Can we call it Failure?

Digital Equipment Corporation and the minicomputer as an engineering success story

Introduction

Some of the most interesting engineering stories seem to revolve around failures: failures to heed warnings and concerns, failures to take proper measures against catastrophic outcomes, failures to make timely decisions, and failures to predict the future direction of technologies and markets. Failures of different types often result in dramatic, memorable, and sometimes tragic events, and prompt thorough investigations that can shed light on how specific failures occurred and how to prevent them in the future.

Over the course of the semester, failure has been a key part of many of our discussions. But how do we define the term and its implications? Is failure (or success) a useful category when considering large-scale engineering projects? How do we bound what we consider to be success or failure? In certain cases, failure seems clear. It would be hard to argue, for example, that the last Columbia mission was not a failure, given the disastrous outcome. In other cases, the lines have been more blurry. Packet switching technology, for example, succeeded in its original intended purpose of allowing messages to be transmitted and received across a network without disruption if part of that network was lost. But packet switching and the architecture it has engendered have failed in many ways as a secure and reliable communication system in the flood of unanticipated uses of the Internet. Nuclear power in the United States, while still providing as much as 20% of the nation's electricity, has failed to gain a greater share of the energy market because of regulatory limitations, safety concerns, and lack of sufficient solutions for waste storage.[1] Seldane succeeded in providing relief for millions of allergy sufferers, then failed when concerns about its interactions with other drugs became more widely known and better

[1] Estimates for nuclear's share of total electricity production are available from the Energy Information Administration (an arm of the Department of Energy): http://www.eia.doe.gov/oiaf/aeo/electricity.html.

understood by doctors and pharmacists (but also provided a platform for a more successful drug later).

The reasons for success and for failure naturally vary a great deal from case to case. It is clear from the cases we've studied over the course of the semester that, more often than not, drawing hard and fast lines between success and failure is no simple task. "Failure" as a broad categorical description is difficult to apply without so many caveats that it ceases to be meaningful. Additionally, when the argument is broadened beyond the technological and engineering decisions to consider the social and historical outcomes, the grey areas expand.

One example is the Concorde supersonic commercial transport program. The Concorde flew for decades and, except for the tragic crash outside Paris in July of 2000, with great technological success. As a proof-of-concept, the Concorde demonstrated that SST was possible, and that despite the $12,000 round-trip price tag, a small but reliable market did in fact exist. Still, with only a small number of planes flying on routes limited by the aircraft's fuel capacity, and environmental and social concerns regarding pollution and noise, Air France and British Airways were up against a substantial number of "cons" to go with the "pros" of continuing flight. In the final analysis, regardless of the safety improvements made after the crash, the Concorde program was deemed unfeasible, and eventually service was cancelled. Commercial jet technology improved greatly while the Concorde was flying, but took its cues not from the supersonic technology of the Concorde but from other areas, particularly those associated with fuel economy and passenger comfort. When the Concorde stopped flying, very little of its legacy could be seen in the technology of standard air travel, unlike the ways that early aircraft designs like the Boeing 707 are reflected in modern jet aircraft. While the Concorde was operational, it succeeded in what was meant to do – transport passengers at supersonic speeds – but failed to live up to its promise of transforming air travel and ushering in a new age of supersonic transport that would eventually displace the traditional jet as the jet itself had displaced the propeller-driven plane.

In the case of the Concorde, it clearly succeeded in some ways and failed in others, and would be easy to call a failure simply because it is no longer in service and no other like-minded supersonic transport system or technology is in line to replace it at the current time. But supersonic transport projects do receive a small degree of research and development attention from several nations, particularly Japan, and a more successful project may unfold in the future that owes a great deal to what the Concorde accomplished during its lifetime. In other words, the future may be a better determinant of the Concorde's success or failure than the present.

In this essay, I will discuss some of the history and technology of another "failed" technology vision, and specifically of the pioneering company that brought it about, Digital Equipment Corporation (DEC). The purpose of my discussion will be to explore the meaning of failure in the context of DEC's rise and fall as a business, and the technological and social outcomes of their tenure as one of the leading US computer manufacturers, and a leader in minicomputer and IT technology. A key part of the DEC story is how they responded to emerging technologies, and how the corporate culture helped to shape those responses over time. A reconfigured notion of failure based on the DEC case may be usefully applied to other cases, and help to generate ideas about how emerging technologies and technological trends can both complicate and enhance a company's business.

What is a minicomputer?

While DEC's innovations were not limited to technology, it is best known for developing and manufacturing technologies that were at the core of a paradigm shift in computing during the 1960s. That shift occurred around the minicomputer. Minicomputers were just as much an "attitude" or "state of mind" as they were a particular type of computer, and as such they enabled both technological and conceptual leaps that helped to usher in a wide range of IT technologies which are now commonplace in industrial, academic, and scientific computing, as well as personal computing.[2]

In chronological and technological terms, the minicomputer occupies the space in the spectrum of 20th century computing between the mainframe computer and the

---

[2] The minicomputer as an "attitude" was described T.G. Lewis, professor at Oregon State University, in 1977. See Lewis, T. G. (1977). "Evolving Minicomputer Architecture." SIGMINI Newsletter. 3: 6-44.

microcomputer. The minicomputer was, in many ways, a response to the unwieldy, expensive and monolithic mainframe computers that dominated the computing world from the early postwar years to the 1970s. Several large corporations, including Remington Rand, Control Data Corporation, and notably IBM, led mainframe computer development, producing machines that cost millions of dollars, required their own elaborate cooling systems and other infrastructure, and took up entire large rooms. Mainframe computers at this time relied on batching techniques for processing data and running programs. Users or groups of users would input data through an interface – usually punched card readers – that would be added to a queue and processed by the computer in the order the data was received. Users would often need to come back the next day or several days later to get the results, which were sometimes simply that the punched cards had errors or had not been processed properly, and the procedure had to be started over.

IBM dominated the mainframe computer market, controlling as much as 70% of it by 1970. Their equipment was leased rather than sold to customers, which enabled IBM to maintain tight control over their systems and how they were used, and ensure customer dependency on IBM for maintenance and repair as well as operation.[3] IBM even built special serial numbers into the processors that were required to run customized versions of software applications – when a customer wanted to purchase software, that software was compiled by IBM for their specific computer, ensuring that it couldn't be run anywhere else.[4] The architecture of IBM mainframes was closed in the sense that it was not possible for users to modify their mainframes, and it was difficult to understand exactly how the machines operated. IBM did not allow users to learn about the low-level functionality of the computer, restricting them instead to using basic functions and following protocols IBM had established.

In many ways, DEC could not have been a more different company than IBM. Founded by Ken Olsen and Harlan Anderson in 1957 with $70,000 in seed money, DEC released their first commercial computer, the PDP-1, in 1959. Olsen came to the

[3] See Paul Ceruzzi's excellent *A History of Modern Computing* for more detail on the operation of IBM mainframes and IBM's leasing policy. Ceruzzi, Paul. 1998. *A History of Modern Computing*. Cambridge, MIT Press.

[4] This is a story I heard from Frank that I have not seen verified elsewhere, but it seems to fit with IBM's general business model.

computer business through years of research and development work at MIT, where he had helped to design and operate one of the world's first computers based on transistor technology: the TX-0. The PDP-1 was the first commercial computer to be designed from the ground up with transistors, which at the time were brand-new technology and vastly more expensive than the vacuum tubes used in most computers. Ken Olsen recalled that one type of transistor they wanted to use for the PDP-1 cost $12.50 a piece (about $82 a piece in 2005 dollars). They bought 1000 of them, and before they were built into any products, the price went down to $8 a piece, which resulted in a $4000 inventory loss before they even built anything. Such are the dangers of innovation.[5]

But the PDP-1 was unique in other ways as well. DEC was a small company, and lacked the personnel necessary to send salespeople and technicians all over the world to run and maintain their computers, as IBM did. Additionally, Olsen wanted to bring some of the spirit of open cooperation that had guided his work at MIT to his new business. Computers, he thought, shouldn't be closed-off, monolithic devices, but rather the kinds of tools that allowed and encouraged users to get their hands dirty and figure out how they worked. In contrast to IBM, DEC not only sold rather than leased the PDP-1 to customers, but provided highly detailed technical manuals, printed on cheap newsprint and given away for free, and encouraged users to tinker with the machines.[6]

Though not as powerful as the IBM mainframes of the time, the PDP-1 had a few other features that made it attractive to users. It was much smaller than a standard mainframe, and did not require an entire large room for storage and operation. More importantly, perhaps, was its ability to process data in real-time rather than in batches, which gave users timely responses to their data processing and programming requests. Students at MIT "flocked" to the PDP-1 that Ken Olsen had donated to the school in the early 1960s, bypassing MIT's IBM mainframe, because of its open architecture and ability to process data in real-time. This open, user-centric model of computing exemplified by the PDP-1 is said to have inspired the hacker culture that emerged from MIT and elsewhere.[7]

---

[5] Olsen, Kenneth. 1982. "Digital Equipment Corporation: The First Twenty Five Years." Address to the Newcomen Society in North America, Boston, September 21, 1982.

[6] Ceruzzi, *A History of Modern Computing*, 129.

[7] Ibid.

The PDP-1 blazed a trail not only for DEC, but for computing in general. By using transistors, DEC took a significant chance on an expensive new technology that had not been put to commercial use in computers up to that point. In spite of their initial inventory loss (which illustrates how rapidly the landscape was changing for this technology), the success of the PDP-1 proved not only DEC's model of open architecture and real-time operation, but the viability of transistors in computing applications writ large. Though the number of PDP-1s sold was small compared with IBM's sales, its impact was significant.

DEC's most popular machine was the PDP-8, released in 1964. The PDP-8 was designed in the same spirit as the PDP-1 (open architecture, detailed manuals, etc) but was substantially cheaper, easier to program, and much smaller. The PDP-8 weighed approximately 250 pounds, took up only eight cubic feet of space, and could be set up on a desk. Several new technologies were designed and included as part of the PDP-8 system, including time-sharing abilities and a variety of advanced I/O devices. The primary input device was the ASR-33 Teletype machine, which conformed to the then-new ASCII standard, and allowed users to store programs on punched paper tape that could be read and sent to remote computers via telephone lines. The PDP-8's architecture was open enough, and it was small enough, to be built into other devices, which gave rise to the now-ubiquitous original equipment manufacturer (OEM) phenomenon. DEC eventually sold 50,000 of the machines.[8]

The PDP-8 was the first machine to be called a "minicomputer." Its small size and the time of its release put it into the same cultural milieu as other "mini" items that were becoming pervasive in American and European societies, such as the miniskirt and the Morris Mini Motor automobile. Utilizing smaller and faster transistors and other components, DEC had started a miniaturization trend throughout the industry. Though they controlled a great deal more of the computing marketplace, IBM did begin to see the writing on the wall, and released their on transistorized computer, the 7090. But unlike DEC's computers, its architecture was the same as its tube-driven counterpart, the 709.

[8] See Paul Ceruzzi, *A History of Modern Computing*, and Christian Wurster, *Computers: An Illustrated History*. Wurster, Christian. 2002. *Computers: An Illustrated History*. Hohenzollernring, Taschen.

The 1960s was a time of tremendous economic growth in the United States, as well as a decade which saw many large-scale technology and industry projects undertaken by the government and by various industries that required computing power and data storage. Many of the smaller businesses that were starting to take off during this period couldn't afford the high costs of an IBM mainframe, so the relatively cheap PDP-8 was a perfect choice.

DEC's rise and fall

The success of the PDP-8 and subsequent minicomputers such as the PDP-11 allowed DEC to grow rapidly. By 1977, the company had grown to 36,000 employees with sales over $1 billion. By 1982 the workforce had increased to 67,000 employees, and sales topped $4 billion. DEC's northeast business was spread out across nineteen separate buildings in and around their headquarters in Maynard, Massachusetts, and they used a fleet of helicopters and shuttle buses to facilitate employee movement back and forth between the different offices. During this period of unprecedented growth, DEC's managers – especially Ken Olsen – worked hard to maintain the "small business culture" that had defined their early years. Egalitarianism was the order of the day, and Olsen refused to allow executives and other high-powered employees to lord over the administrative staff, going as far as making a rule that senior executives were not allowed to take seats away from secretaries who had reserved places on helicopter flights between offices.[9]

Olsen had a strong vision for the company. After Harlan Anderson left in 1966, Olsen was the singular visionary force behind DEC, with computer designer Gordon Bell leading the engineering efforts. In his book about DEC's business and technical culture, *DEC is Dead, Long Live DEC,* Edgar Schein notes that there were three important themes that drove DEC culture during its formative years in the 1960s and 1970s. First, computing needs at that time, while being more and more influenced by consumer demand, were still largely driven by the massive defense industry in the United States. That industry required tremendous computing power, but also the real-time data

---

[9] Schein, Edgar. 2003. *DEC is Dead, Long Live DEC: The Lasting Legacy of the Digital Equipment Corporation*. San Francisco, Berrett-Koehler, 152-153.

processing that DEC's computers offered. Second, Olsen was greatly attuned to the needs of scientists, engineers, and students, who at that time made up the largest contingent of computer users, and whose needs for computing (particularly students) were round-the-clock. Time-sharing, remote access and real-time processing were thus key components of DEC's technology strategy as ways of fulfilling the needs of these users. Lastly, Olsen felt strongly that computing should be "fun, exciting, involving, and stimulating." This emphasis on the exciting attributes of computing was a substantial part of DEC's cultural identity.[10]

But with this identity came difficulty. Schein notes, as have other authors, that Olsen developed an early distaste for marketing, and hired young engineers who subscribed to his vision of providing support and technology to the scientific and engineering communities, reinforcing his bias towards those communities to DEC's detriment later. Additionally, DEC developed a business strategy during its later years that, ironically, came to mirror that of IBM. Though DEC technology followed emerging standards (such as ASCII) and introduced a variety of technologies that led to the formation of standard technologies (such as DECTape, which was a tape-storage drive developed for the PDP-8 that read the tape in both directions, not unlike the floppy disks that followed), they also grew into a full-solutions company. As they met with greater and greater success in their computer systems development, DEC expanded their hardware to include tape drives and other I/O devices, storage devices, teletype machines with expanding communication abilities, and eventually operating systems and software products, including databases, that ran only on their own hardware. When integrated circuits and semiconductors began to appear, DEC adopted them and started developing their own (such as the Alpha chip). The following graph illustrates the broad range of products and technologies DEC produced over its life span:

[10] Ibid., 38.

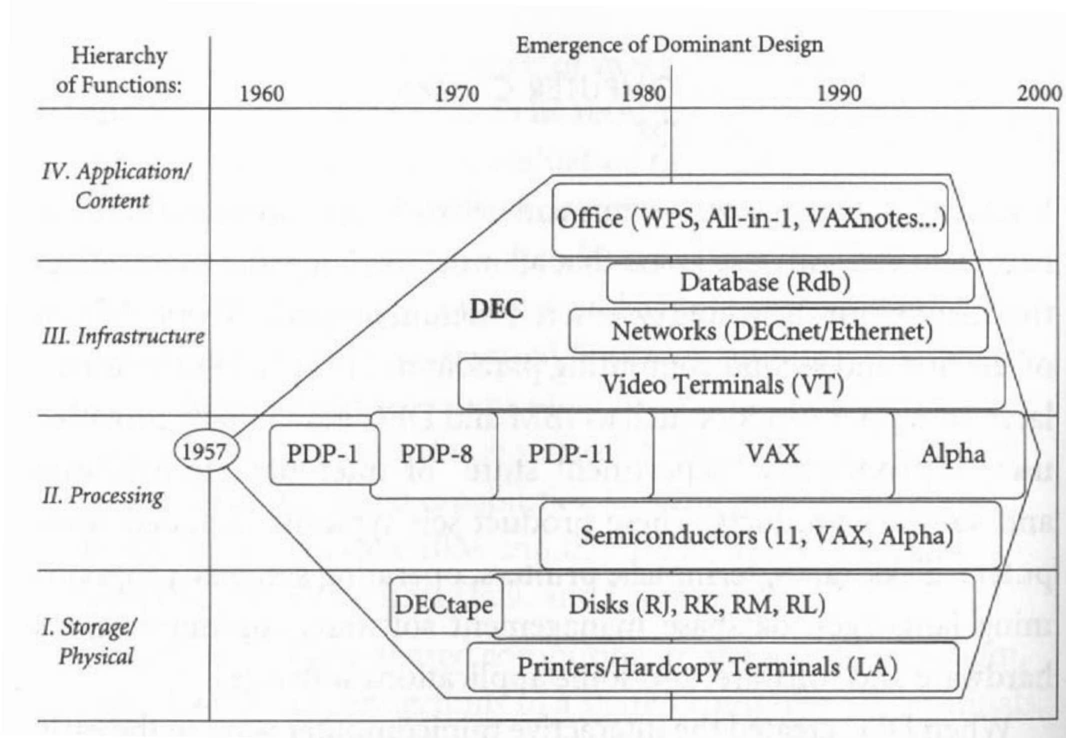

Figure 1 - from Schein, *DEC is Dead, Long Life DEC*, 136 .

Though DEC continued to succeed as its business and product line expanded, it was soon undermined by broader transitions in computing technology and the technology marketplace that Ken Olsen and other DEC leaders failed to anticipate or respond to adequately.

One of the most significant technological developments in modern computing was the invention of the integrated circuit. Though DEC used ICs in an upgraded version of the PDP-8 (the PDP-8/I) and in its successful and powerful PDP-11, released in 1970, its primary business and technology focus remained on institutional users. During the 1970s, decreasing prices and increasing power of key components, particularly ICs, helped to bring about the next stage in computer miniaturization: the microcomputer. The microcomputer just as much a "state of mind" as the minicomputer had been in the sense that it followed a path of miniaturization that the minicomputer had established. Though early microcomputers lacked the computing power of minicomputers and mainframes, they were inexpensive, accessible, easy to program, and fun to use.

Miniaturization also enabled ICs and eventually microprocessors to make up for technological limitations, such as word length, with speed and size, much as the original DEC minicomputers had done in the face of IBM mainframes. Intel released the world's first microprocessor, the 4004, which had 6,000 transistors and was advertised as a "computer on a chip."[11]

The microcomputer also led to the emergence of an entirely new community of users: the hobbyists. Hobbyists were the first to enthusiastically embrace the microcomputer upon the release of the Altair 8800, a kit-based microcomputer with an Intel microprocessor, in 1974. The Altair had neither a monitor nor a keyboard, but allowed users to input simple programs from a series of switches, and included input options with an ASR 33 teletype machine and other I/O and storage devices. Bill Gates and Paul Allen wrote the programming language BASIC for the Altair, which they made freely available prior to the founding of Microsoft. Though the microcomputer's value to the home user was not yet clear, what was clear (to some) was that a major change in how computing was conceived by users was underway.[12]

Though these developments did not immediately undermine DEC's position as a leader in computing technology, they did challenge the fundamental conception of computing held by Ken Olsen and DEC's other business leaders. Through the latter half of the 1970s and into the 1980s, microcomputers in the form of the now-ubiquitous PC made there way in to more and more business and institutional settings, as well as into users' homes. Their relatively low costs and ease of operation made them attractive to entire groups of users DEC had never taken very seriously, and although DEC continued to prosper into the 1990s, their lack of attention to the PC marketplace helped bring about a rapid decline in their business. Ironically, this pioneer in early emerging computing technology was unable to respond to that technology as the emergence continued.

What happened to DEC?

Many authors and pundits have speculated that DEC's downfall as a business was largely the fault of Ken Olsen, who held fast to outmoded ideas, lost control of his own

[11] Wurster, *Computing: an Illustrated History*, 133.
[12] Ibid.

company, and lacked the vision necessary to lead DEC into the future. To lay the blame at the feet of a single individual is both unfair and inaccurate, but it would also be a mistake to underestimate the influence Olsen had on the business and, ultimately, on the entire computing marketplace. Olsen's vision of computing, and the way he ran his company, had long-lasting and broad effects on the IT industry that persisted after DEC was purchased by Compaq in 1998. Many of the technical innovations brought about by DEC engineers were adopted by other technology companies, as were a variety of the cultural aspects of DEC's business model by a wide range of companies both inside and outside of the technology marketplace.

To gain some perspective on Olsen's point of view, it pays to look back to his most (in)famous – and often misconstrued – quotation. In a 1977 article in *Time* magazine, Olsen was quoted as saying, "there is no reason for any individual to have a computer in their home." This, of course, came after the revolution sparked by the Altair 8800, and in the same year as the release of Apple's wildly successful Apple II. Many have held up Olsen's comment as a example of why DEC failed as a business – he simply couldn't conceive of the world as we now know it. In some ways, this comment was indeed indicative of Olsen's point of view, which emphasized a scientific and engineering computing model that relied on time-sharing and networking rather than fully-capable computers in user's homes. Unfortunately, the true meaning of Olsen's comment is often lost in the ridicule. According to Schein, what Olsen himself claims to have meant was that the idea of computers in every home, running everyone's lives, was a bad one that should be avoided. This was the dominant imagined future at the time when Olsen entered the computing business. As Schein points out, computers were not seen as "benign and helpful" back then as they are today. It was not, in other words, intended as a comment against the PC so much as it was a comment against a certain conceptualization of the PC. On the contrary: making PCs, according to Olsen, was DEC's business.[13]

But Olsen's idea of what a PC was turned out to be different than how it was conceived by the greater marketplace. Olsen held fast to the idea that a "personal computer" was "one where a person interacts with the computer directly, where every

[13] Schein, *DEC is Dead, Long Live DEC*, 38-39.

move the person makes the computer reacts, and every move the computer makes the person reacts, which often leads to an intense, often exciting, relationship."[14] This definition left Olsen believing that the comparatively large, expensive, and difficult-to-program DEC computers were in fact PCs, while most others thought of the PC as the increasingly small and user-friendly desktop machines that IBM and others had started to manufacture in greater and greater numbers in the 1980s.

While DEC focused most of its business strategy from the 1970s into the 1980s on large-scale, full-solutions technologies, a variety of smaller, more nimble firms began to emerge offering specialized services in a small number of product categories. Schein refers to these companies as "category killers" that ate into DEC's business at a variety of points, as illustrated in the graph below:

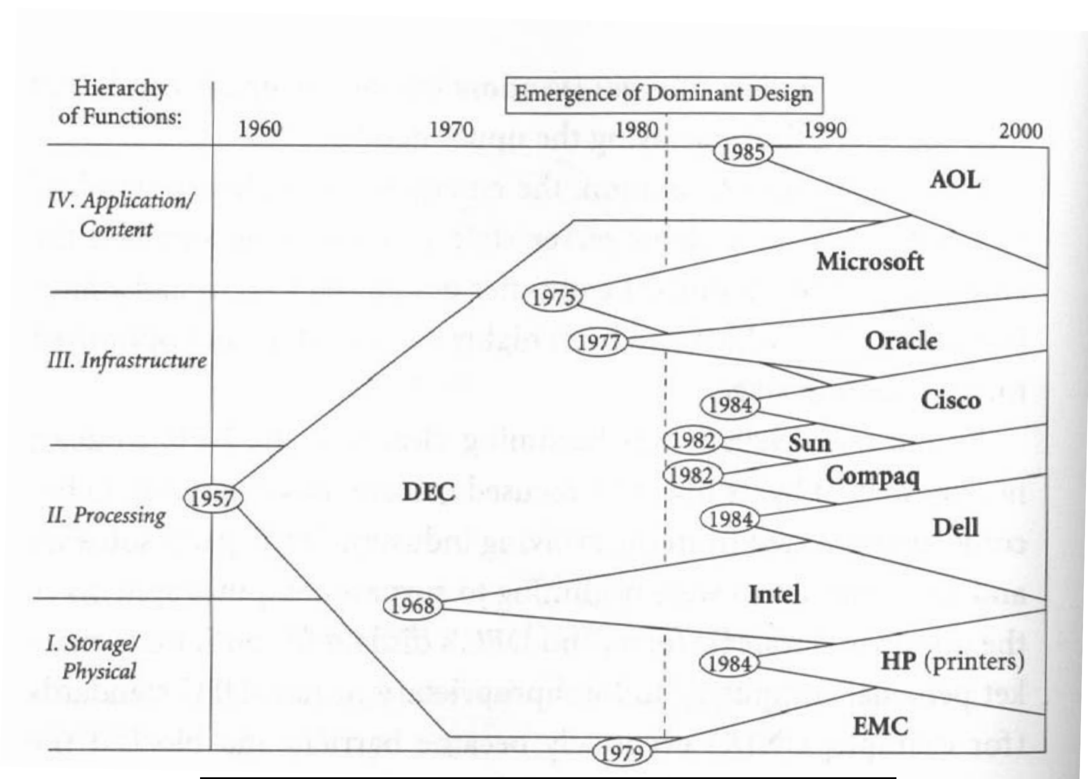

Figure 2 - from Schein, *DEC is Dead, Long Life DEC*, 140.

[14] Ibid., 40.

Products made by these smaller firms were often based upon technology standards or de facto cooperation (Microsoft software ran on Intel processors in computers made by Compaq and Dell, etc) while DEC attempted to remain a full-solutions vendor whose core products operated only with other DEC products, be them hardware or software. Even IBM reoriented their business strategy towards providing standard services and away from large-scale systems, and like many other businesses, focused more and more attention on the smaller-scale business and personal computing markets. Naturally, the competition among these smaller firms was fierce, but for DEC, their increasing power in the computing marketplace seriously undermined the foundation of their business.

But Olsen also found that the culture he had helped to create did not, in many ways, scale up very well. Among other things, Olsen had encouraged the autonomy of various units of DEC's business, in part so that different groups, and individual employees, would feel they had some stake in the success of the company as a whole. This method encouraged people to have new ideas and share them freely rather than needing to navigate a rigid hierarchy, and worked very well when DEC was a small company during the 1960s and early 1970s. The irony of this strategy was that as DEC grew, it became more and more difficult for Olsen to exert control in certain circumstances. Without a more well-established channel of decision-making, he often found himself battling his employees, and thus experienced some difficulty in establishing a unified strategy for the future direction of the company. At the same time, he maintained the final word on most projects, and could kill a project without consulting with the larger teams, or push projects in directions that didn't make sense to his employees (such as developing DEC's word processor, DECMATE, for the PDP-8 rather than the PDP-11, which, according to general manager Andy Knowles, effectively "dead-ended" the software by anchoring it to old technology).[15]

[15] Schein, *DEC is Dead, Long Live DEC*, 186.

Where DEC succeeded

As much as the culture of DEC may have contributed to its failure as a business, it was undeniably successful in terms of fostering an innovative environment that resulted in a wide range of influential and important technologies throughout DEC's lifetime – including the time up to its sale to Compaq. Bob Metcalfe, the inventor of Ethernet, wrote in a letter to Edgar Schein that "success has many fathers, but failure is an orphan. . . but I'm here to testify that we beneficiaries of modern information technology owe much, if not everything, to DEC."[16] He follows this bold statement with a description of numerous modern IT mainstays that originated out of DEC projects or were developed by former DEC employees, such as Microsoft's Windows NT operating system (the precursor to Windows 2000 and Windows XP, which Metcalfe suggests was essentially DEC's VMS operating system written to run on Intel processors and designed by DEC's operating system guru, David Cutler).[17] Employees who left DEC went on to other technology companies including Data General (founded by former DEC employees), SUN Microsystems, Microsoft, Adaptec, Autodesk, Marathon, California Micro Devices and others. Bob Supnik, a former DEC engineer, notes that in spite of DEC's unsuccessful response to the personal computer market, they developed the world's first microprocessor to run at greater than 100Mhz, and suggests that an entire generation of engineers and computer scientists cut their teeth on DEC technology in the 1970s and 1980s.[18] The now ubiquitous UNIX operating system was written on DEC computers, and the original ARPANET included more DEC computers than IBM computers. Later generation DEC VAX computers, running UNIX, helped to open the architecture of the ARPANET to create the familiar Internet of today.[19]

DEC culture influenced more than just technology – it had significant social impacts as well. Ken Olsen's vision of an egalitarian business model extended beyond a unique approach to engineering. DEC was particularly ahead of its time in terms of employee diversity and affirmative action, and had strong anti-discrimination policies

[16] Ibid., 261.
[17] Ibid., 262.
[18] Ibid., 263.
[19] Ceruzzi, *A History of Modern Computing*, 282-284.

that would be emulated by a variety of businesses later. DEC established support groups for gay and lesbian employees, and provided management and executive opportunities to African American employees at a time when this was almost unheard of in American business.[20] Though not specifically about technology, these business and cultural practices were important components of the overall engineering enterprise at DEC that resulted in so many innovative technologies. Egalitarianism and equal access fed back into DEC's approach to technology in interesting ways as the business grew and matured. As Bob Metcalfe recalled:

> In 1968, while still an MIT undergraduate, I wrote a paper for DEC's user group (DECUS) about my Project ASC – the application of small computers in education. DEC had lent me a PDP-8S for my work at MIT in teaching computers to high school students. Right afterward, my PDP-8S was stolen. Instead of making me pay them back the $30,000 a PDP-8S cost, DEC turned my misfortune into a promotion: "DEC has made the first computer small enough to be stolen."[21]

Employees who left DEC prior to its purchase by Compaq helped to develop now ubiquitous technologies at Oracle, Intel, Data General, Quantum, Tandem and elsewhere. Former DEC employees held a reunion in 2001 to reminisce about the time they spent there and what they accomplished. At least 200 people attended, many of whom had not worked for DEC for several years prior to its sale in 1998. Most of the people Schein interviewed for his book, while offering various explanations for why the company did not ultimately succeed, all seemed to have a unique appreciation for their time there and for what the business was able to accomplish both from cultural and technological standpoints.[22]

Ironically, Ken Olsen's beliefs about the uselessness of computers controlling everything in the home, and his attachment to the networked, time-sharing minicomputer-with-dumb-terminals paradigm, have proven to be more forward-thinking than many realized during DEC's heyday or after its collapse. The rise of the Internet validated the importance of networks and powerful servers, while businesses and educational institutions have turned their attention to a server/terminal model for computing if for no other reason than because of its cost effectiveness. MIT's Athena system runs on this paradigm, and Apple's OS X Server operating system allows Macintosh computers with

[20] Schein, *DEC is Dead, Long Live DEC*, 260.
[21] Ibid., 262.
[22] Ibid., 12-13.

no internal hard drives to boot up the OS from a server and create user accounts with storage options, run applications, browse the Internet and otherwise perform all the basic functions that a free-standing computer can without the added hassles associated with the maintenance of individual computers. Additionally, SUN Microsystem's JAVA programming language, originally released with a variety of speculation about its being built into every appliance in the home that would then be networked with every other appliance to create a fully computer-controlled environment, has never delivered on this promise. In some ways, Ken Olsen was right.

What can we learn from DEC and the minicomputer?

Taking into account the impact of DEC technology on the world of IT, it becomes more difficult to conceive of the company as a failure. Though Bob Metcalf's assertion that we owe "everything" about how modern IT works to DEC may be a bit over-the-top, it would be hard to argue against its influence. The technological legacies of DEC reach far and wide, though it may be difficult for a modern-day user of Windows XP to appreciate that the basis of its functionality was first conceived at a now-defunct company headquartered in an old mill in Maynard, Massachusetts.

The history of DEC and of the minicomputer broadly construed can teach us valuable lessons about how businesses and institutions respond to changing technologies, and how innovations in both technology and culture can be spread beyond the boundaries of a single company and even a single industry. But does the DEC story teach us something valuable as well about failure? Does it complicate what we mean by failure? How can we evaluate the decisions made by DEC in the context of its lasting influences?

Contrary to some popular assumptions, DEC can be viewed largely as a success, but in some ways also a victim of the success it helped create in the computing industry. One could even make the argument, as some former DEC employees have, that DEC did *not* fail as a business. They did not simply close up shop and disappear, but rather sold off different parts of their business to other companies. The degree to which those other companies integrated DEC's technology and culture into their own businesses is unclear, but given former employees own statements about the impact they believe DEC has had across the board, it is reasonable to assume integration and adoption of DEC technology and culture was significant.

It is also reasonable to say that many businesses – technology-focused and otherwise – have adopted many of the cultural elements that DEC pioneered (including the business that I used to work for). The idea of forming semi-autonomous teams to work on projects, and fostering a sense of personal responsibility on the part of employees for the success and stability of the company, are standard practices throughout the business world, in addition to the open employment and non-discrimination policies (some of which, granted, are now federally mandated). The monolithic and hierarchical business models practiced by large firms such as IBM during the period of DEC's rise have mostly fallen out of fashion in the technology world in favor of the more open and egalitarian model pioneered by DEC.

Still, DEC was unable to sustain itself when the PC began to dominate the computer market, and most historians and former employees agree that DEC's culture had a role in its demise as much as it played a part in its success. Edgar Schein believes that Ken Olsen's efforts at maintaining a "culture of innovation" were fundamentally incompatible with the ways the company grew, and the rate at which that growth took place. But difficulties associated with rapid and substantial growth are not uncommon in contemporary businesses, and given what happened to so many companies during the Internet boom and bust, one could argue that as easy as it is in some ways to point fingers at Ken Olsen, a huge number of other CEOs and their businesses have suffered similar fates. A more appropriate question than "how did Ken Olsen cause DEC to fail" might be "what is the role of CEOs and company visionaries in general in the success or failure of businesses?"

Rethinking the story of DEC and of the minicomputer in terms of success rather than failure can help broaden our view of engineering success and failure by, among other things, providing an interesting cultural perspective, and taking into account how technological innovation spreads throughout the marketplace. Though not all of the cases we've reviewed this semester would necessarily change dramatically by reconsidering them in this light, it is a worthwhile exercise nonetheless. Perhaps the most challenging case to think of as a success is, of course, the Columbia disaster. The destruction of the orbiter and NASA's response, particularly their latest ideas for the next generation space vehicles, suggest that not just Columbia but the Shuttle program as a whole has largely been a failure. The Shuttle was misconceived by politicians and oversold by NASA, and the destruction of Columbia was clearly the result not just of an isolated incident, but rather a tragic outcome of long-term and deeply rooted problems in NASA's engineering and management practices. The fact that NASA's new designs for future space vehicles look on the surface to be a lot more like the old vehicles of the Apollo generation than the Shuttle suggests that even NASA is trying to distance itself from the program both politically and technologically.

But the influence of DEC has, at least in part, emerged only after DEC disappeared as an independent business. Its history, especially given the growth of the Internet, shows it to have succeeded in a number of ways that perhaps were not clear to those considering its status in 1998. The Shuttle, and perhaps even some aspects of the Columbia disaster, may in the future prove to have positively impacted the technology of space travel in ways that are difficult to conceive of in the wake of its more immediate failures. As discussed in the introduction, a similar situation may emerge for SST as the technologies change and new players enter the market to attempt to provide what the Concorde and other projects could not. The DEC case provides scholars with an opportunity to challenge and broaden our ideas of success and failure, and reconceptualize engineering and business cases that may otherwise seem closed.