JARON LANIER

YOU ARE NOT A GADGET

A MANIFESTO

It's early in the twenty-first century, and that means that these words will mostly be read by nonpersons – automatons or numb mobs composed of people who are no longer acting as individuals. The words will be minced into atomized search-engine keywords within industrial cloud computing facilities located in remote, often secret locations around the world. The vast fanning out of the fates of these words will take place almost entirely in the lifeless world of pure information. Real human eyes will read these words in only a tiny minority of the cases.

And yet it is you, the person, the rarity among my readers, I hope to reach.

The words in this book are written for people, not computers.

I want to say: You have to be somebody before you can share yourself.
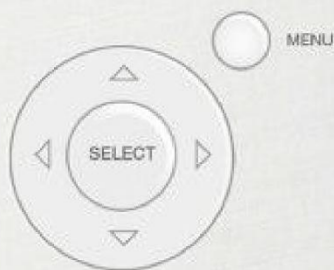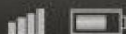
you are not a gadget

A MANIFESTO

JARON LANIER

ALLEN LANE
an imprint of
PENGUIN BOOKS

contents

Preface

IT'S EARLY in the twenty-first century, and that means that these words will mostly be read by nonpersons—automatons or numb mobs composed of people who are no longer acting as individuals. The words will be minced into atomized search-engine keywords within industrial cloud computing facilities located in remote, often secret locations around the world. They will be copied millions of times by algorithms designed to send an advertisement to some person somewhere who happens to resonate with some fragment of what I say. They will be scanned, rehashed, and misrepresented by crowds of quick and sloppy readers into wikis and automatically aggregated wireless text message streams.

Reactions will repeatedly degenerate into mindless chains of anonymous insults and inarticulate controversies. Algorithms will find correlations between those who read my words and their purchases, their romantic adventures, their debts, and, soon, their genes. Ultimately these words will contribute to the fortunes of those few who have been able to position themselves as lords of the computing clouds.

The vast fanning out of the fates of these words will take place almost entirely in the lifeless world of pure information. Real human eyes will read these words in only a tiny minority of the cases.

And yet it is you, the person, the rarity among my readers, I hope to reach.

The words in this book are written for people, not computers.

I want to say: You have to be somebody before you can share yourself.
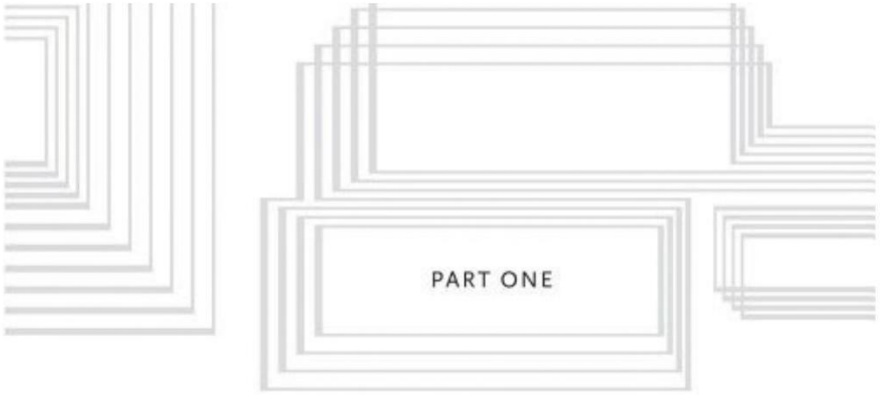
WHaT IS a PERSON?
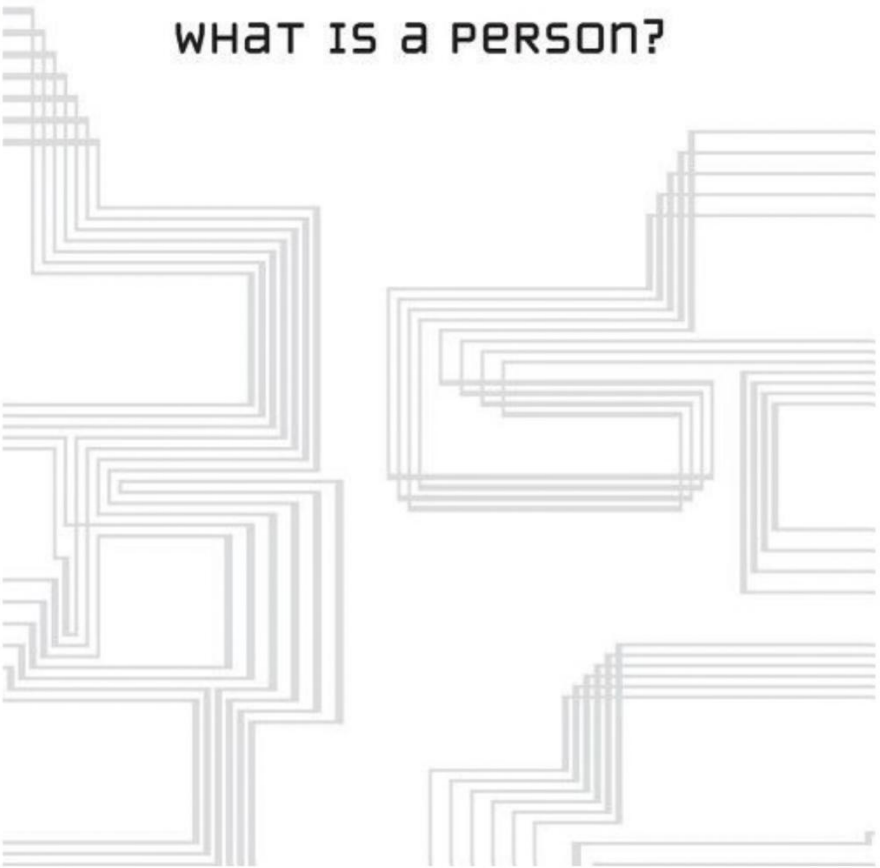
MISSING PERSONS

SOFTWARE EXPRESSES IDEAS about everything from the nature of a musical note to the nature of personhood. Software is also subject to an exceptionally rigid process of "lock-in." Therefore, ideas (in the present era, when human affairs are increasingly software driven) have become more subject to lock-in than in previous eras. Most of the ideas that have been locked in so far are not so bad, but some of the so-called web 2.0 ideas are stinkers, so we ought to reject them while we still can.

Speech is the mirror of the soul; as a man speaks, so is he.

PUBLILIUS SYRUS

Fragments Are Not People

Something started to go wrong with the digital revolution around the turn of the twenty-first century. The World Wide Web was flooded by a torrent of petty designs sometimes called web 2.0. This ideology promotes radical freedom on the surface of the web, but that freedom, ironically, is more for

machines than people. Nevertheless, it is sometimes referred to as "open culture."

Anonymous blog comments, vapid video pranks, and lightweight mashups may seem trivial and harmless, but as a whole, this widespread practice of fragmentary, impersonal communication has demeaned interpersonal interaction.

Communication is now often experienced as a superhuman phenomenon that towers above individuals. A new generation has come of age with a reduced expectation of what a person can be, and of who each person might become.

The Most Important Thing About a Technology Is How It Changes People

When I work with experimental digital gadgets, like new variations on virtual reality, in a lab environment, I am always reminded of how small changes in the details of a digital design can have profound unforeseen effects on the experiences of the humans who are playing with it. The slightest change in something as seemingly trivial as the ease of use of a button can sometimes completely alter behavior patterns.

For instance, Stanford University researcher Jeremy Bailenson has demonstrated that changing the height of one's avatar in immersive virtual reality transforms self-esteem and social self-perception. Technologies are extensions of ourselves, and, like the avatars in Jeremy's lab, our identities can be shifted by the quirks of gadgets. It is impossible to work with information technology without also engaging in social engineering.

One might ask, "If I am blogging, twittering, and wikiing a lot, how does that change who I am?" or "If the 'hive mind' is my audience, who am I?" We inventors of digital technologies

A single person, Tim Berners-Lee, came to invent the particular design of today's web. The web as it was introduced was minimalist, in that it assumed just about as little as possible about what a web page would be like. It was also open, in that no page was preferred by the architecture over another, and all pages were accessible to all. It also emphasized responsibility, because only the owner of a website was able to make sure that their site was available to be visited.

Berners-Lee's initial motivation was to serve a community of physicists, not the whole world. Even so, the atmosphere in which the design of the web was embraced by early adopters was influenced by idealistic discussions. In the period before the web was born, the ideas in play were radically optimistic and gained traction in the community, and then in the world at large.

Since we make up so much from scratch when we build information technologies, how do we think about which ones are best? With the kind of radical freedom we find in digital systems comes a disorienting moral challenge. We make it all up—so what shall we make up? Alas, that dilemma—of having so much freedom—is chimerical.

As a program grows in size and complexity, the software can become a cruel maze. When other programmers get involved, it can feel like a labyrinth. If you are clever enough, you can write any small program from scratch, but it takes a huge amount of effort (and more than a little luck) to successfully modify a large program, especially if other programs are already depending on it. Even the best software development groups periodically find themselves caught in a swarm of bugs and design conundrums.

Little programs are delightful to write in isolation, but the process of maintaining large-scale software is always miserable. Because of this, digital technology tempts the

programmer's psyche into a kind of schizophrenia. There is constant confusion between real and ideal computers. Technologists wish every program behaved like a brand-new, playful little program, and will use any available psychological strategy to avoid thinking about computers realistically.

The brittle character of maturing computer programs can cause digital designs to get frozen into place by a process known as lock-in. This happens when many software programs are designed to work with an existing one. The process of significantly changing software in a situation in which a lot of other software is dependent on it is the hardest thing to do. So it almost never happens.

Occasionally, a Digital Eden Appears

One day in the early 1980s, a music synthesizer designer named Dave Smith casually made up a way to represent musical notes. It was called MIDI. His approach conceived of music from a keyboard player's point of view. MIDI was made of digital patterns that represented keyboard events like "key-down" and "key-up."

That meant it could not describe the curvy, transient expressions a singer or a saxophone player can produce. It could only describe the tile mosaic world of the keyboardist, not the watercolor world of the violin. But there was no reason for MIDI to be concerned with the whole of musical expression, since Dave only wanted to connect some synthesizers together so that he could have a larger palette of sounds while playing a single keyboard.

In spite of its limitations, MIDI became the standard scheme to represent music in software. Music programs and synthesizers were designed to work with it, and it quickly proved impractical to change or dispose of all that software

and hardware. MIDI became entrenched, and despite Herculean efforts to reform it on many occasions by a multi-decade-long parade of powerful international commercial, academic, and professional organizations, it remains so.

Standards and their inevitable lack of prescience posed a nuisance before computers, of course. Railroad gauges—the dimensions of the tracks—are one example. The London Tube was designed with narrow tracks and matching tunnels that, on several of the lines, cannot accommodate air-conditioning, because there is no room to ventilate the hot air from the trains. Thus, tens of thousands of modern-day residents in one of the world's richest cities must suffer a stifling commute because of an inflexible design decision made more than one hundred years ago.

But software is worse than railroads, because it must always adhere with absolute perfection to a boundlessly particular, arbitrary, tangled, intractable messiness. The engineering requirements are so stringent and perverse that adapting to shifting standards can be an endless struggle. So while lock-in may be a gangster in the world of railroads, it is an absolute tyrant in the digital world.

Life on the Curved Surface of Moore's Law

The fateful, unnerving aspect of information technology is that a particular design will occasionally happen to fill a niche and, once implemented, turn out to be unalterable. It becomes a permanent fixture from then on, even though a better design might just as well have taken its place before the moment of entrenchment. A mere annoyance then explodes into a cataclysmic challenge because the raw power of computers grows exponentially. In the world of computers, this is known as Moore's law.

Computers have gotten *millions* of times more powerful,

and immensely more common and more connected, since my career began—which was not so very long ago. It's as if you kneel to plant a seed of a tree and it grows so fast that it swallows your whole village before you can even rise to your feet.

So software presents what often feels like an unfair level of responsibility to technologists. Because computers are growing more powerful at an exponential rate, the designers and programmers of technology must be extremely careful when they make design choices. The consequences of tiny, initially inconsequential decisions often are amplified to become defining, unchangeable rules of our lives.

MIDI now exists in your phone and in billions of other devices. It is the lattice on which almost all the popular music you hear is built. Much of the sound around us—the ambient music and audio beeps, the ring-tones and alarms—are conceived in MIDI. The whole of the human auditory experience has become filled with discrete notes that fit in a grid.

Someday a digital design for describing speech, allowing computers to sound better than they do now when they speak to us, will get locked in. That design might then be adapted to music, and perhaps a more fluid and expressive sort of digital music will be developed. But even if that happens, a thousand years from now, when a descendant of ours is traveling at relativistic speeds to explore a new star system, she will probably be annoyed by some awful beepy MIDI-driven music to alert her that the antimatter filter needs to be recalibrated.

Lock-in Turns Thoughts into Facts

Before MIDI, a musical note was a bottomless idea that transcended absolute definition. It was a way for a musician to think, or a way to teach and document music. It was a

mental tool distinguishable from the music itself. Different people could make transcriptions of the same musical recording, for instance, and come up with slightly different scores.

After MIDI, a musical note was no longer just an idea, but a rigid, mandatory structure you couldn't avoid in the aspects of life that had gone digital. The process of lock-in is like a wave gradually washing over the rulebook of life, culling the ambiguities of flexible thoughts as more and more thought structures are solidified into effectively permanent reality.

We can compare lock-in to scientific method. The philosopher Karl Popper was correct when he claimed that science is a process that disqualifies thoughts as it proceeds— one can, for example, no longer reasonably believe in a flat Earth that sprang into being some thousands of years ago. Science removes ideas from play empirically, for good reason. Lock-in, however, removes design options based on what is easiest to program, what is politically feasible, what is fashionable, or what is created by chance.

Lock-in removes ideas that do not fit into the winning digital representation scheme, but it also reduces or narrows the ideas it immortalizes, by cutting away the unfathomable penumbra of meaning that distinguishes a word in natural language from a command in a computer program.

The criteria that guide science might be more admirable than those that guide lock-in, but unless we come up with an entirely different way to make software, further lock-ins are guaranteed. Scientific progress, by contrast, always requires determination and can stall because of politics or lack of funding or curiosity. An interesting challenge presents itself: How can a musician cherish the broader, less-defined concept of a note that preceded MIDI, while using MIDI all day long and interacting with other musicians through the filter of MIDI? Is it even worth trying? Should a digital artist just give

Indeed, today's computers are millions of times faster, and UNIX has become an ambient part of life. There are some reasonably expressive tools that have UNIX in them, so the speed increase has sufficed to compensate for UNIX's problems in some cases. But not all.

I have an iPhone in my pocket, and sure enough, the thing has what is essentially UNIX in it. An unnerving element of this gadget is that it is haunted by a weird set of unpredictable user interface delays. One's mind waits for the response to the press of a virtual button, but it doesn't come for a while. An odd tension builds during that moment, and easy intuition is replaced by nervousness. It is the ghost of UNIX, still refusing to accommodate the rhythms of my body and my mind, after all these years.

I'm not picking in particular on the iPhone (which I'll praise in another context later on). I could just as easily have chosen any contemporary personal computer. Windows isn't UNIX, but it does share UNIX's idea that a symbol is more important than the flow of time and the underlying continuity of experience.

The grudging relationship between UNIX and the temporal world in which the human body moves and the human mind thinks is a disappointing example of lock-in, but not a disastrous one. Maybe it will even help make it easier for people to appreciate the old-fashioned physical world, as virtual reality gets better. If so, it will have turned out to be a blessing in disguise.

Entrenched Software Philosophies Become Invisible Through Ubiquity

An even deeper locked-in idea is the notion of the file. Once upon a time, not too long ago, plenty of computer scientists thought the idea of the file was not so great.

The first design for something like the World Wide Web, Ted Nelson's Xanadu, conceived of one giant, global file, for instance. The first iteration of the Macintosh, which never shipped, didn't have files. Instead, the whole of a user's productivity accumulated in one big structure, sort of like a singular personal web page. Steve Jobs took the Mac project over from the fellow who started it, the late Jef Raskin, and soon files appeared.

UNIX had files; the Mac as it shipped had files; Windows had files. Files are now part of life; we teach the idea of a file to computer science students as if it were part of nature. In fact, our conception of files may be more persistent than our ideas about nature. I can imagine that someday physicists might tell us that it is time to stop believing in photons, because they have discovered a better way to think about light—but the file will likely live on.

The file is a set of philosophical ideas made into eternal flesh. The ideas expressed by the file include the notion that human expression comes in severable chunks that can be organized as leaves on an abstract tree— and that the chunks have versions and need to be matched to compatible applications.

What do files mean to the future of human expression? This is a harder question to answer than the question "How does the English language influence the thoughts of native English speakers?" At least you can compare English speakers to Chinese speakers, but files are universal. The idea of the file has become so big that we are unable to conceive of a frame large enough to fit around it in order to assess it empirically.

What Happened to Trains, Files, and Musical Notes Could Happen Soon to the Definition of a Human Being

It's worth trying to notice when philosophies are congealing into locked-in software. For instance, is pervasive anonymity or pseudonymity a good thing? It's an important question, because the corresponding philosophies of how humans can express meaning have been so ingrained into the interlocked software designs of the internet that we might never be able to fully get rid of them, or even remember that things could have been different.

We ought to at least try to avoid this particularly tricky example of impending lock-in. Lock-in makes us forget the lost freedoms we had in the digital past. That can make it harder to see the freedoms we have in the digital present. Fortunately, difficult as it is, we can still try to change some expressions of philosophy that are on the verge of becoming locked in place in the tools we use to understand one another and the world.

A Happy Surprise

The rise of the web was a rare instance when we learned new, positive information about human potential. Who would have guessed (at least at first) that millions of people would put so much effort into a project without the presence of advertising, commercial motive, threat of punishment, charismatic figures, identity politics, exploitation of the fear of death, or any of the other classic motivators of mankind. In vast numbers, people did something cooperatively, solely because it was a good idea, and it was beautiful.

Some of the more wild-eyed eccentrics in the digital world had guessed that it would happen—but even so it was a shock when it actually did come to pass. It turns out that even an optimistic, idealistic philosophy is realizable. Put a happy philosophy of life in software, and it might very well come true!

Technology Criticism Shouldn't Be Left to the Luddites

But not all surprises have been happy.

This digital revolutionary still believes in most of the lovely deep ideals that energized our work so many years ago. At the core was a sweet faith in human nature. If we empowered individuals, we believed, more good than harm would result.

The way the internet has gone sour since then is truly perverse. The central faith of the web's early design has been superseded by a different faith in the centrality of imaginary entities epitomized by the idea that the internet as a whole is coming alive and turning into a superhuman creature.

The designs guided by this new, perverse kind of faith put people back in the shadows. The fad for anonymity has undone the great opening-of-everyone's-windows of the 1990s. While that reversal has empowered sadists to a degree, the worst effect is a degradation of ordinary people.

Part of why this happened is that volunteerism proved to be an extremely powerful force in the first iteration of the web. When businesses rushed in to capitalize on what had happened, there was something of a problem, in that the content aspect of the web, the cultural side, was functioning rather well without a business plan.

Google came along with the idea of linking advertising and searching, but that business stayed out of the middle of what people actually did online. It had indirect effects, but not direct ones. The early waves of web activity were remarkably energetic and had a personal quality. People created personal "homepages," and each of them was different, and often strange. The web had flavor.

Entrepreneurs naturally sought to create products that would inspire demand (or at least hypothetical advertising opportunities that might someday compete with Google)

where there was no lack to be addressed and no need to be filled, other than greed. Google had discovered a new permanently entrenched niche enabled by the nature of digital technology. It turns out that the digital system of representing people and ads so they can be matched is like MIDI. It is an example of how digital technology can cause an explosive increase in the importance of the "network effect." Every element in the system—every computer, every person, every bit—comes to depend on relentlessly detailed adherence to a common standard, a common point of exchange.

Unlike MIDI, Google's secret software standard is hidden in its computer cloud[*] instead of being replicated in your pocket. Anyone who wants to place ads must use it, or be out in the cold, relegated to a tiny, irrelevant subculture, just as digital musicians must use MIDI in order to work together in the digital realm. In the case of Google, the monopoly is opaque and proprietary. (Sometimes locked-in digital niches are proprietary, and sometimes they aren't. The dynamics are the same in either case, though the commercial implications can be vastly different.)

There can be only one player occupying Google's persistent niche, so most of the competitive schemes that came along made no money. Behemoths like Facebook have changed the culture with commercial intent, but without, as of this time of writing, commercial achievement.[*]

In my view, there were a large number of ways that new commercial successes might have been realized, but the faith of the nerds guided entrepreneurs on a particular path. Voluntary productivity had to be commoditized, because the type of faith I'm criticizing thrives when you can pretend that computers do everything and people do nothing.

An endless series of gambits backed by gigantic investments encouraged young people entering the online

conservative who writes for journals like *Commentary* and teaches at Yale. And yet I find inspiration in the work of them both.

Trap for a Tribe

The intentions of the cybernetic totalist tribe are good. They are simply following a path that was blazed in earlier times by well-meaning Freudians and Marxists—and I don't mean that in a pejorative way. I'm thinking of the earliest incarnations of Marxism, for instance, before Stalinism and Maoism killed millions.

Movements associated with Freud and Marx both claimed foundations in rationality and the scientific understanding of the world. Both perceived themselves to be at war with the weird, manipulative fantasies of religions. And yet both invented their own fantasies that were just as weird.

The same thing is happening again. A self-proclaimed materialist movement that attempts to base itself on science starts to look like a religion rather quickly. It soon presents its own eschatology and its own revelations about what is really going on—portentous events that no one but the initiated can appreciate. The Singularity and the noosphere, the idea that a collective consciousness emerges from all the users on the web, echo Marxist social determinism and Freud's calculus of perversions. We rush ahead of skeptical, scientific inquiry at our peril, just like the Marxists and Freudians.

Premature mystery reducers are rent by schisms, just like Marxists and Freudians always were. They find it incredible that I perceive a commonality in the membership of the tribe. To them, the systems Linux and UNIX are completely different, for instance, while to me they are coincident dots on a vast canvas of possibilities, even if much of the canvas is

all but forgotten by now.

At any rate, the future of religion will be determined by the quirks of the software that gets locked in during the coming decades, just like the futures of musical notes and personhood.

Where We Are on the Journey

It's time to take stock. Something amazing happened with the introduction of the World Wide Web. A faith in human goodness was vindicated when a remarkably open and unstructured information tool was made available to large numbers of people. That openness can, at this point, be declared "locked in" to a significant degree. Hurray!

At the same time, some not-so-great ideas about life and meaning were also locked in, like MIDI's nuance-challenged conception of musical sound and UNIX's inability to cope with time as humans experience it.

These are acceptable costs, what I would call aesthetic losses. They are counterbalanced, however, by some aesthetic victories. The digital world looks better than it sounds because a community of digital activists, including folks from Xerox Parc (especially Alan Kay), Apple, Adobe, and the academic world (especially Stanford's Don Knuth) fought the good fight to save us from the rigidly ugly fonts and other visual elements we'd have been stuck with otherwise.

Then there are those recently conceived elements of the future of human experience, like the already locked-in idea of the file, that are as fundamental as the air we breathe. The file will henceforth be one of the basic underlying elements of the human story, like genes. We will never know what that means, or what alternatives might have meant.

On balance, we've done wonderfully well! But the challenge on the table now is unlike previous ones. The new

designs on the verge of being locked in, the web 2.0 designs, actively demand that people define themselves downward. It's one thing to launch a limited conception of music or time into the contest for what philosophical idea will be locked in. It is another to do that with the very idea of what it is to be a person.

Why It Matters

If you feel fine using the tools you use, who am I to tell you that there is something wrong with what you are doing? But consider these points:

- Emphasizing the crowd means deemphasizing individual humans in the design of society, and when you ask people not to be people, they revert to bad moblike behaviors. This leads not only to empowered trolls, but to a generally unfriendly and unconstructive online world.

- Finance was transformed by computing clouds. Success in finance became increasingly about manipulating the cloud at the expense of sound financial principles.

- There are proposals to transform the conduct of science along similar lines. Scientists would then understand less of what they do.

- Pop culture has entered into a nostalgic malaise. Online culture is dominated by trivial mashups of the culture that existed before the onset of mashups, and by fandom responding to the dwindling outposts of centralized mass media. It is a culture of reaction without action.

- Spirituality is committing suicide. Consciousness is

attempting to will itself out of existence.

It might seem as though I'm assembling a catalog of every possible thing that could go wrong with the future of culture as changed by technology, but that is not the case. All of these examples are really just different aspects of one singular, big mistake.

The deep meaning of personhood is being reduced by illusions of bits. Since people will be inexorably connecting to one another through computers from here on out, we must find an alternative.

Next to the many problems the world faces today, debates about online culture may not seem that pressing. We need to address global warming, shift to a new energy cycle, avoid wars of mass destruction, support aging populations, figure out how to benefit from open markets without being disastrously vulnerable to their failures, and take care of other basic business. But digital culture and related topics like the future of privacy and copyrights concern the society we'll have if we can survive these challenges.

We have to think about the digital layers we are laying down now in order to benefit future generations. We should be optimistic that civilization will survive this challenging century, and put some effort into creating the best possible world for those who will inherit our efforts.

Every save-the-world cause has a list of suggestions for "what each of us can do": bike to work, recycle, and so on.

I can propose such a list related to the problems I'm talking about:

- Don't post anonymously unless you really might be in danger.

- If you put effort into Wikipedia articles, put even more effort into using your personal voice and expression outside of the wiki to help attract people who don't yet realize that they are interested in the topics you contributed to.

- Create a website that expresses something about who you are that won't fit into the template available to you on a social networking site.

- Post a video once in a while that took you one hundred times more time to create than it takes to view.

- Write a blog post that took weeks of reflection before you heard the inner voice that needed to come out.

- If you are twittering, innovate in order to find a way to describe your internal state instead of trivial external events, to avoid the creeping danger of believing that objectively described events define you, as they would define a machine.

These are some of the things you can do to be a person instead of a source of fragments to be exploited by others.

There are aspects to all these software designs that could be retained more humanistically. A design that shares Twitter's feature of providing ambient continuous contact between people could perhaps drop Twitter's adoration of fragments. We don't really know, because it is an unexplored design space.

As long as you are not defined by software, you are helping to broaden the identity of the ideas that will get locked in for future generations. In most arenas of human

through immersion is that I will direct a sustained stream of negativity onto the ideas I am criticizing. Readers, be assured that the negativity eventually tapers off, and that the last few chapters are optimistic in tone.

an apocalypse of
self-abdication

THE IDEAS THAT I hope will not be locked in rest on a philosophical foundation that I sometimes call cybernetic totalism. It applies metaphors from certain strains of computer science to people and the rest of reality. Pragmatic objections to this philosophy are presented.

What Do You Do When the Techies Are Crazier Than the Luddites?

The Singularity is an apocalyptic idea originally proposed by John von Neumann, one of the inventors of digital computation, and elucidated by figures such as Vernor Vinge and Ray Kurzweil.

There are many versions of the fantasy of the Singularity. Here's the one Marvin Minsky used to tell over the dinner table in the early 1980s: One day soon, maybe twenty or thirty years into the twenty-first century, computers and robots will be able to construct copies of themselves, and these copies will be a little better than the originals because of intelligent software. The second generation of robots will then make a

third, but it will take less time, because of the improvements over the first generation.

The process will repeat. Successive generations will be ever smarter and will appear ever faster. People might think they're in control, until one fine day the rate of robot improvement ramps up so quickly that superintelligent robots will suddenly rule the Earth.

In some versions of the story, the robots are imagined to be microscopic, forming a "gray goo" that eats the Earth; or else the internet itself comes alive and rallies all the net-connected machines into an army to control the affairs of the planet. Humans might then enjoy immortality within virtual reality, because the global brain would be so huge that it would be absolutely easy—a no-brainer, if you will—for it to host all our consciousnesses for eternity.

The coming Singularity is a popular belief in the society of technologists. Singularity books are as common in a computer science department as Rapture images are in an evangelical bookstore.

(Just in case you are not familiar with the Rapture, it is a colorful belief in American evangelical culture about the Christian apocalypse. When I was growing up in rural New Mexico, Rapture paintings would often be found in places like gas stations or hardware stores. They would usually include cars crashing into each other because the virtuous drivers had suddenly disappeared, having been called to heaven just before the onset of hell on Earth. The immensely popular *Left Behind* novels also describe this scenario.)

There might be some truth to the ideas associated with the Singularity at the very largest scale of reality. It might be true that on some vast cosmic basis, higher and higher forms of consciousness inevitably arise, until the whole universe becomes a brain, or something along those lines. Even at much smaller scales of millions or even thousands of years, it

is more exciting to imagine humanity evolving into a more wonderful state than we can presently articulate. The only alternatives would be extinction or stodgy stasis, which would be a little disappointing and sad, so let us hope for transcendence of the human condition, as we now understand it.

The difference between sanity and fanaticism is found in how well the believer can avoid confusing consequential differences in timing. If you believe the Rapture is imminent, fixing the problems of this life might not be your greatest priority. You might even be eager to embrace wars and tolerate poverty and disease in others to bring about the conditions that could prod the Rapture into being. In the same way, if you believe the Singularity is coming soon, you might cease to design technology to serve humans, and prepare instead for the grand events it will bring.

But in either case, the rest of us would never know if you had been right. Technology working well to improve the human condition is detectable, and you can see that possibility portrayed in optimistic science fiction like *Star Trek.*

The Singularity, however, would involve people dying in the flesh and being uploaded into a computer and remaining conscious, or people simply being annihilated in an imperceptible instant before a new super-consciousness takes over the Earth. The Rapture and the Singularity share one thing in common: they can never be verified by the living.

You Need Culture to Even Perceive Information Technology

Ever more extreme claims are routinely promoted in the new digital climate. Bits are presented as if they were alive, while humans are transient fragments. Real people must have left

all those anonymous comments on blogs and video clips, but who knows where they are now, or if they are dead? The digital hive is growing at the expense of individuality.

Kevin Kelly says that we don't need authors anymore, that all the ideas of the world, all the fragments that used to be assembled into coherent books by identifiable authors, can be combined into one single, global book. *Wired* editor Chris Anderson proposes that science should no longer seek theories that scientists can understand, because the digital cloud will understand them better anyway.[*]

Antihuman rhetoric is fascinating in the same way that self-destruction is fascinating: it offends us, but we cannot look away.

The antihuman approach to computation is one of the most baseless ideas in human history. A computer isn't even there unless a person experiences it. There will be a warm mass of patterned silicon with electricity coursing through it, but the bits don't mean anything without a cultured person to interpret them.

This is not solipsism. You can believe that your mind makes up the world, but a bullet will still kill you. A virtual bullet, however, doesn't even exist unless there is a person to recognize it as a representation of a bullet. Guns are real in a way that computers are not.

Making People Obsolete So That Computers Seem More Advanced

Many of today's Silicon Valley intellectuals seem to have embraced what used to be speculations as certainties, without the spirit of unbounded curiosity that originally gave rise to them. Ideas that were once tucked away in the obscure world of artificial intelligence labs have gone main-stream in tech culture. The first tenet of this new culture is that all of

brick is prodded to fall, the energy is revealed. That is only possible because it was lifted into place at some point in the past.

In the same way, stored information might cause experience to be revealed if it is prodded in the right way. A file on a hard disk does indeed contain information of the kind that objectively exists. The fact that the bits are discernible instead of being scrambled into mush—the way heat scrambles things—is what makes them bits.

But if the bits can potentially mean something to someone, they can only do so if they are experienced. When that happens, a commonality of culture is enacted between the storer and the retriever of the bits. Experience is the only process that can de-alienate information.

Information of the kind that purportedly wants to be free is nothing but a shadow of our own minds, and wants nothing on its own. It will not suffer if it doesn't get what it wants.

But if you want to make the transition from the old religion, where you hope God will give you an afterlife, to the new religion, where you hope to become immortal by getting uploaded into a computer, then you have to believe information is real and alive. So for you, it will be important to redesign human institutions like art, the economy, and the law to reinforce the perception that information is alive. You demand that the rest of us live in your new conception of a state religion. You need us to deify information to reinforce your faith.

The Apple Falls Again

It's a mistake with a remarkable origin. Alan Turing articulated it, just before his suicide.

Turing's suicide is a touchy subject in computer science circles. There's an aversion to talking about it much, because

we don't want our founding father to seem like a tabloid celebrity, and we don't want his memory trivialized by the sensational aspects of his death.

The legacy of Turing the mathematician rises above any possible sensationalism. His contributions were supremely elegant and foundational. He gifted us with wild leaps of invention, including much of the mathematical underpinnings of digital computation. The highest award in computer science, our Nobel Prize, is named in his honor.

Turing the cultural figure must be acknowledged, however. The first thing to understand is that he was one of the great heroes of World War II. He was the first "cracker," a person who uses computers to defeat an enemy's security measures. He applied one of the first computers to break a Nazi secret code, called Enigma, which Nazi mathematicians had believed was unbreakable. Enigma was decoded by the Nazis in the field using a mechanical device about the size of a cigar box. Turing reconceived it as a pattern of bits that could be analyzed in a computer, and cracked it wide open. Who knows what world we would be living in today if Turing had not succeeded?

The second thing to know about Turing is that he was gay at a time when it was illegal to be gay. British authorities, thinking they were doing the most compassionate thing, coerced him into a quack medical treatment that was supposed to correct his homosexuality. It consisted, bizarrely, of massive infusions of female hormones.

In order to understand how someone could have come up with that plan, you have to remember that before computers came along, the steam engine was a preferred metaphor for understanding human nature. All that sexual pressure was building up and causing the machine to malfunction, so the opposite essence, the female kind, ought to balance it out and reduce the pressure. This story should serve as a cautionary

tale. The common use of computers, as we understand them today, as sources for models and metaphors of ourselves is probably about as reliable as the use of the steam engine was back then.

Turing developed breasts and other female characteristics and became terribly depressed. He committed suicide by lacing an apple with cyanide in his lab and eating it. Shortly before his death, he presented the world with a spiritual idea, which must be evaluated separately from his technical achievements. This is the famous Turing test. It is extremely rare for a genuinely new spiritual idea to appear, and it is yet another example of Turing's genius that he came up with one.

Turing presented his new offering in the form of a thought experiment, based on a popular Victorian parlor game. A man and a woman hide, and a judge is asked to determine which is which by relying only on the texts of notes passed back and forth.

Turing replaced the woman with a computer. Can the judge tell which is the man? If not, is the computer conscious? Intelligent? Does it deserve equal rights?

It's impossible for us to know what role the torture Turing was enduring at the time played in his formulation of the test. But it is undeniable that one of the key figures in the defeat of fascism was destroyed, by our side, after the war, because he was gay. No wonder his imagination pondered the rights of strange creatures.

When Turing died, software was still in such an early state that no one knew what a mess it would inevitably become as it grew. Turing imagined a pristine, crystalline form of existence in the digital realm, and I can imagine it might have been a comfort to imagine a form of life apart from the torments of the body and the politics of sexuality. It's notable that it is the woman who is replaced by the

computer, and that Turing's suicide echoes Eve's fall.

The Turing Test Cuts Both Ways

Whatever the motivation, Turing authored the first trope to support the idea that bits can be alive on their own, independent of human observers. This idea has since appeared in a thousand guises, from artificial intelligence to the hive mind, not to mention many overhyped Silicon Valley start-ups.

It seems to me, however, that the Turing test has been poorly interpreted by generations of technologists. It is usually presented to support the idea that machines can attain whatever quality it is that gives people consciousness. After all, if a machine fooled you into believing it was conscious, it would be bigoted for you to still claim it was not.

What the test really tells us, however, even if it's not necessarily what Turing hoped it would say, is that machine intelligence can only be known in a relative sense, in the eyes of a human beholder.[*]

The AI way of thinking is central to the ideas I'm criticizing in this book. If a machine can be conscious, then the computing cloud is going to be a better and far more capacious consciousness than is found in an individual person. If you believe this, then working for the benefit of the cloud over individual people puts you on the side of the angels.

Historian George Dyson suggests that Turing might have sided *against* the cybernetic totalists. For instance, here is an excerpt from a paper Turing wrote in 1939, titled "Systems of Logic Based on Ordinals": "We have been trying to see how far it is possible to eliminate intuition, and leave only ingenuity. We do not mind how much ingenuity is required, and therefore assume it to be available in unlimited supply."

The implication seems to be that we are wrong to imagine that ingenuity can be infinite, even with computing clouds, so therefore intuition will never be made obsolete.

Turing's 1950 paper on the test includes this extraordinary passage: "In attempting to construct such machines we should not be irreverently usurping His power of creating souls, any more than we are in the procreation of children: rather we are, in either case, instruments of His will providing mansions for the souls that He creates."

But the Turing test cuts both ways. You can't tell if a machine has gotten smarter or if you've just lowered your own standards of intelligence to such a degree that the machine seems smart. If you can have a conversation with a simulated person presented by an AI program, can you tell how far you've let your sense of personhood degrade in order to make the illusion work for you?

People degrade themselves in order to make machines seem smart all the time. Before the crash, bankers believed in supposedly intelligent algorithms that could calculate credit risks before making bad loans. We ask teachers to teach to standardized tests so a student will look good to an algorithm. We have repeatedly demonstrated our species' bottomless ability to lower our standards to make information technology look good. Every instance of intelligence in a machine is ambiguous.

The same ambiguity that motivated dubious academic AI projects in the past has been repackaged as mass culture today. Did that search engine really know what you want, or are you playing along, lowering your standards to make it seem clever? While it's to be expected that the human perspective will be changed by encounters with profound new technologies, the exercise of treating machine intelligence as real requires people to reduce their mooring to reality.