Chapter 9
The Importance of Ted's Vision

Belinda Barnet

Sometimes a journey makes itself necessary, as Anne Carson puts it in *Autobiography of Red* [3, p. 46]. For Nelson it has been a long journey, at times a very difficult one, but it has been necessary. Necessary for him personally—as he puts it in *Possiplex*, "I have no alternative but to go on. Like Shackleton of Antarctica I find myself enmeshed in a harsh duty that was not the original plan… I will fight for it to my last breath" [10, p. 339]. He has survived the journey so far, his ideals held high above the mud. But as the *Intertwingled* conference has demonstrated, as the people who have spoken here have demonstrated, the journey has also been necessary for the computing world.

Nelson's vision is, I think, the most important vision in the history of computing. That's a tall call, so I'll need to explain my reasoning. In my book *Memory Machines* [1], I argue that the *idea* of a universal digital publishing system, an "open hypermedia" system, originated with Nelson. Bush's vision, though it was about connected items, was analogue. Engelbart's vision was also profoundly important. He was first to build many of the things we take for granted in modern computing (and I don't mean the mouse; I mean the whole concept of real-time, networked, interactive computing). His contributions would take hours to elaborate. But he was not thinking about hypertext as a domestic thing–something your mom and pop would consume over latte and a cheeseburger. Nelson saw that hypertext on a computer screen would spread around the globe like electronic kudzu vine. He wrote singing commercials for it; he even sketched a quirky interior for a Xanadu café.

What Nelson saw was an anarchic, global hypertext publishing system: a "digital repository scheme for world-wide electronic publishing" [9, p. 3/2]. While working at

B. Barnet (✉)
Media and Communications, Faculty of Health, Arts and Design, School of Arts,
Social Sciences and Humanities, Department of Media and Communication,
Swinburne University of Technology, Hawthorn, Australia
e-mail: BBarnet@groupwise.swin.edu.au

D.R. Dechow, D.C. Struppa (eds.), *Intertwingled*, History of Computing,
DOI 10.1007/978-3-319-16925-5_9

Harcourt in 1966 he dubbed the project Xanadu. Although he was not the first to build it, he predicted that hypertext would have *domestic penetration* before anyone else.

Nelson had a rich vision for what Xanadu should look like and the experience it should deliver to the public. The design went through many iterations, which are best explained in *Literary Machines*. They varied in their details, but all contained intercomparison or the re-use of elements (later dubbed transclusion) in some form. I'll talk about how these ideas evolved briefly here.

The vision started in 1960, and Nelson tends to get cranky at me when I begin in 1965 because that is the first published record (historians like bits of paper published at the time in addition to human memory). The designs, he told me in 2011, 'had been swirling in my head for 5 years' before then (Nelson 2011, pers. comm.). I will be starting in 1960 today, but I want to stress that this is based on Nelson's recollections in *Literary Machines,* on his autobiography *Possiplex,* and on our personal interviews. I'd love to go through Nelson's archives and find a term paper or file card from 1961. Nelson has shown me tantalizing video footage of a box of file cards from the 1960s. That's like watching porno for a media historian.

According to *Literary Machines* (and his autobiography), Nelson did a term writing project for the IBM 7090 in 1960. The IBM, the only computer at Harvard at the time, was stored in a big, air-conditioned room at the Smithsonian Observatory [9, p. 1/25]. That setup would have cost over two million dollars back then, and it wouldn't have had a screen.

Nelson believes he saw a screen in a manual at one point – as he told me in an email a couple of weeks ago, "I remember it very clearly. A round CRT and a flat desk surface, a light pen" (Nelson 2014, pers. comm.). This was apparently not Ivan Sutherland's Sketchpad, which was built in 1963, and Nelson has been unable to find the image again in the IBM 7090 manual). In *Possiplex*, he writes:

> A few words, a few pictures of people at computer screens, and the understanding that computer prices would fall—these gave me all I needed to know, a crystal seed from which to conjure a whole universe [10, p. 100].

In 1960 Nelson proposed a machine-language program to store documents in the computer, change them on a screen with various editorial operations, and print them out. This was no mere word processor, which in any case didn't exist at the time; Nelson envisioned the user would be able to *visually compare* alternative and prior versions of the same document on-screen. That's an important strand of Nelson's thinking that would continue on, comparing documents in parallel, or "intercomparison."

Intercomparison was a radical idea to dream up in 1960, when computers were for serious people (like engineers) to solve serious problems. At that time people were computing with cards. A "terminal" was a teletypewriter, used by an "operator", who put in the cards and tore off the printouts. In the mid 1960s you could, if you were rich enough, acquire a "glass teletype", a video-type display that acted like a teletype, but I don't think that would have been available in 1960.

Back then computers were million-dollar things that demanded refrigeration and respect. You "talked" to them with punch cards and got your answer on a printout. Nelson conjured a different universe from a crystal seed. Fourteen years later, in

Computer Lib, he would write, "if computers are the wave of the future, displays are the surfboards" [8, p. 22].

The second part of Nelson's vision took shape in the early 1960s, when there was "a lot of talk around Cambridge about Computer-Assisted Instruction, for which there was a lot of money" according to *Literary Machines* [9, p. 1/26]. At this point, Nelson's project was not so much a design, he stressed when I asked for records once again (I must annoy him with my desire for bits of paper), he said, "it was an idea that may have been on only one file card" (Nelson 2011).

At this stage he devised what he called "the thousand theories program," an explorable computer-assisted instruction program that would allow the user to study different subjects by taking different trajectories through a network of information. In *Literary Machines* he writes, "This idea rather quickly became what I would eventually call hypertext" [9, p. 1/26]. He thought of the system as incorporating many separate, modularized paragraphs, each with branching choices: writing as a graph, not a single line or sequence.

This led to another idea, which Nelson drafted as an academic paper while teaching sociology at Vassar College in 1965. That would become the transclusive relationship—and eventually ZigZag. It also got him published for the first time, which is why historians like to use that date.

In his 1965 paper for the ACM, "A File Structure for the Complex, the Changing and the Indeterminate" [5], Nelson proposed a display-based computer system that permitted linking between documents and the re-use of elements called Zippered Lists. Again, this was at a time when the whole idea of text on a screen was seen as a waste of processing power, let alone bizarre "nonsequential" text.

In 1965, unless you were working on Doug Engelbart's team or could afford a system with video-type display (Nelson reasons it "would cost less than a secretary" in his paper, at $37,000 in 1965 money, which would be a well-paid secretary), computers were expensive things with more important jobs to do. For most organisations they *still* didn't have screens.

I should also stress that, in 1965, text was not data—it was something academics and journalists manipulated with typewriters. As Doug Engelbart told me in 1999, the whole concept of a human being sitting in an interactive feedback loop with a computer, manipulating symbols on a screen, was foreign to most people in the 1960s.

> It was wacky even in the seventies, when we had it working–real hypermedia, real groupware working. (Engelbart 1999, personal communication)

As some eminent speakers discussed during the conference, this was a "paradigm problem" Nelson would later take on in *Computer Lib*. In fact, he did more than just take it on. He declared outright war on the established computer religion—particularly the idea that computers belong to a rarified priesthood. So although that declaration was best made with a raised fist (and capital letters) in *Computer Lib*, it had been brewing for him since 1960.

For the time being, in 1965, he ignored the dominant paradigm and published his zippered list design. That paper, when you think about what a computer was meant to be used for at the time, had chutzpah. He called the design the “Evolutionary List File” or ELF.

In this system, items in one sequence could become part of another like the teeth in a zipper—except the two sides of the zipper didn’t have to be in the same order. Versions of a document could be visually intercompared, and all items could be written or retrieved in a nonsequential fashion. Links could be made between large sections, small sections or single paragraphs. Most importantly, however, chronological stages and sections in a document could be retrieved and compared. You could trace the evolution of an idea.

In his autobiography, Nelson reflects that the ELF design was “strange and hard to understand. In fact, it was quite bad” [10, p. 151]. It did, however, include facilities to compare versions of a document and reuse elements from these versions. Both of these ideas would make their way into Xanadu in some form, but the zippered list in particular would eventuate in a “deliverable” 30 years later: ZigZag.

In that same 1965 paper, Nelson claimed that computers would eventually “do the dirty work of personal file and text handling” [5, p. 85]. Needless to say, personal computing has happened. The dirty work is now done by clunky programs like Microsoft Word—programs Nelson has a go at every now and then.

Not because he’s a cranky “‘one-note-samba’ fellow who can’t find another idea to push” and “should have moved on by now” as one programmer put in on a *Lambda the Ultimate* blog post (this programmer may not be happy that I’m including him here—[14]), but because the computing world could be completely different. That’s always been his message, even in the 1960s.

In 1967, having seen Engelbart’s NLS, Nelson went on to predict a networked structure of information that would “be read from an illuminated screen; the cathode-ray display; it will respond or branch upon actions by the user. It will be a succession of displays that come and go according to his actions” [6, p. 195]. That succession of displays that respond and branch has also happened. It’s hard to imagine a world without it, actually.

The idea that the Xanadu system should be an open or shared access publishing system, what this talk is really about, also started in the 1960s. Although in 1967 he envisioned a sort of “super Executive’s Console,” which was self-contained. He writes in *Literary Machines* that in 1967 “the idea of communicating between such consoles was beginning to get through to me, and the nagging issue of shared access began to grow on me” [9, p. 1/31].

It may have been growing on Nelson in 1967, but as I’ve said, the computing world really wasn’t about to swallow the idea of a global hypertext publishing system. Work had not even started on the ARPANET (though Ivan Sutherland and Bob Taylor had been thinking about it for some time). The computing establishment was still trying to grapple with the concept of a person sitting in front of a screen and exploring information in real-time after Doug’s mother of all demos in 1968. That demo took years—over 20 years—to filter through properly.

There was, however, an attempt to build part of Nelson’s vision at Brown University in 1967, and that resulted in a unique and historically important standalone system called the Hypertext Editing System. I’m not going to go into that here, however—this is Nelson's party and I don't want to poop it. If you are interested you can find it in my book [1], and the implementation notes are published in the Xuarchives [7]. I'll just say that it didn't happen the way he wanted it to happen.

Nelson had also met Engelbart and seen his landmark NLS system by 1967. The friendship would last until Engelbart's recent death. (Although Nelson had initially thought this meeting happened in 1967, he found some notes while writing *Possiplex* and realized it was in 1966. Those notes have since been lost!)

He thought Engelbart was warm and wonderful upon first meeting him, though he did not like the "hierarchical" structure of Engelbart's system. He went on to dedicate his book on Xanadu, *Literary Machines*, to this "visionary of The Augmentation of Human Intellect…and (what this book is largely about) THE TEXT LINK" ([9]—"1987 Dedication" included in 1993 edition).

Nelson and Engelbart were close: they understood each other on multiple levels. In his passionate eulogy for his friend, Nelson said, "I don't just feel like I've lost my best friend. I feel like I've lost my best planet" [11]. Although their visions were different, they shared some similar life experiences. Firstly, Engelbart and Nelson watched their ideas spread around the globe then re-emerge as someone else's interpretation, an *approximation* of a vision. That has upsides and downsides.

I remember putting it to Nelson once, in Melbourne, when he was getting a bit despondent about his life, that he has "inspired" people. He told me immediately that this was never his intention, and that the problem with inspiring people is that they then try to credit you with things you don't like (Nelson 2011, personal communication). He never set out to "inspire" people: he wanted to create an entirely new computer world. He wanted to actually build that world, not watch other versions of emerge. The fact that it has not yet been built drives him to continue.

Englebart and Nelson also lived through resistance: resistance to those original visions in the 1960s. I don't want to dwell on this, but I think it does need to be said. There have not always been conferences like this one held in Nelson's—or Engelbart's—honour.

Although Engelbart, as an engineer with a prestigious post at SRI, had more basis for conversation with the computing mainstream, what he was doing was not seen as "science" back then either. As the Head of Engineering at SRI told a young Bill Duvall (and Duvall later recounted to me), "You don't really think what they're doing up there is science, do you?" (Duvall 2011, personal communication).

That kind of resistance has dogged Nelson for many years. People didn't understand what he was going on about, and neither Ted nor his vision seemed to fit in any one nice explanatory box. As *The Economist* put it in 1986, "Boon or boondoggle, nobody is quite sure" (cited in [9], preface). Discussion of Xanadu still positions his work in left field.

As others have discussed, in 1974, in *Computer Lib*, he took his idea to the public, in the hope that he may have better luck there. He argued that computers are mere changeable devices for twiddling symbols that should power this new all-singing, all-dancing media experience for everyone. It was a rallying cry to that cause. Some of the conference attendees were actually around when that book was published, and they have unpacked its importance for personal computing. I just want to talk about hypertext.

One of the main things I want to emphasise is that for many years it was up to Nelson to promote the idea of a world-wide hypertext publishing system. It may be self-evident, even pedestrian today, but it certainly wasn't in the 1960s and 1970s—right into the 1980s people were still building workstation-based hypertext systems.

HyperCard, the elephant in the pre-Web hypertext room that introduced the concept of linking to the general public, was a stand-alone system. NoteCards, Guide, etc., none of these were globe-spanning open publishing systems. Even in the 1980s, it seemed wacky.

In a 1988 paper given at Oxford that Nelson provided to the participants of this conference (I hadn't seen it before) called "Hypertext: the Manifest Destiny of Literature" Nelson writes, hopeful as ever:

> The key problem is…to create a universal literary medium, an unbounded storage and delivery system as simple in concept as the book and library, unrestricted as to what screens you may see it on, unrestricted in its organization, unimpeachable in its authenticity, and as quickly available as a phone call. (Nelson 1988)

So it wasn't obvious even then, either, in 1988. It wasn't obvious that it was needed and that it was about to happen on a massive scale.

It was not until the Web that people really saw and understood, as Jay Bolter put it in our interview, that "the really interesting things happen when your links can cross from one computer to another," from one continent to another (Bolter, Jay David, Interview with the author, 2011). Then it all became rather obvious.

Nelson had been arguing for a global hypertext for a long time before the web. The thing is, he was not the first to build it, and that must have been deeply frustrating (particularly when they built it WRONG, he would add).

The Web is not Xanadu. It just looks a lot like what he'd been talking about all that time. It is also, crucially, what hypertext looks like to at least two billion people around the world. But Nelson won't "move on" and find "another idea to push" because he can see that it could be so much better.

In 1999, Nelson told me:

> The web is a universal, world-wide, anarchic publishing system. It completely vindicated my 35 years of saying that a universal, worldwide, anarchic publishing system would be of enormous human benefit. There is no question of whether it is of benefit. It just does it all wrong, that's all (Nelson 1999, pers. comm.).

Somewhere around 1993 Nelson found he no longer had to convince people that such a network was possible; you just had to switch on your machine to see that it was. His task changed to convincing people that Xanadu would be better. I should add here, if it isn't obvious by now, that I believe him.

But I've learnt some things from studying the history of hypertext. The first is that hypertext is not the Web. The Web is but one implementation of hypertext. People had been building and designing hypertext systems decades before the web arrived, and many of them did things the web just doesn't do. Xanadu in particular still has much to offer us. The Web is great in that it actually works, for most people, most of the time—and it has stayed the course for 25 years. It is not, however, the only way hypertext can be done.

So back to the title of my talk: the importance of a guiding vision.

The remarkable thing about Xanadu is that, despite countless setbacks, it refuses to die. Its logo is, appropriately enough, the Eternal Flaming X. Paisley and Butler (cited in [12, p. 262]) have noted that "scientists and technologists are guided by 'images of potentiality'—the untested theories, unanswered questions and unbuilt devices that they view as their agenda for 5 years, 10 years, and longer." Often accused of hand waving and lucid dreaming, Nelson's Xanadu has nonetheless become inherited vision.

But engineering discourse has always privileged prototypes over ideas – things that are concrete. Working prototypes, working algorithms, real deliverables that you can see in action. The same might be said for computing science. As Vincent Childress put it, the "main criteria applied to engineered technological solutions is that they work" [4], or more precisely, that they are seen to work.

For this reason, I think, people have been able to write off the Xanadu design as a pipe dream, particularly Gary Wolf in his scathing Wired article [13]. But the thing about the process of invention and innovation is that vision and prototype work in unison; they work together. Without an image of potentiality – the untested theory, the unbuilt device, the unanswered questions – innovation becomes a process of stabbing around in the dark.

You have to pick your vision, and your visionary, carefully though. You would want that vision to solve a problem that you think people are facing, for example (incidentally the reason why Engelbart had "flashes" of himself sitting at a Memex-like machine while driving home in 1950, a flash that changed the world). You would want that visionary to be right, at least some of the time, about what the future might hold. You would also want it to be technically feasible. And by that I mean, extrapolating from the devices, technologies and ways of doing things available to me right now, can I build it one day? Most importantly, you would also want it to be guiding you somewhere beneficial.

On that note, I'll leave the last piece of my talk to Engelbart. As Nelson put it in his eulogy, "No one ever had such a soaring view of human potential as Douglas Carl Engelbart—and he gave us wings to soar with him, though his mind flew on ahead, where few could see" (2013).

This is an excerpt from the 1995 Vannevar Bush Symposium. Engelbart was on the stage with Nelson, Alan Kay and Tim Berners-Lee. A member of the audience (referred to in the transcript as "Bob Franston"—this was probably Bob Frankston, co-creator of VisiCalc) asks a question of the panel, that starts with:

> I'm not going to defend Windows, but what I want to try and understand is why Windows is such a problem. If you have to change the world all at once and you can't coexist with what exists, you've got a problem…. Do you really feel like you have to change all the world at once?

Engelbart, who had in fact already changed the world by that point, though not all at once, answered him.

> The only thing I can say is that you have to pilot software, there has to be some sort of conscious pursuit of that future that you can't really guarantee is there, but [you need to] look… (Vannevar Bush Symposium [2]).

We have to consciously pursue a future that is beneficial; we have to pilot ourselves towards it; we have to look. There is no other way. I think that is what Ted has been doing since 1960.

References

1. Barnet B (2013) Memory machines: the evolution of hypertext. Anthem Press, London
2. Brown University/MIT Vannevar Bush Symposium (1995) Notes from the panels. http://cs.brown.edu/memex/Bush_Symposium_Panels.html
3. Carson A (1999) Autobiography of red: a novel in verse. Vintage Books, Random House, New York
4. Childress V (1998) Engineering problem solving for mathematics, science, and technology education. J Technol Educ 10(1). http://scholar.lib.vt.edu/ejournals/JTE/v10n1/childress.html
5. Nelson TH (1965) A file structure for the complex, the changing and the indeterminate. In: Proceedings of the ACM 20th national conference. ACM Press, New York, pp 84–100
6. Nelson TH (1967) Getting it out of our system. In: Schlechter G (ed) Information retrieval: a critical review. Thompson Books, Washington, DC, pp 191–210
7. Nelson TH (1968) Hypertext implementation notes, 6–10 March 1968. Xuarchives. http://xanadu.com/REF%20XUarchive%20SET%2003.11.06/hin68.tif
8. Nelson TH (1974) Computer lib: you can and must understand computers now/dream machines. Hugo's Book Service, Chicago
9. Nelson TH (1993) Literary machines. Mindful Press, Sausalito
10. Nelson TH (2010) Possiplex: movies, intellect, creative control, my computer life and the fight for civilization. n.p., available at Lulu*:* http://www.lulu.com/shop/ted-nelson/possiplex/paperback/product-14925222.html
11. Nelson TH (2013) Eulogy for Douglas Engelbart. Speech at Technology legend: honoring Douglas Engelbart, computer history museum, mountain view California, December 9th 2013. http://www.youtube.com/watch?v=FNCCkhADpiw
12. Smith LC (1991) Memex as an image of potentiality revisited. In: Nyce J, Kahn P (eds) From memex to hypertext: vannevar bush and the mind's machine. Academic, London, pp 261–286
13. Wolf G (1995) The curse of Xanadu. Wired 3(6). http://www.wired.com/wired/archive/3.06/xanadu.html
14. xeo_at_thermopylae (2004) Comment on a Lambda the ultimate blog post at Wed, 2004-09-01 18:52 http://lambda-the-ultimate.org/node/233#comment-1729