. Said his brother Nicholas, "We studied *Faust* in school very thoroughly, both parts, in original and in Hungarian translation. And we discussed it for years and rereading it occasionally thereafter, throughout our respective lifetimes."⁸⁶ One of the three passages Nicholas describes as particularly important to his brother was Faust's grappling with logos: "Faust's monologue at the opening of the First Part: 'In the beginning was the Act,' and the corresponding statement in Part II: 'The deed is everything, the Glory naught.'" This we discussed in the context of the redeeming value of action.⁸⁷ According to Nicholas, this passage led "ultimately to John's views emphasizing the redeeming value of practical applications in his profession."⁸⁸ John von Neumann as an unredeemed (although not yet fallen) Faust.

This passage, however, has other resonances, intersecting with the question of logos weaving through this book. Faust, seeking to translate the Bible into German pauses over "in the beginning was the Word":

This Is Not a Dance

U.S. NEWS AND FAN MEDIA recently descended upon the debunking of a prevailing cultural icon: the lone video game player (chubby, pasty-faced, at home in a dark room, planted in front of the computer or television with his eyes locked on the glowing screen, totally immobile except for the furious button-punching and joystick-twisting of his fingers) was triumphantly supplanted in media accounts by pairs of teenage boys frenetically hopping in synchronized rhythm to techno music atop the lighted stage of an arcade game, in public space, while crowds gathered to gawk, cheer, and queue up so that they too could play.¹

Sustained by on- and offline fan communities, this “arcade craze” revolved around the dance simulation game *Dance Dance Revolution*, or *DDR*. *Dance Dance Revolution* was officially released to U.S. arcades in 2000 and quickly became the most successful of a genre of rhythm action games called Bemani.² Employing specialized input devices rather than the traditional joystick-based game controller, Bemani video games turn players into performers—requiring them to dance, shake maracas, scratch DJ turntables, beat drums, and strum guitars in synchronicity with the game machine. The arcade version of *Dance Dance Revolution* works

like this: as strobe lights flash and music thumps out of large, neon-lit speakers, each player follows a dance sequence, symbolized by a pattern of arrows scrolling up the game screen, by stepping on the corresponding quadrants of a raised dance platform. Scores are based on the accuracy and timing of the player’s steps. The game ends if the player misses too many steps.

Dance Dance Revolution has gone through multiple versions and modes of play since its original release as an arcade game in Japan in 1998, moving between a highly visible arcade-based “dance craze” in public leisure space and a home-console game experienced in private. Far-flung renditions of *Dance Dance Revolution* include versions that are played with your fingers on a handheld mini-dance pad, ones that allow live, networked gameplay in shared virtual space, and ones that integrate a digital camera so that players can see themselves dancing onscreen. Official full releases of the game mingle with song and character upgrades, as well as a profusion of region-specific, third-party, pirated, and hacked game hardware and software.³ Rather than each new release replacing the old in linear progression, all of the versions continue to circulate through both authorized and unauthorized global channels. Certainly,



9. Plot summary of Hans Dominik's *The Trail of Genghis Khan, A Novel of the 21st Century* in Lindqvist 2001, section 128.
10. Dower 1987.
11. Lindee 1994.
12. Hecht 2009.
13. Borstelmann 2001; McNay 2001.
14. Kuisel 1993; de Grazia 2005; Krige 2006.
15. Adas 1989.
16. Latham 2000; Gilman 2003; Engerman et al. 2003; Engel 2007.
17. Adas 2005; Westad 2005.
18. Cooper 2005; Cooper and Packard 1997; Escobar 1995; Mitchell 2002.
19. Abraham 1998.
20. Moon 1998.
21. Diouf 1997.
22. Prashad 2007.
23. Westad 2005: 396. A similar argument, albeit with different empirical emphasis, lies at the core of Mamdani 2004.

2 Islands: The United States as a Networked Empire

Ruth Oldenziel

In the spring of 2003, US President Bush, British Prime Minister Blair, Spanish Prime Minister Aznar, and their host, Portuguese President Barroso, landed on the island of Terceira to hold a press conference and present Iraq with an ultimatum for war. The location of their press conference in the Portuguese Azores—a constellation of nine Atlantic islands far from mainland Europe—puzzled commentators. Reporters speculated that this far-flung setting, best known as an exotic holiday destination rather than a convincing projection of US power, symbolized the marginal European support for an invasion of Iraq.

In ways that commentators did not realize, the choice of the Azores brought into focus a projection of American power rooted in networks and islands. Washington's geographical selection was not a remote launch pad for war, but a manifestation of power that often and purposefully remains hidden from view.¹ The Azores transformed from a remote Portuguese outpost into a US hub of information, communication, and military systems. The process of transformation reconfigured the Azores and several other colonial island chains into nodes in the American projection of power. That process fundamentally remapped the globe during the Cold War.

The character of American power has been widely discussed. The historian Arthur Schlesinger Jr., articulating the Cold War consensus, once argued that the United States, though "richly equipped with imperial paraphernalia [such as] troops, ships, planes, bases, proconsuls, local collaborators, all spread around the luckless planet," should be understood as an "'informal' empire, not colonial in polity."² That argument became a dominant narrative frame during the Cold War as the US faced ideological competition from the USSR.³ Since then, others have refined the idea to claim the US is a reluctant empire, an empire by invitation, or the world's indispensable nation.⁴ All of these arguments turn on the notion that the US wields a strikingly different kind of power because it lacks overseas possessions. Indeed