## The Rise and Fall of Silicon Graphics

or How a Rebellious Youth Briefly Conquered the World Apr 04, 2024 Abort Retry Fail

James Henry Clark was born on the 23rd of March in 1944 in Plainview, Texas. Clark’s family was far from wealthy. His father was fond of drinking and couldn’t keep a job. His mother worked at a local doctor’s office making about $225 per month (around $2605 in 2024). Clark’s parents divorced while Clark was still young, and while that salary may seem fine if low adjusted for inflation, Clark’s mother would only have received $175.50 ($2032) after income tax and social security tax, and it was the sole income for a woman and her three children. For himself, Clark was a bit rowdy. His high school highlights include setting off a smoke bomb on the band bus, smuggling a skunk into a school dance, telling his English teacher to go to Hell, drinking, and drag racing. Given the era, I imagine that the drinking was accompanied by chain smoking.

That times were different is… inadequate verbiage. For all the unruly behavior, Clark was only suspended from school twice. On his second suspension, young Clark decided he’d not be returning to school. He chose to join the US Navy and convinced his mother to sign the permission forms. Of course, this is Jim Clark, and the initial days of his naval career didn’t exactly go well. Clark had never taken a multiple choice test. He thought that for many questions more than one of the answers were at least partially true and therefore selected them. The officers in charge of test administration thought that Clark was attempting to fool the computer that checked the answers, and he was immediately sent out to sea with other delinquent recruits where he was given poor treatment, and rough and disgusting chores. The experience of Naval life lit a fire in Clark, and he chose to advance his station in life. He began learning about electronics, taking some general educational courses, and offering loans to other sailors at up to forty percent interest.

His first step was to get his General Education Diploma, which he did. He then enrolled at Tulane. Clark did well at Tulane but transferred to the University of New Orleans from which he received his BS and MA in Physics. He then attended the University of Utah where he earned his Ph.D. in computer science in 1974. From 1974 through 1978, Clark was employed as an assistant professor at UC Santa Cruz, but he left to become an associate professor at Stanford in 1979.

[[Jim Clark]] Early in his time at Stanford, Clark worked on a project with Xerox PARC with support from ARPA to develop three dimensional graphics. This led to the creation of the Geometry Engine. In “The Geometry Engine: A VLSI Geometry System for Graphics,” Clark also makes specific reference to Marc Hannah and Lynn Conway as being valuable contributors to the effort. What was the Geometry Engine? It was a special purpose microprocessor that handled matrix math along with point mapping. It featured an instruction set suitable both to 2D and 3D graphics, could generate quadratic/cubic curves and conic sections, worked with both vector and raster based systems, and operated in either integer or floating point systems as needed. In fewer words, Jim Clark and his team at Stanford along with the folks of PARC invented the GPU.

[[The Geometry Engine, image from ACM via Stanford]]

Clark founded Silicon Graphics Inc on the 9th of November in 1981, and he left Stanford early in 1982 to pursue building the company full time with just $25000 in funding (around $85000 in 2024) from a friend and the contents of his own accounts. Accompanying Clark in this adventure were Kurt Akeley, Dave Brown, Tom Davis, Mark Grossman, Marc Hannah, Herb Kuta, Rocky Rhodes, and Abbey Silverstone. While SGI knew they would deal in computers outfitted with a powerful GPU, they did not know precisely what else those computers should feature. As a result, Clark asked potential customers what they’d like to see in a workstation. While at least one potential customer was interested in VMS, NASA’s new Advanced Supercomputing division was very interested in UNIX and they were willing to pay. The division’s director at the time spoke with Clark, and (verbally) committed to purchasing at least eighteen workstations in their first order.

As things began to come together around a product plan, Mayfield invested in the young company. As the development and production of workstations is rather expensive, Clark and SGI’s other founders were forced to sell more and more of the company’s ownership to keep operating. The first product to ship was the IRIS 1000, where IRIS meant Integrated Raster Imaging System, in November of 1983. This machine was intended for use as a terminal for a VAX-11 and featured a Motorola 68000 clocked at 8 MHz with 768K RAM, a Geometry Engine clocked at 6 MHz capable of over six million geometric floating point operations per second, and a 10 Mbps ethernet NIC. The cabinet of the IRIS 1000 was ten inches wide, twenty one inches tall, twenty seven inches deep, and when fully assembled weighed in at one hundred pounds with a ten slot backplane. This machine was followed by the IRIS 1200 which was the same machine but with a twenty slot backplane. These were followed by workstation models 1400 and 1500 in April of 1984 which upgraded the CPU to the Motorola 68010 clocked at 10 MHz with 1.5M of RAM. These machines were differentiated from one another in the size of HDD they featured with the 1500 having been larger. The 1400 featured a 72MB winchester disk, while the 1500 featured 474MB of SMD. Both of these ran a UNIX SVR4 variant with BSD enhancements called GL2, and they featured twenty slot backplanes. The main system boards in these four machines were licensed from Andy Bechtolsheim just before he founded Sun Microsystems. The 1000 and 1200 used the PM1 and the 1400 and 1500 used the PM2. These were not cheap systems with the IRIS 1000 having a price of $22500 (around $67200 in 2024) and the 1400 having a price of $35700 in 1984 (around $106600 in 2024). These twenty slot machines were eighteen inches wide, twenty nine inches tall, and twenty seven inches deep, and fully assembled weighed in at two hundred pounds. By the time the first of these machines sold to Carnegie-Mellon University’s Electronic Imaging Lab, the founders of SGI owned very little of their company.

From nearly the first day that SGI’s hardware was on the market, software developers began trying to exploit the machines’ graphics capabilities. A rather prominent example of this was Wavefront Technologies in Santa Barbara led by Bill Kovacs, Larry Barels, and Mark Sylvester. Their first product was called Preview and launched in 1984 on SGI’s hardware. Their customer list included Universal Studios, NBC, NASA, and Electronic Arts. Naturally, this also informs us that these companies were using SGI hardware.

Given the outline of his youth, it isn’t very surprising that Clark was a hands-off kind of manager. He would hire the brightest minds he could, set a general target, and then let people go after it however they saw fit. There are two narratives for what follows. The first and most common that I’ve read was that Mayfield didn’t much care for Clark’s management style and they brought Ed McCracken formerly of HP in as CEO. The second narrative states that Clark didn’t care for running the company and brought McCracken in on his own accord. Whatever the case, McCracken stated of Clark:

Jim's not a day-to-day person. He works in his own time frame. He takes complex things and makes it simple. It might take a month, a day, or a year. He gets in these moods for a while where he's almost unavailable. He's most effective when he's in that mood.

In August of 1985, the company introduced the IRIS 2000 series of workstations. These machines were all based upon the the PM2 system board featuring the Motorola 68010 clocked at 10 MHz with a floating point coprocessor (SKYFPM-M-03). Naturally, these all featured the graphics engine as well. The IRIS 2000 and 2200 were ten slot backplane, shipped without a disk, and were intended for use as terminals. The 2300 and 2400 were twenty slot backplane and shipped with winchester disks. The IRIS 2500 was rackmount and used SMD disks. The 2000 series used a Geometry Engine clocked at 8 MHz. A few months after the initial launch of these upgraded machines, SGI launched the turbo line. This included the 2300T, 2400T, and 2500T which featured the IP2 system board with a Motorola 68020 clocked at 16 MHz, an FP1 floating point unit, and 2MB to 16MB of RAM. The RAM of the turbo units used a newer, faster, local bus. As a result, the RAM between turbo and non-turbo systems could not be mixed. This was an important bit of information as SGI did offer turbo upgrades for non-turbo systems that would then require the purchase of expensive proprietary memory.

In January of 1986, SGI made their initial public offering raising $17.2 million (nearly $49 million in 2024) with trading having started at $3 per share and topping $30 on the day. The following month, the company introduced the IRIS 3000 line. These are very similar to the IRIS 2000 turbo machines but with Enhanced IRIS Graphics. These featured either ten or twelve Graphics Engines clocked at 10 MHz with either eight or thirty two bitplanes depending upon configuration. The 3000 line could be ordered with either winchester disk drives, ESDI drives, or SMD.

Also in 1986, Control Data Corporation and Silicon Graphics signed a deal under which CDC would resell IRIS machines under CDC’s own branding. As far as I know, no complete listing of which models sold under what naming survives today, but it is known that the IRIS 3130 was resold as the CDC Cyber 910. This would make it a machine with twelve GEs at 10 MHz and ESDI drives.

[[IRIS 3130, rebadged by Control Data, image from sgistuff.net]]

In March of 1987, Silicon Graphics announced a new machine that marked a major transition for the company. The Professional Iris was a RISC machine built around the R2000 from MIPS Computer Systems (another project started at Stanford and spun out as its own company) clocked at 8 MHz. The company’s press release read:

The first member of the Iris line is the 4D/60, a RISC superworkstation with a 32-bit 8 MHz CPU from MIPS Computer Systems. It offer performance three times that of the Silicon Graphics Iris 3100 series. The graphics performance has been enhanced with 38 custom and semicustom graphic chips. It performs 140,000 32 bit three dimensional floating point transformations per second and renders over 4,500 100-pixel polygons per second with smooth shading and hidden surface removal. It offers 24 colour bit-planes for more than 16 million colours; four user-accessible system planes for overlay or underlay, menu and windowing functions; a 24-bit Z-buffer enabling hidden surface removal with greater accuracy and realism; high-level primitives such as splines and surfaces for more accurate renderings; and a multi-mode graphics windowing environment.

Standard configuration includes 4Mb CPU, eight colour bit-planes for 256 colours); four system planes, a Weitek-based floating point accelerator board; a 170Mb ESDI disk and controller; a 19″ 1,280 by 1,024 60Hz non-interlaced colour monitor; keyboard and mouse; and a floor-standing chassis with 12 VME slots and a 1,000-watt power supply.

Software compatible with the previous generation, it runs Unix System V.3 with a base price of $74,000.

The Professional Iris line included the 4D/60 mentioned in the press release followed by the 4D/50, 4D/70, 4D/80, and 4D/85. All of these featured the R2000 CPU with a floating point coprocessor. The 50 and 60 had an R2000 clocked at 8 MHz, while the 70 was at 12.5 MHz, and the 80 and 85 were clocked at 16.7 MHz. For comparison to other architectures, the 4D/50 was capable of seven million instructions per second, the 70 was capable of ten million, and the 80 was capable of thirteen million. The 50 and 60 had memory configurations starting at 4MB and upgradeable to 12MB. The rest of the lineup started at 8MB and could be upgraded to a maximum of 144MB. The first of the 4D/60, 50, and 70 systems to ship utilized the Clover 1 graphics system. Later models shipped with Clover 2 branded as IRIS GT. IRIS GT brought hardware support for lighting, smooth shading, antialiasing, pan/zoom of images, arbitrarily shaped windows, and other rather modern capabilities. Importantly, the bus for this system was a proprietary 64 bit bus. The actual chips powering all of the graphics capabilities were still the Graphics Engines, but these were updated some and they were capable of twenty million floating point operations per second. The Professional Iris series brought an end to the disk anarchy of the previous lineup and all systems utilized SCSI hard disks, and QIC-120 tape drives were also available. These systems were resold by both Control Data Corporation and Prime Computers. The UNIX version mentioned in the press release was SGI’s 4D1 which would later be renamed IRIX.

On the 29th of March in 1988, Control Data Corporation announced that it would be acquiring twenty percent of Silicon Graphics for $68.9 million (nearly $181 million in 2024) and extending its licensing deal for reselling SGI’s machines with an agreement to purchase $150 million (around $393 million in 2024) in hardware over the next three years.

On the 16th of September in 1988, SGI announced that IBM would be purchasing graphics cards and licensing IRIS GL, the software library for SGI’s graphics, for use in the IBM RS/6000 POWERStation. McCracken commented:

We are pleased to establish a relationship with IBM and look forward to working with them. The agreement reinforces our long-time conviction that three-dimensional graphics will become a mainstream technology in the computer industry. As real-time 3D graphics is made more affordable, the rapid growth that the 3D workstation industry is experiencing will continue to escalate.

The card in question was the IrisVision, and while I refer to it as a card, it was really two cards. The primary card held the Graphics Engine and daughter cards held the framebuffer and z-buffer memories totaling 5MB for the framebuffer and 3.75MB for the z-buffer. The primary card connected to the computer via its MCA bus edge connector, and it provided a DE-15 connector for display attachment. Overall, the IrisVision MCA card’s hardware was extremely similar to the graphics system in the SGI Personal Iris series introduced in 1987. It featured SGI’s fifth generation geometry processing pipeline (referred to as GE5, or Graphics Engine five), either an eight or twenty four bit per pixel frame buffer, and twenty four bits per pixel z-buffer. Also, just as the workstations’ hardware did, the IrisVision implemented the entire IrisGL API in hardware. The primary difference in IrisVision was the presence of a VGA (DE-15) passthrough for 2D graphics. In the course of the IrisVision’s development, an IBM PS/2 running OS/2 was used for testing and development. This resulted not only in a minimal OS/2 driver, but also in an ISA version of the IrisVision being developed. Ultimately, the only major customer SGI had managed to obtain was IBM for the MCA card for the RS/6000 UNIX workstations. Their struggle may have been that the card was priced at $4995 (just over $13000 in 2024). The company ultimately spun off the entire project as a separate company, Pellucid, which didn’t fare well. The former SGI employees who started Pellucid still managed to change the world when they founded 3dfx which used similar technology as well as the passthrough for 2D graphics.

SGI held a rather firm grasp on high-end graphics workstations, but hadn’t yet made a push into the entry level market. This changed with the introduction of the Personal Iris lineup. The line started with the 4D/20 which made use of a R2000 CPU from MIPS clocked at 12.5 MHz achieving ten million instructions per second. The other three machines made use of the R3000. In the 4D/25 the R3000 was clocked at 20 MHz achieving sixteen million instructions per second. In the 4D/30, the clock speed was pushed to 30 MHz and the performance was bumped to twenty seven million instructions per second. The highest performance model was the 4D/35 at 36 MHz and thirty three million instructions per second. Maximum memory supported on these systems was 128MB. Personal Iris systems were sold by both SGI and Control Data as expected, but these systems were also offered rebadged by the somewhat newly reconstituted Groupe Bull. From what I can find, Bull’s sales of rebadged SGI machines weren’t great; they had better luck with NEC hardware. For the naming “Personal Iris” and the thought that SGI would be attacking the “low-end” of the workstation market… the pricing wasn’t all that reflective unless one were to compare to “high-end” SGI machines which could reach lofty prices of about $100000 (about $262000 in 2024). The Personal Iris line started at $20000 (roughly $52000 in 2024).

[[SGI Personal Iris, press image from SGI]]

The other, much higher end and far more expensive, SGI lineup introduced at this time was the PowerSeries which were multi-processor systems (up to eight CPUs) and could be deskside or rackmount. These systems could also support higher clocks at up to 40 MHz which in combination with up to eight processors could mean performance over two hundred thirty million instructions per second. The power of these systems was put to use in the movies The Abyss, Terminator 2, and Jurassic Park among many more.

In March of 1991, Compaq acquired thirteen percent of SGI for $135 million (around $307 million in 2024) along with an agreement to invest another $50 million (about $114 million in 2024) in the development of a new workstation that would be priced at around $7500 (roughly $17100 in 2024).

The most famous and beloved SGI systems were introduced from 1991 to 1995. These models were the Indigo, Indigo 2, and the Indy. The corresponding high-end systems were the Crimson, and Challenge series. The first Indigo system released in 1991 featured a MIPS R3000 CPU clocked at 30 MHz. The Indigo (and Crimson) moved SGI’s systems to 64 bit MIPS CPUs starting with the R4000 at 100 MHz and the R4400 at 150 MHz in 1992. The 150 MHz part in an Indigo could achieve one hundred twenty million instructions per second. The Indigo 2 was first introduced in 1993 with the MIPS R4400 CPU and “Extreme” graphics. The Indy was lowest end SKU of the three, and it was introduced in July of 1993 with a 100 MHz R4000PC CPU, 24 bit graphics system, 16MB of RAM, the IRIX operating system, a fifteen inch monitor, and a price of $4995 (about $10700 in 2024).

[[SGI Indigo 2 10000 IMPACT, from my personal collection]]

On the 13th of March in 1992 announced that it was acquiring MIPS Computer Systems via a stock swap worth about $333 million (around $737 million in 2024). This followed MIPS having had financial problems, high employee turnover, and the exit of the company’s president, Charles Boesenberg, one month earlier. For SGI, the acquisition ensured their part supply. MIPS Computer Systems became MIPS Technologies. The combined company had revenues at around $1 billion (about $2.21 billion in 2024). However, the large acquisition did mean that SGI posted a loss on the year of about $118 million (or $261 million in 2024). This move also briefly brought SGI into the ACE alliance that aimed to build a workstation standard on the MIPS CPU and the UNIX operating system as well as the 80386/486 and NT. This group was built of Compaq, MIPS, Microsoft, DEC, SCO, Acer, CDC, Kubota, Olivetti, NKK, Prime Computer, Pyramid Technology, Siemens, Sony, Sumitomo, Tandem, Zenith, and Wang. SGI and Compaq left the alliance rather promptly. This could be due to their own arrangement not long before, but ACE fell apart completely not much later anyway. I suspect that no strong alliance of fierce competitors would last long in a market that was shrinking due to low-cost commodity hardware and software consistently improving year over year in the PC compatible market. Yet, the SGI Indigo 2, Indy, Challenge and a few more were mildly compliant with the ACE ARC (Advanced RISC Computing) standard.

On the 30th of June in 1992, Silicon Graphics released OpenGL. This was a cross-platform API for both 2D and 3D graphics allowing hardware acceleration of rendering via one or more GPUs descended directly from IRIS GL. Unlike its predecessor, OpenGL did not have windowing, and it didn’t offer a mouse or keyboard API. IRIS GL had been developed before X and other graphical environments were available, and therefore had needed those features, but OpenGL had no such requirements. Another major change in the transition to OpenGL regarded feature availability. IRIS GL presupposed the use of SGI’s hardware. OpenGL could not make such an assumption, and as a result it allowed features not supported by a GPU to be rendered in software by the CPU. One customer this would positively affect was Microsoft who’d licensed IRIS GL for inclusion in NT in 1991.

At the end of 1992, Jim Clark met with Nintendo CEO Hiroshi Yamauchi to discuss bringing 3D graphics to Nintendo’s next game console. In many ways, the Nintendo 64 was an SGI workstation in miniature with a MIPS R4300 CPU clocked at 93.75 MHz offering one hundred twenty five million instructions per second, 4MB of Rambus DRAM at 250 MHz (actually 4.5MB but 512K is visible only to the GPU) which could be doubled with a RAM expansion pack, and the Reality coprocessor clocked at 62.5 MHz which offered the SGI GraphicsEngine (though a more modest version). The system supported 16.8 million colors, a maximum resolution of 640x480, and audio sampled at up to 44.1 KHz. Unfortunately, the design of the system meant that the full capabilities would almost never be fully realized. For example, there was no dedicated sound chip, so high sample rates would tax the CPU, and while the R4300 is 64 bit, the Nintendo 64 had a 32 bit data bus. Yet, showing the nature of the hardware packed into the Nintendo 64 is the Nintendo 64DD. This offered 64MB read/write magnetic disks (similar to Zip), a real time clock, internet connectivity via a 28.8 kbps modem, keyboard, mouse, and audio/video capture effectively transforming the Nintendo 64 into a small workstation. The expansion, after significant delays and a one year two and a half month life on the market, was a commercial failure. The Nintendo 64 itself, however, was a huge success following its release in 1996.

Industrial Light and Magic had been using SGI hardware since 1987, and on the 8th of April in 1993, they announced a partnership with SGI to create the Joint Environment for Digital Imaging, or JEDI. This allowed the two companies to gain insight from each other’s work. SGI got access to much of ILM’s software expertise while ILM got access to the latest and greatest hardware at a discount.

In 1994, Jim Clark left SGI, sold his shares in the company, and went on to partner with Marc Andreessen and start Netscape.

In 1995, SGI spent about $500 million (or $1 billion in 2024) acquiring Alias Research, Kroyer Films, and Wavefront technologies. At roughly the same time, SGI worked with DreamWorks SKG to form DreamWorks Digital Studio where these newly acquired companies’ products could be put to good use.

On the 26th of February in 1996, Silicon Graphics acquired Cray Research for $740 million (or $1.47 billion in 2024). This gave SGI control of around forty percent of the high performance computing market at the time. While many industry analysts speculated about SGI’s motives, Cray was struggling to survive and they had multiple installations at NASA. While SGI had been successful in entertainment, that sector accounted only for something around ten percent of SGI’s annual revenues. The bulk of SGI’s customers were governmental, so much so that SGI created the wholly owned subsidiary Silicon Graphics Federal Inc to hold those contracts and provide service and support for governmental organizations. In this way, SGI was essentially making sure they couldn’t lose one of their largest and most valuable customers, NASA, as they’d be the provider of not only workstations but also the support and service of NASA’s supercomputers. The supercomputer relationship benefited SGI all the way to 2008 with Pleiades.

[[Cray Y-MP Model D installation at NASA’s Glenn Research Center, image from Clive England via cray-history.net]]

The new SGI workstations of 1996 were the O2 and O2+ series. These systems were very different from both their predecessors and successors in that they utilized a unified memory architecture via the Memory & Rendering ASIC (MRE). The MRE had direct paths to all parts of the O2 such as the CPU, memory, I/O, compression, display, and imaging. Due to this structure, graphics hardware wasn’t optional but rather integral to the system’s design. The O2 could come equipped with an R5000, RM5200SC, RM7000A, R10000, or R12000 CPU. Frequencies ranged from 180 MHz to 400 MHz, all options had on-board floating point support, and could support up to 1GB of unified memory via eight 128MB DIMMs of one hundred thirty nine pin SDRAM.

The high-end deskside and rackmount options made available at this time were the Origin 2000, Origin 200, and Onyx 2 series. These were multiple CPU systems with distributed, shared-memory architecture called S2MP. The Origin 200 was the entry level system, the Onyx 2 was a step up, and the Origin 2000 was the premium SGI branded system and was rackmount. This series also had Cray Origin at the super-premium level with up to one hundred twenty eight R10000 CPUs. The IRIX operating system shipped with these models supported SMP.

On the 14th of May in 1997, SGI announced the acquisition of ParaGraph International Inc. ParaGraph was a vendor of VRML and web graphics software, and after the acquisition the company and its assets were moved to Mountain View with the new name of Cosmo Software. McCracken commented:

One of the most important long-term growth opportunities for Silicon Graphics is to empower the designers, developers, and service providers of the Second Web. With the acquisition of the leading PC 3D Internet company and the formation of Cosmo Software, we are increasing our investment and reinforcing our leadership in the market for the software and services that will bring about this new interactive medium.

Bringing the technologies of Onyx 2 series to the midrange workstation was the Octane, released in January of 1997. This was the a desktop machine instead of deskside, but it supported dual CPUs. This line featured the crossbar switch that debuted in the high-end and server machines of the prior year. The concept was that instead of a traditional shared bus, each subsystem could communicate with any other without interference. The crossbar switch had seven ports: HEART ASIC (CPU and memory), graphics (Impact [first or second generation] or VPro), XIO B, XIO C, XIO D, built-in I/O, PCI bridge. The Octane did have a higher-end version, the Octane 2, which featured more powerful CPUs and GPUs, higher density memory support, and a beefier PSU. CPUs in the Octane ranged from the R10000 at 175 MHz to the R14000A at 600 MHz, and RAM ranged from 64MB to 2GB.

Silicon Graphics didn’t do too well in 1997 overall. For revenues of $3.6 billion (or $7 billion in 2024) the company posted a loss of $78.6 million (roughly $152 million in 2024). On the 29th of October in 1997, Ed McCracken resigned as did the executive vice president of sales and marketing, Gary Lauer. The company then laid off around nine percent of its employees (about seven hundred people). Richard Belluzzo (formerly at HP) took over as CEO and Robert H. Ewald who was already the executive vice president of computer systems (formerly president of Cray) took over Lauer’s job duties. Some sources claim that McCracken was forced out, but this isn’t accurate. At the annual shareholder meeting in Palo Alto, McCracken announced his resignation stating: “after a great deal of thought, the time is right for me and the company to make a change.” He then proceeded to find and to hire his replacement himself.

Around this time, Silicon Graphics filed a lawsuit against a startup called ArtX. ArtX was founded by Dr. Wei Yen and around nineteen other SGI employees who’d worked on the Nintendo 64. The company’s original goal was to develop a PC graphics chip that would rival 3dfx. Then, in May of 1998, the company gained a contract to develop a graphics processor for Nintendo’s next generation game console, the GameCube. At COMDEX in the autumn of 1999, the company unveiled the Aladdin 7 chipset which shipped as integrated GPUs on K6-2 and K6-3 motherboards made by Acer Labs. ArtX was bought by ATI in February of 2000. ArtX’s technology was incorporated into ATI’s GPUs from 2002 until roughly 2005. SGI’s lawsuit against ArtX was quietly dropped in 1998 without any settlement having been reached.

On the 1st of January in 1998, shortly after taking over as CEO, Belluzzo sold two of SGI’s PCB factories and restructured the company from twenty six groups to just five. SGI then setup MIPS Technologies as its own legal entity (though SGI maintained a majority ownership), terminated the Cosmo software business, and proceeded to make customers hesitant to continue investments into the MIPS architecture by announcing SGI’s intent to migrate to Itanium (and collaborating on projects Monterey and Trillian) while simultaneously launching an IA-32 series of machines running NT known as the Visual Workstation. Additionally, the company began outsourcing the manufacturing of their computers, and cut the operating budget by about $200 million (or $381 million in 2024). In Spring of 1998, the company announced a lawsuit against NVIDIA for patent infringement.

None of this helped to change the overall direction of the company. Revenues fell to $3.1 billion and the company posted a loss of $460 million for 1998. On the 20th of July in 1999, without adequate funding to continue the lawsuit against NVIDIA, SGI and NVIDIA agreed to license one another their respective patent portfolios. The company continued to lose money, and Belluzzo left on the 22nd of August in 1999 to lead Microsoft’s MSN division.

As Bob Bishop took the reigns of SGI, things looked dark. AMD announced their 64 bit architecture in October, PC graphics had made massive strides while remaining significantly less expensive than SGI’s offerings, NT was proving to be a solid and less expensive competitor to UNIX, Linux was eating away at traditional UNIX market segments, and Itanium still hadn’t launched. By this point, the company had no fall back as they’d mostly stopped investment into new MIPS CPUs.

On the 2nd of March in 2000, SGI sold Cray to Tera Computer for $22 million (or $40 million in 2024). Tera promptly renamed itself to Cray Inc as it took on an installed base of six hundred supercomputers and two hundred customers across thirty different countries.

SGI’s final MIPS workstations were the Fuel and Tezro lines. Fuel was introduced in 2002 with the R14000 clocked at 500 MHz, up to 4GB of DDR SDRAM, and SGI’s VPro graphics. Models were available with up to an R16000 CPU clocked at 900 MHz. The Tezro was launched in 2003 starting at $20500 (or $34574 in 2024). This model featured only the R16000 and could be configured at clock speeds from 600 MHz to 1 GHz with 512MB to 8GB of DDR SDRAM and SGI’s VPro graphics. Fuel workstations were single CPU, but Tezro was offered with one to four CPUs. While SGI’s IA-32, Itanium, and Xeon workstations and servers sold, they didn’t make much money. On the 10th of July in 2003, SGI vacated and leased their headquarters to Google.

[[SGI’s headquarters after becoming the Googleplex, by Coolcaesar at English Wikipedia]]

As SGI’s fortunes continued to decline, the company sold Alias Systems (formerly Alias|Wavefront) for $58 million on the 16th of April in 2004 to Accel-KKR (roughly $95 million in 2024). Then, in November of 2005, SGI was delisted from the NYSE due its stock price sinking below the minimum required. In January of 2006, Dennis McKenna was hired as president and CEO, and named chairman of the board. Bishop remained on the board of directors and served as vice chairman. On the 8th of May, the company filed for bankruptcy protections. The campus leased to Google was sold to Google in June of 2006 for $319 million (or $491 million in 2024). It’s prior home in Mountain View had been sold to the Computer History Museum in 2002. The company emerged from bankruptcy and was relisted in October. The official end of SGI’s MIPS and IRIX came on the 29th of December in 2006 with the final orders being fulfilled in March of 2007. Bob Ewald replaced McKenna as CEO on the 9th of April in 2007. In December of 2008, SGI was again delisted. On the 1st of April in 2009, the company filed for bankruptcy, and was subsequently purchased by Rackable Systems for around $42 million on the 11th of May in 2009 (roughly $65 million in 2024). Rackable renamed itself Silicon Graphics International following the acquisition, and it was later bought by Hewlett Packard Enterprise.

@@ The Rise and Fall of Silicon Graphics (abortretry.fail)

345 points by BirAdam 6 days ago | 365 comments

***martinpw*** 5 days ago Whenever this topic comes up there are always comments saying that SGI was taken by surprise by cheap hardware and if only they had seen it coming they could have prepared for it and managed it.

I was there around 97 (?) and remember everyone in the company being asked to read the book "The Innovator's Dilemma", which described exactly this situation - a high end company being overtaken by worse but cheaper competitors that improved year by year until they take the entire market. The point being that the company was extremely aware of what was happening. It was not taken by surprise. But in spite of that, it was still unable to respond.

***mrandish*** You highlight one of the most interesting (and perhaps less understood things) about the key Innovator's Dilemma insight. Even if the senior management have read the Innovator's Dilemma books, know they are being catastrophically disrupted and desperately trying to respond - it's still incredibly difficult to actually do.

Not only are virtually all organizational processes and incentives fundamentally aligned against effectively responding, the best practices, patterns and skill sets of most managers at virtually every level are also counter to what they must do to effectively respond. Having been a serial tech startup founder for a couple decades, I then sold one of my startups to a valley tech giant and ended up on the senior leadership team there for a decade. I'd read Innovator's Dilemma in the 90s and now I've now seen it play out from both sides, so I've thought about it a lot. My key takeaway is that an incumbent's lack of effective response to disruption isn't necessarily due to a lack of awareness, conviction or errors in execution. Sure, there are many examples where that's the case but the perverse thing about I.D. is that it can be nearly impossible for the incumbent to effectively respond - even if they recognize the challenge early, commit fully to responding and then do everything within their power perfectly.

I've even spent time sort of "theory crafting" how a big incumbent could try to "harden" themselves in advance against potential disruption. The fundamental challenge is that you end up having to devote resources and create structures which actually make the big incumbent less good at being a big incumbent far in advance of the disruptive threat appearing. It's hard enough to start hardcore, destructive chemo treatment when you actually have cancer. Starting chemo while you're still perfectly healthy and there's literally no evidence of the threat seems crazy. It looks like management incompetence and could arguably be illegal in a publicly traded company ("best efforts to maximize/preserve shareholder value" etc).

***duxup*** Even just changing simple behaviors across a large company can be impossible. I worked at a company with several thousand employees all required to attend mandatory training on "effective meetings", several hours long spread out across thousands of employees. Rule 1 was to have an agenda for the meeting. This was something nobody did at this company and you would regularly attend meetings unprepared.

After all that training, still nobody did it (ok I did and one other guy). That company couldn't change anything. It was amazing.

They had a project to change a department into more proactive than reactive. The solution was to create a lot of bureaucracy surrounding being proactive. As you can imagine bureaucracy about being proactive was really just institutionalizing ... not being proactive.

I eventually left and work at a smaller company now. It's been refreshing for years now when we can decide "this process doesn't work, let's not do it anymore" and it just happens. Even just new coworker: "I won't be in tomorrow, who do I tell?", me: "you just did, have a great time off" seems revolutionary after being at a big company for so long.

I'm convinced that as the sheer numbers of humans increases the friction to making real change in a company decreases and there's not much you can do. Fundamental change to respond to real disruption, nigh impossible.

***rbanffy*** I think SGI failed to understand that there was a point where desktop PCs would be good enough to replace dedicated workstations. Continuing to make hardware that's much better than the best PCs wasn't going to save them after PCs crossed the good-enough line - whatever they had, would be relegated to increasingly rarefied niches - the same way IBM now only makes POWER and mainframes - there is no point of making PCs, or even POWER workstations anymore for them, as the margin would be too narrow.

SGI could double down on their servers and supercomputers, which they did for a while, but without entry-level options, their product lines becomes the domain of legacy clients who are too afraid (or too smart) to port to cheaper platforms. And being legacy in a highly dynamic segment like HPC is a recipe for disaster. IBM survived because their IBMi (the descendant of the AS/400) and mainframe lines are very well defended by systems that are too risky to move tied to hardware that's not that much more expensive than a similarly capable cluster of generic and less capable machines. As the market was being disrupted from under them, they retreated up and still defend their hill very effectively.

The other movement they could do was to shift downwards, towards the PC, and pull the rug from under their workstation line. By the time Microsoft acquired Softimage and had it ported to NT, it was already too late for SGI to even try that move, as NT was solidified as a viable competitor in the visual computing segment, running on good-enough machines much, much cheaper than anything SGI had.

***panick21\_||| IBMs market was also just far larger. SGI sales were around 3 billion, Sun were around 3x that. IBM was even more. SGI and Sun were based Unix so Linux could disrupt that far more easy then IBMs systems.***

***IBM also came into the game more vertically integrated. Having your own fab is expensive, but if you read on what Sun and SGI had to do in order to get chips, that route also wasn't great.***

***In the beginning there was a chance that Sun and SGI could have merged, the reason it didn't happen was mostly because leadership couldn't agree on who would lead the company. Between them they duplicated a lot of technology while sitting right next to each other. Both doing their own RISC chips, at times Sun was doing their own graphics cars, competing in 'low' priced and mid priced work stations, incomparable Unix developments, competing in the large SMP market against each other. If they had been together and things could have been different a larger install base and more investment into the architecture might have given them a better chance.***

***|||giantrobot*** > I think SGI failed to understand that there was a point where desktop PCs would be good enough to replace dedicated workstations.

I think the real problem here is PC workstations, with Windows NT (later Linux) and an Intel chip, could do 90% of the SGI workstations for a fraction of the price. By workstation I mean an honest workstation with a high end graphics card, 10k RPM SCSI drives, and hundreds of megs of RAM.

The price of SGI workstations was "call us" which translates to tens of thousands of dollars per workstation. PC workstations didn't and couldn't replace all uses of SGI workstations. What SGI was not able to handle was the fact their customers suddenly having a viable option besides paying them tens of thousands of dollars for their workstations. Even their Visual Workstations weren't a real improvement cost wise as those were still overpriced compared to competitors' PC workstations.

***smackeyacky*** This is my recollection of the era as well. A quality PC like a Compaq was already a good alternative during the Motorola era. In a lot of ways the whole RISC thing was a dead end as all that happened was SGI, IBM, HP and Sun cannibalised each others sales.

ARM is the only one left standing from that era which with hindsight seemed so unlikely.

***vidarh*** Keep in mind several of the others survived long past their public visibility. There were MIPS smart phones for a while. Both PPC and MIPS long outsold x86 in number of CPUs - just at low margin licenses with the bulk going into embedded uses, like ARM.

ARM had the advantage in that space of starting from the very low end, and being able to squeeze margins instead of being squeezed.

***rbanffy*** Don't forget IBM is still selling (and improving) their POWER family, running AIX (which had the low-end eaten away by Linux) and IBMi (which is the only minicomputer family still standing). IBMi is extremely well defended as it's not even fully documented (unlike mainframes) and, therefore, doesn't even have a way to be emulated outside IBM. And I would not be surprised if that "secret sauce" to be kept in an underground facility under a mountain in the middle of a desert, in a completely offline system behind multiple biometric locks.

ARM survived this long because it had a niche others couldn't break into (and still can't) as the highest performance per watt anywhere.

***giantrobot*** I think it would be more accurate to say the RISC workstation built entirely of custom low volume components was a dead end. Even if SGI decided to cut their margins to compete on cost their lowest prices would still be way above a PC workstation. Compaq benefitted from economies of scale that SGI could not.

RISC architectures live on today. Your house likely has dozens of MIPS chips in various components. You've got more ARM chips for sure but no shortage of other RISC chips in various components.

***rbanffy*** “Custom low volume components” is a marketing strategy. They could sell them to competitors and make money off the components. Late in its history SGI owned MIPS and sold their hardware to others. I have a WindowsCE laptop running on a MIPS R4000 processor. Runs NetBSD with effort, but with dignity.

***mrandish*** I think your analysis of the shifting technology landscape is largely on target. However, I'm not convinced that the true root of SGI's failure was the technology. Clearly their tech did need to evolve significantly for them to remain competitive but that's a transition which many companies successfully make. Even though SGI chose not to evolve the tech soon enough, fast enough nor far enough, I suspect they still would have failed to survive that time period due to an even more fundamental root cause: their entire corporate structure wasn't suited to the new competitive environment. While the "desktop transition" was most obviously apparent in the technology, I think the worst part for SGI was that the desktop shift changed the fundamental economics to higher volumes at lower costs.

SGI had invested in building significant strengths and competency in its sales and distribution structure. This was one of their key competitive moats. Unfortunately, not only did the shift in economics make this strength irrelevant, it turned it into a fundamental weakness. All that workstation-centric sales, distribution, service and support infrastructure dramatically weighed down their payroll and opex. This was fine as long as they could count on the higher margins of their existing business. While it's easy to say they should "just layoff all those people and relaunch as a desktop company" that can't be done in one quarter or even one year. It requires fundamentally different structures, processes, systems and skill sets. Hiring, training and integrating all that while paying for massive layoffs and shutting down offices, warehouses etc takes time and costs a lot of money.

Worse, once their existing workstation customers saw them shutting down the SGI the customers had bought workstations and service contracts from to become a different kind of company entirely, sales revenue would have taken an overnight nosedive. SGI's stock would also have tanked far more immediately than it did as the fickle stock market investors sold stock they'd bought because SGI offered a specific risk/return expectation which just became much more "risk" and much less "return" (at least in the near-term). In making such a dramatic move SGI would have effectively dumped much of their current quarterly revenue and the value of one of their core strengths - all at the same moment. Thus turning them into one of their emerging startup competitors with all of a startup's disadvantages (no big ongoing revenue streams, no big cash pile (or high stock valuation to leverage for cash)) yet none of a startup's strengths (nimble, lower-paid staff and more patient venture investors).

The point of my earlier post was mainly that a true disruptive market shift is nearly impossible for a large, established incumbent to successfully survive because they basically have to rapidly turn into someone else almost overnight. How can a champion sumo wrestler survive a shift so dramatic that their sport quickly turns into a track meet? Even seeing it coming doesn't help. How does one even prepare for such a shift since losing mass turns you into a bad sumo wrestler long before you even start being a viable sprinter? As Christiansen observed, such disruptions are often enabled by technology but the actual cause of incumbent death is often due to the shift turning an incumbent's own strengths into weaknesses almost overnight.

***panick21***| I don't think you need to shut down that distribution. Because fundamentally, existing costumers often continue to buy existing product at existing prices, as long as they get a faster product. This was something that Gordon Bell pointed out. Existing costumers are design to work at that level of expense. Expensive graphics workstations and rendering servers and so on still exist. Sure it would go down eventually as all business does in the long run.

The real failure is not picking up new business along the way. With the N64 they showed they could design a consumer product. But outside of that they were in no future consoles. 3dfx and ArtX both came from former SGI. You don't need to stop selling workstations just because you make chips for consoles and other such devices. Nvidia barely survived and might not have if not for consoles. There are other markets where their expertise could have been applied.

Surviving changes like that often requires finding other markets. And then when it comes to making hard choices you need to cut part of the thing that is unprofitable. But this is really hard to do and in some ways it goes against the 90s US corporate philosophy of focusing only on the 'Core' business. DEC for example sold profitable business units that might have been helpful to have. DEC had a printer business and had the potential for a Database business. Oracle calls RDB one of the best acquisitions.

***vidarh*** > the N64 they showed they could design a consumer product. But outside of that they were in no future consoles. 3dfx and ArtX both came from former SGI. You don't need to stop selling workstations just because you make chips for consoles and other such devices.

Commodore tried for a play where their - still much lower end - newest generation chipset would have scaled (with the same chips) from being a full low end computer, console, or set-top box, computer (it had a PA RISC core on chip, so could run standalone), a high end graphics card for PCs, and the chipset for a higher end Amiga at the same time.

They ran out of money - short maybe a few million - before being able to test if that would have worked as a way to widen their market.

I wish we'd have gotten to see how that'd unfolded, though Commodore's deep management dysfunction probably still would have made it fail.

***mrandish*** During the Amiga era Commodore's board was controlled by old-school big business / Wall Street suits who had literally zero clue about computers or even technology in general. For a while toward the end the inside rumors were that the powers controlling the board were primarily benefiting from international currency and tax games and were cutting costs to harvest whatever cash they could into offshore tax havens while refusing to invest a cent back into saving the company.

***rbanffy*** Legend says Commodore was at some point approached by Sun willing to OEM their Amiga 3000/UX machines as entry-level Unix workstations and sell them through their distribution channels. The same accounts say Commodore management refused (maybe because the deal had a non-compete part).

They also missed an (earlier) boat with the Commodore 900 workstation, which would run Mark Williams' Coherent OS (a Unix-like system).

***vidarh*** Commodore is one long history of management failures... As a big fan at the time, without insight into how dysfunctional it was, it was thoroughly depressing both to see from the outside, and then reading up on the internal chaos years later. It's a miracle they survived as long as they did, and at the same time there were so many fascinating lost opportunities.

ChuckMcM was at Sun at the time, and mentioned a while back he tried to get Sun to buy Commodore outright:

https://news.ycombinator.com/item?id=39585430 (his later replies in that sub-thread are also worth reading)

***panick21***\_||| > They also missed an (earlier) boat with the Commodore 900 workstation

They didn't really miss the boat on that. The C900 wasn't really an attractive machine. They would have sold it at a pretty high price. At the same time you could buy a PC with Unix on it that was arguable better. It would have just been another market where they got clobbered by the PC. Not like Zilog was the right horse to bet on anyway.

***|||pjmlp*** Many Amiga engineers were UNIX heads, and even Amiga DOS /Workbench started as what might be another take into UNIX like systems with multimedia hardware, when they pivoted away from a games machine.

There are some vintage computer club talks where they dive into this.

***rbanffy*** Commodore could have used Coherent as the Amiga OS.

***pjmlp*** On the other hand, thankfully they haven't taken that approach, as otherwise wouldn't have been the multimedia machine it turned out to be, or have OS design decisions like its use of Libraries, Datatypes and everything scriptable that are yet to be mainstream.

With exception of a few UNIX systems like Irix, Solaris with NeWS, NeXTSTEP, Apollo, everything else tends to be the same deck of cards reshuffled.

***gregw2*** Your analysis except for your first sentence is good.

But it is not true that SGI failed to understand there was a point where desktop PCs would be good enough to replace dedicated workstations. They had built a $3500 PC graphics card (IrisVision) way early on, and did the Nintendo deal before PC 3D really became a thing, they partnered in 1991 with Compaq on an investment and a $50-million joint effort to develop a workstation (https://www.encyclopedia.com/books/politics-and-business-mag...), and they were themselves taking advantage of the ability to squeeze more and more 3D into a single chip; the tech trends were obvious.

SGI was a client of tech research a firm I joined at the time and it was heartbreaking to see them lose and very hard to figure out what they could/should do. It wasn't my explicit role to solve their problem but I spent a lot of time thinking about it.

You do capture some of the dynamics well but you don't capture the heart of it. The heart of it is point #1. The rest below are also inhibitors.

1) SGI had revenues of $2 billion/year and the 3D market revenue was, say, $50/million/year. (OK, maybe a bit more than that; Matrox 2D king had what, $125 million in revenue?) How do you trade the former for the latter? And on top of that trade high margins for low margins? When 95% of your stakeholders don't care about the new (PC games) market?

2) Engineering pride/sophistication/control. The company started out focused on Graphics but, being in Silicon Valley, had grown/acquired huge engineering strengths in other areas besides graphics, CPUs design (MIPS), NUMA cache-coherent SMP hardware+SMP UNIX design, etc and that's before you get to the Cray acquisition and bits. They were the "Apple" of graphics workstation vendors but there was no "iphone" integrated vertical play for them downmarket (except maybe consoles and they half-tried that with N64 and even Nvidia while using that has minimized that due to its low margins and low opportunity for upside.) It was hard technically to give up/deprioritize all those levels of engineering sophistication in favor of competing on graphics performance and price-performance when the PCI(/AGP) bus was fixed, the software API layer was fixed, the CPU layer was fixed, and you had to compete with 80+ fledging companies to win in both performance and price/performance in a low margin PC 3D games graphics which is just a much lower value-added play for engineering.

3) Compensation. Employees with knowledge of how to make a 3D graphics chip had a bigger upside outside of the company than inside with that knowledge. They left. 3dfx founder left. ArtX guys left. Later other guys left for Nvidia.

4) Slow/different engineering cycle times/culture. SGI cycle times for new products were 3-4 years. Sure they'd do a speed bump at the half-way point. Some volume 2D chip companies would have tweaks to their designs coming out every 2 weeks. PC graphics vendors needed a new product every 6-12 months. Nvidia's most critical life-saving measure in their product history was to cut their cycle time radically by using simulation because there was no other option, and it left them with 2/3rds of their target blend modes not working. https://www.acquired.fm/episodes/nvidia-the-gpu-company-1993...

5) Pride around owning/controlling the 3D software layer. Having developed OpenGL, they didn't/wouldn't figure out how to let Microsoft control/change/partner with it for gaming markets at the API level. Yes they licensed OpenGL, and eventually they caved and cross-licensed patents, but Microsoft was never going to let them own/control the API nor was Microsoft ever going to fully support an open API. And there was no love lost between Silicon Valley engineers and Microsoft in the late 90s. Hard to partner.

6) Executive leadership (I don't know any of this firsthand.) The founder who cared about graphics, Ed Clark, by some accounts saw the workstation/PC threat a ways away. When the critical timeframe came to deal with it though (92-94), he also saw the web and saw how big that wave was, and PC 3D graphics was a small wave in comparison so he switched focus and left behind him was the CEO who was an ex-HP executive who had for a decade grown SGI from $5m revenue to billions by remaking SGI into the image of HP rather than focusing on graphics, graphics, graphics and was not equipped to bet the company on rebirthing/growing the company out of a nascent PC 3D games market growing.

As an industry analyst serving SGI as a client (and its competitors) and seeing the forces facing them in the mid/late-90s, what was known at the time was: - 3D games were going to fuel consumer usage, on both consoles and PCs, thanks to one-chip texture mapping (+ over time geometry processing) - Wintel controlled the PC space but did allow third-party graphics cards manufacturers and were fine with that - Linux was good enough, but was for a transition period not nearly as good as conventional UNIX - There was a huge room for improvement in 3D graphics on the PC at first, but at some point you would get good-enough bang for your buck and then the market for your graphics processors would stagnate. Screen resolutions grow more slowly than Moore's law, and once you can do enough shading for every pixel on the screen 4x over and enough polygons matrix operations for one polygon on every pixel on the screen, how much more compute do you really need?

But in 92-95 it was hard to advocate for SGI downsizing/betting-the-company based on this alone.

In mid-1993 Windows NT 3.1 marked "good enough" Windows to compete with UNIX (process isolation) and Windows 3.5 in Nov 1994 solidified that. In Nov 1995, with Pentium Pro specInt figures coming out, it was clear RISC was going to die but just cognitively hard to recognize.

SGI clearly saw the problem and tried to diversify out of it Alias/Wavefront (1995) and going upmarket (buying Cray whom they were cannibalizing) but neither of those "saved it".

I remember thinking at some point charting out with a colleague how the industry consolidation might happen that they really should join up with Apple (both really rode the 'halo' effect of high-end sexy products to sell bread and butter products effectively and both were very vertically-integration oriented); I don't know if that was ever considered. Apple was 100x weaker then and I don't recall if the actual market cap economics would have worked.

What wasn't fully recognized (at least by me or others I read voraciously at the time), but is clearer in hindsight was that: - while SMP parallelism was part of the wave of the future (and required both work on both CPU+OS layers of the stack), and GPUs contained a mixture of ASIC-based parallelism for both shading and vertex operations, that one could construct a general purpose coprocessing engine that would have real market uses beyond graphics (first in HPC then in AI) and a longer-term value proposition that would outlast a gamer-centric niche market and be a general compute engine.

The term GPU alone that Nvidia used early on in marketing now implies and delivers on that vision, but it didn't imply it (to my eyes) at the time; SGI and others had used the term GPU informally even before Nvidia marketing did, to refer to chips with geometry/matrix computations on them and Nvidia was entering that game (and of course talking up their book.) But the true vision of creating a general-purpose parallelism compute engine coprocessor and API along with it was really fleshed out at Stanford PhD work by Ian Buck a decade later in 2004 and he then graduated and took it to Nvidia and it became CUDA. https://graphics.stanford.edu/papers/brookgpu/brookgpu.pdf At least as far as I can tell.

It was always possible as a graphics chip company in the 1990s to see the next 1-2 generations of possible technical improvements, but it was very hard to see a multi-billion dollar business. Tens of millions absolutely. Hundreds of millions, sure. But billions? And this goes back to my point #1.

There was a very real possibility they would enter the 3D PC gaming fray, not be able to compete and lose, and then all that Silicon Valley graphics marketing halo they benefited from so much have would faded. But it faded anyway.

I suppose they could have cross-licensed their texture mapping patents to any startup giving them 5-10% equity in return for capital and then tried to buy them out as they grew. They tried fighting over that issue with Nvidia later. But they would have hemoraged engineers to that approach and I'm less sure that would have worked.

In my view, they should have just bit the bullet, rolled the dice, and played the PC 3D graphics gaming game and stayed true to their name, "Silicon" "Graphics". Not "Silicon Software" (alias/wavefront) or even "Silicon parallelism" (Cray). It can be true that if you stay in a niche (3D graphics for the masses), you end up in a ditch. But they lacked the courage of their potential and went the wrong way. In hindsight, someone probably should have kicked out McCracken in 1992, not 1997 and they could have gotten a more visionary leader less tied to the HP executive way of looking at problems/opportunities. But I don't know how they could have transitioned to a leader with better vision or where they could have found one.

I'd be interested if there was ever an HBS case study on this based on internal SGI documents. Or if others have pointers to internal SGI discussions of this dilemma. It still bothers me 30 years later as you can see by the length of this post. Too bad I posed this on hn a day late.

***ForOldHack*** Your analysis is amazing. Thank you.

***gregw2*** Thank you! It encourages me that someone got to see and appreciate it.

What is hard to convey to people outside the 3D hardware space is that the chief problem once you have the 3d pipeline down is really a market-development problem.

How do you sell an ever increasing amount of coprocessor compute?

Because the moment you hit the inevitable S-curve flattening out of your 3D value proposition, your coprocessor gets integrated into the (Intel) CPU die, just like the intel 387sx/dx floating point unit or earlier generations of IO coprocessors. Hence a frantic strategic push always into raytracing, HPC, AI, etc.

In hindsight it looks visionary, but the paranoia is at least as much a driving factor.

It’s now, for now, incredibly lucrative and the mastery of hardware parallelism may last as a moat for Nvidia, but I can sympathize at SGI not wanting to squeeze themselves back into a coprocessor-only business model. It’s a scary prospect. We can see only with hindsight it could have worked. Both business and technical leadership had such huge success diversifying their graphics hardware expertise into full system skills that they couldn’t refocus. They would have had to die in order to live again. Or perhaps slightly less exaggerated, to survive they would have to forsake/kill off a huge part of what they had become.

***throwaway4good*** Everyone has been reading that book since the late 90es.

I remember a talk by Clayton Christensen talking specifically about Intel and how they setup the Celeron division to compete with themselves (based on his advice).

A key property of tech in economics lingo is that it is “natural monopolies” - all fixed cost and no variable cost.

This creates these winner takes all games. In this case both Intel, SGI plus others knew the rules and it just ended up with Intel taking the prize and it all becoming Wintel for a decade or so - basically until the smart phone allowed enough capital to be accrued to challenge the old monopoly.

***pfdietz*** Arguably, the shareholders don't want the management to try to harden the company. If the shareholders wanted to invest in a new approach, they could do that on their own. Rather, the shareholders expect the company to squeeze what they can out of what they are, even if that means winding down the firm.

***Pet\_Ant*** Companies and management would rather a slow controlled descent into the ground than unexpected success. They balance their risk at the portfolio level not at the company level.

During good times focus on your core and then peter out. Just follow a nice parabolic arc.

A lot of success is just timing and attempting to fight all threats look foolish. A remember, for every up and comer that “we all saw coming” there were lots that didn’t make it. If you waste time fighting those you wasted money.

***ghaff*** Right. It's not actually in anyone's interest most of the time to deliberately destroy the company in the short term to have a glimmer of hope of maybe successfully reinventing it in the longer term. Shareholders always have the option of moving on and employees/management can probably collect paychecks for more time while they possibly look elsewhere. Corporate legal continuity is probably overrated.

***bsder*** "Maximally efficient is minimally robust."

A company that optimizes for efficiency will get stomped flat when the environment changes.

The problem is that there are no incentives in business to optimize for robustness.

***roughly*** Well, that’s partially because of the converse: a company that optimizes for robustness will get stomped flat before the environment changes to require robustness. Short term games are bad in the long term, but often good enough in the short term to win before the long term arrives.

***Kamq*** > and could arguably be illegal in a publicly traded company ("best efforts to maximize/preserve shareholder value" etc).

There's no way the shareholders win this lawsuit. Violating fiduciary duty involves doing things you know won't provide value. Long term planning would be a defense for this.

The shareholders could absolutely oust those executives, though. And they may very well do so.

***kjkjadksj*** I wonder if any major companies have experimented with an internal “red team” startup? It would be on level playing field with your upstart competitors, free from existing hierarchy and structure to develop disruptive tech like any other startup, only you are bankrolling them which gives them a head and shoulders boost relative to any other startup. Eventually you can let this startup grow to be more dominant than the old company and repeat the process with the next disruptive technology.

***telesilla*** I've seen this in several mid-size companies but the trick to success is that you make a start-up in another vertical so your existing customers aren't dropping their contracts for the cheaper thing you disrupt yourself with. You're building a new business for the same market that already know you, so you can capitalize on your brand, using revenue from a dying but profitable model.

***JustLurking2022*** As a thought experiment, let's say Facebook did this.

The red team designs a new streamlined look with less click baity posts and fewer ads. Users flock to it, abandoning the existing platform. The new platform isn't monetized as effectively as the old one so revenue drops by billions per year - maybe an order of magnitude more than the new product brings in. Shareholders demand an answer as to what management are doing about it.

There might be some golden path, but that typically relies on finding a new market that doesn't interfere with the existing business (e.g. AWS and Amazon). However, the far easier options are a) shutdown the new service or b) increase its monetization, thereby starting the enshitification cycle.

***datavirtue*** Does this apply to the US response to Chinas lead in battery and solar production? The Inflation Reduction Act is essentially a protectionist policy for manufacturing in response to Asian and Chinese tech manufacture. I feel like we are trying to fight our way out of one of these situations. I'm concerned because every time I see something happening that excites me (the massive policy shift from enacting the IRA and the surge in American manufacturing) it usually means something really bad is about to happen.

***gregw2*** I agree. You might appreciate my elaboration on the SGI specifics of this elsewhere on this thread.

***ghaff*** Having worked longtime for a minicomputer company--which actually survived longer than most mostly because of some storage innovations along with some high-end Unix initiatives--it's really hard. You can't really kick a huge existing business to the curb. Or otherwise say we're going to largely start over.

Kodak was not actually in a position to be big in digital. And, of course, the digital camera manufacturers mostly got eclipsed by smartphones anyway a decade or so later.

***aurizon*** On the contrary, Kodak was well placed to do well by anticipating 'Moore's Law' as pertinent to sensor pixel density and sensitivity versus film. Film resolution was towards the end of intense development in pixel terms - not much further to go. They had pioneering patents and ongoing R&D would have enabled a long period of dominance during the transition and to this day!! The board and scientists were asleep on a mountain of cash, and they sold their future for a few crumbs left for shareholders after bankruptcy. Blackberry did much the same with fewer excuses. I met with some board members of Kodak in the 80's and they were like old English gentlemen - long on pomp and procedure, but they wore blinders and a vision bypass - TRIH.

***ianburrell*** Kodak did fine in the transition to digital. They made some popular compact cameras and tried to make DSLRs. They were wiped out by compact cameras being killed by smartphones. The survivors are the old camera makers like Canon and Nikon that have ecosystems. The other big survivor is Sony, which bought a camera company and makes most of camera sensors.

Fuji is interesting, they weren't that successful in first digital cameras, but now have some interesting mirrorless ones. They still make film.

***ghaff*** Fujifilm is a much smaller company than Kodak was. They also applied a lot of their expertise in emulsions to medical applications.

And, yes, they have some interesting if somewhat niche cameras.

***ghaff*** Kodak was essentially a chemical company at one point. They even spun off an actual chemical company. Kodak could probably have played a better hand even if they did probably before their time things like PhotoCD. But they could have been Apple or maybe Instagram? That's a stretch.

I'm not a particular Kodak apologist but suggesting that a company should have been able to anticipate and correct for their business collapsing by 90% in a decade or so seems to need a lot of particulars.

***xcv123*** > But they could have been Apple? That's a stretch.

They could have been a Sony. The iPhone camera sensor is made by Sony.

***ghaff*** And Sony has certainly had rough patches too. And that's for a company coming from an electronics manufacturer angle.

Kodak could have spun off a consumer electronics or semiconductor manufacturing company. But it's not clear why that is actually a better model than someone else just spinning up similar entities.

I don't need all the chemical engineers and a lot of other people connected with the old business anyway. And I'm sure not turning them into semiconductor experts.

So you're one of the 10% of employees in HR who snuck through to the other side. Is that really a big deal?

***xcv123*** That's right. The chief executives and the HR lady basically get transferred over to a new startup funded with Kodak's money and everyone else is fired.

***maire*** Kodak was well aware of what was going to happen. Company culture killed digital photography.

I was at Apple when we worked with engineers from Kodak who were working to change various format standards to allow digital photos. This was in the late 1980s or early 1990s.

***ghaff*** But, from the perspective of today, Kodak would have had to basically eclipsed Apple.

Even displacing the big Japanese camera manufacturers, who by then had dominated high-end photography, would have required reversing decades of a shift away from high-end cameras like the Retina line.

I don't doubt there was company DNA against digital photography but it's not like non-smartphone photography, especially beyond relatively niche pro/prosumer level, has had such a good run recently either.

***nradov*** There is still a lot of business opportunity in supplying image sensors and lenses to smartphones.

***chiefgeek*** But it is nowhere near as profitable as the 35mm film system was.

***ghaff*** The 35mm system was a huge consumables business down through the food chain. That basically doesn't exist with digital. (Aside from ink jet supplies and I'm not sure how true even that is any longer.)

***loloquwowndueo*** Data General?

***ghaff*** Yes. CLARiiON eventually enabled a sale to EMC (which arguably saved EMC for a time) and the Unix business (especially NUMA servers) were sufficient revenue producers for a while to keep the lights on. ThinLiiNe (or whatever the capitalization was) never went anywhere but neither did a lot of things in the dot.com era.

***panick21\_||| Seems like Sun really shot itself in the foot by not buying DataGeneral. A Unix storage business that fits pretty well with their larger datacenter portfolio. And a start to a real x86 business.***

***I just finished reading 'LIFE UNDER THE SUN: My 20-Year Journey at Sun Microsystems' that talks about Sun and the storage businesses a bit. Sun was never happy with how well they did in storage. Sun was part of defining Fibre Channel.***

***For some reason that still doesn't make sense to they bought StorageTek for an incredible $4 billion at the time when Sun wasn't exactly flying high. The explanation from the CEO given in the book mentioned above is baffling.***

***Edit:***

***Seems they bought:***

***1999: MAXSTRAT Corporation, a company in Milpitas, California selling Fibre Channel storage servers.***

***Never heard of them. Not sure how comparable it was.***

***|||loloquwowndueo*** I knew it :) thanks for confirming! And for sharing.

***ghaff*** I was the PM for a bunch of the minicomputers from the mid-80s on. Then I was PM for the initial Unix AViiONs and later the NUMA servers including being one of the main liaisons with CLARiiON.

***AnimalMuppet*** Someone at SGI wrote a paper/web page/blog post titled "Pecked To Death By Ducks", claiming that x86 chips could never compete with SGI, and claiming to refute all the arguments that they could.

Then Intel introduced dual core (or maybe just two chips in one housing sharing a bus), and that generated a lot of buzz. So he wrote a follow-up titled "Pecked To Death By Ducks With Two Bills".

I don't recall the timing, though, how it related to the timing of asking everyone to read The Innovator's Dilemma. But at least some of the time, there was a pretty deep denial (or at least a pretty deep effort to keep the customers in denial).

***rbanffy*** A bit like Seymour Cray's plowing a field with 1024 chickens instead of two oxen.

***MichaelZuo*** That’s really funny for some reason.

***HeyLaughingBoy*** IIRC, "Pecked to Death by Ducks" is the title of either a short (nonfiction) story or a book by Gerald Durrell, one of my favorite childhood authors.

***jfk13*** I don't recall that one, and I thought I knew Gerald Durrell's work pretty thoroughly. There is a book of that title by Tim Cahill, though; maybe that's what you're remembering?

***HeyLaughingBoy*** Yesss! Thank you. I'm embarrassed because I used to have that one, too :)

On the upside, I've never known anyone else who had even heard of Durrell.

***bluenose69*** I would have thought that lots of folks had watched the "The Durrells in Corfu" TV series (PBS Masterpiece Theatre) and then did some research on the family.

I don't know whether this is available online, but I can recommend it as a pleasant programme, with lovely scenery, interesting storylines, and engaging actors.

***selimthegrim*** Lawrence is probably better known.

***VelesDude*** It was clear they were trying to do more consumer grade things, just look at the N64. Couldn't get more mainstream than that. Seeing how the graphics market ended up, it looks obvious from here but in the mid 90's it was still the wild west and everybody was throwing mud at the wall seeing what would stick.

I have never really said that they where "taken by surprise", but a part of it felt like (from the outside) that management had been a little blinded by their pass success and the profit margins from their workstations combined with no clear path forwards for the whole industry. Nvidia could have very easily been just a curiosity of the past but they managed to strike it lucky standing on the shoulders of others.

If SGI had always been a company that could provide graphics workstations the worked with x86/Windows PC's early for example - maybe they would have fared better. Would have gone with the flow of technology at the time rather than fighting uphill no matter the potential technical brilliance. But being saddled to their MIPS processors and custom OS meant that once people left, they almost never came back. One can have the best tech and still fail.

***the\_mitsuhiko*** > It was clear they were trying to do more consumer grade things, just look at the N64.

Yes, but the team that did that also left SGI, then worked directly with Nintendo for the GameCube and are acquired by ATI. I’m not sure how SGI managed to not support that effort within itself.

***cuno*** Nintendo wasn't loyal to the company it was loyal to the team, so when they just decided to leave and form ArtX they took the customer with them... SGI was happy with the Nintendo contract. They earned $1 in additional royalties for every single N64 cartridge sold worldwide. Losing the team was a big blow.

***rbanffy*** > provide graphics workstations the worked with x86/Windows PC's

Integraph started making PCs with high-end graphics at one point, when they abandoned CLIX and gave up on their (Fairchild's, really) Clipper processor. It didn't work for them either. SGI did their own "Visual Workstation" that ran Windows and had a Pentium, but that too was a huge disappointment.

***coredog64*** The 320 and 540 had a few nice things going for them: You could option them with the same 1600SW monitor that the higher-end O2 workstations used without having to use the fiddly multilink. SGI paid Microsoft for a HAL license and did a better job with multiprocessor than what you got from installing the vanilla version.

***timc3*** They had decent bandwidth internally allowing them to playback uncompressed realtime standard definition video which normal PCs running Windows couldn’t do at the time.

***rbanffy*** The moment your product runs Windows, it'll compete with thousands of others. Being able to do one single thing better than the others won't help you much unless that one single thing can drive all your sales.

Even for video rendering, if your box is twice as fast, it'll be outcompeted by machines that cost half as much or less. At times my desktop PC was not fast enough, it was simpler to get another PC and run time-consuming things on it while I did other things on the other.

***hintymad*** I remember Clayton Christensen mentioned that Andy Grove invited him to Intel to talk about how to deal with the dilemma, and interrupted Christensen while he was talking and said something like "I know the problem, and I need you to tell me the solution". Similarly, Peter Drucker repeatedly mentioned one of the biggest challenges in business is "killing the cash cow". Along that line, Netflix's Reed Hasting is really amazing. He somehow managed to kill the DVD business and used it to milk the streaming business, when almost everyone in the industry and some of his lieutenants in Netflix didn't believe him.

***specialist*** Yes and:

These years later, while the innovator's dilemma thesis describes what, there's still little treatment of why and how.

I keep wanting someone to account for the roles of investment and finance.

Amazon's innovation was lower cost of capital. They convinced investors to wait for returns. And they got a massive tax holiday. (How could they not succeed?)

Ditto Tesla, with its saavy moves like govt loans, prepurchases, tax incentives, and selling direct.

That cheap capital was necessary, but not sufficient. Both still had to create products customers wanted.

I keep coming back to Apple. How'd Apple avoid the trap? Despite their terrible position. My guess is better financial strategy (if that's the right phrase). Apple focused on margins (and monosophony) instead of market share. And then leveraged their war chest to implement monosphony.

***system7rocks*** Apple also recognized one of its big problems was marketing to the end user, in particular in big box stores where not many people knew and appreciated what Macs could do. Creating their own stores created innovation and an experience, rather than less than knowledgeable staff and price tag comparisons against the latest Dell PC. That meant letting go of some of the traditional stores and the chain of legacy Mac resellers. Of course, now, you can get Macs at Costco.

***Spooky23*** Apple’s limited SKUs let them maximize their economies of scale. When Jobs took over, despite teeny market share, products like iMac and iBook were #1 in shipments, compared to Dell and their 300 models.

***microtherion*** Apple also did not hesitate to kill their cash cow, supplanting the iPod with the iPhone (an observation made, I think, by Horace Dediu).

***varjag*** iPod was killed quite late into iPhone era, when the sales have plummeted to unsustainable levels.

***Pet\_Ant*** By “killed” GP means “cannibalised” not “cancelled”. They out-competed the iPod while it was still hot.

***panick21\_||| Investors only 'wait' if you can show revenue growth and large market.***

***Tesla was very capital efficient compared to how hard the task was, only just enough to get the next product out. By the time they were making big loses, you could see the Model 3 was gone turn around once they reached scale. There was always a clear story of if we do X next, that means Y.***

***> I keep coming back to Apple. How'd Apple avoid the trap?***

***I think by changing CEO radically at the right time. Whatever was gone work for Apple, it wasn't any of the typical things you would think off. I'm not at all a Jobs fan, but I don't think many people could have turned it around.***

***Jobs had the ability to come back and radically shift the company. They also had the benefit of 20 years of loyal costumers, maybe the biggest asset Apple had was their loyal costumer base (or captured market). Personally I can say my dad was a complete Apple fan, he would just buy Apple. He simply didn't care the higher performance or any of the other things.***

***|||froonly*** For a while you could view Netflix online and rent DVDs from them.

***RandallBrown*** Oh dang, I thought you still could. Looks like they shut down the DVD rentals about 6 months ago.

***chiph*** There are now a couple of smaller disc-by-mail startups who aim to serve that market. I've signed up with DVDInBox out of Florida and they've done a good job thus far.

From what I've seen their biggest current challenge is their mailer. Netflix spent a lot of time designing their signature red envelopes, working with the USPS on ensuring they would survive a trip through the postal sorting machines. DVDInBox has yet to reproduce it - their envelopes sometimes arrive fairly mangled, or have not arrived at all (lost to the bowels of the machines).

***Octokiddie*** > It was not taken by surprise. But in spite of that, it was still unable to respond.

This is even a major point of discussion in the book. The incumbents always see it coming a mile away. They can't respond because doing so breaks their business model. Employees can't go to managers and say, "We need to enter this low-margin market for low-end products." Doing so is a good way to get sidelined or fired.

The "dilemma" is that either traditional option, compete with the upstarts or move further up-market, leads to the same outcome - death.

***appstorelottery*** I was making crazy money in the dot-com boom and bought a SGI 540 in 1999 (with an SGI monitor).

With money to burn SGI was a childhood brand, legends in 3D. Such wonderful memories. 15k on a desktop setup - it was loose change, however it shows how clueless I was back then. However I'd felt like I'd "arrived".

SGI with Windows NT - lol - I wrote my first OpenGL game in Visual Basic... I've always been somewhat of an outlier ;-) God help me.

The point? My personal experience says something about the strength of the SGI brand - even in the face of what was happening at the time (3DFX and so on - my previous company was one of the few 3DFX api devs - illustrating how clueless I was...)... it all happened so quickly... I'm not surprised SGI couldn't respond - or more importantly understand the strength of Microsoft/OpenGL/DirectX in the boiling pot of 3DFX / Nvidia and the rest... From memory it took three years and SGI was done - shared memory architecture? No longer worth the cost. :-(

Looking back, I was such a kid - a complete fool.

Greybeard advice: bet on the monopoly. Be smart. Brands like SGI are nothing in the face of install base. Think about how crazy it was to spend 15k on a desktop SGI back then... nostalgia is insanity, vanity.

***VelesDude*** The one thing that I always have to point out to folks who didn't live through it. The pace of change in everything was so rapid, especially in the graphics space.

It is wild to think that in games for instance, we went from Quake in 1996 running software rendering to Quake 3 requiring a GPU only 3 years later and that becoming the standard in a matter of months.

***lizknope*** Yeah but it still sounds really cool!

From 1991 when I first saw SunOS I wanted a SPARCstation. I started college in 1993 and the school was full of DEC Ultrix machines, Suns, HP PA-RISC, and a handful of IBM RS/6000 and SGIs.

I just thought DOS/Windows PCs were such garbage. Single user, no preemptive, multitasking, no memory protection. Then Linux came out and it changed everything. I bought a PC just to run Linux. My dream of a Unix RISC workstation faded away.

My roommate in 1996 bought a DEC Alpha. Not the cheaper Multia but an Alpha that could run OSF/1 Digital Unix. He actually ran NetBSD on it.

In 1997 I took the computer graphics class and we used SGIs. There was just one lab of them reserved for that class and grad students. I was so excited and it was really cool but I didn't think I could ever afford one. It's still really cool though that you had one.

***panick21\_||| Linux on RISC-V ... Its happening ... eventually ... probably.***

***|||yndoendo*** When statements like X was better than Y come up. I always think of "All models are wrong, but some are useful", from statistician George E. P. Box, and rephrase it as "All models are flawed, but some are useful" so that model becomes ambiguous, such as a smartphone, TV, computer, car, programing language, programing design pattern, social media platform, and so on.

Price-point, SGI technology was a financially flawed model pertaining to the growing market and more useful than flawed performance of the low cost technology market.

Did anyone at SGI try to simply buy the low tech products, play with them a bit, and see about slowly integrating your tech to make that low tech product just a little better than the competition and cost effect for the market?

***blackoil*** I believe having a talented dictatorial leader at top may be only solution. Like Steve Jobs, Bill gates or Jeff Bezos. Once they believe in a path, they have methods to get it done. Internet Tidal Wave memo is a good example of it. Zuckerberg is able to invest 100s of billions on a future he believes in.

Obviously the observation has a confirmation bias.

***pixelfarmer*** Companies through their usual hierarchical structuring are authoritarian by nature. However, many leaders are no actual leaders, they are just bureaucrats, and that is the reason why things get stuck.

***hintymad*** I guess an interesting question would be whether Nvidia is SGI in the late 90s or Intel of 80s.

***VelesDude*** I get the vague feeling they are more like Intel in the 80's. Nvidia has had a real talent partially by their own smart moves and partial just pure luck for stumbling from one win to another. Seeing them fall into the Crypto boom and then straight onto the AI boom has been wild to see.

***geor9e*** SGI was a pretty small 3D workstation company at the bottom of the S&P500, even at it's peak. Microsoft was at least a hundred times as big as SGI. (And Nvidia was even smaller, selling GPUs for both SGI worthstations and Windows PCs at the time.) Now Nvidia is a titan at #5 in the S&P. They could certainly be taken down a notch or two as CUDA alternatives mature and people start buying cheaper AMD/Intel/FPGA(groq, etc) hardware. But they're the best at what they do, with the best people. They don't really have a doomed business model the way SGI's "take out of a second mortage for the Onyx workstation to make N64 games" business model was doomed. I don't buy Nvidia stock, personally, but I especially wouldn't bet against them long term either.

***leetrout*** I have been comparing them to SGI of the late 90s but the next 18 months will prove me right or wrong as Intel, Apple, Google and AMD try to compete.

***foobarian*** I think it's a near impossible situation - the status quo is literally something that should not exist given the new market realities. Pivoting is pretty much asking a company to commit seppuku - asking the layers of leadership to basically replace themselves and quit in many cases. Which is pretty much what happens anyway.

***bunderbunder*** And, just like every Unix workstation vendor of the 1990s, they got hit with a perfect storm. They had their hardware being disrupted by x86 really coming into its own as a viable option for higher-end computing at the exact same time that Linux was becoming a serious option for the operating system.

"Literally something that should not exist" is the perfect way of putting it. In 1990, lots of people needed boutique workstation vendors. In 2000, nobody did.

***bunabhucan*** It's worse than that. Instead of "nobody" it was conservative slow moving vendor locked clients that could convince you to keep selling at those prices ("look at the margins!") instead of focusing on alternatives. I remember $5,000+ PCs being considered "cheap" workstations when those clients finally migrated.

***rbanffy*** Even Apple, which became the unlikely last of the Unix workstation vendors, was disrupted by Intel (and moved from PowerPC to x86 for a while). Ironically, Apple is now the very last Unix workstation vendor in existence.

***grumpyprole*** Yes, although they have clearly not always been committed to the Mac Pro over the years, it still exists and the pricing is certainly workstation like.

***musicale*** They are also well-established in certain markets like video and music production. And using their own silicon gives them some nice features like unified memory and power efficiency (and some not-so-nice features like lower system memory limits and non-expandability.) Apple also has some experience with AI, and hardware acceleration, so perhaps they will weather (or surf) the AI wave.

Apple abandoned both intel and Nvidia (not to mention AMD/ATI) which is interesting. Apple seems to be vertically integrated like IBM or SGI - but better at differentiation and at locking in new customers.

Apple seems particularly good at differentiated product design; their devices often just look better than the competition and seem to offer a better user experience.

(That being said, SGI's designs look pretty cool too.)

***rbanffy*** Every Mac is a Unix workstation these days. I work on a Macbook Pro and spend an inordinate amount of time running Unixy stuff in the terminal.

***AlbertCory*** > Pivoting is pretty much asking a company to commit seppuku

This is conventional wisdom (and thus, usually correct).

However, it's always interesting to look at counterexamples: Beretta, for example (in business for 500 years).

https://www.albertcory.io/lets-do-have-hindsight

or the IBM PC, which cannibalized IBM's business, at least in IBM's mind. Thus, they screwed it up and let Wintel make the real billions. So it worked, until they took your advice and decided that had to stop.

***ghaff*** And, at some point, what does it matter if the leadership and most of the employees turn over, typically involuntarily?

Is there any significance really to Foot Locker basically being a reorganized Woolworth's as opposed to being a brand-new company?

If you're big enough and have some product lines that still bring in a lot of money and didn't totally collapse like IBM you can sometimes pull it off. But it's hard.

***neuralRiot*** Probably the lack of vision is not just failing to turn into the direction of new “products” but not acquiring, digesting and eliminating those busines who start to grow before they’re too big. See Microchip for example, how many relatively small semiconductor and technologies manufacturers have already eaten.

***foobiekr*** I was at one of SGI's competitors and we had teams doing cheap HW - ATi and other cards, like IBM's GPUs at the time - and yet the company as a whole was like "ALL GOOD LET'S KEEP BUILDING MASSIVE CUSTOM GRAPHICS SYSTEMS!"

They were as dead as SGI in the same timeframe.

***bigboy12*** Walking the streets of nyc 19 years ago. I saw a sgi indigo on the street and sadly I just kept walking on. Like wow.

***mcculley*** I worked a lot with SGI hardware and IRIX in the 90s and very early 2000s. I ported a lot of code to and from Linux in that time.

I remember trying to explain to SGI reps that while we loved the SGI hardware, Linux on commodity x86 was the increasingly more effective choice for our projects. We wanted to buy SGI x86 boxes but they were pushing NT.

It was very apparent that SGI salesmen knew which way the wind was turning and they were milking the last bits of the commission gravy train.

Even when everyone understands “The Innovator’s Dilemma”, the incentives of salesmen and shareholders can be to optimize for extracting value out of the dying company. If I am a shareholder in a company being out innovated, it might make sense to want them to maximize short term profits while reallocating capital to the innovators instead of trying to improve the dying company.

***Pet\_Ant*** Seems like issue is a shift from high margin sales representative lead growth to low margin mass where you don’t need sales people to convince you to buy it. Then there is no way to justify the sales team, so they don’t support it. The company is beholden to them, and they sabotage the move to mass market.

***mcculley*** That is certainly one case, maybe the most likely. But another case I was involved with involved sales just not understanding the innovation. They could have potentially made just as much or more money with the new thing, but were too comfortable selling the old thing.

***bombcar*** The way to survive is to eat your own lunch. Be the low cost competitor and cannabilize your own market.

Otherwise you don’t build the iPhone because you don’t want to lose iPod sales.

***VelesDude*** I not so obvious but similar example of this would be with Nintendo and the Switch. They had always hand their handheld and consoles as separate things but eventually decided to just combine them into a single thing. So instead of double dipping on the hardware and potentially the software sales, it just become a single entity. And to say that has worked out well for them is an understatement. Sold 139 million units and are still taking their time of getting to a successor. They are probably going to take the #1 highest selling games machine of all time and they did it by going against their old business practices.

***musicale*** > Be the low cost competitor and cannabilize your own market. Otherwise you don’t build the iPhone because you don’t want to lose iPod sales.

The iPhone was a high-cost competitor to the iPod. And Apple knew that multi-function devices usually offered poorer user experiences than more focused, (mostly) single-function devices like the iPod. But they still did the iPhone.

***Animats*** They sort of tried. Around then they had a Windows NT machine that cost around US$12,000. But it was too late. The first serious graphics cards for PCs were appearing, from Matrox and others, with prices of a few thousand dollars.

(I tried some early NT graphics cards on a Pentium Pro machine. This was before gamer GPUs; these were pro cards from tiny operations. Fujitsu tried unsuccessfully to get into that business, with a small business unit in Silicon Valley. At one point they loaned me a Fujitsu Sapphire graphics card prototype. When I went back to their office to return it, the office had closed.)

Also, there was a bad real estate deal. SGI owned a lot of land where Google HQ is now. They sold it to Goldman Sachs in a sale and lease-back transaction, selling at the bottom of the market. That land, the area north of US 101 in Mountain View had, and has, a special property tax break. It's the "Shoreline Regional Park Community", set up in 1969. The area used to be a dump. Those hills near Google HQ are piles of trash. So there was a tax deal to get companies to locate there. That made the land especially valuable.

***msisk6*** SGI tried its hand at the PC video card business as early as 1990. I was at Autodesk at the time and got one of these to beta test on a DOS 486 running AutoCAD. It was an impressive product. But huge; it took up two full-length ISA slots. And the display drivers were a bit buggy.

***sillywalk*** Here's a brochure for the IrisVision boards - uses 2 ISA or Microchannel slots.

Prices start start at $3,495

https://www.1000bit.it/js/web/viewer.html?file=%2Fad%2Fbro%2...

***rbanffy*** I wish they ported IRIX to x86. You can make more money by making stuff for Windows, but it won't protect you from market erosion.

***nradov*** I doubt that would have helped. Customers didn't love Irix as an OS. They tolerated it in order to use SGI's superior hardware and specialized applications.

Competitors such as Sun did port their proprietary Unix flavors to the x86 PC platform but never achieved traction. It was impossible to compete with free Linux, and lack of device drivers was always an obstacle.

***rbanffy*** Kind of. Oxide uses what's left of OpenSolaris as their OS of choice and Oracle still sells Solaris SPARC servers. It's the retreat up - they can defend their hill for a long time that way and, in Oracle's case, they don't even innovate the same way IBM does with POWER and Z.

***Y\_Y*** Sounds just like Nvidia in 2024

***theandrewbailey*** Except Nvidia's modern cards are even bigger.

***masfoobar*** "taken by suprise"

When looking at 3d Acceleration cards on PCs, I believe PowerVR (and 3dfx) was released in 1996. Beforehand would have demonstrated their technology at conferences, shows.. with demos and adverts before they were released... this means SGI must have seen this coming as early as 1994.

In as little as 4 years... Graphics Cards were required for PC gamers!

***hulitu*** > The point being that the company was extremely aware of what was happening. It was not taken by surprise. But in spite of that, it was still unable to respond.

They had the best processor on the market, yet they decided to sell Intel and Windows. I really don't underestand what were they smoking.

***tambourine\_man*** I remember reading how Andy Grove, IIRC, spent two years trying to exit the DRAM business. He would flat out order people to stop working on it and they wouldn’t listen, believe or understand. The amount of inertia for large bodies is remarkable.

***Pet\_Ant*** And yet there are stories of people ignoring the company direction, continuing work, and saving the company.

See: The Blue LED https://youtu.be/AF8d72mA41M?feature=shared

So it’s not clear that the company knows better. Feels like educated guesses but a lof of luck involved.

***tambourine\_man*** Very true

***AtlasBarfed*** Why was exiting the DRAM business such an obvious positive?

Intel is basically at its root PC hardware. Yes, it doesn't dominate the entire industry, but memory is pretty key to a PC and the OS that runs on it.

In fundamentally it's another fab product. Like chipsets.

***tambourine\_man*** They were being crushed by the Japanese in price and went from the dominant player to barely registering the sales chart in one decade

https://anthonysmoak.com/2016/03/27/andy-grove-and-intels-mo...

***jiggawatts*** In the late 90s I was in the last year of high school. Silicon Graphics came to do a demo of their hardware for students that were interested in taking a computer science course at university in the following year.

The graphics demos looked like trash, basically just untextured and badly shaded plain colored objects rotating on the screen. For reference I was playing Quake III around the time which had detailed textures and dynamic lighting.

I asked the SGI presenter what one of his Indigo workstations cost. He said $40,000, not including the graphics card! That’s extra.

I laughed in his face and walked out.

***dekhn*** In the late 90s, SGI demos were much more impressive than what you describe. It was used by technical folks to do real stuff, with stringent criteria.

More importantly, the things that made Quake III so great were state-of-the-art for gaming. But those things couldn't render lines quickly and well (a mainstay of CAD at the time), or render at very high resolution (which IIRC was 1280x1024 in that era).

Here's what Carmack said abotu the SGIs a few years before: """SGI Infinite reality: ($100000+) Fill rate from hell. Polygons from hell. If you don’t trip up on state changes, nothing will come within shouting distance of this system. You would expect that.""" SGI was also key for map builds before PCs were capable.

But yes, 1999-2000 was just around the cusp of when SGI went from "amazing" to "meh".

***jiggawatts*** If I remember correctly, their cards had a high compute rate even at double precision, but had tiny memory buffers and basically couldn’t do texturing at all in practice.

It turned out that double precision was a mistake that was sold as a “professional” feature. By sharing edges correctly and using the correct rounding modes, single precision provides pixel-perfect rendering. Efficiencies like this allowed the consumer GPUs to run circles around SGI hardware.

***dekhn*** They did texturing just fine.

***jiggawatts*** Most SGI graphics cards had no texture ram at all, and had zero texturing capability. At the time there was one card that had 4 MB of texture RAM, but in the same year a typical PC GPU had between 16 or 32 MB of shared memory, most of which was used for textures.

A “budget” SGI card cost more than my parents’ car. I bought three different GPUs by that point with my pocket money.

***dekhn*** I honestly don'tt know what you're talking about- I wrote many programs that did high performance texture mapping on SGIs and they had both texturing capability and RAM. When you say "SGI card", it makes me sound like you're talkijng about something other than an Octane2 or InfiniteReality.

***sillywalk*** The Indy/Indigo2 graphics - the Elan/Extreme/XZ - never had texture hardware, but would fall back to software. It wasn't until the High IMPACT series was released for Indigo2 in 1995 they had hardware texture mapping on their "low end" - i.e. desktop sized systems.

***dekhn*** 1995 is when I started using SGIs in earnest (on Indigo2 and RealityEngine). I do recall a lab next to ours that had $15K indys that were absolutely useful for doing anything with graphics.

***sillywalk*** I always wanted an Indy. Preferably the WebForce one with all the extra software.

***epcoa*** You’re very confused. In 1992 you could get an SGI system equipped with 16MB of just texture RAM. Long before 3D accelerators for PCs. Later iterations such as InfiniteReality of course had much more texture memory. The SGI VPro could be configured with 108MB for texture memory.

***rbanffy*** The curve that maps fucking around with finding out is not linear. By the time you start finding out, it's very hard to stop finding out much more than you would like to.

***szundi*** Thanks for this comment, very much appreciated.

***dclowd9901*** Can you guess why? Was it maybe outmoded or outdated sales models that didn’t take reasonable competition into account.

I’m so sick of dancing around this topic. Managers and business types destroy companies. It never stops. Company after company. How many times do we have to pretend this isn’t the case before we see the truth.

Innovators make things worth selling. Business types drop it on the floor.

***voidmain0001*** Could this eventuality happen to Google, Nvidia, or Apple?

***infinite8s*** Why not? Google's only been a powerhouse for the last decade or so.

***davepeck*** I was there near the end. First, as a summer intern in 1998, and then in 1999 as a full time engineer on what is now Google's Mountain View campus. SGI had always been a dream company for me. I'd first learned about them in high school; now, right out of college, I'd somehow managed to land a dream job.

SGI's hardware was cutting-edge and exotic. IRIX was killer (sorry Solaris). Cray was a subdivision. My coworkers used emacs, too. They put an O2 on my desk!

The dream didn't last long. Major layoffs hit just a few months after I started full time. I wrote about the experience here: <https://davepeck.org/2009/02/11/the-luckiest-bad-luck/>

*The Luckiest Bad Luck? February 11, 2009*

*A few days ago I was speaking with a friend of mine who works at Microsoft. He joined direct from college; Microsoft is the only employer he’s ever known. He has now spent nearly ten years at one company. He confided that the recent cutbacks and layoffs were terrifying. For the first time in his ten-year career, he was forced to accept that nothing is forever. In his words, it felt like a “gut punch to the stomach.”*

*I’ve experienced a similar “gut punch to the stomach,” but it happened only a few months after I graduated from college. At the time, it seemed like terrible fortune. But it occurs to me now that perhaps this is the single luckiest thing that’s happened to me in my career.*

*After college, I joined Silicon Graphics as a full-time employee. My initial exposure to SGI dated back to a high-school internship at the Navy Research Labs in Washington, DC. Ever since then, I knew that SGI was the coolest computer company on the planet and I knew that I was going to work there someday. When I was a junior in college, I worked hard to get an internship at SGI; it was a fantastic summer in California. When I was a senior, SGI turned around and offered me a full-time position. This was my dream job! I signed and moved out to the bay area shortly after graduation.*

*It was 1999, and SGI was in dire shape. Not three months after I joined, nearly my entire division — all the way up to Rick Belluzzo, then the CEO — was axed. I was spared, as was another friend who joined at the same time, because we were recent college hires and HR didn’t want to piss off our university’s computer science department. Of course, our comrades vanished, as did our teams and products.*

*It was a miserable time. I was somewhere far away from home and far away from many of my friends. My dream job had fallen apart before I even got a chance to really dig in. What was I going to do next? Certainly, staying at SGI seemed like a mistake.*

*In retrospect, I think SGI’s downfall was just the sort of lesson I needed from the world of work. I’m lucky it happened ten years ago, not two weeks ago. It seemed like crazy times, so I decided to make a crazy decision. A few friends of mine were forming a little startup in San Francisco and wanted an engineer to join them. Given my recent experience, it seemed that working at a startup was about as sure a thing as working for an established company. I had enough money to last a few months and no real desire to go through another round of big corporate interviews. So, why the heck not join?*

*The rest, as they say, is history. EQuill became an exciting organization full of brilliant people and lots of forward motion. Microsoft acquired EQuill in late 2001 and in January 2002 I moved up to Seattle to join Microsoft as a full-time employee. Not a bad outcome at all.*

*Most importantly, I think SGI’s kick in the pants nearly a decade ago granted me the ability to leave Microsoft in 2008 and strike out on my own. The last nine months as a consultant have been an exciting ride full of new people, technologies, and possibilities. They’ve been a little scary, of course, what with the wintry economic climate. But as my friend recently discovered, safety is never a guarantee — not even at Microsoft. I’m excited to see if I can put together a long-term-sustainable consultancy on my own, and I believe I owe it all to the luckiest bad luck in my career.*

***dxbydt*** > SGI had always been a dream company

It was a dream company for pretty much every siggraph person at that time. I was in grad school, eagerly awaiting a very popular 3-semester course in computer graphics. It had been devised and taught by a young promising professor who had published some pioneering siggraph papers. I signed up for the course. On the first day of class, the head of the department walked in and said the professor had been recruited by his dream company SGI for an ungodly sum of money to work on some Jewish director’s movie about a dinosaur themepark. I thought ok, whatever, someone else will teach the course. The bastards scrapped the entire 3 series computer graphics module because there wasn’t anyone else who could teach that. So we had to pick from one of the usual dumb options - databases, OS, Networks, Compilers. Since then I’ve always held a grudge against sgi.

***krapp*** > to work on some Jewish director’s movie about a dinosaur themepark

I assume you mean Steven Spielberg and one of the Jurassic Park films?

If so, why can't you just say so? Why are you referring to Steven Spielberg, one of the most famous directors of all time, as "some Jewish director?" Do you think people won't recognize the name? I promise people know who Steven Spielberg is.

***Kamq*** > If so, why can't you just say so?

Based on the comment, it sounds like that's the way the head of the department phrased it.

Presumably the department head didn't know the title as it hadn't been released yet.

***theonething*** I think the GP was telling their story in the context of that time. It's a technique to help the reader more fully understand the context. I'm almost sure there is a term for this literary technique.

***IntelMiner*** "Jewish director" is an...interesting description

***brcmthrowaway*** Jewish director? Hrmph

***Y\_Y*** Spielberg had a bar mitzvah, what more do you want?

***krylon*** I don't anyone is disputing that, but why did the department head see a need to point that out?

***oaktowner*** I worked at Google from 2013 to 2020. There were definitely employees (maybe a majority) who assumed that Google would always be the dominant force in technology. Those of us who were a bit older always understood that everything changes in Silicon Valley.

Those buildings represented that change to me. I can remember coming to concerts at the Shoreline in the 90s and looking at those Silicon Graphics buildings: they looked so cool, and they represented the cutting edge of technology (at the time). And yet...it all disappeared.

Same goes for the Sun campus which is where Meta/Facebook is now. Famously, the Facebook entrance sign is literally the same old Sun sign, just turned around! [0]

So I always cautioned co-workers: this too, shall pass. Even Google.

[0] <https://www.businessinsider.com/why-suns-logo-is-on-the-back>...

***There's A Hidden Message On The Sign Outside Facebook's Campus — It Reminds Employees To Stay Motivated***

Alyson Shontell Dec 7, 2014, 8:24 AM EST www.businessinsider.com

Facebook's sprawling campus in Palo Alto, California was previously owned by Sun Microsystems.

Sun Microsystems was acquired by Oracle in 2009.

When Facebook moved into the office, Mark Zuckerberg didn't replace Sun Microsystem's sign. Instead, he flipped it over and put Facebook's name on the front.

Why?

The Sun logo reminds employees to stay motivated. It demonstrates what can happen when you're on top but fail to innovate.

Meta's secret plan to 'clone' Snapchat revealed in court documents

Lev Grossman interviewed Mark Zuckerberg for Time. Grossman writes about the sign's symbolism here:

Because of the limits of space and time, a lot of Silicon Valley companies don’t build new headquarters; they just take over the discarded offices of older firms, like hermit crabs. Facebook’s headquarters used to belong to Sun Microsystems, a onetime power-house of innovation that collapsed and was acquired by Oracle in 2009. When Facebook moved in, Zuckerberg made over the whole place, but he didn’t change the sign out front, he just turned it around and put Facebook on the other side. The old sign remains as a reminder of what happens when you take your eye off the ball.



sun microsystems sign at facebook campus

Here's the back of the sign:



***dbreunig*** Meta still has the Silicon Graphics logos on a few glass conference room doors in building 16, I believe. At least they were there in 2012.

Great memento mori.

***samatman*** Presumably you mean the Sun logo: <http://www.logobook.com/logo/sun-microsystems/> Which is one of the all-time greats IMHO. I'd keep it around too.

***dbreunig*** I do! Thanks!

***ska*** > SGI's hardware was cutting-edge and exotic.

This was their downfall, trying to scale out adoption with esoteric hardware.

I remember being quoted $18k ish for memory upgrade on a O2 or origin, same amount of memory I had just bought for $500 for an intel Linux box at home.

Sure, it wasn’t apples to apples, but I remember thinking very clearly that this wasn’t going to end well for SGI.

***alecco*** I had to support an open source library for all major unixes and the Irix compiler was by far the best one. It took years for the rest to catch up. But it took ages to compile with optimizations on. Good times.

***ryandrake*** I graduated undergrad in 1998 and can confirm that SGI was the company to go to. I felt so jealous of those few guys who had SGI offers, where I had to settle for a more generic PC graphics company. History is what it is but the SGI really had that luster that only a handful of companies ever boasted.

***e40*** Beg to differ on IRIX. I always hated it as an ISV. Solaris was way better to work with.

***lowbloodsugar*** That's funny because my reaction to the O2 was "oh, this is far too expensive for what it is". Was workin on N64 game, and the other teams were using the Indy devkits, while we had PCs with the SN systems dev kits. Writing was on the wall at that point.

***davepeck*** Yeah, the O2 definitely was too expensive for what it was. And while it was the least cool and powerful of the lineup by far, as a recent college grad, it was still the coolest computer I had ever had on my desk. ;-)

***mrpippy*** What did you work on at SGI during your brief stint?

***davepeck*** MineSet, their data mining and visualization package.

***bcantrill*** If a may, can I fact check a story conveyed to me through a mutual acquaintance of ours? The story was that SGI was trying to sell off MineSet, and needed the team to stick around long enough to sell them off -- so a bonus was to be given after a short period of time (a month maybe?). The bonus was significant enough to get people to at least defer a job search ($10K?), but SGI didn't manage to find a buyer. The check was to hit bank accounts on a particular day; the team waited to hear word that the literal money was in the bank -- and then all quit simultaneously.

Is there at least some truthiness to it? Or has this just become Silicon Valley urban legend in my head?

***davepeck*** That rings a bell although fuzzily: as the new kid from school, I was pretty disconnected from the politics of the moment. I do seem to remember that the MineSet team departed en masse, but IIRC that departure roughly coincided with broader layoffs in the org.

(With apologies for reviving 90s IRIX/Solaris snark in my earlier post. :-)

***bcantrill*** Ha -- no worries on the snark; Irix probably was a better system in ~1998, as ZFS, DTrace, Zones, SMF, FMA, etc. were all still in the future...

***dekhn*** I was in love with SGI when I was an undergraduate just over the hill at UC Santa Cruz in the early to mid 90s. Everything about the machines from their industrial designed cases but wonderfully colorful cases, and the sexy desktop OS ("This is UNIX. I know this!") and the way IrisGL rendered molecular graphics.

Driving to a Phish show at Shoreline, we passed the low-slung office buildings of SGI which seemed like the sexiest place to work. When I graduated, I thought I was "too dumb in CS" to get a job in Mountain View and went to grad school in biophysics instead.

By the time I was a few years into grad school, I worked in a computer graphics lab outfitted with Reality Monsters and Octanes and other high end SGIs (when you maxxed out an SGI's graphics and RAM, they were really fast). I was porting molecular graphics code to Linux using Mesa (much to the derision of the SGI fans in the lab). When we got a FireGL2 card it had a linux driver and could do reasonable molecular graphics in real time and the SGI folks looked real scared (especially because the SGI Visual Workstation had just come out and was a very expensive turkey).

Less than a decade after that I was working in those very buildings for Google. Google took over SGI's old HQ (Jeff Dean told me there was a period where Google and SGI overlapped in the GooglePlex and the SGI folks looked very sad as they paid for their lunches and teh googlers got free food). There was still plenty of SGI signage strewn about. And now Google has gone dumb and also built their own HQ next door (note the correlation between large SV companies building overly fancy HQs and then going out of business).

Such is the cycle of sexy tech.

***dalke*** We've talked before about our respective molecular graphics background.

I started with Unix on a Personal IRIS as an undergrad working in a physics lab which used it for imaging capture and analysis. I was the nominal sys admin, with one semester of Minix under my belt and just enough to be dangerous. (I once removed /bin/cc because I thought it was possible to undelete, like on DOS. I had to ask around the meteorology department for a restore tape.)

The summer before grad school I got a job at the local supercomputing center to work on a parallelization of CHARMm, using PVM. I developed it on that PI, and on a NeXT. That's also when I learned about people at my future grad school working on VR for molecular visualization, in a 1992 CACM article. So when I started looking for an advisor, that's the lab I chose, and I became the junior co-author and eventual lead developer of VMD.

With a Crimson as my desktop machine, a lab full of SGIs and NeXTs, and the CAVE VR setup elsewhere in the building. Heady times.

I visited SGI in 1995 or so, on holiday, thinking that would be a great place to work. They even had an Inventor plugin for molecular visualization, so I thought it would be a good lead. I emailed and got an invited to visit, where the host kindly told me that they were not going to do more in molecular visualization because they wanted to provide the hardware everyone uses, and not compete in that software space.

In the early 1990s SGIs dominated molecular modeling (replacing Evans & Sutherland), so naturally the related tools, like molecular dynamics codes, also ran on them. But we started migrating to distributed computing, where it didn't make sense to have 16 expensive SGIs, leaving them more as the head .. which as you pointed out, was soon able to run just fine on a Linux machine.

***DaiPlusPlus*** Pardon my ignorance, but what is so unique or special about molecular visualisation compared to, say, Quake - or CAD? If you’ll permit me to reduce it down to “just” drawing organo-chem hexagons, lines, and red/grey/black spheres connecting those lines (and a 360-degrees spin animation for the investor-relations video) - where’s the room for the rest of CG? E.g., texture-mapping, fragment shaders, and displacement mapping?

***dalke*** Quakes came out in, what, 1996? And was written by some of the foremost practitioners of computer graphics?

We were a couple of physics grad students working on a side project in late 1993. My background was a semester course based on Foley & van Dam. Hardware gave us a 5-10 year lead over what we could have done with consumer tech.

There wasn't really a "rest of CG". Only the highest-end SGI machines at the time had hardware texture mapping - most did it in software (see https://en.wikipedia.org/wiki/Extreme\_Graphics).

We aren't talking 2D organo-chem hexagons, but 3D spheres and cylinders. Back around 1995 I posted some benchmarks to Usenet about the different approaches I tried (including NURBS), but I can no longer find a copy of it.

The straight-forward way is to render the spheres as a bunch of triangles, so, what, 50 polygons per sphere? Times 100,000 spheres = 5 million polygons. That was large for the time, but doable. Plus, during movement we used a lower level of detail.

What was Quake's polygon count?

Oh, and we're displaying animated molecules, including interacting with a live physics simulation, so no pre-computed BSP either.

Rastering spheres quickly on a PC was also possible then, which was RasMol's forte, but it was flat compared to having a couple hardware-based point lights plus ambient lighting.

Interestingly, AutoCAD (RIP Walker) tried to get into molecular modeling, but it didn't work out. https://www.fourmilab.ch/autofile/e5/chapter2\_82.html

***justsomehnguy*** 3D games make a lot of smokescreen and mirrors to make you believe you see a lot.

Not looking it up right now but the original Q1 had a very low poly count.

***theideaofcoffee*** Oh how I lusted over the Challenges, the Octanes, the Indigo2s of the time. It was a revelation when I finally was able to sit down at a console of an Octane (with two, count 'em TWO R14000 and a whopping 2.6G of RAM), tooling around in IRIX via 4dwm was so much more satisfying than today's UIs. It was snappy and low-latency unlike anything I've used since.

Later on, I was able to do some computational work on an Altix 3700 with 256 sockets and 512G of RAM spread over four full-height cabinets with the nest of NUMAlink cables at the back), at the time running SuSE linux and that was wild seeing the 256 sockets being printed out with a cat /proc/cpuinfo. Now the same capabilities are available in a 4U machine.

The corporate lineage story is also just as interesting as the hardware they made as well. Acquisition, spinoff, acquisition, rename, acqusition, shutter, now perhaps just a few books and binders and memories in the few remaining personnal at HPE are all that's left (via Cray, via Tera, via SGI, via Cray Research).

RIP SGI

***bitbckt*** I still keep a maxed out Octane2 in running order for posterity. Occasionally logging in to it reminds me just how a desktop environment should feel. We truly have lost something since then.

***ofrzeta*** If you feel nostalgic you can run https://docs.maxxinteractive.com/ on Linux.

***hypercube33*** I really wish they'd do movies like they made for RIM about: Cray, DEC, Compaq, SGi, Pixar. sounds like these places were either wild or strait up IBM culture or some clash or both either inside or outside. Raven and id Software would be neat too. Westwood Studios.

***latchkey*** Back in 1993, I was in college and working for the extended education department running their all their computer infrastructure.

One day, someone wheeled this approx. 3x3 foot sized box to my door and asked me if I wanted it. It was a SGI Onyx with a giant monitor sitting on top, with a keyboard and mouse.

I plugged it in and it sounded like an airplane taking off. It immediately heated up my entire tiny office. It was the 4th Unix I had ever played with (Ultrix, NeXT and A/UX were previous ones). It had some cool games on it, but beyond that, at the time, I had no use for it because A/UX on my Quadra950, was so much more fun to play with.

I don't even think I ever opened it up to look at it. I don't know what I was thinking. lol.

After realizing it did not have much going for it, I ended up just turning it on when the office was cold and using it as a foot rest.

Oh yea, found a video... https://www.youtube.com/watch?v=Bo3lUw9GUJA

***Firmwarrior*** Holy shit, that thing is amazing compared to the computers I had access to in the early 90s

***vrinsd*** I believe nVidia was started with a lot of SGI's core technology ; not "I have a good idea and I can't do it here" and more like "let me just take this stuff I doubt anyone will notice". I think SGI sued but didn't really pursue the matter because they didn't really see nVidia as a threat. I think Jensen was pivotal in this "technology transfer".

Regarding computing cycles, boom/bust, I recently re-read Soul of New Machine and was struck by how much the world has NOT changed. Sure we're not talking about micro/mini-computers and writing micro-coded assembly but the whole "the market is pivoting and we need to ride this wave" and "work like a dog to meet some almost unobtainum goal" seems to still underpin being an engineer in "tech" today.

***alecco*** (1999) https://www.eetimes.com/sgi-graphics-team-moves-to-nvidia/

***markus\_zhang*** I love "Soul of New Machine" too. It was a blast to read. It even made me ponder the possibility to start over at 40+ and do something hardware-wise (or very low-level software). Of course I then found myself drown by 2 mortgages and dropped the thought.

***formerly\_proven*** From my reading SGI was already dead and falling apart by that time. If you look at 3D, SGI had two graphics architectures in the 90s: RealityEngine from 1992 and InfiniteReality from 1996. They never managed to release a follow-up to IR. Similarly everything that came after about 1996-97 was a refresh of a prior product with only marginal changes. And then they went bankrupt in the early 2000s. So SGI had really only a very brief productive period that was over by the second half of the 1990s.

SGI also never had a presence in business critical applications which gave some of the other vendors more momentum (HP-UX/PA-RISC, VMS/Alpha, Solaris/SPARC).

***cuno*** I worked at SGI on the next generation (code named Bali) in 1998 (whole year as an intern) and 1999 (part time while finishing my degree, flying back and forth from Australia). Bali was revolutionary. The goal was realtime Renderman and it really would. I had an absolute blast. I ended up designing the highspeed data paths (shader operations) for world's first floating point frame buffer (FP16 though we called it S10E5) with the logic on embedded DRAM for maximum floating point throughput. It was light years ahead of its time. But the plug got pulled just as we were taping out. Most of the team ended up at Nvidia or ArtX/ATI. The GPU industry was a small world of engineers back then. We'd have house parties with GPU engineers across all the company names you'd expect, and with beer flowing sometimes maybe a few secrets could eh spill. We had an immersive room to give visual demos and Stephen Hawking came in once pitching for a discount.

For team building, we launched potato canons into NASA Moffet field, blew up or melted Sun machines for fun with thermite and explosives. Lots of amazing people and fond memories for a kid getting started.

***mrpippy*** Very cool. Was Bali going to be the next high-end architecture after InfiniteReality? (I think IR was code-named “Kona” so the tropical codenames fit)

Why did they cancel it, money running out? It’s sad to think they were close to a new architecture but then just kept selling IR for years (and even sold a FireGL-based “Onyx” by the end).

Also was it a separate team working on the lower-end graphics like VPro/Odyssey?

***cuno*** Yes Bali was the next gen architecture and incredibly scalable. It consisted of many different chips connected together in a network that could scale. The R chip was so big existing tools couldn't handle it and ppl were writing their own tools. As a result it was very expensive to tape out so many hefty chips and I think that's why when it came time, and with a financial crisis, upper management pulled the plug.

Yes there were separate teams working on the lower-end graphics.

***phonon*** Why was Bali cancelled?

***vrinsd*** Well,

Most Hollywood effects were all done on SGI systems before the slow migration to Linux. Renderman, Maya, were all SGI first-party programs.

Also SGI made huge advances in NUMA and machines with dozens of CPUs/processors before most other companies ventured into this space.

But not business critical like IBM CICS or Java.

1. https://en.wikipedia.org/wiki/NUMAlink

2. https://www.cs.ucr.edu/~bhuyan/CS213/2004/numalink.pdf

3. https://cseweb.ucsd.edu/classes/fa12/cse260-b/Lectures/Lec17...

***sllabres*** The large Origin servers and the nice indigo workstations at trade fairs with their cool real time visualizations comes into my mind. Also applications like Softimage, the 4Dwm desktop ...

Later the large Altix NUMA systems with core counts in unprecedented sizes (and problems booting due to lock contention ;)

And of course their donation of the XFS filesystem to the linux world!

***GPUrender123*** The N and the V in Nvidia stood for "New Version" of SGI firmware.

***miohtama*** The Nvidia lawsuit is discussed in the article.

***DeadFred***\_ I mean there's more to it. NVidia literally just took SGI's IP. Only more blatant start was Cisco, where they straight stole a University computer.

***takinola*** Cisco was started by a husband/wife team who were the heads of IT for the Stanford Electrical Engineering School and Business School respectively. Anecdotally, they first developed the technology trying to connect the networks for both schools.

***meekaaku*** where can i read about this cisco thing?

***vrinsd*** https://www.tcracs.org/tcrwp/1origin-of-cisco/

***assimpleaspossi*** I was a system engineer for SGI in 1992 working mainly with McDonnell-Douglas in St Louis. It was thrilling to be sitting in the cafeteria and have Jim Clark plop down next to me for lunch. Just one of the guys.

As an outsider--cause I didn't live and work in California--this was the go-go atmosphere of such companies back then where they thought they could do no wrong. And the after work parties were wild (how the heck do you break off half a toilet bowl?).

One of the buildings had plastic over the windows cause that's where they were working on the plugin GL card for the PC. (Ssh! No one's supposed to know that!)

Being the first system engineer in St Louis, my eyes lit up when my manager told me he had ordered an 16-core machine for my office--just for me!

I was hired as a video expert. The company re-org'ed and my new boss decided he needed a Fortran expert so that was the end of my job with SGI.

***ChrisMarshallNY*** I remember having a Personal Iris, at the company I worked at, and, later, an Indigo. We never used them. I think they were really there, to impress the visitors (They were in our showroom).

I remember the colors as being very different, from the photos, though.

The Personal Iris was a deep purplish-brown, and the Indigo was ... indigo.

Jim Clark sounds like my kinda guy. I made a hash of my teenage years, and barely squeaked in, with a GED, myself. It has all worked out, OK, in the end, though.

***randomdata*** > We never used them.

When I was in high school we had a lab full of SGI machines. They also never got used. Hundreds of thousands of dollars of computing equipment, and probably that much again in software licenses (at the commercial rate), just sitting there doing nothing. It was heartbreaking.

On a happy note, the SGI bus (a semi-trailer full of SGI machines demoing their capabilities) came to school one time. As a teenage nerd, getting to play with a refrigerator-sized Onyx2 was a good day.

***mrpippy*** My goodness, at a high school? Like Indys, or O2s? Was this a private school?

***randomdata*** They were O2s. Rural public school.

There were all kinds of toys, though. There was a dedicated classroom setup for video-based remote learning some 30 years before COVID - that got used for one semester, from what I gather (was never used while I was there). The school was even host to a dialup ISP at one point.

The administrators were all in on technology. The teachers, not so much...

Eventually, in my last year, the government changed the funding model and the party ended.

***fuzztester*** >The Personal Iris was a deep purplish-brown

>Personal

>Iris

Must have been the color of their lover's eyes.

https://en.m.wikipedia.org/wiki/Iris\_(given\_name)

***fuzztester*** >The Personal Iris was a deep purplish-brown, and the Indigo was ... indigo.

Nice.

I once worked at a startup that had a Cobalt Qube in the server room, and the Cobalt was ... cobalt blue.

https://en.m.wikipedia.org/wiki/File:Cobalt\_Qube\_3\_Front.jpg

https://en.m.wikipedia.org/wiki/Cobalt\_Qube

***selimthegrim*** Every time I get depressed about being at Tulane I think about Jim Clark.

***tombert*** There's a few cases in the history of computers where it feels like the world just "chose wrong". One example is the Amiga; the Amiga really was better than anything Apple or Microsoft/IBM was doing at the time, but for market-force reasons that depress me, Commodore isn't the "Apple" of today.

Similarly, it feels like Silicon Graphics is a case where they really should have become more standard. Now, unlike Amiga, they were too expensive to catch on with regular consumers, but I feel like they should have become and stayed the "standard" for workstation computers.

Irix was a really cool OS, and 4Dwm was pretty nice to use and play with. It makes me sad that they beaten by Apple.

***jandrese*** SGI dug their own grave. Not only were the workstations expensive, but they demanded outrageously priced support contracts. This behavior drives people nuts and will insure that the switch to a competitor the instant it becomes an option. Despite the high cost, the support contracts had a pretty lousy reputation as well, with long wait times for repairs from a handful of overworked techs. Even worse is the company turned away from its core competencies to focus on being an also-ran in the PC workstation market.

There was a window in the mid-90s where it would have been possible for SGI to develop a PC 3D accelerator for the consumer market using their GE technology, but nobody in the C-Suite had the stomach to make a new product that would undercut the enormous profit margins on their core product. It's the classic corporate trap. Missing out on the next big thing because you can't see past next quarter's numbers. Imagine basically an N64 on a PCI card for $150 in 1996. The launch versions could be bundled with a fully accelerated version of Quake. The market would have exploded.

***cduzz*** Ugh.

Worked at a university in the early 90s.

Maybe irix was okay to use if you were just sitting in front of it doing rando user / graphics things, but administering it was unbearable. The license fees to get OS updates were exorbitant; you'd have to get wacky new licenses to enable NFS or NIS and you'd need new kernels for just about anything.

As far as I could tell they were a cursed company that hated their users. "Here's a pretty thing that does one thing well but is otherwise insane and will ruin you when you need it most."

Good riddance.

***grumpyprole*** > The market would have exploded

Absolutely, they could have been where Nvidia is now!

***Keyframe*** I'd argue Nvidia is ex-SGI, and so is ex-ATI. It's all their crew in the beginnings.

***foobarian*** I wish we could have a debugging view of the universe, draw a diagram with clusters of people labeled with company names, and watch them change over time. :-)

***jmtulloss*** This view would certainly explain to people outside of Silicon Valley/ SF why the Bay Area has been so dominant in our industry for so many years.

***christkv*** Or they could have 3dfxed themselves.

***snakeyjake*** >One example is the Amiga; the Amiga really was better than anything Apple or Microsoft/IBM was doing at the time

Amiga was only better 1985-1988.

I still have my original Amiga and A2000. I was an Amiga user for a decade. They were very good. I was platform agnostic, caring only to get work done as quickly and easily as possible so I was also an early Macintosh user as well as Sun and PA-RISC. And yes, I still have all of those dinosaurs too.

By 1987 PC and Mac caught up and never looked back.

But by 1988 the PS/2 with a 386 and VGA was out and the A2000 was shipping with a 7MHz 68000 and ECS.

By 1990 the 486s were on the market and Macs were shipping with faster 030s and could be equipped with NuBUS graphics cards that made Amiga graphics modes look like decelerated CGA.

After the A2000 the writing was on the wall.

Note: my perspective is of someone who has always used computers to do work, with ALMOST no care for video games so all of the blitter magic of Amiga was irrelevant to me. That being said when DOOM came out I bought a PC and rarely used my Amigas again.

What I can confidently assert is that I upgraded my A2000 many times and ran into the absolute configuration nightmare that is the Amiga architecture and the problems with grafting upgrades onto a complex system with multiple tiers of RAM and close OS integration with custom chips.

One more bit of heresy is that I always considered Sun's platform to be superior to SGI's.

***icedchai*** I think you are mostly right, I just think your timing is off. Those early 386 machines and Mac II systems were very expensive, at least 2 to 3x the cost of an Amiga. The average home user wasn't going to drop $8K on a PS/2 model 80 with a 386/16.

By the early 90's the Amiga just wasn't competitive. The chip set barely evolved since 1985. ECS barely added anything over the original chip set. By around 1992 or 1993, 386 systems with SVGA and Soundblaster cards were cheap. Amiga AGA was too little, too late. Also consider the low end AGA system (Amiga 1200) was totally crippled with only 2 megs of slow "chip" RAM.

I was an Amiga fan until 1993. Started with an A500, then A3000. Eventually I moved on to a 486 clone w/Linux. Later on I had a Sun SparcStation 10 at home, so I agree with you on Sun and SGI.

***logicprog*** > Amiga was only better 1985-1988. By 1987 PC and Mac caught up and never looked back.

Oh indubitably! I don't think even the most committed Amiga fan, even the ones that speculate about alternate histories, would deny that at all.

The thing is, though, that only happened because Commodore essentially decided that since it had so much of a head start, it could just rest on its laurels and not really innovate or improve anything substantially, instead of constantly pushing forward like all of its competitors would do, and so eventually the linear or even exponential curve of other hardware manufacturers' improvements outpaced its essentially flat improvement curve. So it doesn't seem like IBM PCs and eventually even Macs outpacing the power of Amiga Hardware was inevitable or inherent from the start.

If they had instead continued to push their lead — actually stuck with the advanced Amiga chips that they were working on before it was canceled and replaced with ECS for instance — I certainly see the possibility of them keeping up with other hardware, and eventually transitioning to 3D acceleration chips instead of 2D acceleration chips when that happened in the console world, eventually perhaps even leading to the Amiga line being the first workstation line to have the gpus, and further cementing their lead, while maintaining everything that made Amiga great.

Speculating even further, as we are seeing currently with the Apple M-series having a computer architecture that is composed of a ton of custom made special purpose chips is actually an extremely effective way of doing things; what if Amiga still existed in this day and age and had a head start in that direction, a platform with a history of being extremely open and well documented and extensible being the first to do this kind of architecture, instead of it being Apple?

Of course there may have been fundamental technical flaws with the Amiga approach that made it unable to keep up with other hardware even if Commodore had had the will; I have seen some decent arguments to that effect, namely that since it was using custom vendor-specific hardware instead of commodity hardware that was used by everyone, they couldn't take advantage of the cross-vendor compatibility like IBM PCs, could and also couldn't take advantage of economies of scale like Intel could, but who knows!

***panick21***\_||| The thing with Commodore was that as a company it was just totally dysfunctional. The basically did little useful development between C64 and the Amiga (the Amiga being mostly not their development). The Amiga didn't sell very well, specially in the US.

The company was going to shit after the Amiga launched, it took a competent manager to save the company and turn the Amiga around into a moderate success.

Commodore didn't really have money to keep up chip development. They had their fab they would have need to upgrade that as well, or drop it somehow.

Another example of that is the Acorn Archimedes. Absolutely fucking incredibly hardware for the price. Like crushing everything in price vs performance. But ... literally launched with a de-novo operating system with 0 applications. And its was a small company in Britain.

The dream scenario is for Sun to realize that they should build a low cost all costume chip device. They had the margin on the higher end business to support such a development for 2-3 generations and to get most software ported to it. They also had the software skill to make the hardware/software in a way that would allow future upgrades.

***logicprog*** Imagining Sun buying Amiga and making it a lower end consumer workstation to pair with its higher end ones, with all the much-needed resources and interesting software that would have brought to the Amiga is a really cool thought experiment!

***panick21***\_ Sun did actually approach Commodore to license its technology for low end work station. However the Commodore CEO at the time declined for unknown reasons.

I don't know what Sun had planned for this tech.

A even more interesting approach for Sun would have been to cooperate or acquire Acorn. The Acorn Archimedes was an almost perfect low end work station product. Its incredible weakness was its lack of OS and it total lack of applications.

Acorn spend an absolutely absurd amount of money to try to get the OS and application on the platform. They spend 3 years developing an new OS, and then realized that this was going nowhere. So they rushed out another new OS. And then they realized that nobody want to buy a machine with a compromise OS and no application. So they had to put up huge effort to try to fix that. The company simply couldn't sustain that kind of effort on the Software side while at the same time building new processors and new machines. Its surprising what they achieved but it wasn't a good strategy.

Had they just adopted SunOS (BSD) it would have been infinitely better for them. And for Sun to release new high and and low end RISC workstations at the same time would have been an absolute bomb in the market.

Even if you added all the bells and whistles to the system (Ethernet, SCSI, extra RAM), you could be very low priced and absolutely blow pretty much every other system out of the water.

***logicprog*** That's really interesting information!

Re Acorn though — As much better from a market perspective as buying Acorn and releasing RISC- and BSD-based low-end workstations might have been for Sun, I still prefer to imagine a world where the Amiga's unique hardware and software got to live on — perhaps with compatibility layers to run Sun software, but nevertheless preserving a UNIX-like but still non-UNIX OS lineage and non-generic-PC hardware lineage.

***pjmlp*** From retrogaming talks from former Commodore engineers, the issues were more political and management than technical alone.

***AnimalMuppet*** That's kind of typical, though, isn't it? When a company falls off, it's almost always not just technical.

***logicprog*** That's definitely how it seems to me, which is why I focused on Commodores poor management decisions first and only mentioned the possible technical issues second

***pjmlp*** It took a bit more than 1990, for PC 16 bit sound card, Super VGA screens, with Windows 3.1 to be widely adopted for the PC to out perform the Amiga, specially in European price points.

My first PC was acquired in 1992, and still only had a lousy beeper, on a 386SX.

***geophile*** I was similar, not really interested in graphics, just a nice programming environment. PCs had that stupid segmented address space (which was not ignorable at the programming language level), expensive tools, and crappy OSes. My Amiga 2000 had a flat address space, a nice C development environment, and multitasking actually worked. It really was ahead of its time, in combining a workstation-like environment and an affordable price.

***snakeyjake*** >My Amiga 2000 had a flat address space

Chip ram, fast ram, cpu ram, expansion board ram, or slow ram? Did too much ram force your zorro card into the slooooooooooow ram address space (mine did)? Tough cookies bucko!

Macintosh, pounding on table: "RAM is RAM!"

***geophile*** One thing I do remember about Amiga RAM is that some (all?) of it would survive reboot! That was very handy.

***snvzz*** It's the same in most computers. Wiping RAM is effort.

The feature here is that AmigaOS will try and reuse the ExecBase structure if found.

Such structure has a checksum, which is checked. If the check fails, a new one is made. This happens e.g. on power on, or after running games that are not system friendly (i.e. most games).

But if the check passes, this structure has important information, such as a list of memory regions, the "cold/cool/warm" vectors, which are function addresses that get called if non-zero at different points of the boot process (non-surprisingly a virus favorite), as well as and a list of reset-resident modules, which become allocated memory, thus protecting them.

A popular such device implements a reset-resident memory-backed block device, which the Amiga is able to boot from.

***logicprog*** As someone trying to get into Amiga retro competing as a hobby in today's day and age, I find it keeping all the different types of ram straight very confusing lol

***geophile*** This was a loooooong time ago. I have no clue.

***dylan604*** We kept our A2000 viable longer by adding the CPU board with the 030 chip. We went from 7MHz to somewhere around 40MHz or whatever. It meant that my Lightwave render went from 24 hours per frame to a few hours per frame.

***qqtt*** My main problem with Silicon Graphics (& have the same problem with Sun Microsystems) is that they just tried to do too much in propriety hardware and completely resisted standards. Microsoft & IBM "won" because they made computers with actual upgrade paths and operating systems with wide support among upgrade paths. With SGI/Sun you were very much completely locked in to their hardware/software ecosystem and completely at the mercy of their pricing.

In this case, I think the market "chose right" - and the reason that the cheaper options won is because they were just better for the customer, better upgradability, better compatibility, and better competition among companies inside the ecosystems.

One of the most egregious things I point to when discussing SGI/Sun is how they were both so incredibly resistant to something as simple as the ATX/EATX standard for motherboard form factors. They just had to push their own form factors (which could vary widely from product to product) and allowed almost zero interoperability. This is just one small example but the attitude permeated both companies to the extent that it basically killed them.

***dekhn*** The big exception here is that SGI took IrisGL and made it into OpenGL which as a standard lasted far longer than SGI. And OpenGL played a critical role preventing MSFT from taking over the 3D graphics market with Direct3D.

***pjmlp*** Except that OpenGL only mattered thanks to Carmack and id Software mini-GL drivers.

It hardly matters nowadays for most game developers.

***dekhn*** When I say "hardware graphics market" I'm referring to high performance graphics workstations, not gaming. There is a whole multibillion dollar market there (probably much smaller than games, but still quite significant). It's unclear what carmack's influence on the high performance graphics workstation environment is, because mini-GL left out all the details that mattered to high performance graphics (line rendering would be a good example).

In my opinion, Mesa played a more significant role because it first allowed people to port OpenGL software to run on software-only cheap systems running Linux, and later provided the framework for full OpenGL implementations coupled with hardware acceleration.

Of course, I still greatly enjoyed running Quake on Windows on my 3dfx card with OpenGL.

***pjmlp*** Well, put that way it is a market that runs on Windows with OpenGL/DirectX nowadays, or if using GNU/Linux, it is mostly with NVIDIA's proprietary drivers, specially when considering VFX reference platform.

***thisislife2*** > With SGI/Sun you were very much completely locked in to their hardware/software ecosystem and completely at the mercy of their pricing.

How is that in anyway different from Apple today with it's ARM SoCs, soldered SSDs and an OS that requires "entitlements" from Apple to "unlock" features and develop on?

***Gracana*** You can buy a cheap Mac and easily write programs for it. You don't have to spend $40k on a computer, you don't have to buy a support contract, you don't have to buy developer tools.

***fuzztester*** >You can buy a cheap Mac and easily write programs for it.

Interesting. How cheap? Never used Macs, only Windows and Unix and Linux.

***cryptoxchange*** Every time I’ve checked over the last decade (including today), you can buy a mac mini that supports the latest macOS for under $250 on ebay. You can also test your app using github actions for free if your use case fits in the free tier.

There is no way to do this for an IBM z16, which is the kind of vendor lock in that people are saying Apple doesn’t have.

***icedchai*** You can get a Mac Mini for $600-ish. Never get the base model though. (FYI, macOS is Unix.)

***fuzztester*** Yes, I did know. Darwin, etc. Thanks.

***fuzztester*** Thanks, guys.

***mcculley*** Are there entitlements or unlockable features other than when talking about App Store distribution?

***CountHackulus*** Thanks to web browsers and web apps it's not QUITE as bad of a lock-in nowdays. At least from a general consumer point of view.

***hn\_throwaway\_99*** > Similarly, it feels like Silicon Graphics is a case where they really should have become more standard. Now, unlike Amiga, they were too expensive to catch on with regular consumers, but I feel like they should have become and stayed the "standard" for workstation computers.

I think you highlighted very correctly there, though, why SGI lost. It turned out there were cheaper options, which while not on par with SGI workstations initially, just improved at a faster rate than SGI and eventually ended up with a much better cost/functionality profile. I feel like SGI just bet wrong. The article talks about how they acquired Cray, which were originally these awesome supercomputers. But it turned out supercomputers essentially got replaced by giant networks of much lower cost PCs.

***bunderbunder*** Hypothesis:

What smaller businesses are using will tend to be what takes over in the future, just due to natural processes. When smaller businesses grow, they would generally prefer to fund the concurrent growth of existing vendors that they like using than they are to switch to the existing "industrial-grade" vendor.

At the same time, larger organizations that can afford to start with the industrial-grade vendors are only as loyal as they are locked in.

***01HNNWZ0MV43FF*** I see the same trend in programming languages. Say a really solid career lasts from about 20 to 60, 40 years long. Say that halfway through your career, 20 years in, you're considered a respectable senior dev who gets to influence what languages companies hire for and build on.

So in 20 years in, the current batch of senior devs will be retiring, and the current noobies will have become senior devs.

\*Whatever language is easy to learn today will be a big deal in 20 years\*

That's how PHP, Python, and JavaScript won. Since JavaScript got so much money poured on it to make it fast, secure, easy, with a big ecosystem, I say JS (or at least TS) will still be a big deal in 20 years.

The latest batch of languages know this, and that's why there are no big minimal languages. Rust comes with a good package manager, unit tester, linter, self-updater, etc., because a language with friction for noobies will simply die off.

One might ask how we got stuck with the languages of script kiddies and custom animated mouse cursors for websites. There's no other way it could turn out, that's just how people learn languages.

***chuckadams*** Back in the old days there was a glut of crappy bloated slow software written in BASIC. JS is the BASIC of the 21st century: you can write good software in it, but the low bar to entry means sifting through a lot of dross too.

My take: that’s just fine. Tightly crafted code is not a lost art, and is in fact getting easier to write these days. You’re just not forced into scrabbling for every last byte and cpu cycle anymore just to get acceptable results.

***tombert*** I mean, there are corporations who only sell to very large corporations and have had plenty of success doing so. Stuff like computational fluid dynamics software, for example, has a pretty-finite number of potential clients, and I don't think I could afford a license to ANSYS even if I wanted one [1], since it goes into the tens of thousands of dollars. I don't think there are a ton of startups using it.

But I think you're broadly right.

[1] Yes I know about OpenFOAM, I know I could use that if I really wanted.

***gspencley*** I still dream of having a Beowulf Cluster of Crays.

One day ...

***analognoise*** https://github.com/DarkwaveTechnologies/Cray-2-Reboot

I'm on board for this project?

***cmrdporcupine*** This betting wrong on specialization happened over and over again in the late 70s and 80s. The wave of improvements and price reduction in commodity PC hardware was insane, especially from the late 80s onwards. From Lisp machines to specialized graphics/CAD workstations, to "home computer" microcomputer systems, they all were buried because they mistakenly bet against Moore's law and economies of scale.

In 91 I was a dedicated Atari ST user convinced of the superiority of the 68k architecture, running a UUCP node off my hacked up ST. By the end of 92 I had a grey-box 486 running early releases of Linux and that was that. I used to fantasize over the photos and screenshots of workstations in the pages of UnixWorld and similar magazines... But then I could just dress my cheap 486 up to act like one and it was great.

***kazinator*** Atari ST and Intel PC are not distant categories. Both are "'home computer microcomputer' systems". Not all home computer systems can win, just like not all browsers can win, not all spreadsheets can win, not all ways of hooking up keyboards and mice to computers can win, ...

***cmrdporcupine*** They were distant on market tier but most importantly on economies of scale. The Intel PC market grew exponentially.

***kazinator*** Sure, but the economy of scale came from the success. The first IBM PC was a prototype wire-wrapped by hand on a large perf board.

When you switched to Intel in 1992, PC's had already existed since 1981. PC's didn't wipe out most other home computers overnight.

***tombert*** Yeah, I'm more annoyed about Amiga than SGI. They were priced competitively with Apple and IBM offerings.

I guess it's just kind of impossible to predict the future. I don't think it's an incompetent decision to try and focus entirely on the workstation world; there are lots of businesses that make no attempt to market to consumers, and only market to large companies/organizations, since the way budgeting works with big companies is sort of categorically different than consumer budgets.

But you're absolutely right. Apple and Windows computers just kept getting better and better, faster and faster, and cheaper and cheaper, as did 3D modeling and video editing software for them. I mean, hell, as a 12 year old kid in 2003, I had both Lightwave 3D (student license) and Screenblast Movie Studio (now Vegas) running on my cheap, low-spec desktop computer, and it was running fast enough to be useful (at least for standard definition).

***mike\_hearn*** Of course, the reason they got better so fast is volume. There was just way more investment into those platforms. Which means this explanation is somewhat circular: they were successful because they were successful.

I think a more useful explanation is that people rate the value of avoiding vendor lockin extraordinarily high, to the extent that people will happily pick worse technology if there's at least two competing vendors to choose from. The IBM PCs were not good, but for convoluted legal reasons related to screwups by IBM their tech became a competitive ecosystem. Bad for IBM, good for everyone else. Their competitors did not make that "mistake" and so became less preferred.

Microsoft won for a while despite being single vendor because the alternative was UNIX, which was at least sorta multi-vendor at the OS level, except that portability between UNIXen was ropey at best in the 90s and of course you traded software lockin for hardware lockin; not really an improvement. Combined with the much more expensive hardware, lack of gaming and terrible UI toolkits (of which Microsoft was the undisputed master in the 90s) and then later Linux, and that was goodbye to them.

Of course after a decade of the Windows monopoly everyone was looking for a way out and settled on abusing an interactive document format, as it was the nearest thing lying around that was a non-Microsoft specific way to display UI. And browsers were also a competitive ecosystem so a double win. HTML based UIs totally sucked for the end users, but .... multi-vendor is worth more than nice UI, so, it wins.

See also how Android wiped out every other mobile OS except iOS (nobody cares much about lockin for mobile apps, the value of them is just not high enough).

***HarHarVeryFunny*** The reason SGI failed, and eventually Sun too, isn't because the world "chose wrong", but because their performance simply did not keep up with x86.

When these RISC-based workstations were initially released their performance, especially at graphics, was well beyond what a PC could do, and justified their high prices. A "workstation" was in a class by itself, and helped establish the RISC mystique.

However, eventually Intel caught up with the performance, at a lower price, and that was pretty much the end. Sun lived on for a while based on their OS and software ecosystem, but eventually that was not enough especially with the advent of Linux, GCC, etc, as a free alternative.

***cduzz*** Ivan Sutherland described the reason [1] why PCs won a long time ago. Basically a custom tool may do a thing "better" than a general purpose tool for a while, but eventually, because more resources are spent improving the general tool, the generalized tool will be able to do the same thing as the specialty tool, but more flexibly and economically.

[1] <http://www.cap-lore.com/Hardware/Wheel.html>

This is a story told by Ivan Sutherland. It is fun to tell the story and I will try not to embellish it. The story serves now as an interesting review of early display technology and also the best description of a design pitfall that afflicts modern software design practice even more.

I now have the reference:

On the Design of Display Processors T. H. Myer, I. E. Sutherland;

Communications of the ACM, Vol 11, No. 6, June 1968

This paper is the original and much more complete. It also tells how the cycle was broken.

The time was the early 60’s. The milieu was Digital Equipment Corporation and MIT. Early computer displays plotted points. The DEC PDP 1 and IBM 704 had remarkably similar hardware for which the program would load an x and y value into common registers and issue a plot command. (140 μs for the 704) The hardware would briefly display a small point on the screen at those coordinates. If the screen had a persistent phosphor and the program plotted points fast enough and the room lights were low then a very useful image might appear on the screen. While television was already common the cost of refresh memory ruled out today’s raster scan. The PDP-1 and the 704 had neither the megabit memory required to serve as a raster scan refresh buffer, nor adequate memory bandwidth. When the program was not plotting points, the screen was black. The PDP-1 had a light pen that was a hand held photo cell that would provide a program testable state that would be true briefly after plotting a point within view of the light pen. This design followed the work on the Tx-10 at Lincoln Labs which had supported the pioneering work of Ivan Sutherland and his famous Sketchpad program.

Then as now there was a continuing decrease in hardware costs. Architects wondered how they could overcome the awkward fact that the screen would go black whenever the CPU diverted its attention from the consuming task of refreshing the screen. The solution was the display list. Hardware was built to read an array of point coordinate pairs from core memory and plot them on the screen. The array was thus processed until the program told it to stop. The CPU could then construct display lists and go on to other tasks while the user examined the static image.

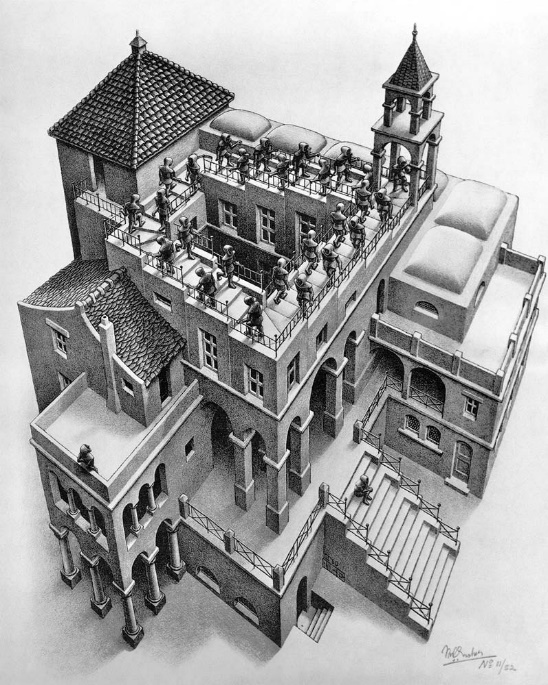
The program was most likely to want to modify part of the image while avoiding the cost of recomputing, or even copying the entire display list. The display refresh hardware was enhanced with an ability to recognize a pattern, stored among the plot data, directing it to another area of memory for more display data. The program could then construct a cycle of several lists which it could modify in a dialog with the user.

It soon became evident that some patterns were repeated at different parts of the screen. The letter A might be required several times in the display list with different coordinates. Each occurrence of A on the screen would require a list of points with x and y adjusted for that location. A subroutine like function was added to the display hardware that passed an x and y coordinate offset parameter value as it displayed points from another location in memory. A return instruction would go back to the saved location of the routine invocation command.

A common pattern was the line segment. The CPU spent too much time doing the simple arithmetic required to construct the display list for the points of a segment. Arithmetic and conditional capability were added to the display hardware.

With conditionality it was discovered how to plot a circle if only the hardware had one more register. A new yet familiar problem arose. The display hardware began to be occupied with complex algorithms beyond keeping points on the screen. Should we add a simple display list mechanism to off load the display processor?

At this point it had become clear that the display hardware design had grown to become a general purpose computer. The idea of multi processing was already well known but seldom practiced. The conventional wisdom regarding multi processing was that it interfered with the economy of scale that then ruled computer design; bigger computers were more cost effective than simpler ones. The foregoing design cycle had seemed at each step to yield an improved design. A paradox like Escher’s circling monks. It would seem that ‘better’ is not transitive. More likely there were a variety of concepts of ‘better’ in play.



The resolution of the paradox was highly contingent on the hardware design rules that are no longer relevant. The pit fall of reincarnation of function is still with us. Sometimes the layers are inhomogeneous which leads to conceptual costs, namely it is too hard to understand the layered systems. Sometimes the inhomogeneities linger in legacy architectures long after their original justifications are forgotten.

The wheel turns still:

From what I hear of current (2005) GPU design we are now in the midst of another display reincarnation cycle! See “General-Purpose Computation Using Graphics Hardware” [] too.

***sys\_64738*** Sun had the perfect opportunity with Utility Computing around the mid-2000s but when cloud took off we had Oracle buying SUNW. They killed Sun Cloud which had the opportunity to be big, vast, and powered by JAVA hardware.

Sun Microsystems was a company like no other. The last of a dying breed of "family" technology companies.

***msisk6*** I was at the MySQL conference when it was announced that Oracle was buying Sun. It just took all the life out of the conference. All the Sun folks were super pissed off. Truly the end of an era.

***hodgesrm*** I was there too. It certainly felt "timed" to maximize the sense of deflation for people working on MySQL. Perhaps it was just coincidence. IIRC Larry Ellison said that the crown jewel in the deal was actually Java.

***icedchai*** I remember that time. It felt like Sun was on death's doorstep since the dot-com crash. On the hardware side, the market was flooded with used Sun hardware. On the software side, Linux was "good enough" for most workloads.

***hinkley*** Sun really struggled to make full use of their multicore systems. That m:n process model is coming back with fibers and libuv, but we have programming primitives and a deeper roster of experienced devs now than we did then. Back then they caused problems with scalability.

There were times when Java ran better on Intel than on Solaris.

***mtillman*** The Amiga couldn't handle the performance requirements of Doom at the time (Game Engine Black Book Doom). Workbench was more fun than Windows and at least the install process that was early linux.

As much as I loved my O2 (my first professional computer), it was underpowered for the time for anything other than texture manipulation. The closed source nature of that time period and the hardware sales motion meant that you were paying through the teeth for compilers on top of already very expensive hardware. The Cray-linked Origin 200's ran Netscape web server with ease but that's a lot of hardware in a time period when everything went out of date very quickly-donated ours! Irix still looks better than the new Mac OS UIs IMO but no-Motif is a small price to pay for far cheaper access to SDKs IMO. Also, Irix was hilariously insecure due in part to its closed source nature. https://insecure.org/sploits\_irix.html

***downut*** "... hardware sales motion meant that you were paying through the teeth for compilers..."

For Fortran? My memory is hazy but at NASA NAS a bunch of us were using gcc/g++ starting ~1990. g++ was... an adventure. Building my own (fine!) compiler for free(!) got me hooked on OSS to the point that when Linux/FreeBSD launched I jumped in as fast as I could.

I really loved my various SGI boxen. Magical times. I was a NASA "manager" so had the Macintosh "manager" interface box that I solved by keeping it turned off.

***nyrikki*** We had 6 O200s Cray-linked into three nodes in an CXFS cluster to run an appletalk server backed by clarion arrays. While there were serious limits caused by the single metadata server, XVM and CXFS were better than anything provided by Veritas or the other major UNIX vendors of the day.

The Fibre Channel XIO boards were really needed back then for that application as PCI was still way too slow.

I was sad to know that when I left that job that SGI server was being replaced and the support personal at SGI were going to lose their jobs too.

***axpvms*** >Also, Irix was hilariously insecure due in part to its closed source nature.

That was in addition to having three default accounts with well known passwords and a telnet server.

***icedchai*** Some versions of IRIX (4.x, maybe?) also defaulted to having X11 authentication disabled. Anyone in the office could "xmelt" your screen... or worse.

***nyrikki*** Oracle EBS required root X11 authentication be disabled to use the compositor for PDF generation even 10 years later so still pretty common.

Oracle threatened to not support us when I used an unprivileged Xvfb instqance instead.

Still stupid but not that uncommon back then.

***JohnBooty*** If Amiga really "deserved" to win, I think they wouldn't have been eclipsed by the PC ecosystem in terms of performance.

They leapt out ahead of the competition with an advanced OS, purpose-built for graphics and sound in a way that PCs and Macs weren't.

Which was great. But they weren't really better than the competition. They were just doing something the competition wasn't. And when the competition actually started doing those things they got eclipsed in a hurry.

I wonder if Tesla will suffer the same fate. They were obviously around a decade ahead of the established players when it came to electric cars. But once the other established players actually got serious about electric cars, Tesla largely stopped being special, and upstarts like Lucid and Rivian are neck and neck with them (in terms of compelling products, not sales) as well.

***cduzz*** "the future is already here, it just isn't evenly distributed."

This means there are products out there with futuristic features that will be seen as requirements for all things going forward and right now those features are niche elements of some product.

The Amiga was a fantastic device but not a general purpose device. Lots of things are fantastic at a niche but not general, and those almost always fail.

Is this also the "worse is better" truism?

***hinkley*** Tesla will also suffer a reverse cult of personality problem.

I don’t know anyone at Rivian so my opinion of them is neutral. Meanwhile Tesla is run by the jackass who ruined twitter.

***JohnBooty*** There are definitely a lot of people who would consider Musk's personality a huge minus. I'm one of them.

I'm less clear on whether or not he's a net minus. I don't know if his detractors outnumber his fans or vice-versa.

***ip26*** We've seen again and again that the high end of the computer market can't sustain itself; the mass market outruns it. The result is that the high end works best when leveraging the mass market instead of trying to compete with it.

See the dominance of Threadripper in workstations, which is built on top of mainstream desktop and server parts bin. Or look at the Epyc based supercomputers, rumored to be the only supercomputers to turn a net profit for the suppliers, thanks to leveraging a lot of existing IP.

***hnhg*** The people that created the Amiga weren't the same people as the ones leading Commodore. Apple's success seems to have been heavily based on the company's leader being very involved in product development and passionate about it.

Along the same lines, there is an alternate timeline where the Sharp X68000 took over the world: https://www.youtube.com/watch?v=OepeiBF5Jnk

***randomdata*** I'm not sure Apple did continue to succeed after its early success. It eventually gave up its name to NeXT, which is who found later success.

***samatman*** The standard quip here is that NeXT purchased Apple for negative $400 million.

***tombert*** I've actually seen that video!

Yeah, I think that would also have been a better timeline; I'm just stuck in the anglo-world and thus my knowledge is mostly limited to what was released in the US or Europe.

***epcoa*** > the Amiga really was better than anything Apple or Microsoft/IBM was doing at the time

At the time. A brief moment in time, and then they had no path forward and were rapidly steamrolled. Nothing was "chosen wrong" in this aspect.

***tombert*** Well, wait, the Amiga had preemptive multitasking way before Apple or Windows got it, like the mid 80s. I don't think Windows got it until Windows NT, and it didn't become mainstream until Windows 95. Macs had bizarre cooperative multitasking that would freeze if you just thought it about it funny [1] all the way until OS X.

There's other stuff too; they had better color graphics in the 80s while DOS was still dealing with CGA and EGA, and decent sound hardware. Even by 1990, the video toaster was released, well before it got any port to DOS.

[1] I'm sure it got better, my first exposure to it was System 7 and that thing was an unholy mess. I didn't touch macOS again until OS X.

***epcoa*** Long before Windows 95 there was DOOM and DOOM would not run on an Amiga.

> 80s while DOS was still dealing with CGA and EGA, and decent sound hardware.

And then the 80s ended. What point did I make that you are contradicting?

> Even by 1990, the video toaster was released,

And if you wanted to do CAD? Would you use an Amiga? Probably not. What about desktop publishing? Pointing out that Amiga had carved out a niche (in video editing) when that was the norm back in those days doesn't make any strong comment about the long term superiority or viability of the platform.

Also, I don't buy into the idea that just because a company had something "superior" for a short period of time with no further company direction that they didn't lose fair and square. That Amiga had something cool in the 80s but didn't or couldn't evolve isn't because the market "chose wrong". Commodore as a company was such a piece of shit it made Apple of the 80s look well run. Suffering a few more years with the occasional bomb on System 7 was not a market failure.

> Macs had bizarre cooperative multitasking

What was bizarre about it, compared to any other cooperative multitasking system of the time? Also you seem to be fixated on preemptive multitasking to the neglect of things like memory protection.

***tombert*** > Long before Windows 95 there was DOOM and DOOM would not run on an Amiga.

Yeah fair. I do wonder if a port like the SNES version would have been possible if id would have greenlit it, but that's a "what if" universe. Alien Breed 3D would run on a 1200, but IIRC it ran pretty poorly on that.

> And then the 80s ended. What point did I make that you are contradicting?

I mean, yes, VGA cards and Soundblaster cards were around in 1990, but they weren't really standard for several years later.

> And if you wanted to do CAD? Would you use an Amiga? Probably not. What about desktop publishing? Pointing out that Amiga had carved out a niche (in video editing) when that was the norm back in those days doesn't make any strong comment about the long term superiority or viability of the platform.

Also fair. I'll acknowledge my view is a bit myopic, since I don't really do CAD or desktop publishing, but I do some occasional video editing, and I do think Amigas were quite impressive on that front. You're right in saying it was a "niche" though.

> Commodore as a company was such a piece of shit it made Apple of the 80s look well run.

No argument here. Still think that the hardware was pretty cool though.

> What was bizarre about it

I guess "bizarre" was the wrong word. It was just really really unstable, and System 7 would constantly freeze for seemingly no reason and I hated it.

> Also you seem to be fixated on preemptive multitasking to the neglect of things like memory protection.

I feel like if Commodore had been competently run, they could have done work to get proper protected memory support, but again that's of course a "what if" universe that we can't really know for sure.

I guess what frustrates me is that it did genuinely feel like Commodore was really ahead of the curve. I think the fact that they had something pretty advanced like preemptive multitasking (edit: fixed typo) in the mid 80s was a solid core to build on, and I do kind of wish it had caught on and iterated. I see no reason why the Amiga couldn't have eventually gotten decent CAD and Desktop publishing software. I think Commodore didn't think they had to keep growing.

***icedchai*** The Amiga OS was designed in a way that protected memory support was basically impossible. Message passing was used everywhere. How did it work? One process ("task", technically) sent a pointer to another, a small header with arbitrary data, which could contain anything, including other pointers. Processes would literally read and write each other's memory.

***sys\_64738*** Commodore's story is more about achieving the impossible with 1-2 engineers building each computer. Commodore was a company built around Jack Tramiel who wanted his widgets to ship in volumes to "the masses, not classes". When he left then it was a lifestyle sucking cash machine for Irving Gould who appointed incompetent CEO after incompetent CEO after Tramiel. The miracle is it staggered on ten years post-Jack.

But the reality is the Commodore 64 kept Commodore going during most of that period rather than Amiga sales. It's similar to Apple where the Apple 2 kept Apple afloat during the 80s and 90s until Steve returned.

***cmrdporcupine*** Times changed though, too, and Tramiel couldn't replicate his success w the C64 at Atari Corp, despite bringing the same philosophy (and many key engineers) over there.

By the late 80s the "microcomputer" hobby/games market was dead and systems like the ST and Amiga (or Acorn Archimedes, etc.) were anachronisms. You had to be a PC-compat or a Mac or a Unix workstation or you were dead. Commodore and Atari both tried to push themselves into that workstation tier by selling cheaper 68030 machines than Sun, etc, but without success.

***logicprog*** > they had no path forward

This is I think the premise that you and people like me who think Amiga could have gone on to do great things disagree on, I think. Most Amiga fans would say that it totally had a path forward, or at least there is no evidence that it didn't, and the failure to follow that path therefore it wasn't an inherent technical problem, but a problem of politics and management. Do you have any evidence to the contrary?

***prpl*** It’s just a lesson in worse is (often) better. If you can do some most of the job with something that is either cheaper, easier to build, or easier to iterate on, then it will often overtake a better engineered solution.

***cameldrv*** I used some SGIs in the mid-late nineties, and they did have cool 3D graphics capabilities. I found 4dwm to be kind of cool but mostly gimmicky and it was really slow on the Indy and O2. Windows 95/NT were much snappier on contemporary hardware.

By '97 or so SGI actually had essentially given up competing when they shut down the team that was developing the successor to InfiniteReality.

In a sense though, Silicon Graphics did become more standard, in that their original 3D framework was Iris GL, which then evolved into OpenGL, which became the main 3D graphics standard for many years.

***causi*** "Revolutionaries rarely get to live in the societies they created"

I think it's a combination of a skillset/culture needed to create a paradigm shift isn't the same one needed to compete with others on a playing field you built, and of complacency. It happens over an over. We saw it happen with RIM, and we're watching it happen right now with Prusa Research.

***itronitron*** Both Prusa and SGI are (and were) probably largely unknown to 90% of their potential market. The globally recognized companies tend to spend far more on marketing than anyone in a STEM field would consider remotely reasonable.

***fuzztester*** True. In the early to middle days of Java, I read that Sun spent millions of dollars on marketing it, and related stuff around it.

***bluedino*** They were destined for eventually dying like the rest of the high end UNIX workstation market. Linux and x86 got better and better every year.

***tombert*** Yeah, and OS X more or less mainstream-ized consumer UNIX as well. It gave you access to the UNIX tools in the command line if you wanted them, had a solid UNIX core, but was a lot cheaper than an SGI and also easy to use.

***knorker*** Well, for SGI that's like saying the world "chose wrong" that long distance travel is not done by Saturn 5 rockets.

The Saturn 5 was clearly a technical marvel better than any plane, and it'd get you anywhere much faster.

If you spare no expense, you get a better product. Sure. I'm also not surprised that a $100k BMW is more comfortable than a Renault Clio.

***cladopa*** I never had an Amiga, but I had friends that had it. It was a superior tech only for a very small period of time.

What happened was Intel, they took great decisions like automating the design of their processors and this made them grow at an incredible pace. The Amiga depended on a different processor that stagnated.

***KerrAvon*** The 68k CPU lineup at the heart of the Amiga was competitive well into the 90's; the Amiga had run out of juice by 1989. The Amiga was only as good as the custom chips. If Commodore kept investing in R&D for the custom chips, they would have at least remained competitive.

***sys\_64738*** Intel never pulled ahead until the Pentium but by then Motorola weren't interested in the 68K series.

***pjmlp*** Yes, Irix is one of the few UNIX based OSes that I actually find cool.

***sys\_64738*** People are always passionate about various UNIX systems and their derivatives like Linux. Windows is so utilitarian.

***pjmlp*** Outside Irix, Tru64, Appolo, Solaris with NeWS, NeXTSTEP, all other UNIXes are pretty meh.

Regarding Windows, some time reading the excellent Windows Internals book series is recommended.

***hinkley*** It certainly got fewer complaints than HP-UX.

***fuzztester*** What were the complaints that HP-UX used to get?

I used it for a while earlier at work, and don't remember many problems with it. One did have to apply OS patches fairly regularly to it, but IIRC, that process was somewhat smooth.

***hinkley*** In the time of SGI, I believe it had a lot of posix compliance problems.

And if memory serves, the Bible (https://www.goodreads.com/book/show/603263.Advanced\_Programm...) didn’t cover it, which was a problem.

***fuzztester*** Okay, got it. :)

***pjmlp*** On HP-UX 10, back in 2000, the C compiler version I was using still wasn't fully ANSI C, and needed K&R C function declarations, but hey at least we had containers (HP Vault), and 64 bit file system access.

***jefflinwood*** I worked on the SGI campus as a consultant/vendor to them in 1999/2000 during the dot.com boom. I really wanted one of those 1600SW flat screens (everything was CRT back then), but they weren't really in use at the time.

One of the neatest things is that they let us (Trilogy/pcOrder) put together a sand volleyball team to compete in their company intramurals.

Their cafeteria was also top notch.

***dekhn*** That cafeteria went on to be known as "Charlie's" at Google and was the main HQ cafeteria (serving great food, and then later, extremely meh food). TGIF was also held there. If there ever was a place that was "Google central", that was it.

***fuzztester*** >"Charlie's" , and its chef, Charlie, are mentioned in the book called The Google Story: https://en.m.wikipedia.org/wiki/The\_Google\_Story

***rongenre*** I played with SGI machines in college and they felt like.. the future. I really hoped they would hire me when I graduated.

Incredible, though, how the relatively cheaper Windows NT machines and 3dfx cards and graphics software just killed them. I was a little sad when I wandered around the campus of an employer in Mountain View and noticed the fading sign that had what was left of the SGI logo.

***jandrese*** The awesome old cube logo or the new "we spent millions of dollars on a professional marketing department to design a new logo" that is just the initials in a boring font and off center?

I co-oped for SGI onsite in the sales/marketing/support for a major ISP of the day back in the late 90s and the buzz around the office was that the company (at this point experimenting with overpriced Windows NT boxes and generic Linux servers) was experiencing massive brain drain to some brand new startup that was going to make something called a "GeForce" card for cheap PCs that was going to avoid the pitfalls of the then popular Voodoo cards. Apparently the engineers were unhappy with the direction the company was taking under the new leadership and thought that there was still an interest in graphics acceleration.

***mrpippy*** The "sgi" logo was a big step down from the cube, but it was a lot more attractive than the Rackable/Silicon Graphics International "sgi" logo that looked like a cheap knockoff of the previous one.

https://en.wikipedia.org/wiki/Silicon\_Graphics\_International

***theideaofcoffee*** It really was a letdown when Rackable resurrected SGI and then brought about that ... thing of a logo. It just felt it hollowed out the brand even more, even if SGI itself was still making some interesting hardware at the time-namely the Altix 4700, UV and ICE, the soul just wasn't there anymore.

***rongenre*** The awesome cube, which apparently didn't look good when faxed...

***nullindividual*** 3Dfx didn't play in the SGI space. But Matrox (for 2D), 3Dlabs (another RIP), Orchard (used 3Dlabs chip), STB (again, 3Dlabs chip...), and Diamond (uh... 3Dlabs!).

3Dfx grew up in the arcade market. They were always consumer-focused.

***technothrasher*** > I played with SGI machines in college and they felt like.. the future.

I had a couple of Indigos that I supported while an undergraduate (I had a student job with the University's Unix group in their computing center), and the SGIs felt to me exactly like the Amiga- Really cool, but kind of lopsided. I tended to do most of my work on the SPARCstations and ignore the SGIs unless I specifically wanted to play with the graphics stuff.

I actually still have an Indigo XS24 that I collected at one point over the years. Tried to get it to boot a bit ago but it's dead, unfortunately.

***fernly*** I appreciate the article. As a lowly member of technical staff 92-97 I didn't see or understand a fraction of this. I \*loved\* my Indy workstation and the IRIX desktop UI, and I admired the brilliance of some of the engineers I worked with.

Remember the Lavarand[1]? Random number generator based on an array of lava lamps?

[1] https://en.wikipedia.org/wiki/Lavarand

***browningstreet*** I had an Indigo2 on my desk in college. I moved back and forth between that an a NeXT cube that a colleague had in their lab. The NeXT was nice but SLOW. The Indigo2 wasn't especially fast but it was nice and could do visual things that just weren't available as readily on our alternatives. We had SunOS and Solaris systems that were mostly used for network and engineering projects, and I was engaged in some visualization work. When the O2 was announced I was quite sure it would be the solution to our speed issues.. around the same time, another colleague was the first to install a beta of Win95, and it did seem awfully pretty.

***mobilio*** This was explained here: https://vizworld.com/2009/04/what-led-to-the-fall-of-sgi-chapter-1/ (whole series here, excerpt below)

<begin> *SGI’s lavish parties during the time were the stuff of Legend. Being invited to an SGI party was the geek equivalent of being invited to the Playboy Mansion. No expense was spared as Huey Lewis and the News, Natalie Cole, Kenny G, Penn & Teller, Tony Robbins, and more were present. $10,000 bar tabs and $8,000 dinners were the norm, and expensed directly to SGI. Ed also missed obvious signs of what was happening with SGI (They lost money for 4 of 5 consecutive quarters remember? That’s a full year of losing profits), probably because he spent too much time hanging around with Bill Clinton & Al Gore. [4] He spent so much time and money hanging around with popular people that some analysts christened SGI as “the new Apple”. And that was all find with Ed, as he had his $5.3M severance payment to mend his sorrows. [5]* <end>

***cf100clunk*** I'd never met Rick Belluzzo, but having known SGI insiders of his era I gathered that he'd driven the company strongly towards NT on x86 at the cost/risk of losing their corporate and personal UNIX competencies, causing some internal outrage. When his time was up at SGI he quickly popped up at Microsoft running MSN in what I saw as a synecure meant as a pat on the back for a job well done. Am I right on this?

***shrubble*** That is a widely held view among Unix nerds like me, at least.

***jra\_samba*** My recollection of that time (I was at SGI when Belluzzo was there).

<https://www.linkedin.com/posts/jeremyallison_wither-google-f>...

***Jeremy Allison | Distinguished Engineer at CIQ. Board member of Software Freedom Conservancy. Board member of GNOME Foundation.***

*Wither Google ?*

*FYI. This is my recollection of events, but it's been twenty or so years now, so if my memory is fuzzy on the details, please forgive the ramblings of an old engineer :-).*

*Many years ago, I was working at SGI when the company started to do poorly under the leadership of Rick Belluzzo. In those days the company sponsored the Friday afternoon "beer-bust", when all the employees hung out on what is now the Google volleyball court, drank beer and socialized. Rick cancelled the beer busts as a cost cutting measure.*

*One of SGI's lead engineers, Casey Leedom (known internally as Mr. IRIX, for SGI's version of UNIX - a codebase he knew inside-out) took it upon himself to self-organize replacement beer-busts with the rest of the engineering staff. We all chipped in to buy the beer, and the Friday afternoon festivities continued as normal.*

*Rick Belluzzo heard about this, and embarrassed, called Casey into a meeting. "Things aren't that bad", he told Casey. "We can fund the beer-busts again". "No", said Casey. "We know the company is having some hard times, and we're all in this together, so the engineering staff are happy to continue to self-fund the beer-busts".*

*"There is one thing you could do", Casey added. "I know you are a busy man, but if you could take it upon yourself to occasionally attend a beer-bust, it would be much appreciated by the employees".*

*"Great idea !", said Rick, and promised to do so. Six or so months later, Rick Belluzzo ran away from SGI over a weekend, and on Monday it was announced that Microsoft had a new VP. "We appear to have lost our CEO" was the email announcement from now acting CEO Bob Bishop.*

*In the time since Casey asked him to the time he joined Microsoft, Rick attended zero self-sponsored beer busts at SGI.*

*For some reason I can't see Sundar attending any self-sponsored Google beer-busts either.*

***masfoobar*** I was about 10 years old when films like Terminator 2 and Jurassic Park came out. Although I had games consoles I never had a proper computer. I had friends who owned an Amiga or Amstrad but they always wanted to come to mine and play Sega or Nintendo.

Being into art at the time, was facinated with Silicon Graphics computers. If I could "wish" a xmas present, it would have been (something like) an SGI Indigo2 system.

It wasn't long before a 486-PC entered our home. It was certainly a learning experience trying to get Windows 3.1 to work but it was DooM that completely changed my views on video games. I never expected graphics to make such a huge leap when I considered Mortal Kombat 2 to be peak graphics! How game graphics changed for the rest of the 90s is insane!!!

When the Nintendo 64 came about (originally Ultra 64 and Project Reality) I honestly had a childish attitude that NOTHING would touch it for a long time because it was SGI under the hood! In defence of my childish attitude, I was still a child.

The harsh reality is... by the time the Nintendo 64 was on the shelves I knew it was already "old tech" after watching (and playing) a demo of Tomb Raider 2 in a computer shop. It was likely a Pentium 90 mhz computer with Windows 95, maybe 500GB Hard drive space and running on VooDoo graphics. Once you get over the awesome graphics you realise that these machines can be anywhere from £600-£1500.

My defensive side kicks in... but.. but.. SGI is still better, right? They cost so much more they are not about games, they are for Hollywood movies! A year or two after, seeing what 3D animation software can do... wouldn't be surprised if rendering speed was competitive on Windows 98 machines to SGI ones.

Move on to today. Our phones are more powerful than those really expensive SGI systems from 1992. It is crazy when you think about it.

***Keyframe*** Steve Jobs on Silicon Graphics: youtube.com/watch?v=iQKm7ifJpVE

***vondur*** It's pretty simple why these Unix vendors all died. Linux and Intel chips. Sure you could get a really nice Sun system at the time with all of the redundancy tech built in, which cost around $50k. Or you can go and get 4 or 5 Linux servers from Dell running RedHat which by the early 2000's were faster too.

***alecco*** Interesting insider comment from a previous thread:

<https://news.ycombinator.com/item?id=30920824>

**<<INSERT BEGIN>>**

**oppositelock** on April 5, on: The cult of Amiga and SGI, or why workstations mat...

Oh, it's a lot longer story than that. I worked as SGI from just around its peak, to its downfall, seeing the company shrink to a tenth of its size while cutting products.

At the time, I was a fairly junior employee doing research in AGD, the advanced graphics division. I saw funny things, which should have led me to resign, but I didn't know better at the time. Starting in the late 90's, SGI was feeling competitive pressure from 3DFx, NVIDIA, 3DLabs, Evans and Sutherland (dying, but big), and they hadn't released a new graphics architecture in years. They were selling Infinite Reality 2's (which were just a clock increase over IR1), and some tired Impact graphics on Octanes. The O2 was long in the tooth. Internally, engineering was working on next gen graphics for both, and they were both dying of creeping featureitis. Nothing ever made a deadline, they kept slipping by months. The high end graphics pipes to replace infinite reality never shipped due to this, and the "VPro" graphics for Octane were fatally broken on a fundamental level, where fixing it would mean going back to the algorithmic drawing board, not just some Verilog tweak, basically, taping out a new chip. Why was it so broken? Because some engineers decided to implement a cool theory and were allowed to do it (no clipping, recursive rasterization, hilbert space memory organization).

At the same time, NVIDIA was shipping the GeForce, 3DFx was dying, and these consumer cards processed many times more triangles than SGI's flagship Infinite Reality 2, which was the size of a refrigerator and pulled kilowatts. SGI kept saying that anti-aliasing is the killer feature of SGI and that this is why we continue to sell into visual simulation and oil and gas sector. The line rendering quality on SGI hardware was far better as well. However, given SGI wasn't able to ship a new graphics system in perhaps 6 years at that point, and NVIDIA was launching a new architecture every two years, the reason to use SGI at big money customers quickly disappeared.

As for Rick Beluzzo, man, the was a buffoon. My first week at SGI was the week he became CEO, and in my very first allhands ever, someone asked something along the lines of, "We are hemmoraging a lot of money, what are you going to do about it"? He replied with, "Yeah, we are, but HP, E&S, etc, are hemmoraging a lot more and they have less in the bank, so we'll pick up their business". I should have quit my first week.

**panick21** Trying to be both sell a seller of very high end computer products while also doing your own chips and graphics at the same time is quite the lift. And at the same time their market was massively attacked from the low end.

The area where companies could do all that and do it successfully kind of ended in the late 90s. IBM survived but nothing can kill them, I assume they suffered too.

What do you think, going back to your first day, if you were CEO could have been done?

I always thought for Sun OpenSource Solaris, embracing x86, being RedHat and eventually Cloud could have been the winning combination.

**rbanffy** > What do you think, going back to your first day, if you were CEO could have been done?

Not quite sure. You correctly pointed out SGI (HP, Sun, everyone else in the workstation segment) was suffering with Windows NT eating it from below. To counter that, SGI would need something to compete in price. IRIX always had excellent multiprocessor support and, with transistors getting smaller, adding more CPUs could give it some breathing room without doing any microarchitectural changes. For visualization hardware the same also applies - more dumb hardware with wider buses on a smaller node cost about the same while delivering better performance. To survive, they needed to offer something that's different enough from Windows NT boxes (on x86, MIPS and Alpha back then) while maintaining a better cost/benefit (and compatibility with software already created). I'd focus in low-end entry-level systems that could compete with the puny x86's by way of superior hardware-software integration. The kind of what Apple does, when you open the M1-based Air and it's out of hibernation before the lid is fully opened.

> I always thought for Sun OpenSource Solaris, embracing x86, being RedHat and eventually Cloud could have been the winning combination.

I think embracing x86 was a huge mistake by Sun - it helped legitimize it as a server platform. OpenSolaris was a step in the right direction, however, but their entry level systems were all x86 and, if you are building on x86, why would you want to deploy on much more expensive SPARC hardware?

Sun never even tried to make a workstation based on Niagara (first gen would suck, second gen not so much), and OpenSolaris was too little, too late - by then the ship had sailed and technical workstations were all x86 boxes running Linux.

panick21 > IRIX always had excellent multiprocessor support and, with transistors getting smaller, adding more CPUs could give it some breathing room without doing any microarchitectural changes.

That kind of exactly what Sun did and likely gave them legs. This might not have made it out of the 90s otherwise.

> I think embracing x86 was a huge mistake by Sun - it helped legitimize it as a server platform.

x86 was simple better on performance. I think it would have happened anyway.

> OpenSolaris was a step in the right direction, however, but their entry level systems were all x86 and, if you are building on x86, why would you want to deploy on much more expensive SPARC hardware?

That's why I am saying they should have dropped Sparc already in the very early 2000s. They waste so much money on machines that were casually owned by x86.

**rbanffy** > That's why I am saying they should have dropped Sparc already in the very early 2000s

They had two SPARC architectures - the big core, high-performance one and Niagara, the many wimpy core one and Sun never thought about combining both on the same machine, which is more or less what x86 is doing now because they are being forced to do it by Apple and its M1. Sun was there in the early 2000's.

There's no fundamental reason why x86 has to be faster than SPARC, in fact, SPARC machines trounced x86 ones.

Another thing that killed Sun was that it could never decide whether they were Apple or Microsoft - they never decided whether they wanted to make integrated hardware or a become plain software company.

panick21 > There's no fundamental reason why x86 has to be faster than SPARC, in fact, SPARC machines trounced x86 ones.

Other then Intel having 100x more money and more architects with better nodes ...

SPARC was already worse by 1998.

Its really only continued to make money because companies couldn't figure out how to scale vertically and rather paid absolutely absurd amounts for these multi-core machines.

Some of the Supercomputer people showed how they could totally destroy Crey/Sun and so on with simple clusters of x86.

Sure had they perfectly executed SPARC and hit on every investment they might have done a lot better, but that just wasn't the reality. Intel just executed much better. SPARC had all kinds of development failure in the 90s and in the late 90s Intel just had better nodes in addition to better architecture.

> Another thing that killed Sun was that it could never decide whether they were Apple or Microsoft - they never decided whether they wanted to make integrated hardware or a become plain software company.

I think they did want to be Apple but they simply were not that god at making products. They were actually better at making software, but then didn't do very well at making products based on that.

The did some good stuff like Fishworks, AMD x86 Servers and so on.

They should really have turned into Open Apple, RedHat and AWS. With OpenSolaris and Zones on x86 costume machines they could have dominated the Cloud space (they even had products going into that direction early in the 2000s).

**jacquesm** SGI also offered x86 based machines, of all things running NT or WIN 2K. That was when the writing really was on the wall.

**rbanffy** on April 6, 2022 Precisely. When you start offering x86 boxes with Windows, it's obvious your own architecture and OS are dead.

But I don't remember those. I remember Intergraph did (and they were quite good, but died quickly)

**jacquesm** Here is a news article about it:

https://www.itprotoday.com/windows-8/sgi-debuts-killer-new-n...

**rbanffy** Oh wow... In my brain I was confusing it with the Intergraph one.

I also was not surprised to see Rick Belluzzo's name in the press release... The devastation that man caused has no parallel in the history of personal computing. It's comparable to what Steven Elop did in the mobile space.

I use to joke Microsoft had perfected the use of executive outplacement as an offensive weapon.

**oppositelock** I think some kind of discipline around releasing products in a timely way by cutting features would have done wonders. However, the kinds of computers SGI built were on the way out, so they couldn't have survived without moving in the direction that people wanted. Maybe it was a company whose time had come. SGI wasn't set up to compete with the likes of NVIDIA and Intel.

**panick21**\_ on April 5, 2022 | root | parent | next [–]

Why couldn't they compete with NVIDIA? Were the not just as big?

**digisign** on April 5, 2022 | root | parent | next [–]

The PC market grew bottom up to be 10x the size of the workstation market during the 90s. Even with thinner margins, eventually workstation makers couldn't compete any longer on R&D spend.

The book The Innovator's Dilemma describes the process.

**digisign** ^meant thinner margins of PC industry.

**oppositelock** on April 5, 2022 | root | parent | prev | next [–]

Engineering culture. SGI was not pragmatic in building hardware, more of an outlet for brilliant engineers to ship experiments.

**jacquesm** on April 5, 2022 | root | parent | next [–]

I can see how that was your view if you came in on the tail end but it definitely wasn't always so. I've owned quite a few of them and if you had the workload they delivered - at a price. But for what they could do they would be 3 to 4 years ahead of the curve for a long time, and then in the space of a few short years it all went to pieces. Between NVIDIA and the incredible clock speed improvements on the x86 SGI was pretty much a walking zombie that did not manage to change course fast enough. But CPU, graphics pipeline, machine and software to go with it is an expensive game to play if the number of units is smaller than any of your competitors that have specialized.

I'm grateful they had their day, fondly remember IRIX and have gotten many productive years out of SGI hardware, my career would definitely not have taken off the way it did without them, in fact the whole 'webcam/camarades.com/ww.com' saga would have never happened if the SGI Indy did not ship with a camera out of the box.

**gugagore** I wasn't familiar, so I searched and found your great account of the history! <https://jacquesmattheij.com/story-behind-wwcom-camaradescom/>

Posted by Jacques Mattheij July 7, 2009

This is all done from memory without any fact checking, chances are I’m misremembering stuff or leaving out important pieces, in that case please forgive me, it’s late!

I may revisit this at some point to correct and revise it.

Personal note, I’m a high school drop-out, have been coding since roughly 1978 (when I was 13 or so) and if it weren’t for a few very nice people would have never ever gotten where I am today. Piet Tacx gave me a job in the mailroom of a bank in Amsterdam when I was 18, had no house and no income, another (Eddy de Leeuw) gave me my first ‘real’ programming job for that same bank simply because he got totally sick and tired of telling me that he wasn’t going to hire me. He literally gave me one chance, go to ‘Volmac’, learn Cobol in the next three months and don’t bother coming back if you don’t succeed. (it took only a month and I passed with the minimal grade because they suspected me of cheating because I aced the tests…).

So, on to the story of the live webcam:

In the dark ages of the web one of my former partners (Michael Erkelens) bet me for a bar of chocolate that I couldn’t come up with a way to make ‘animated gifs’ ‘live’. I figured it had to be possible somehow and in one of those famous allnighters I produced a very primitive version of the first streaming webcam software. Basically it produced an endless animated gif, which the browser dutifully downloaded and animated. It wasn’t much to look at but it was years ahead of the competition.

That first black and white image showed a crossroads in Hoofddorp near schiphol airport.

Before long we found a way to do it with jpegs (which gave a tremendous boost in quality), this must have been somewhere around ‘95 or so. With every iteration of the software the quality, speed or features improved and before long the software was sold on a license basis to lots of companies.

As a demonstration I set up a little box with two relays controlling a fan and a lamp. The fan blew a mobile of small paper cranes around my desk and the lamp would light up the scene, to create a bit of variation and proof that the thing was real and not just some elaborate hoax (in spite of that there were still plenty of people claiming that it must be a hoax…).

The first real breakthrough came when Gilbert Cattoire who was working for a French media company spotted the software and suggested Yves St. Laurent should use it to broadcast their annual fashion show. This was quite the event for me, going to Paris with my trusty SGI hardware in the back of the car and being treated to the finest french hotels and cuisine :)

In turn, this webcast was spotted by someone working for NetNitco, an American ISP who had contacts with NASA. The next thing we knew we were transmitting the launches and flights of two space shuttle missions. One of these was to repair the Hubble space telescope.

Suddenly the live webcam software started selling like hotcakes, and we went from a 1 man 1 woman show to several employees. Licenses were sold to just about every large ISP and webcams started popping up all over the place. Rembert Oldenboom did a first rough port to the windows platform (up to here it was strictly SGI) which opened up a big market, the first commercially available colour webcams were reaching the market but people did not have software.

The Michael mentioned above became a partner in the company and suggested we give away the software. This scared me very much, after all, our livelihood depended on selling those licenses but I relented and we started giving the software away. This was somewhere in December of ‘97. Another new partner in the business, Taco Scargo came up with the idea of making a ‘live index’ so that the page where we linked to our customers would not show so many dead links. From there it was a small step to make it a condition that if you wanted to use the free software your webcam had to be in the live index and on the 1st of March ‘98 Camarades.com was born.

The next night the server was a smoking slag heap, over 10,000 downloads that first night and no way were we prepared to handle the traffic. With a lot of help from friends in various places we managed to weather this first crisis and we invested a ton of money in a large dell server (which was already too small a couple of weeks later when it was delivered).

A downside of this free webcam business was that it seemed to attract a different kind of audience than what we had hoped for and we had to establish 24 hour oversight in order to not end up with a porn site instead of a webcam site. (even today this is still a problem, I recall clearly the first time we had someone strip on the site and we all went like ‘What?, did I see that right?’, so much for our naivity I guess…).

A full time designer (Jonathan Kraij) was hired who made the site look good and came up with the idea of ‘chico’, our little webcam like mascotte.

We attracted some investors but this was quite a rocky ride, one of them basically wanted to leverage his investment into a large amount of cash within a few weeks of buying in (he did get us an advertising deal with 24x7 though), another used his shares and some smarts and a crooked notary public to gain control of a joint venture and sold the software source code to a German porn company. This was a busy year for the lawyers, we learned a lot about how not to do business. Around this time our contacts with Logitech started to warm up, and their subsidiary (spotlife) had a look at us for a potential takeover. For years (even after spotlife died) logitech sent us tons of traffic from people that were looking to getting software with their new and shiny webcams.

In the meantime we had opened up an office in Toronto because we were literally kicked off the international backbone because we were saturating it with traffic, keep in mind that up to this point most of the content on the internet was static and we were pumping video over that line, at some point we had more than 1200 live cameras online at any time of the day, each of those serving a large number of viewers.

Toronto offered room for expansion, front street was only a block away and we managed to get some bandwidth on a fibre-optic line that had just been installed with a company run by friends of ours.

This worked well for about a year and a half, the site got remade by Julian Kreho and kept on growing. At some point in this period we were roughly in the top 300 or so of all the sites active at the time, but you have to keep in mind that the web was a lot smaller then than it is today (I think we had about 100,000 uniques per day).

Then the internet started creaking, in december of ‘99 or so I had my first warning when a cheque from 24x7 was late. I didn’t like it one bit because 24x7 was all of our turnover at this point (the license business had dried up completely once we started giving away the software) so behind the scenes we started to work on what is now known as the ‘freemium’ model, a basic service that would give everything that we did already away for free with a premium package that you had to pay for. Still, the speed with which the walls came crashing down was more than I had bargained for, and with our half baked version ready 24x7 media went bust leaving us with 0 income. We had to lay off a lot of the people that worked for us and I really hated that bit of it, it certainly wasn’t their fault.

Somehow we managed to survive this period, but it was pretty scary. Every month we ended up with a few more paying subscribers than the month before, and after half a year or so we were back on a fairly solid financial footing.

We even managed to save a bit and when ww.com came up for auction on Ebay we managed to buy it for a bunch of cash, I always thought camarades.com was an ok name (and a nice pun) but it was too long and too hard to spell, ww.com is much easier to remember (it stands for webcam world).

I sold my house to buy out my partners and emigrated to Canada with my wife & kid, we ended up on St. Josephs Island in rural Ontario and ran the seriously downsized business from our house there over a bunch of modems that were bonded to give us reasonably fast internet access.

After several years of trying to get our permanent resident status we gave up on that and moved back to the Netherlands.

Since then it has been an uphill battle, the company is still making some money, not enough to make it the full time interest of two people so we used what little money we still had to restart our lives here and have invested some of the savings in to other companies and projects, some of which are doing well and others which are doing not so well.

Since december ‘08 we’ve been busy working our way to a complete revamping of the site, new software, a new website layout but this is a major project and it will still take some effort and time before it is completed.

The take home lessons from all this for me are that timing really is everything and that it is better to not have done a deal than to have done the wrong deal.

Your partners are a bigger deciding factor than the product and you have to make sure that everybody plays by the rules.

Also that it is better to hire people that may not have the qualifications but that are willing to learn than people with lots of papers and no desire to give their 100%.

**jacquesm** Fun times! Also frustrating but an excellent school and all is well that ends well.

**mrpippy** Do you know anything about the rumor that an O2 successor was prototyped that used NVIDIA graphics? (I think I read that on Nekochan long ago).

The slow pace and poor execution of CPU and graphics architectures after ~1997 is crazy to think about. The R10000 kept getting warmed over, same for IR, and VPro, and the O2.

The Onyx4 just being an Origin 350 with ATI FireGL graphics (and running XFree86 on IRIX) was the final sign that they were just milking existing customers rather than delivering anything innovative.

**rasz** >NVIDIA was launching a new architecture every two years

at the peak of nvidia 3dfx war new chips were coming out every 6-9 months

Riva 128 (April 1997) to TNT (June 15, 1998) took 14 months, TNT2 (March 15, 1999) 8 month, GF256 (October 11, 1999) 7 months, GF2 (April 26, 2000) 6 months, | 3dfx dies here |, GF3 (February 27, 2001) 9 months, GF4 (February 6, 2002) 12 months, FX (March 2003) 13 months, etc ...

**anamax** In many cases, an executive's behavior makes sense after you figure out what job he wants next.

**unixhero** Thank you so much for your inside story. Hilbert space memory organization sounds great :)

**Beecafe** Texture memory is still stored like that in modern chips (presuming they meant Hilbert curve organization). It's so that you can access 2D areas of memory but still have them close by in 1D layout to make it work with caching.

**beagle3** Is it really Hilbert?

A project I worked on a couple of decades ago had interleaved the bits of the x and y indexes to get that effect “for free”, I imagine a Hilbert curve decode would take quite a bit of silicon.

**beecafe** Ah right, youre right it's probably the interleaved bits (aka Morton code) in actuality. Or more likely, something tuned to the specific cache sizes used in the GPU.

**buescher** I have no clue what hilbert space memory organization could possibly be - arbitrarily deep hardware support for indirect addressing? - but it sounds simultaneously very cool and like an absolutely terrible idea.

***oppositelock*** the framebuffer had a recursive rasterizer which followed a hilbert curve through memory, the thinking being that you bottom out the recursion instead performing triangle clipping, which was really expensive for the hardware at the time.

The problem was that when you take some polygons which come close to W=0 after perspective correction, their unclipped coordinates get humongous and you run out of interpolator precision. So, imagine you draw one polygon for the sky, another for the ground, and the damn things Z-fight each other!

SGI even came out with an extension to "hint" to the driver whether you want fast or accurate clipping on Octane. When set to fast, it was fast and wrong. When set to accurate, we did it on the CPU [1]

1 - https://www.khronos.org/registry/OpenGL/extensions/SGIX/SGIX...

**vardump** Nowadays all GPUs implement something similar (not necessarily Hilbert but maybe Morton order or similar) to achieve high rate of cache hits when spatially close pixels are accessed.

3D graphics would have terrible performance without that technique.

buescher Got it. I was imagining something else entirely.

**<<INSERT END>>**

***ben7799*** I was kind of right the exact age to see all this happen all while I was in college.

Fall 95 enter freshman year and we had Indys and IBM RS6000s as the main workstations on campus. Really great setup where you could sit at any workstation and all your stuff just worked and your whole environment seamlessly migrated. The only thing you had to do was if you were compiling your own stuff you'd have to recompile it for the machine you sat down at.

SGI brought a demo truck to campus in the spring of my Freshman year (Spring 96) and blew us all away. They were there for interviews, obviously I was a freshman but we all went to check it out.

Summer 96 I get an internship and for kicks they gave me an Indy with a 21" CRT (huge at the time) and the silly video camera that was like 10+ years ahead of it's time.

Fall 96 we got labs full of O2s.

Fall 1997 I bought a 3DFX card. MS/Intel somehow made a donation to the school and got them to start phasing out the Unix workstations. The windows NT setup was terrible, they never had the printing and seamless movement of files down till after I graduated. Video games in the Fall of 1997 on the 3DFX were basically as impressive as the demos on the $100k refrigerator sized machine SGI showed in 1995.

Probably fall 1998 I remember my Dad got a computer with an Nvidia Riva 128.

Spring 99 I graduated and that fall I rebuilt my PC with a Geforce 256.

I'm not sure when I last saw an SGI, but I did briefly use one of their NT machines IIRC.

Last time I had a Sun machine at work was probably 2004. I remember maybe 2007-2008 at work deciding for the first time we were going to support Linux, then by 2010-11 we had dropped support for Sun.

Most of the commercial Unix workstations had tons of Unix annoyances I never found Linux to have. Irix was maybe the best. HP-UX was super annoying I remember. I didn't use DEC Unix and Tru64 much. Closed source PC Unix like SCO I remember being horrible.

***rdtsc*** > announcing SGI’s intent to migrate to Itanium (and collaborating on projects Monterey and Trillian) while simultaneously launching an IA-32 series of machines running NT known as the Visual Workstation.

That's pretty wild, I had completely forgotten about that. It was too late of course by then.

I was working at a CAD/CAE the company which leased SGI workstations in the early 2000s. Was just a fresh grad wondering why were they leasing them, then realized they were more expensive than some cars.

Some of the developers were starting to use NT workstation with 3D graphics, they were not cheap, but they could run circles around Octanes and Indigos at a fraction of the cost. They were rushing to port everything to NT as writing had been on the wall for a while by then.

***kelsey98765431*** A close friend of mine growing up had a parent that worked at SGI. I'm not trying to start a holy war, but I just have to let it be known that emacs was the recommended editor at SGI. Just saying. Maybe other things contributed to the fall but, in my heart I will always remember emacs.

***nonrandomstring*** I agree that these machines and their OS were too proprietary and over-engineered to weather the PC revolution. But when I think back to my days using Suns and SG (Indigo) the memory feels like driving a Rolls Royce or Daimler with leather seats and walnut panels.

***canucker2016*** There's the "SGI Irix bloat" internal email that got posted to Usenet. Took me awhile to find a copy on the net, Here's a formatted-for-HTML version: <http://www.art.net/%7Ehopkins/Don/unix-haters/tirix/embarrassing-memo.html>

Date: Tue, 26 Apr 94 08:42:32 EDT

From: Jerry Leichter <LEICHTER@lrw.com>

Subject: Stress Analysis of a Software Project [long]

The following, which claims to be an internal Silicon Graphics memo, has already seen fairly broad network distribution. I have no way of verifying that it is what it claims to be, but (a) I'm told by someone with close dealings with SGI that it fits with what he's heard; (b) if it's a fake, someone put a huge amount of effort into producing it.

I forward it to RISKS as a wonderful record of what goes wrong with large software projects, and why. It would be as useful if all the names, including the company and product names, were removed. This memo should not be seen as an indictment of SGI, which is hardly unique. There is good evidence that Sun, for example, had very similar problems in producing Solaris; and I watched the same thing happen with the late, unlamented DEC Professional series of PC's, and something like it almost happen with firmware for DEC terminals a number of years back.

I hope that Tom Davis's position hasn't been badly hurt by the broad distribution of his memo - but based on the traditional reaction to bearers of bad news, especially when the bad news becomes widely known, I can't say I'm sanguine about it. -- Jerry

Software Usability II | October 5, 1993 | Tom Davis

Last May, I published my first report on software usability, which Rocky Rhodes and I presented to at Tom Jermoluk's staff meeting (with Ed, but without Tom). Subsequently, I made it available to quite a few other people.

This sequel is to satisfy all those people who have urged me to bring it up to date. I begin with a summary; details follow.

Please read at least the summary.

**SUMMARY**

Release 5.1 is a disappointment. Performance for common operations has dropped 40% from 4.0.5, we shipped with 500 priority 1 and 2 bugs, and a base Indy is much more sluggish than a Macintosh. Disk space requirements have increased dramatically.

The primary cause is that we attempted far too much in too little time. Management would not cut features early, so we were forced to make massive cuts in the final weeks of the release.

What shall we do now? Let's not look for scapegoats, but learn from our mistakes and do better next time.

A December release of 5.1.2 is too early to fix much -- we'll spend much more time on the release process than fixing things. Allow enough time for a solid release so we don't get: 5.1.2.1, 5.1.2.2, 5.1.2.3, ...

Let's decide ahead of time exactly what features are in 5.1.2. If we pick a reasonable set we'll avoid emergency feature cuts at the end.

Nobody knows what's wrong -- opinions are as common as senior engineers. The software environment is so convoluted that at times it seems to rival the US economy for complexity and unpredictability. I propose massive code walk-throughs and design reviews to analyze the software. We'll be forced to look closely at the code, and fresh reviewers can provide fresh insights.

For the long term, let's change the way we do things so that the contents and scheduling of releases are better planned and executed. Make sure marketing and engineering expectations are in agreement.

**INTRODUCTION**

We've addressed some of the problems presented in the original May report, but not enough. Most of the report's warnings and predictions have come true in 5.1. If we keep doing the exact same thing, we'll keep getting the exact same results.

I'm preparing this report in ASCII to make it widely available. It's easy to distribute via news and mail, and everyone can read it.

An ASCII version of the May 12 report can be found in:

bedlam.asd:/usr/tmp/report.text

The included quotations are not verbatim. Although the wordings are inexact, I believe they capture the spirit of the originals.

**BLOAT UPDATE**

*"Do you want to be a bloat detective? It's easy; just pick any executable. There! You found some!" -- Rolf van Widenfelt*

In the May report, I listed a bunch of executable sizes, and pointed out that they were unacceptable if we intended to run without serious paging problems on a 16 megabyte system. Between May and the 5.1 release, many have grown even larger. IRIX went up from 4.8 megabytes to 8.1 megabytes, and has a memory leak that causes it to grow. Within a week, my newly-booted 5.1 IRIX was larger than 13.8 megabytes -- a big chunk of a 16 megabyte system. It's wrong to require our users to reboot every week.

There are too many daemons. In a vanilla 5.1 installation with Toto, there are 37 background processes.

DSOs were supposed to reduce physical memory usage, but have had just the opposite effect, and their indirection has reduced performance.

Programs like Roger Chickering's "Bloatview" based on Wiltse Carpenter's work make some problems obvious. The news reader "xrn", starts out small, but leaks memory so badly that within a week or so it grows to 9 or 10 megabytes, along with plenty of other large programs. But what's really embarrassing is that even the kernel leaks memory that can't be recovered except by rebooting!

Showcase grew from 3.2 megabytes to 4.0 megabytes, and the master and status gizmos which are run by default occupy another 1.7 megabytes. Much of this happened simply by recompiling under 5.1 -- not because of additional code.

The window system (Xsgi + 4Dwm) is up from 3.2 MB to 3.6 MB, and the miscellaneous stuff has grown as well. As I type now, I have the default non-toto environment plus a single shell and a single text editor, jot. The total physical memory usage is 21.9 megabytes, and only because I rebooted IRIX yesterday evening to reduce the kernel size. Luckily, I'm on a 32 megabyte system without Toto, or I'd be swamped by paging.

Much of the problem seems to be due to DSOs that load whole libraries instead of individual routines. Many SGI applications link with 20 or so large DSOs, virtually guaranteeing enormous executables.

In spite of the DSOs, large chunks of Motif programs remain unshared, and duplicated in all Motif applications.

**PERFORMANCE UPDATE**

"Indy: an Indigo without the 'go'".  
-- Mark Hughes (?)

"[X](http://www.art.net/%7Ehopkins/Don/unix-haters/x-windows/disaster.html) and [Motif](http://www.art.net/%7Ehopkins/Don/unix-haters/x-windows/motif.html) are the reasons that [UNIX](http://www.art.net/%7Ehopkins/Don/unix-haters/handbook.html) deserves to die."  
-- Larry Kaplan

The performance story is just as bad. I was tempted to write simply, "Try to do some real work on a 16 megabyte Indy. Case closed.", but I'll include some details.

In May, I listed some unacceptable Motif performance measurements. Just before 5.1 MR, someone reran my tests and discovered that the performance had gotten even worse. Some effort was expended to tune the software so that instead of being intolerable, it was back to merely unacceptable performance.

We no longer report benchmark results on our standard system. The benchmarks are not done with the DSO libraries; they are all compiled non-DSO so that the performance in 5.1 has not declined too much.

Before I upgraded from 4.0.5 to the MR version of 5.1, I ran some timings of some everyday activities to see what would happen. These timings were all made with the wall clock, so they represent precisely what our users will see. I run a 32 megabyte R4000 Elan.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **4.0.5** | **5.1** | **% change** |
| C compile of a small application | 25 sec | 35 sec | 40% |
| C++ compile of a small application | 68 sec | 105 sec | 54% |
| Showcase startup, May report file | 13 sec | 18 sec | 38% |
| Start a shell | <2 sec | ~3 sec | ~50% |
| Jot 2 MB file | <2 sec | ~3 sec | ~50% |

What's most frightening about the 5.1 performance is that nobody knows exactly where it went. If you start asking around, you get plenty of finger-pointing and theories, but few facts. In the May report, I proposed a "5% theory", which states that each little thing we add (Motif, internationalization, drag-and-drop, DSOs, multiple fonts, and so on) costs roughly 5% of the machine. After 15 or 20 of these, most of the performance is gone.

Bloating by itself causes problems. There's heavy paging, there's so much code and it's so scattered that the cache may as well not be there. The window manager and X and Toto are so tangled that many minor operations like moving the mouse or deleting a file wake up all the processes on the machine, causing additional paging, and perhaps graphics context swaps.

But bloat isn't the whole story. Rocky Rhodes recently ran a small application on an Indy, and noticed that when he held the mouse button down and slid it back and forth across the menu bar, the (small) pop-up menus got as much as 25 seconds behind. He submitted a bug, which was dismissed as paging due to lack of memory. But Rocky was running with 160 megabytes of memory, so there was no paging. The problem turned out to be Motif code modified for the SGI look that is even more sluggish than regular Motif. Perhaps the problem is simply due to the huge number of context swaps necessary for all the daemons we're shipping.

The complexity of our system software has surpassed the ability of average SGI programmers to understand it. And perhaps not just average programmers. Get a room full of 10 of our best software people, and you'll get 10 different opinions of what's causing the lousy performance and bloat. What's wrong is that the software has simply become too complicated for anyone to understand.

**WHAT WENT WRONG IN 5.1?**

The one sentence answer is: we bit off more than we could chew. As a company, we still don't understand how difficult software is.

We planned to make major changes in everything -- a new operating system, new compilers, a new user environment, new tools, and lots of new features in the multi-media area. Not only that, but the new stuff was promised to do everything the old software had done, and with major enhancements. (Early warning: version 6.0 promises to be even more disruptive.)

About 9 months ago, Rocky and I pointed out the impossibility of what we were attempting. Rather than reduce the scope of the projects, a decision was made to hire a couple of contractors (who know nothing about our system) to handle the worst user interface problems in the Roxy project. In addition, promises were obtained from various executives that a significant effort would be made to improve software performance.

Management was basically afraid to cut any features, so we continued to work on a project that was far too large. The desperate attempt to do everything caused programmers to cut corners, with disastrous effects on the bug count. And the bug count was high simply because 5.1 was so big.

Only when the situation was beyond hope of repair did we start to do something. Features and entire products were removed wholesale from the release, and hundreds of high-priority bugs were classified as exceptions, so that we could ship with "no priority 1 and 2 bugs". We did, however, ship with over 500 "exceptions". The release was deemed too crummy to push to all our machines, but was restricted to the Indys, the high-end machines, and a few others where new hardware required the new software. Due to the massive bug count, virtually no performance tuning was done.

When the schedule is impossible as it was in 5.1, the release process itself can get in the way. The schedule imposes a code freeze long before the software is stable, and fixing things becomes much more difficult. If you know you're going to be late, slip before the code freeze, not after. We're trying to wrap up the box before the stuff inside is finished, and then trying to fix things inside the box without undoing the wrapping -- it has to be less efficient.

**Management Issues:**

There was never an overall software architect, and there still is not, and until Way Ting was given the job near the end, there was no manager in charge of the 5.1 release, either.

I wrote a note in sgi.bad-attitude about the "optimist effect", which I believe is mostly true. In condensed form:

Optimists tend to be promoted, so the higher up in the organization you are, the more optimistic you tend to be. If one manager says "I can do that in 4 months", and another only promises it in 6 months, the 4 month guy gets the job. When the software is 4 months late, the overall system complexity makes it easy to assign blame elsewhere, so there's no way to judge mis-management when it's time for promotions.

To look good to their boss, most people tend to put a positive spin on their reports. With many levels of management and increasing optimism all the way up, the information reaching the VPs is very filtered, and always filtered positively.

The problem is that the highly filtered estimates are completely out of line with reality (at least in recent software plans here at SGI), and there are no reality checks back from the VPs to the engineers on the bottom. I think it's great to have aggressive schedules where you try to get things out 20% or so faster than you'd expect. The problem is that in 5.1, the engineers were expected to get things out 80% faster, and it was clearly impossible, so many just gave up.

We certainly didn't win any morale prizes among the engineers with 5.1. It's the first release here at SGI where most of the engineers I talked to are ashamed of the product. There are always a few, but this time there were many. When engineers were asked to come in over the weekends before the 5.1 release to fix show-stopper bugs, I heard a comment like: "Why bother? SGI's going to release it anyway, whether they're fixed or not."

I'm not blaming the engineers. Most of them worked their hearts out for 5.1, and did the best they could, given the circumstances. They'll be happy to buy into a plan where there's a 20% stretch, but not where there's an 80% stretch. They figure: "It's hopeless, and I'll be late anyway, and I'm not going to get rewarded for that, so why kill myself?"

**Marketing - Engineering Disconnect**

*"Marketing -- where the rubber meets the sky."  
-- Unknown*

There's a disconnect between engineering and marketing. It's not surprising -- marketing wants all the whiz-bang features, it wants to run in 16 megabytes, and it wants it yesterday. Although engineering would like the same things, it is faced with the reality of time limits, fixed costs, and the laws of nature.

It's great to have pressure from marketing to do a better job, but at SGI, we often seem to have deadlocks that are simply not resolved. Marketing insists that Indy will work in 16 megabytes and engineering insists that it won't, but both continue to make their plans without resolving the conflict, so today we're shipping virtually useless 16 MB systems. Similarly for feature lists, reliability requirements, and deadlines.

Well, at least we met the deadline.

**WHAT TO DO -- SHORT TERM (5.1.2)**

*"We should sell 'bloat credits', the way the government sells pollution credits. Everybody's assigned a certain amount of bloat, and if they go over, they have to purchase bloat credits from some other group that's been more careful."  
-- Bent Hagemark*

There are problems in both performance and bugs, and we'd like to fix both. In addition, the first thing we should do is decide exactly what's going into release 5.1.2.

If we are serious about a December all-platforms release, there may be very little we can do other than keep stumbling along as we have been. Three months isn't much time to do anything, considering the overhead of a release, where perhaps half of the time will be spent in "code freeze". After 5.1, many engineers are exhausted, and it's unreasonable to expect them to start hard work immediately. 500 outstanding priority 1 and 2 bugs is a huge list, and we haven't even begun to hear about customer problems yet.

**What Should be in Release 5.1.2:**

I'm afraid the answer is going to be "everything that didn't make it into 5.1". I know that won't be the case, but I hope that we will carefully select what goes in now, rather than hack things out in a panic in December. The default should be "not included", and we should require a good reason to include things. Let's make sure that there's a minimal, solid, working set before we start adding frills.

**Improving Performance:**

*"SGI software has a cracked engine block, and we're trying to fix it with a tune-up."  
-- Mark Segal*

As stated above, we don't even know exactly what's wrong. We probably never will, but we should start doing things that will have as much of an impact on the problem as possible. I don't think we have time to study the problem in detail and then decide what to do -- we've got to mix the research with doing something about it.

Before we begin, we should have definite performance goals -- lose less than 5% wall-clock time on compiles of some known program over 4.0.5, have shells come up as fast as in 4.0.5, or whatever.

Some people claim that we need new software debugging tools to look at the problem, and that may be true, but it's not a short-term solution, and it runs the risk of causing us to spend all our time designing performance measurement tools, rather than fixing performance.

In fact, I don't really believe that simple "tuning" will make a large dent. To get things to run significantly faster, we've got to make significant changes. And we can't beat the "5% rule" by just speeding up all the systems by 5% -- if everything is exactly 5% faster, the overall system will be exactly 5% faster.

There's a strong tendency to look for the "quick fix". "Get the code re-arranger to work", or "Put all the non-modifiable strings in shared code space", for example. These ideas are attractive, since they promise to speed up all the code, and they should probably be pursued, but I think we're not going to make a lot of progress until we identify the major software architectural problems and do some massive simplification. Remember that DSOs were the last "quick fix".

There's got to be more to it than tuning; there must be some amazingly bad software architecture -- from a novice's point of view, a 4 MB Macintosh runs a far more efficient, interesting system than a 16 MB Indy. The Mark Segal quote above sums it up.

Code walk-throughs and design reviews are in order for most of our software. The attendees should include not only people working in the same area, but a small cross-section of experienced engineers from other areas. Get a pool of, say, 20 experienced engineers and perhaps 3 at a time would sit in on code reviews together with the other people working in that area.

Code reviews will help in many ways -- the engineer presenting the code will have to understand it thoroughly to present it, others will learn about it, and outside observers will provide different ways to look at the problems.

The most important thing should be the focus -- we're trying to make the code better and faster, not to make it more general, or have new features, or be more reusable, or better structured.

For complex problems, the walk-through should also include some general design review. Are these daemons really necessary? Do we really need this feature? And so on.

**Fixing Bugs:**

The code walk-throughs will obviously tend to turn up some bugs, so they'll serve a dual purpose.

With 500 or so priority 1 and 2 bugs, we must prioritize these as well. A bug that causes a system crash only on machines with some rare hardware configuration is properly classified priority 1, but it's probably less important than a bug in a popular program like Showcase that causes you to lose your file every tenth time, which would normally rank as priority 2. The effort involved in the fix should also be taken into account. For bugs of equal frequency of occurrence, it's probably better to fix 20 priority 2 bugs than 1 priority 1 bug if the priority 2 bugs are 20 times easier to fix.

A bunch of bugs can be eliminated by getting rid of features. Let's have the courage to cut some of the fat.

**WHAT TO DO -- LONG TERM**

*"Software quality is not a crime." -- Unknown (seen on a poster in building 7)*

It's easy to go on forever here, but I'll try to limit it to a few key ideas. We don't have to do all these at once, but we'd better start.

Have an overall SGI software plan.

Let's get an architect, or at least a small group of highly technical people, not just managers, to agree on plans for releases. In fact, since the release is a company-wide project, there ought to be company-wide participation in the decisions of what's in a release. The group should include marketing, documentation, engineering, and management and should come up with a compromise that's reasonable to all.

In every case, some attempt must be made to check reasonableness all the way to the bottom. There's a long series of excuses, "Well, that's what my junior VPs told me.", or "That's what my directors/managers/lead engineers/engineers told me." We get killed by the optimist effect, and a disinclination to listen seriously to anyone but our direct reports. Try to imagine the guts it takes for an engineer to go to his director and say: "My manager's out of his mind -- I can't possibly do what he's promised."

Let's try to concentrate on performance and quality, not on new features, especially for the 5.1.2 release. I know from my own experience that when I write good code, I spend 10% of the time adding features, and 90% debugging and tuning them. It's the only way to make quality software. In SGI's recent releases, the opposite proportions are often the rule. It's much easier to add 100 really neat features that don't work than to speed up performance by 1%.

Aim for simplicity in design, not complexity. Make a few things work really well; don't have 1000 flaky programs.

Be willing to cut features; who's going to be more pissed off: a customer who was promised a feature that doesn't appear, or the same customer who gets the promised feature, and after months of struggling with it, discovers he can't make it work?

Get better agreement between the top level VPs and the lowest engineers that a given schedule is reasonable.

For new development, continue the formal design reviews and code walk-throughs. These shouldn't just happen once in the development cycle -- things are bound to change, and code reviews can be very valuable, even for our experienced programmers.

**ACKNOWLEDGEMENTS**

I take full responsibility for the opinions contained herein, but I'd like to thank Mark Segal, Rosemary Chang, Mary Ann Gallager, Jackie Neider, Sharon Fischler, Henry Moreton, and Jon Livesey for suggestions and comments.

**The Original Author Responds**

I am the author of the original memo below, which was intended for internal Silicon Graphics use only, and was not for anyone outside the company. But since it has been leaked to the net, and is beginning to be used by competitor's sales people, I feel a response is required.

I don't believe that these problems are unique to Silicon Graphics. From discussions with friends who are insiders in many different companies, I am certain that similar memos could be written about the software of each of our competitors.

What I like about working for Silicon Graphics is that at least here, something is being done about it -- I worked for companies in the past where the response would have been to stick our heads in the sand in hopes that the problems would just fix themselves. If I hadn't thought that the memo would catalyze some change here, I wouldn't have written it.

The details appear as comments to my original article below. Luckily, the article is 6 months old, and we have had a chance to make some significant progress.

Typically, what happens is that each faster generation of hardware is followed by software that more than compensates for the increased speed, but as a result of this memo, Silicon Graphics has been able to skip one of the slowing software cycles, making, instead, a performance and quality based release. The next release is going to be similar, and in the meantime, we get an extra hardware boost from the faster R4600 processors.

-- Tom Davis, Silicon Graphics

**General comments:**

As a fairly direct result of this memo, SGI decided not to continue "business as usual" in software development. The approach we took to the problem was the following:

With the 4.0.5abcdefghi... fiasco, and the fact that the 5.\* releases were still for specific machines, our developers were desperate for an all-platforms release. We decided to make such a release relatively soon -- and 5.2 actually MRed in February. The 5.2 release had two goals -- primarily, all-platform, and given that it went out in February, do as much performance-tuning and bug-fixing as time allowed.

In that period, the performance on 16MB systems was essentially doubled, which improved performance on larger systems as well, but to a lesser degree. Significant numbers of bugs were fixed as well.

Some people hoped that a few quick fixes would bring back all the performance in 5.2, but a little investigation indicated that was a list of things to be done, and that another quality release would be required.

The 5.3 release, not officially scheduled, but which should be MRed around October or November is that quality (performance and bug-fix) release. We'll add a few new features, but they will be the exception rather than the rule. The longer time before the 5.3 release should give us time to do a thorough job of solving our problems.

For 5.3, there's also time to set up solid performance and bug-fixing goals, and these are already being discussed.

And most important -- the worst problems were with 16 MB systems that paged their brains out. They are better now, but not great. But we don't sell them. One of the 5.3 goals is to improve performance (or reduce sizes enough) that it will be acceptable on a 16 MB machine.

The kernel memory leaks are all fixed, and many of the important programs have been reduced in size. For 5.2, 5 or 6 of our most heavily-used programs were subjected to close scrutiny to find out where the performance went, and many were significantly improved.

A lot more work is planned for 5.3 to reduce the sizes of the executables.

Work is continuing on the DSOs to split them up properly so that they don't all have to be loaded, and to improve their performance and start-up time. We're working to make "quick-starting" happen more automatically.

**PERFORMANCE UPDATE**

I don't think it's unusual to do benchmarks with non-standard compiler settings. Both we and our competition have done so for a long time. We do ship all the libraries, et cetera, necessary to duplicate these results so customers for whom speed is the only objective can pay the cost of larger executables in exchange for the added speed.

Unfortunately, I can't re-run some of these tests, but 5.2 is definitely better than 5.1.

I think the 5.1 fiasco has caused a lot of our management to see the light, and in conversations with people at all levels, it's clear that nobody wants to see anything like it happen again. The 5.2 and future 5.3 releases seem to be steps in the right direction.

But there's still a lot of work to do, and we in engineering can use every minute between now and the 5.3 release to improve things.

The 5.3 release is being planned with reasonable beta-cycles, and with enough time between now and "code freeze" to make significant improvements.

**Management Issues:**

I think this sort of disconnect is not too unusual -- there is always enormous pressure to announce a very low entry price-point, and the 16MB system provided that. Everybody does this with the full knowledge that on a minimum system, you won't be able to run many interesting applications, and almost everyone will have to purchase a bit more memory. It's just that in the case of Indy, there were so many new features that the proposed minimal system was embarrassingly slow.

The "fix" is simply not to ship the 16MB systems which will insure that everyone will get a very usable machine. All we really lose is our low entry price point, and the gain is that we won't have to deal with the few irate customers who bought a minimal system.

Although some of our performance loss is due to more complicated features, the vast majority is due to the fact that more memory is required, and without it, the systems page with a consequent dramatic reduction in performance. The 4.0.X -> 5.X change on our large machines was measurable, but not nearly so noticeable as on the smaller ones.

We're still not completely there (as far as I can tell) with respect to better software management. The good thing is that many of our higher-level managers are acutely aware of the problem now -- Forest Baskett and Tom Jermoluk are extremely concerned, for example.

It's too bad it took a shock like 5.1 to make everybody take notice, but at least they did, and we're doing the right sorts of things to correct it.

***icedchai*** Meanwhile, today, we have chat applications taking up almost a gig of memory...

***Keyframe*** It still has that cachet with, well a bit older generation. I have an Indy and Indigo2 (purple - max impact one) and when someone visit and sees the machines it's like you sprawled a vintage ferrari in (our) eyes. I don't think there's anything comparable today when we have it all.

I demonstrated IRIX to younger colleagues and it was - ok, so it's alright I guess, like anything else we have today? Yep.. but contemporary world was NOT like that.

I had an Octane as well, heavy and loud beast. All in storage now waiting for move. In late 90s I worked a lot on SGI machines (vfx).

***danans*** > On the 10th of July in 2003, SGI vacated and leased their headquarters to Google.

I was around to witness the he tail end of that office space transition (on the incoming side at Google). It was surreal to be sitting in the physical carcass of a company I had long fantasized about (in part due to their marketing via Hollywood).

In retrospect it was ironic because a company that was based on selling very expensive high performance compute (SGI) was being physically replaced by a company selling (albeit indirectly) very cheap high performance compute.

***TacticalCoder*** The 'G' in SGI is basically the same 'G' as in GPU. Had SGI still rocked the world, today they'd be at the forefront of AI. All the models would be trained on and run on SGI computers.

The attack on SGI didn't only happen on the lame Windows side, with their crappy software and ultra lame 3D software (compared to what SGI had), which people would love because their lameness was matched by their ultra-cheap pricepoint.

The attack on SGI also came from open-source and Linux: cheap commodity hardware that'd run both mediocre but ubiquitous commercial software and the very same cheap commodity hardware that'd run Linux.

On which OS are most (all?) AI models trained today? What OS powers 500 of the world's top 500 supercomputers? Linux.

That's the tragedy of SGI: as if cheap commodity x86 hardware wasn't enough, they got then attacked by both Windows and Linux on that cheap hardware.

P.S: as a sidenote my best friend (still best friend to this day)'s stepfather was the head of SGI Benelux (Belgium/The Netherlands/Luxembourg) so my friend had this SGI Indy (the "pizza tower" one) at his home. Guess what we'd do every day after school?

***temporarely*** We had a few SGI boxes in arch school (for cad). I learned C++ on those boxes (the GL api, which I thought was beautiful btw). But the main attraction were the two demos (this is '91-2): the flight simulator was awesome and there was this deconstructing cube thing that was pretty wild as well. Compared to what was on PCs those days those SGI machines were truly dazzling.

***fnordpiglet*** I was at SGI and left with Clark to Netscape. That was a fun time in my career. That was also a time that it became clear SGI was about to explode as commodity GPUs were being developed by former SGI engineers and the core SGI graphics teams were attritting hard. Fun to read this story again and learn some of the things that happened later.

***timthorn*** I miss SGI for many reasons, but their industrial design (and that of pre-HPE Cray) is one of the big ones. The 19" rack form factor is the shipping container of the computing world - practical, standardised, sensible... and dull.

The variety in enclosures matched the novelty in architectures of the period. Exciting times to be part of.

***sgt*** The article went very quick from rowdy teen throwing smoke bombs to... boom, he has a PhD in Computer Science.

Really interesting article that goes into depth regarding the SGI products. I didn't know Clark basically invented the GPU.

Speaking of exotic hardware, I'm actually sitting next to an SGI O2 (currently powered off). A beautiful machine!

***junar*** I wish the author had shown their work regarding the mother's after-tax income. They apparently assume a 22% effective tax rate, but that seems unrealistically high.

The SSA says that Social Security tax was a mere 1% before 1950, and would remain below 4% until the 1970s. [1]

A 1949 Form 1040 [2] suggests that a single filer with 3 dependent children, earning $2700 that year, would have a federal income tax liability of only $7. Not 7 percent, 7 dollars.

[1] https://www.ssa.gov/oact/progdata/taxRates.html

[2] https://www.irs.gov/pub/irs-prior/f1040--1949.pdf

***thewileyone*** Wow, this article brought back memories I'd forgotten. I applied for an internship at Wavefront Technologies in Santa Barbara back in the 90's. I was so impressed with their offices and the SGI boxes they were using.

***pixelpoet*** Random trivia which will likely get buried but whatever: the rendering software Octane Render got its name from my refusing to let the author use my rendering software's name, and he had some historical beef with us, the two devs of Indigo Renderer, so he named his renderer after the SGI machine that came after the Indigo, Octane.

He also leant his SGI Indigo2 to the LGR guy on YouTube, and he did a great video on it: https://youtu.be/ZDxLa6P6exc

***krylon*** One thing that set SGI apart from the rest was that their cases were so incredibly pretty. Even today, the most beautiful PC cases barely play in the same league (IMHO). When Apple started building those colorful cases around ~2000, they reached that level of aesthetic appeal, but then they went for brushed metal (which is also very pretty, but not that pretty, IMHO).

Had one of the major PC vendors hired their designers and built just run-of-the-mill PCs, housing them in those amazing cases, I wonder how that would have worked out.

***formerly\_proven*** Interesting - some of them look neat, sure, but they also look fairly cheap. The O2 especially. The construction certainly isn't high quality, they're just stamped coil steel boxes with plastic wrapped around them. Some OEMs still do this, like this Alienware, which is just a squared-off miditower dressed up in a plastic shell: https://www.youtube.com/watch?v=DY1dlVPzUVo&t=8m10s

Of course in the 90s this would've been quite modern I imagine, considering that the competition was gray painted steel boxes and steel cabinets with a gray powder coat.

***lowbloodsugar*** I remember the nVidia CEO saying that nVidia is a software company. Whereas SGI thought of itself as a hardware company. That's why SGI is gone, and nVidia is still making the best graphics hardware.

***davidw*** I remember having access to an Irix box at my first job. It was seen as a real, professional, serious OS, not like the Linux box I set up. Pretty incredible how that all changed in a matter of years.

***ThinkBeat*** I remember when we got the first Indys at the uni. That was magic like. People nearly got into physical fights to use one of them. People came in at night to use them as well.

I wonder if the uni is so locked down now that students can sit in the lab all night.

Being a bit pragmatic in getting my actual thesis done I discovered that there was all of a sudden, a lot more resources available on one of the (older) Sun servers.

It saved me days if not weeks.

***AlbertCory*** The ONLY companies that last more than a generation or two are family-owned. https://albertcory50.substack.com/p/whats-missing-with-ameri...

There are a few exceptions, but for the most part, "professional management" just does not survive for very long.

***Apocryphon*** Low-key iconic machines, Hollywood kept putting them in movies.

http://www.sgistuff.net/funstuff/index.html

https://www.starringthecomputer.com/computers.html#SGI

***Uhhrrr*** The article doesn't mention the reason for the fall: less good but cheaper competitors. First Sun, then Windows NT and Linux.

***cf100clunk*** > The article doesn't mention the reason for the fall: less good but cheaper competitors

The article has this: ''As Bob Bishop took the reigns of SGI, things looked dark. AMD announced their 64 bit architecture in October, PC graphics had made massive strides while remaining significantly less expensive than SGI’s offerings, NT was proving to be a solid and less expensive competitor to UNIX, Linux was eating away at traditional UNIX market segments, and Itanium still hadn’t launched.''

I can agree with almost all of that statement but I object to the ''NT was proving to be a solid and less expensive competitor to UNIX'' part as mostly false in any mixed OS environment over which I'd ever been admin.

***BirAdam*** Well, do remember the cost of a UNIX license at the time (unless you were using BSD). If you didn’t have thousands of dollars on hand, NT was a good choice.

***Uhhrrr*** But that's 1999, and they were already losing money in 1997, and the article doesn't say why. Sun was why.

***pipeline\_peak*** Look harder

***selimthegrim*** The article is a little imprecise, though. Clark enrolled in the night school at Tulane, which is what is now called the School of Professional Advancement, not the regular school.

***KineticLensman*** They had me at ‘Skywriter Reality Engine’. I’d used vaxen at Uni in the 80s, then got into industry using Symbolics Lisp machines, then had several dark years programming on DOS boxes. The return to a truly innovative workstation blew my mind

***johndhi*** When I was a kid I remember my brother and I asking my dad a ton of questions about SG. We viewed them as this amazing awesome company that made cool looking towers and the fastest computers in the world.

***lowbloodsugar*** >Nintendo 64 was an SGI workstation with ... 4MB of Rambus DRAM at 250 MHz (actually 4.5MB but 512K is visible only to the GPU)

It was actually 9-bit RAM, but the CPU could only see 8 of the bits!

***Arathorn*** There's a surprising amount of good info here about very first IRISes. I ended up with 3 of the very first IRIS 1400 workstations mentioned in the post which NASA Ames bought; at least one of which is still in working order. They were my first unix workstations (running a Unisoft-based sysv/bsd hybrid pre-IRIX variant), and is where I first learnt UNIX, C, IRIS GL, vi and lots of other good stuff. Irritatingly they shipped with an XNS rather than TCP/IP network stack (although I did get hold of a beta IP stack, I never got it work). I did get XNS working on a LAN using their EXOS 101 ethernet cards tho.

In case anyone's interested, their graphics card (GE1, the world's first ever hardware 3D graphics card?) looks like:

https://matrix-client.matrix.org/\_matrix/media/v3/download/m...

...and the PM2 68k processor card mentioned in the post looks like:

https://matrix-client.matrix.org/\_matrix/media/v3/download/m...

...and one of the machines itself looks like:

https://matrix-client.matrix.org/\_matrix/media/v3/download/m...

Suffice it to say that I have a very soft spot for these machines :)

***cellularmitosis*** There's a kid on youtube (username 'dodoid') who put together a series of videos on SGI. Neat to see someone young get super enthusiastic about computing history.

***sneed\_chucker*** I remember to thank SGI every time I format an XFS filesystem.

***betaby*** Yes, XFS has been in the Linux kernel since 2001! Time flies.

***pjmlp*** A bit of trivia, SGI used to host the C++ STL documentation based on HP libraries, pre-adoption into the standard.

***jibbit*** i'll never forget my 1st time coming into a high end Flame suite. it was so exciting.. i don't think i could have been more excited if you'd told me i was stepping into the worlds first time machine

***neduma*** Irix OS was my second best OS exposure after Ubuntu during college days.

***HeckFeck*** What did you do with it?

I’ve always wanted to play with Irix . The UI looks very intuitive.

The boxes are hard to find, so I went for emulation but I couldn’t get it any further than a boot screen in MAME.

***fuzztester*** This reminds me of the book The Soul of a New Machine.

***mvkel*** "it is difficult to get a man to understand something when his salary depends on his not understanding it."

If you're the incumbent, a paradigm shift usually forces you to intentionally cannibalize your existing revenue base in service of the not-yet-proven new thing. That's if you want to survive.

It's incredibly difficult, rare, nearly impossible, to pull such a thing off within a company system.

Imagine being Blockbuster, knowing that to survive, you'd need to transition to a content streaming company.

That's a ton to unwind before you can even buy servers, let alone hire the people to -lead the industry- in such a shift. All to simply remain alive.

***matthewmcg*** Much of this history is also described in Michael Lewis’s book the New New Thing (2000) which profiles Clark and his various ventures. It’s really a snapshot of pre-.com crash Silicon Valley.

***CalChris*** My takeaway from that book was that Clark invented the dot com.

reply