



Articles



People



Learning



Jobs



Games



Get the app



Product Management History #01: The Rise and Fall of the First Data and Spreadsheet Software

...



Ali Elouafiq CEng

Chartered Engineer, ISO 42001 Senior Lead Implementer

Published Oct 28, 2022

+ Follow

In these series I will be sharing content from the archives of the [Computer History Museum](#), that is publically accessible, and is the copyright of the museum. I do not claim Authorship of this publication, sharing it on LinkedIn is to make it accessible and sharable. This account has been augmented with hypertext links, to people mentioned in this account, as well as the photographs from that time, and the links to the events mentioned. [Source: The Creation and Destruction of VisiCalc](#)

The Creation and Destruction of VisiCalc, a Personal Account by [Daniel Fylstra](#) Founder of VisiCorp, conducted by the Software History Center
Abstract and Summary

In preparation for a pioneer workshop meeting on PC software conducted by the [Software History Center](#) in 2004 in Natick, Massachusetts, the organizers sent [Daniel Fylstra](#) , [Dan Bricklin](#) and [Bob Frankston](#) questions regarding the history of VisiCalc.

This was to assist them in preparing to participate in workshop sessions on the history of PC spreadsheets and for individual oral histories to be conducted with each of them by historians in attendance at the pioneer meeting.

Dan Fylstra chose to write the following "white paper" giving his remembrance of the VisiCalc history. He covers how he got started with Dan Bricklin and how the product was marketed so successfully and then how differences of opinion between VisiCorp (Fylstra's company) and Software Arts (Bricklin's and Frankston's company) eventually led to the downfall of VisiCalc.



Time & Money. Commodore®, Atari® & Apple® users get more with VisiCalc™ software.

A financial VP in Massachusetts is cutting the time it takes to prepare month-end reports from three days to three hours.
A California company is replacing most of its time-share computer service with a personal computer and VisiCalc,

more alternatives and forecasting more outcomes. It really increases your decision-making batting average!
When you finish, you can print a copy of the worksheet just as it appears on the screen and/or save it on diskette.

saving at least \$30,000 the first year.

Thousands of other personal computer users are also sold on how VisiCalc is increasing their productivity. Besides saving time and money, they're simplifying their work and getting more information that helps them make better decisions. A typical user reaction comes from a New York dentist:

"VisiCalc has become an integral part of my business."

VisiCalc displays an "electronic worksheet" that automatically calculates nearly any number problem in finance, business management, marketing, sales, engineering and other areas. The huge worksheet is like a blank ledger sheet or matrix. You input problems by typing in titles, headings and your numbers. Where you need calculations, type in simple formulas (+, -, ×, ÷) or insert built-in functions such as net present value and averaging. As quickly as you type it in, VisiCalc calculates and displays the results.

"I am extremely impressed with VisiCalc's capability, flexibility and orderly presentation of instructions."

So writes the director of a New York corporation. He appreciates VisiCalc's powerful recalculation feature. Change any number in your model and instantly all numbers affected by that change are recalculated and new results are displayed. You can ask "What if . . .?", analyzing

Commodore is a registered trademark of Commodore Business Machines Inc., Atari is a registered trademark of Atari Inc., Apple is a registered trademark of Apple Computer Inc.

Circle 24 on inquiry card.

"I like VisiCalc's ease of use."

That response comes from a Utah businessman using VisiCalc for production forecasts, financial report ratio analysis and job cost estimating. Ease of use is VisiCalc's best-liked feature. It's designed for a non-programmer, and has an extensive, easy-to-understand instruction manual.

Users also like solving a wide variety of problems with VisiCalc . . . and solving them their way. VisiCalc can even justify the cost of a personal computer, according to a New Hampshire financial analyst:

"VisiCalc is paying for itself over and over."

VisiCalc is available for 32k Commodore PET/CBM, Atari 800 and Apple disk systems. VisiCalc is written by Software Arts, Inc.

See VisiCalc at your Personal Software dealer. For your dealer's name, call Personal Software Inc. at 408-745-7841, or write 1330 Bordeaux Drive, Sunnyvale, CA 94086.

While there, see our other Productivity Series software: Desktop Plan and CCA Data Management System. They're like time on your hands and money in the bank.

PERSONAL SOFTWARE

See us in Chicago at NCC, booth 881



Introduction

This recollection was written May 1-4, 2004 because I'm expected to attend a conference titled "PC Software: The First Decade," held in Boston, Massachusetts May 6-7, 2004. Just one day ago, I received a list of questions from the conference organizers as follows:

- Where did the technology breakthroughs come from?
- Why was Apple the chosen VisiCalc target machine (not the CP/M microcomputers)?
- What were the financial arrangements?
- How did the partners expect to balance the normal conflicts between developers and marketers?
- How did the introduction of the IBM PC and the competition from other spreadsheet programs affect the VisiCalc strategy?
- What happened to VisiCalc in the long run?
- What enabled Lotus 1-2-3 to be able to outsell VisiCalc and make Lotus Development the leading PC software company by 1984?

I will attempt to answer these questions honestly based on my first-hand recollections and point of view as the marketer of VisiCalc. Doing so is not easy for me, because while the creation of VisiCalc was a positive, exciting experience, the destruction of VisiCalc was a negative, painful experience that I have preferred to leave 'in the past where it belongs.' Over many years, I have declined requests to revisit the VisiCalc story, verbally or in writing; this is the first time since about 1984 that I have written anything on the subject.

Since this is a relatively long account, I will provide quick answers to the above questions as 'teasers' for what follows:

- The conception and design of VisiCalc is rightly credited to Dan Bricklin. VisiCalc did leverage one technology factor that distinguished the Apple II and other new personal computers from CP/M-based systems and older minicomputer designs: Directly addressable video memory that gave software a 'wide path' to the user.
- The [Apple II](#) became the target machine for VisiCalc because I personally persuaded Dan Bricklin that it was the best choice, instead of the [DEC PDT](#) minicomputer that was his original target. I was influenced by Apple's marketing prowess and by Steve Jobs, who personally sold me the [Apple II](#) computer that I loaned to Dan Bricklin for VisiCalc's development.
- The financial arrangements included a cash advance from Personal Software to Software Arts to help cover development costs, and a royalty rate of 35.7% on our sales of VisiCalc, in exchange for exclusive worldwide rights to market all versions of the software. At the peak, we were paying royalties to Software Arts of about a million dollars per month.
- The partners certainly failed to anticipate the 'normal conflicts between developers and marketers,' missing a couple of major opportunities to avoid or resolve potential and actual conflicts. In the later stages, the VisiCorp-Software Arts conflict unfolded, as [Richard Melmon](#) once put it, like a Greek tragedy – the participants (at least on the VisiCorp side) foresaw the outcome and tried many things to avoid it, but every change of course only seemed to move us further towards the demise of VisiCalc.
- The introduction of the IBM PC helped VisiCalc – indeed, revenues and royalties from the IBM PC version of VisiCalc were by far the largest of all the VisiCalc versions for different personal computers. But competition from other spreadsheet programs, combined with what I'll call the 'anti-marketing' of VisiCalc by Software Arts, ultimately doomed the brand and the product.

- VisiCalc lost its market position, but its influence on software has continued to this day. It obviously inspired the design of every other spreadsheet program (especially Lotus 1-2-3); it influenced the design of many other software products on PCs and other platforms; and ideas first expressed in VisiCalc are a basic part of the human factors lexicon.
- Several factors enabled Lotus 1-2-3 to quickly outsell VisiCalc – chief among them, a brilliant design by [Mitch Kapor](#), a brilliant implementation by [Jonathan Sachs](#), and a 'huge for the era' marketing budget from [Ben Rosen](#) (*note: now [Sevin Rosen Funds](#)*). But Lotus enjoyed several kinds of unintentional assistance from VisiCorp and Software Arts. 1-2-3 was able to displace VisiCalc in the PC dealer channel with incredible speed, thanks to a big marketing 'push' from Lotus combined with a big marketing 'pull' of VisiCalc out of the dealer channel by Software Arts.

I prefer to talk, if at all, about the creation of VisiCalc – it was a creative time, full of promise and possibilities. But the questions indicate that the historians or the conference organizers also want to know about the destruction of VisiCalc. This is not a pretty story, and I have simply not talked about it for 20+ years, because I cannot recount the story without talking about actions by Software Arts that I feel were foolish and that strongly contributed to the demise of VisiCalc. I've been happy to see them receive accolades for VisiCalc, and I have no desire to 'pin this' on them or to revive a controversy.

But it appears that the upcoming conference "PC Software: The First Decade" will be a definitive source of information for historians, who are the primary attendees, and there will be accounts by many other participants in the early days of the industry, for whom I have considerable respect. So, with reluctance, I'm offering my account of these events, 'calling it as I saw it,' in hopes that it will be compared to other accounts and understood in context, and will help others to learn from our successes and our foibles and mistakes.

What went wrong for VisiCalc? In my view, several factors: the inherent risks of a software publishing arrangement when the software is central to the business of both sides; a breakdown in communications that occurred as both companies grew and sought to professionalize their management; and mismatch of the roles taken by both sides with the motivations of the principals on each side. The Creation of VisiCalc.

The Creation of VisiCalc: Leading Up to VisiCalc

The story of VisiCalc began for me sometime in the fall of 1978, with a phone call from an MBA student at [Harvard Business School](#) (HBS), Dan Bricklin. I had graduated with an MBA from HBS the previous spring, with a concentration in marketing. A good part of my second year at HBS was spent on a “marketing research project” supervised by my retailing professor, [Claudia Malone](#), in which I researched the personal computer industry, and simultaneously started up my company, Personal Software, which later was renamed VisiCorp. At spring graduation, my Section B classmates had given me – as a joke – the ‘Dan Fylstra Computerized Universe Award,’ because I had incessantly talked about the new personal computers, and had predicted that sometime in the future, we’d all be using one at our desktops at work, and later in our homes.

I had been involved in the fledgling personal computer industry since 1975: After graduating from MIT with a degree in electrical engineering and computer science, I worked for an MIT spinoff company, Intermetrics, where I met [Carl Helmers](#), who was publishing hobbyist microcomputer designs in [Popular Electronics](#) in his spare time. Carl became founding editor, and I became founding Associate Editor of Byte Magazine. Later I served briefly as founding Editor of Computer Dealer, one of the first magazines for the emerging computer store channel.

By the fall of 1978, Personal Software was advertising its PC software in Byte Magazine, and gradually building distribution through computer stores. Our first real hit product, MicroChess, was written by [Peter Jennings](#), who became my business partner in Personal Software; we sold more than 50,000 copies, primarily through [Tandy Corporation Radio Shack](#) stores.

Meeting Dan Bricklin

When Dan Bricklin called, he said he had been referred to me by Charles Kelso, his finance professor at HBS, who had also taught my first-year finance class. Dan said he had an idea for a software package that would function as an ‘electronic blackboard,’ useful for building budgeting and financial planning models. Apparently, Prof. Kelso, who was familiar with interactive financial modeling tools such as PLAN and FINPLAN offered on time-sharing systems, felt that the market was already well served; he advised Bricklin to talk to me, apparently expecting that I would agree, since I was also familiar with such time-sharing based tools through my work at Intermetrics. Intrigued, I invited Bricklin to come to my Allston

apartment, home to Personal Software, to talk further.

Though the idea was only a gleam in Bricklin's eye at this point, I saw great potential in it from the first conversation. Up to this point, Personal Software had offered only technical tools and games like MicroChess, but I was looking for significant business applications. I knew that the PC market was in its infancy, and that we could enter it and grow with it; Personal Software already had a strong foothold in distribution through computer stores.

Apple II as the Target

Bricklin had previously worked at Digital Equipment Corporation, where I understand he worked on a full-screen word processing program, and gained familiarity with DEC's products including a recent entry called the DEC PDT. He said that he envisioned writing the 'electronic blackboard' program for the PDT, but he had heard about the new microcomputers, and was curious about whether they could support such an application. I believe that until he met with me, Dan had not actually worked with a personal computer in person.

We had an Apple II, a Commodore PET and a Radio Shack TRS-80 in our apartment / office, and I showed them all to Dan. I had written much software for Data General minicomputers myself at Intermetrics, and had more recently written some of Personal Software's first software products for the 6502 processor used in the Apple II and PET computers. I enthusiastically argued for the PC as the platform for the future, and especially for the Apple II, because I was impressed with both the Apple II's technical design and with Apple Computer's marketing strategy and prowess. Further, I saw emerging an opportunity to work with both Apple and with Computerland, the computer store franchise operation, which carried the Apple products. Personal Software's sales data was already showing that Computerland stores were gaining ground relative to non-franchised stores, and that Apple versions of products like MicroChess were gaining strength in the marketplace.

Bricklin borrowed our first Apple II – the one I purchased directly from Steve Jobs at a negotiated discount – took it home, and wrote a prototype of the 'electronic blackboard' program in AppleSoft Basic, initially using the Apple's handheld game control to move the cell cursor. (I never did get that Apple II back from Bricklin and

Frankston – it was later used as the test machine during the development of the real VisiCalc program.)

When Bricklin came back to our apartment / office and demoed the prototype to me, my enthusiasm grew. I could see that this tool for budgeting and planning would offer a far better human interface than the established, command line oriented financial modeling tools on time-sharing systems. I had the youthful chutzpah at the time to predict that we would make the time-sharing based financial planning products obsolete. As far-fetched as it seemed in 1978, we proceeded to do exactly that with VisiCalc.

Two Companies

Bricklin told me about his MIT classmate Bob Frankston, and his desire to work with Bob to create the software. I believe I had previously encountered Bob, but I can't remember the details; in any case, the three of us met several times in the negotiation state, and later worked together on a number occasions, usually in Bob's attic, on the design and feature set for the electronic blackboard.

Even though I had set up Personal Software to be a software publisher, paying royalties to software authors like Bricklin, I felt that this project might be important enough to warrant a different approach, so I proposed that Dan and Bob should come to work for Personal Software; but Bricklin politely declined, saying that he had wanted for some time to work with Bob and start a company of their own. I accepted this, and proceeded to negotiate with Dan and Bob as owners of the to-be-formed Software Arts, Inc.

But in retrospect, this choice was the first seed of destruction for VisiCalc. If we had agreed to form one company, many problems that later arose might have been avoided altogether. Two separate companies naturally reduced the level of communication between the parties; this was exacerbated later when Personal Software moved across the country to Sunnyvale, California, while Software Arts stayed in Cambridge, and was further exacerbated when both companies brought in outsiders as president, who had no 'equity in the relationship' with the other side.

Software Design and Development

Dan proceeded to work out many specifics of the design of the 'electronic

blackboard,' meeting with Bob, me, and certainly others. The development stage is well described with photos at Dan Bricklin's Website. My role at this stage was that of the classical software 'product manager' that most companies employ today: I was in close touch with the market as it was evolving in 1978-79, and I also had the technical background to appreciate the development issues faced by Bob and Dan. Sitting in Frankston's attic, we discussed many features, adding some and discarding others; I remember an episode where I pushed hard for general floating-point number formats and scientific functions like EXP, LOG and SIN, arguing that even though we were aiming at a business market, scientists and engineers could also use the 'visible calculator' (and they did).

As the product evolved, I remember our own enthusiasm waxing and waning – there were times when we said to each other, "This tool will be so versatile that everyone with a PC will want to buy it!" and other times when we said, "It's too general-purpose and people won't see how they can use it – maybe no one will buy it." As it turned out, this was one of the rare moments in an entrepreneurial venture when our unbridled optimism was fully justified.

But our realization that the general-purpose nature of VisiCalc could be a marketing problem motivated us early on to create numerous example worksheets that illustrated what VisiCalc could do. These weren't random examples – they were aimed at specific markets and applications, including budgeting and financial planning, inventory planning (due to Bricklin), real estate analysis, insurance applications, and the like. We later created a self-running demo for VisiCalc – one of the first tools of its kind – that cycled through a set of these example models. This self-running demo was used in hundreds of computer stores, helping both salespeople and their customers learn what this new tool could do. We worked together on these things, not worrying about 'whose job it was.'

The Publishing Agreement

Sometime in the early stages of development, we agreed on a royalty rate of 35.7% for VisiCalc, during a meeting at the [Joyce Chen](#) restaurant near Fresh Pond in Cambridge. Much has been written elsewhere about the royalty rate, which I won't revisit here, but nearly everyone who has worked in software publishing would say that this royalty rate was a very good deal for Software Arts. Royalty rates in software publishing agreements have varied widely but have clustered in the 10% to

15% range, with outliers ranging from 5% to 25%. But over time, Software Arts apparently became dissatisfied with the deal, sought even higher royalty rates from others, and finally sought to 'break the contract' and market VisiCalc themselves, as recounted below.

The royalty rate was sustainable for VisiCorp only because of VisiCalc's rapid growth in the market, and our diversification into other "Visi" products with much lower royalty rates, which grew to about 50% of our revenue. At the peak, when we were paying royalties to Software Arts of about a million dollars a month, VisiCorp was earning net profits before tax of 17% to 18%, so Software Arts was receiving about a dollar in VisiCalc royalties for every dollar of VisiCorp's profits on all "Visi" products combined.

Actual signing of the publishing agreement, which was prepared by [Fred Anderson](#), a Boston area copyright attorney, took place some months later. I felt then, and still feel that this software publishing agreement, while far from perfect, did a fairly good job of spelling out the duties and responsibilities of the parties, and it covered many of the potential "trouble spots" known to attorneys from previous book publishing agreements. Five years later, the agreement was tested in court in the VisiCorp-Software Arts lawsuit, and some of its weak points became apparent, especially surrounding the VisiCalc trademark. But I believe that the agreement itself did not cause or exacerbate the conflict that later arose between Software Arts and VisiCorp.

I was, however, naive in that I had far too much faith in the agreement as written, and in the law, lawyers' opinions, and the court system. My faith in these things was shattered in 1984, during the VisiCorp-Software Arts lawsuit.

The Name VisiCalc

Bob Frankston wrote, in an email to Daniel Power dated April 15, 1999, "I came up with the name VisiCalc as a shortening of Visible Calculator." In my opinion this is – to put it charitably – a faulty recollection. Dan Bricklin, on his Website, says that "The product name, as I understand it, was coined at a meeting with Bob Frankston and Dan Fylstra (Personal Software) at Vic's EGG on ONE on Mass. Ave. in Cambridge, MA." This is accurate, and Dan has been careful not to say himself that Bob coined the name.

I have a very clear memory about this. We discussed product names on a number of

occasions. Dan Bricklin had suggested “electronic blackboard” early on, and I believe he also suggested and liked the name “CalcuLedger.” Bob Frankston proposed the name “CompuLator,” and I vividly recall explaining to him that this was just not going to work for a serious business product. At the breakfast meeting between Frankston and me that Bricklin refers to, the subject of names came up once again, and I gave Bob my latest ideas, by writing two names for the first time on a napkin: “VisiCalc” and “VALC,” explaining to him that VisiCalc was a contraction of Visible Calculator. Bob was initially lukewarm at best about the name VisiCalc. But I continued to promote this choice of names to both Dan and Bob, and over time they accepted it.

I was astonished when, in 1984 in the course of the VisiCorp-Software Arts lawsuit, Bob Frankston testified, and signed an affidavit claiming that he had coined the name VisiCalc. I began to learn about ‘lawyer-assisted recall’ of in-person events, and began to lose my naive faith in the contract, the law, lawyers and the court system.

Market Development

As the software development and testing work progressed, Peter Jennings and I were busy with the early fieldwork to prepare for the marketing launch of VisiCalc. As Dan Bricklin recounts on his Website, I ran ‘teaser ads’ in Byte Magazine, with the headline “VisiCalc: How Did You Ever Do Without It?” But the real work consisted of personal selling, and attempting to leverage our selling efforts through others. We flew to San Francisco multiple times, meeting with Apple Computer and Commodore in Silicon Valley, and with Computerland in San Leandro. This opened many doors for us to Apple’s dealers and to Computerland franchisees.

We (Peter Jennings and/or I) first introduced and demo’ed VisiCalc to [Steve Jobs](#) and [Mike Markkula](#) (chairman of Apple Computer) early in 1979. We started the demo with a checkbook balancing spreadsheet that I believe Dan Bricklin had originally created, because it was the simplest example of what VisiCalc could do. At first, Markkula thought we had a dedicated checkbook balancing software package, and he pulled out an Apple-labeled software package on cassette tape, a checkbook program that he had written. Of course, in VisiCalc someone could create the checkbook worksheet in ten minutes, and create a completely different application in the next ten minutes. Markkula soon caught on, and became an enthusiastic

supporter of VisiCalc. But we later reported this somewhat amusing misunderstanding of VisiCalc to Bricklin and Frankston.

I believe the above episode is the origin of an odd comment about me on Bricklin's Website: "Fylstra, needing a program to sell that could do checkbooks, and being an MBA himself so he appreciated the value of financial forecasting, made a deal with my friend Bob Frankston and me." I believe this must be a faulty recollection. Personal Software was never interested in actually marketing a checkbook program – this would have been a minor seller in 1979, to the few consumers who had hobby or personal computers. I was interested in VisiCalc precisely because it could address a broad market of serious business users.

The Whole Product

In Personal Software's marketing efforts for VisiCalc, we were trying to create a "whole product" solution – an idea I learned at HBS, that was strengthened and popularized for high technology companies years later by [William Davidow](#), and most recently by [Geoffrey Moore](#). Besides the software itself which was being created by Software Arts, we knew we needed a user guide, example spreadsheets, and business-oriented product packaging, which hadn't really appeared in the PC software industry up to that point – as described by Peter Jennings at www.benlo.com. But we also had to provide pre-sales training and post-sales support – in this era when the whole idea of personal computing was new and unproven in business, we had to create an environment where prospective customers could satisfy themselves that they would be able to successfully use VisiCalc. And of course, in 1979 the vast majority of prospects did not own a PC – they would be looking for a PC, software, and support in a 'whole product' solution. Personal Software could not provide all of these things on its own. We needed to create an alliance with other players who could. So I deliberately focused on Apple Computer and Computerland, seeking to create alliances with the players I felt had the best chances of success.

At Apple, Jobs and [Markkula](#) introduced us to several marketing managers – I especially remember Mike Connor, who became enthusiastic about VisiCalc and who explained to me how Apple worked with several regional distributors, who in turn serviced individual computer stores. With help from Apple, I was able to give presentations of VisiCalc at several regional distributor's meetings, with hundreds of

computer store managers in attendance. This was an exciting time. I remember that I used a big-screen digital projector – today an everyday tool, but in 1979 a rare and expensive resource – to demo VisiCalc in a theatre-like setting. The effect of scrolling the small Apple display to reveal a seemingly infinite worksheet, and of changing one number and seeing all the calculated values change, was electric on the audience. Computer store owners and their salespeople soon realized that they could sell an entire Apple Computer system and VisiCalc software, and many of them learned to use and demo VisiCalc from our example spreadsheets and self-running demo.

With Steve Jobs' help, I was introduced to [Regis McKenna](#) firm, who became our PR agency. Our first press tours introduced many journalists to the spreadsheet and the power of the Apple II for business professionals – the Apple II was not exactly portable, and I remember carrying the full weight of an Apple II system with its external disk drives and monitor around city streets in New York and Boston. I introduced VisiCalc to Ben Rosen and [Esther Dyson](#) ; Rosen soon wrote a visionary piece about VisiCalc, the "tail that may wag the personal computer dog," in his Morgan Stanley Electronics Letter (reproduced on Dan Bricklin's Website). We also ran what I believe were the first full-page, four-color process ads for a software product in the PC industry trade magazines.

Moving Apart

In mid 1979 – realizing that we were constantly dealing with many firms in Silicon Valley – we decided to move Personal Software from Boston to the San Francisco Area, ultimately settling in Sunnyvale, California. This was an excellent move for Personal Software and for the marketing of VisiCalc, but it also meant that Software Arts and Personal Software lost further day-to-day communication and moved further apart. I believe that, as both companies grew and new people joined their teams, we each inevitably developed our own culture, 'stories' and history, and our views of the market and VisiCalc development diverged. People joining each company learned different versions of the story – no doubt emphasizing the role of the company they joined, and tending to minimize the role of the other. This certainly influenced the attitudes of the outside presidents later hired to run each company: Terry Opdendyk at VisiCorp, and Julian Lange at Software Arts. In retrospect, further seeds of the destruction of VisiCalc were being sown.

Success and Frictions

By the time VisiCalc was ready for launch in the fall of 1979, the market was primed, and sales took off from the very beginning – our early sales figures are detailed on Peter Jennings' Website. But I believe that the rapid success of VisiCalc in the marketplace helped sow the seeds of destruction of VisiCalc years later. Removed by the separation of the companies and geographic distance, it probably wasn't apparent to Bricklin and Frankston that we had done so much to prime the market for VisiCalc, and to create a whole product solution. I believe they felt that VisiCalc was so revolutionary that it was selling itself, and Personal Software was simply along for the ride. What's more, our PR efforts bore fruit, with many articles in the national business press and trade press. Even though I personally tried, over and over, to explain to analysts and journalists the key roles played by Bricklin and Frankston, many articles appeared mentioning only VisiCalc and Personal Software, sometimes with photos of me.

Personal Software paid royalties to Software Arts monthly at the 35.7% royalty rate, which meant that Software Arts' financial fortunes zoomed along with ours at Personal Software. But widespread recognition for them was slower in coming; they hired their own PR firm to tell their story independently of our efforts. Even the 35.7% rate apparently wasn't enough: a year later, several people told us that Software Arts had visited other prominent software firms such as Microsoft and MicroPro, offering them TK!Solver, their second product, but insisting on a 50% royalty rate. There were no takers, so Software Arts ended up publishing TK!Solver themselves.

Recommended by LinkedIn

Off The Shelf Methods Don't Work For Us

Ivar Jacobson International · 1 month ago

Paradox of the Active User

Goutham Ravi Kumar Ravutla · 11 months ago

The 'soft' underbelly of Software - Elaboration 2/3:...

Madhav Vattikuti · 1 year ago

Venture Capital, Outside Presidents, and the Merger Attempt

Around this time, I raised about \$500,000 of venture capital for Personal Software from [Arthur Rock](#) and [Venrock](#) Associates (the Rockefeller family's venture arm), both of whom were investors in Apple Computer. [Arthur Rock](#) and [Hank Smith](#) from [Venrock](#) joined our Board of Directors. This first-ever investment in a PC software company by top-ranked VCs drew further attention to Personal Software, and opened doors for many other PC software ventures later.

We also started talking earnestly about the interdependence between Personal Software and Software Arts. We knew that there were risks in having the VisiCalc product split between the two companies. Hank Smith suggested that we should explore a merger; I agreed, and we held a preliminary meeting with Dan Bricklin and Bob Frankston. I was ready to make this move, though I was uncertain about how it would all play out in terms of equity ownership; I learned later that Dan and Bob were also interested and prepared to negotiate. If we had successfully negotiated a merger, I think that the entire VisiCalc story might have had a very different ending. But it did not – why?

In this same time frame, also at Hank Smith's suggestion, we recruited [Terry Opdendyk](#) from Intel to be president of Personal Software. The company was now growing at rates that far outstripped my management experience as a young entrepreneur, so I readily embraced this move. Somewhere in this time frame – I do not know the timing or the details – Software Arts recruited [Julian Lange](#), a professor at Harvard Business School, to serve as president of their firm. These moves changed the dynamic of the relationship between the two companies. Terry Opdendyk felt that a merger was unnecessary and would be expensive; I am not sure whether Julian Lange was involved or had a view at this time – but in any case, the merger idea simply wasn't pursued further.

Unfortunately, both new company presidents quickly seemed to alienate the

principals at the other company: I found Julian Lange not at all informed about what we were doing to market VisiCalc, but full of lectures and critical comments about what we should be doing. Terry Opdendyk visited Bricklin and Frankston, feeling confident that he could re-negotiate terms of our publishing agreement with them, including a lower royalty rate. But Bricklin and Frankston wouldn't even consider this – they wanted to increase the rate – and the episode only served to make them suspicious of Terry's intentions.

Relations between Dan, Bob and I had begun as friendly, and remained at least cordial for several years. But with the arrival of others in various management roles, relations between the two companies grew increasingly formal. The principals rarely spoke, not wanting to "go around" the people they had hired – a mistake in retrospect. I suspect that, on both sides, people often weren't sure who should say what to the other company, especially about problems – so that much communication just didn't happen.

New Platforms

As sales of VisiCalc and the Apple II soared, there was tremendous demand from other personal computer manufacturers for us to implement VisiCalc on their platforms. We wanted to fulfill this demand and capture as much of the PC market for VisiCalc as possible. Personal Software worked with the manufacturers and prioritized the platforms, but the work of actually porting the code fell to Software Arts. Personal Software marketed most of these VisiCalc versions itself, dealing directly with computer dealers who were carrying several PC product lines. Although each version required a development effort, demand was so strong that both companies realized a payoff rather quickly on most of these platforms. The Commodore PET and the Atari 800 came first, because both were based on the same 6502 microprocessor as the Apple II, and both had memory-mapped video displays. Others, like the Radio Shack TRS-80, came later. Through our work with Atari, with gained relationships with Atari's manufacturer's reps, who were exceptionally strong; we later hired the individual salespeople working for the rep firms who were most enthusiastic about VisiCalc, creating a highly effective national sales force calling on computer stores, electronics stores and other retail outlets.

CP/M, the Osborne PC, and SuperCalc

Why was VisiCalc not ported early to the various Intel 8080-based personal

computers running the CP/M operating system? There were two main reasons, one technological and the other market-oriented. First, Software Arts had written VisiCalc in 6502 assembly language, so porting it to CP/M machines meant rewriting all the code in 8080 assembly language. Further, Software Arts had utilized the memory-mapped video display of the Apple II, PET, Atari, and TRS-80 computers to implement the fast screen updating that was a key feature of VisiCalc. CP/M computers generally did not have memory-mapped displays; instead they supported external CRT terminals over a serial line. Implementing VisiCalc for these systems would have required a redesign and partial rewrite of this code. Second, the CP/M marketplace was far more fragmented than the 'appliance computer' marketplace, with scores of manufacturers and many small differences between machines – supporting a wide range of these systems would have been difficult, for two companies that were already strapped to keep up with demand.

But the absence of VisiCalc on CP/M machines created an entry point for competitors, which was soon filled. Adam Osborne called on me at Personal Software to see if we could provide VisiCalc for his to-be-announced portable computer, the Osborne PC. I was certainly intrigued with this idea, but Software Arts couldn't promise a version for this CP/M system in the time frame that Adam needed. So Osborne turned to Sorcim, creators of SuperCalc, which was written for the 8080 processor and which didn't require a memory-mapped display. Sorcim soon became the primary supplier of spreadsheets for CP/M computers.

Other competitors began to appear, notably MultiPlan from Microsoft, with new features not present in VisiCalc. But at this stage, Personal Software's revenues, and Software Arts' royalties, were growing by double-digit percentages every month – in 1981, I believe, our revenues almost quintupled.

The "Visi" Family of Products and the Name VisiCorp

Personal Software had always published other software products, before and after the launch of VisiCalc. Sales of game programs such as MicroChess also grew rapidly, but became an ever-smaller percentage of our revenues, so we finally returned rights to these products to the software authors and concentrated entirely on business software. Using programs written by several software authors that Personal Software published, we created a complete 'family' of software products aimed at analysts and professionals: The "Visi" family, including VisiPlot and

VisiTrend for graphics and statistical analysis; VisiFile, a desktop database program; VisiWord, a word processing program; VisiSchedule, a project management program; and VisiLink and VisiTerm, online database access and terminal emulation programs. We created user guides, packaging, and advertising to enhance the 'product family' effect. Sales of these products also grew rapidly, and the family further strengthened the position of VisiCalc. But VisiCalc remained our best seller. At the peak, our revenues were split roughly 50-50 between VisiCalc and the other Visi products.

With so much market momentum behind the "Visi" name, it was natural for Personal Software to change its corporate name to VisiCorp. The name change was a well-executed marketing program of its own, with powerful effects on our brand position. But in doing so, we became so wedded to the "Visi" name that we could not afford to give it up. This became a serious issue or "seed of destruction" when Software Arts came to desire ownership of the VisiCalc name for itself.

Competition and VisiCalc Features

As marketers, we at VisiCorp were always keenly interested in what our customers wanted, and what competitors were doing. As Peter Jennings describes on his Website, he began building a list of new features desired for VisiCalc almost immediately after the first version for the Apple II shipped in the fall of 1979. I also created a list of desired enhancements to VisiCalc, and we later turned over these lists to the person we hired as product manager for VisiCalc: Mitch Kapor, who had written VisiTrend and VisiPlot and who was eager to come to work for VisiCorp.

By this time, competitive spreadsheet programs were rapidly improving, and most offered features not present in VisiCalc. Our marketing momentum, our strong relationships with computer dealers, and the presence of the Visi family all helped make up for the new features missing from VisiCalc.

There was an unfortunate side of this success. VisiCalc was falling behind, but Software Arts royalties were still growing rapidly. I believe that, without the direct market feedback that would have come from working in the same company as the salesforce in the field, Bricklin and Frankston came to feel that new spreadsheet features, from variable columns widths to charts and graphs, were not critical, compared to ports to new platforms and their efforts to develop other interesting

new products. Our course, we sought to pass on the market feedback to them, and we even “hammered on the table” to them about the most important features. But it seemed that our message fell on deaf ears.

At one point, our development manager, [Eugene Buechele](#), visited Software Arts, which had grown to over 100 employees. He returned to VisiCorp with a report that he had been introduced to only 2 or 3 relatively junior engineers working on VisiCalc; all the others were apparently working on other projects. The nuts and bolts of small, customer-requested improvements to an existing product are rarely very exciting or creative, but they are essential, especially in a hotly competitive market. For whatever reason, these improvements just didn’t happen with VisiCalc.

I remember being personally frustrated when Microsoft launched a new advertising campaign for MultiPlan: A double-page spread in all the popular magazines with the headline: “VisiCalc was a Swell Idea for Then. MultiPlan is a Great Idea for Now.” The ad featured more than a dozen screen images of MultiPlan spreadsheet features, each with corresponding screens for VisiCalc – but these were blank, because VisiCalc didn’t have those features. It was a powerful challenge to our product leadership.

When the IBM PC version of VisiCalc was launched, perhaps two years after the Apple II version, it had essentially the same feature set as the original Apple II version, and supported only 64K of the IBM PC’s 256K of main memory. Meanwhile, our product manager for VisiCalc, Mitch Kapor, had returned to Cambridge – and had begun working on Lotus 1-2-3. I heard second-hand that Bricklin spent many hours with Kapor, walking around Fresh Pond in Cambridge, telling him about all the problems Software Arts was having with VisiCalc and VisiCorp. Mitch Kapor, who had our lists of desired features for VisiCalc, who knew that Software Arts wasn’t implementing them, and who knew that relations between the companies had deteriorated, was ideally positioned to design a competitive spreadsheet: Lotus 1-2-3.

VisiOn

Dan Flystra with VisiOn and VisiCalc © Roger Ressmeyer/CORBIS

VisiCorp had been aware for some time that the competition had caught up with or surpassed VisiCalc, and that the "Visi" family wouldn't last forever. We resolved to build the next generation of desktop applications, based on the new ideas for graphical user interfaces from Xerox Parc and other sources. As Apple Computer worked to develop similar ideas in the Apple Lisa and then the Macintosh, we were working to develop VisiOn.

VisiOn offered a fully graphical user interface, with standard menus and dialogs, point and click control with a mouse, a multi-tasking kernel, and a powerful application programming interface (API) for modern graphical applications. Developed by Scott Warren and Dennis Abbe, two brilliant engineers on a par with Bob Frankston, Jonathan Sachs and Charles Simonyi, it was a technological marvel, with capabilities that wouldn't appear in other systems for years. The first public showing of VisiOn probably prompted Bill Gates to initiate the development of Microsoft Windows.

The full story of VisiOn would take too much space in this already-long account of VisiCalc. But I should say in summary that VisiOn was a too-ambitious undertaking and a strategic mistake for VisiCorp. VisiOn was a 'discontinuous innovation,' like the IBM Micro Channel in the early 1980s and Lotus Improv in the early 1990s, launched at a time when the broad market was rushing to adopt en masse the already-proven technology of individual desktop applications – a strategic marketing mistake. From a business strategy viewpoint, VisiOn carried risks that we didn't appreciate: It was a threat to far too many key players in the PC industry, including Apple, Microsoft, and others – motivating all of these players to turn against VisiCorp. It also led to a direct threat from Software Arts of a lawsuit to block the shipment of VisiOn, which triggered the VisiCorp-Software Arts lawsuit.

VisiCorp developed a full set of GUI, mouse-based applications for VisiOn, including word processing, business graphics, database, and spreadsheet packages. We knew that a spreadsheet for VisiOn was an essential component, so we disclosed our plans early on to Software Arts, and asked them to develop VisiOn Calc. Software Arts wasn't interested. So VisiCorp engineers wrote VisiOn Calc from scratch, using VisiOn's powerful APIs. The result was a GUI spreadsheet that easily surpassed VisiCalc in functionality.

VisiOn was previewed to the public in early 1983, and for a relatively brief time, it

was hailed by analysts and industry observers as the wave of the future. Software Arts apparently changed its view of VisiOn, deciding that it was indeed a threat to VisiCalc.

The Destruction of VisiCalc

Dan Bricklin's Website describes "The End" of VisiCalc as follows: "VisiCalc's publishers, VisiCorp (originally named Personal Software), sued Software Arts in September 1983, much to the surprise (and dismay...) of us at Software Arts. Software Arts countersued. The lawsuit went to a preliminary injunction hearing in 1984, where the judge did not grant the injunction requested by VisiCorp. In the summer of 1984, a settlement was reached. VisiCorp was eventually sold off to various players. Software Arts' assets were sold to Lotus Development Corporation, the creators and publishers of the 1-2-3 spreadsheet, in 1985. Lotus decided not to continue publishing VisiCalc."

Here is where the breakdown of communication between Software Arts and VisiCorp had its most serious consequences.

The Lawsuit

I know for certain that, while we certainly felt that Software Arts had breached our publishing agreement by not delivering enhancements and new features for VisiCalc, VisiCorp wasn't even considering the idea of filing a lawsuit until we received an in-person message from an officer of Software Arts (I believe this was Tracy Licklider, speaking to one of VisiCorp's key people, but not to me) that Software Arts was going to sue VisiCorp to block the first shipment of VisiOn. This news was the equivalent of a thermonuclear first strike to us: We had invested several years and more than \$10 million in VisiOn, trying to leapfrog the competition that had already surpassed VisiCalc, and reclaim market leadership. Software Arts' threat to block the shipment of VisiOn was perceived as a threat to our very survival. We leaned heavily on our attorneys, whose advice was "if there's going to be a lawsuit anyway, it's better to go first with a complaint for breach" – and so the complaint was prepared and filed.

On the day the lawsuit was filed, I paced for hours, wondering how Bricklin and

Frankston would respond. Were they dead set on the idea of suing us to block shipment of VisiOn? Would they understand that our move was defensive? Would they call and ask whether the matter could be resolved out of court, as we hoped? I very much wanted to call Bricklin myself at that moment, to clearly communicate our concerns and intentions. I talked to our in-house attorney, [Valarie McInroy](#), about this idea, but she was vigorously opposed to it. As a lawyer concerned with our litigation tactics, she felt that such a call would be a serious mistake. Others at VisiCorp also opposed the idea, so I ultimately dropped it.

When – much later – I explained our reasons for filing to Dan Bricklin, he expressed much surprise. He denied that Software Arts planned to sue to block the shipment of VisiOn, and expressed disbelief that Licklider had said the things we had heard. So in retrospect, the call that my gut instinct told me to make should have been made. This was ultimately my fault – I was chairman of VisiCorp, and could have made the call despite our attorney's and others' objections – but I chose not to do so. And Bricklin evidently chose not to call me.

Filing of the lawsuit doubtless hardened the feelings of mistrust and hostility that had already developed between the parties. But despite its emotional impact, the lawsuit itself had only a modest impact on VisiCalc sales. This may seem like an incredible statement to Bricklin and Frankston, but it is certainly true. Lawsuits aren't positives, but any number of lawsuits between companies like Apple, Microsoft, Sun, Intel and AMD have wound their way through the courts for years without materially impacting sales. The real action happened in the marketplace, not in the courtroom.

VisiCorp worked hard to minimize the public relations impact of the filing – we issued only a short public statement in response to press inquiries, declined interview requests, and did our best to keep the matter out of the news, since we felt this could only hurt, not help – and the lawsuit did disappear from the news after a brief flurry of coverage. VisiCalc sales by this time were declining from their peak, but were still very substantial, and VisiCorp continued selling VisiCalc aggressively. For many months after filing the lawsuit, we continued to pay many hundreds of thousands of dollars every month in royalties to Software Arts, though we eventually exercised a clause in the contract concerning the withholding of royalties.

(extra Readings: New York Times : [How Software Winner Went Sour](#), [VisiCalc LawSuit is Settled](#))

Software Arts' Anti-Marketing Campaign

By this time, Lotus 1-2-3 had been launched. 1-2-3 was an excellent product; while it was initially billed as an 'integrated application' combining a spreadsheet, charts and graphs, and database functionality, it was received by the market as a spreadsheet, and as a replacement for VisiCalc, since it could open VisiCalc files and it emulated most VisiCalc keystroke commands. Thanks to a multimillion dollar investment from [Sevin Rosen Funds](#), 1-2-3 was introduced with an advertising and promotional budget that exceeded anything we had been able to spend in an equivalent amount of time, with a major effort directly towards the computer dealers who carried VisiCalc and the IBM PC.

But this was not the only marketing program directed against VisiCalc. Amazingly, Software Arts launched a major advertising and public relations campaign of its own that had the effect of destroying dealer distribution of VisiCalc, and with it, VisiCalc sales.

Software Arts launched their campaign with a PR blitz that began at [Esther Dyson](#) PC Forum in (I believe) 1984. Literally on cue, fanning out into the audience of industry executives, analysts and journalists, Software Arts executives announced that they were 'breaking the contract with VisiCorp' and they would begin selling VisiCalc themselves. The same day, FedEx packages from Software Arts arrived at the offices of all the key publications covering the PC industry, with a lead press release entitled "Software Wars: Battle Erupts Between Author and Publisher of VisiCalc," and with 8x10 glossy photos of Bricklin, Frankston and others at Software Arts. This was followed by full-page newspaper ads in major newspapers such as (I believe) the Wall Street Journal, the New York Times, and the Boston Globe, announcing that Software Arts was selling VisiCalc direct to consumers for \$99. At the time, computer stores were selling VisiCalc for \$200, and they were buying it from VisiCorp for \$120.

Software Arts surely did not intend to destroy VisiCalc – they meant to destroy their publisher, VisiCorp, and reap the benefits of VisiCalc for themselves. ('Destroy' may sound unduly harsh, but Software Arts' actions were entirely consistent with a threatened plan to 'undercut our dealers, pressure our employees and shareholders,' and the like, outlined to me at some length in person earlier, in a diatribe laden with cursing and swearing, by their president Julian Lange, in response to a lawsuit settlement overture from VisiCorp.)

But (in my opinion at least), even from Software Arts' viewpoint, the folly of this marketing plan was obvious. Of course the PR campaign created controversy and negative publicity, and of course the ads failed to efficiently reach the desired audience. But the heart of the matter was that this was 1984, not 1995, and people weren't ready to buy a spreadsheet through a direct marketing ad in a newspaper. Most prospects needed to buy a computer at the same time they bought a spreadsheet program, and they needed the sales support they could get at a computer store. For this simple reason, dealer distribution was the sensible strategy for a program like VisiCalc or 1-2-3 in this era. The primary effect of Software Arts' \$99 advertising offer was to cause computer dealers across the country, whose wholesale cost was \$120, to throw up their hands on VisiCalc, and turn to the obvious alternative – Lotus 1-2-3. As Lotus's intensive marketing campaign was pushing 1-2-3 into retail stores, Software Art's 'anti-marketing' campaign was pulling VisiCalc out of the stores. The distribution 'footprint' that VisiCorp had spent five years building was destroyed in perhaps five months.

I can only cite the parable of the man who killed the goose that laid the golden eggs. VisiCorp was the goose, and the golden eggs were the million-dollar-a-month royalties. Just as in the parable, Software Arts' attempt to kill the goose and get all of the eggs at once for itself simply destroyed the market position of VisiCalc altogether.

The Injunction Hearing

Software Arts' anti-marketing campaign was of course opposed by VisiCorp in court. This led to the preliminary injunction hearing that Bricklin mentions in his brief summary of the lawsuit – VisiCorp sought the injunction to prevent Software Arts from misappropriating the VisiCalc trademark, which VisiCorp owned. This was actually the only court hearing that occurred in the course of the lawsuit. I learned the hard way about lawsuits and the courts: After a day of testimony, the judge decided to postpone the rest of the injunction hearing – for about four months. By the time the hearing resumed, most of the damage in the marketplace had been done. With the burden of proof for an injunction on VisiCorp, and the waters muddied by language in the publishing agreement concerning disposition of the VisiCalc trademark (too complex to explain here), and Frankston's sworn affidavit that he had coined the name VisiCalc, the injunction was not granted – but the issue

was practically moot at that point.

VisiCorp fought on in the marketplace for a long time thereafter, but we knew the game was basically over once the VisiCalc brand name and distribution had been destroyed. Terry Opdendyk resigned, and I assumed the role of president, with a primary job of laying off people and selling off assets. We enjoyed something of a renaissance when I managed to raise about \$10 million in cash, by selling both the VisiOn technology and a subsidiary company, Communications Solutions. But a dilutive venture financing at this juncture was enough to kill off my enthusiasm for continuing. VisiCorp merged with another venture-financed startup, Paladin Software, run by Roy Folk and Lynn Brock; the combined company was named Paladin because [Regis McKenna](#) felt that the "Visi" name had been damaged beyond repair. Soon thereafter, I left the Board of the combined firm. Paladin's story continued for a while thereafter, but this was, for me, "The End."

Summary Comments

What lessons can be learned from the VisiCalc story? I can only offer my own opinions. At the outset, I cited the inherent risks of a software publishing arrangement when the software is central to the business of both sides; a breakdown in communications that occurred as both companies grew and sought to professionalize their management; and a mismatch of the roles taken by both sides with the motivations of the principals.

While software publishing is obviously an appropriate and viable business strategy in many situations, I would caution anyone against a software publishing arrangement when the product involved is too important to the business of either side. [Richard Melmon](#), who was VisiCorp's Director of Marketing, applied this lesson when he and [Trip Hawkins](#) founded Electronic Arts – Rich later told me that they had many disputes with software authors, but no single dispute was ever important enough to seriously hurt the company. In contrast, I have over the years heard stories of many smaller software publishing companies whose business was basically wrecked by disputes with software authors.

Software Arts' decision to pursue its 'anti-marketing' strategy, and kill the goose that laid the golden eggs, is hard to understand from a financial or marketing standpoint, but it makes some sense if recognition was ultimately more important than money

to Dan Bricklin and Bob Frankston. Although we certainly tried, VisiCorp probably did not do enough to ensure that Bricklin and Frankston were recognized for their role as developers of VisiCalc. Better communication between Dan and Bob and me, and actions taken to ameliorate this issue might have made a real difference.

Ironically, I never really wanted the personal recognition and acclaim that came initially as we successfully marketed VisiCalc. I did not look forward to the next interview or press tour, but I was very determined to make VisiCalc successful, and felt it was my duty to use the power of public relations effectively. The motivations of the principals in terms of recognition versus money were mismatched compared to the roles we played. I hope that the past 20 years, when I have declined to talk about the VisiCalc story at all, while Bricklin and Frankston have been honored for their roles on so many occasions, have redressed the issue of recognition.

I have never been good at history, because my attention always has been, and still is, focused on the future. I've put an effort into this account for the sake of the historians who will be attending the upcoming conference, "PC Software: The First Decade." There have been, and doubtless will be other accounts from different points of view, and I certainly do not claim an infallible memory or Mt. Olympus-like objectivity. But I've done my best to be truthful, even about the difficult parts of the story.

My personal desire is to leave the VisiCalc story where it belongs for me – in the past. The future is far too interesting and promising for me to do anything else.

[Original Source: The Creation and Destruction of VisiCalc](#)



Like



Comment



Share



13

To view or add a comment, [sign in](#)

More articles by Ali Elouafiq CEng

Mar 19, 2023

A Blueprint for the Software Engineering Revolution: Taking Cues from Accounting and Law.

To the BCS, The Chartered Institute for IT President, Dear Gillian Arnold FBCS,

As a member of the British Computer...

8 · 2 Comments

Oct 22, 2022

When you find your Homework cited in +40 Patents by AT&T 📢🌐✅

Never under-estimate your class work. While doing a portfolio review, this week I found by chance more than +40 AT&T...

125 · 11 Comments

Oct 20, 2022

5 Coding Tips for Analysts to become better Programmers 💡🧐💻

After a prompt from our team-mate Jessie Miller on how analysts can be better programmers, in a context where coding...

5

Oct 19, 2022

What is Synthetic Data 🚀📊? why is it relevant to your process 🤔

When you want to analyze business data, you are either analyzing it yourself, or you give it to somebody else to do it....

15

Show more

Insights from the community

Library Services

How do you balance accuracy and efficiency in cataloging?

User Experience Design

Here's how you can conduct a post-mortem analysis after a failed project in UED.

Critical Thinking

You have a critical thinking project to complete. What flowchart tools should you use?

Decision-Making

You're building a data-driven business. What's the best tool to make sure you're on the right track?

Decision-Making

How can you improve your team's decision-making with software tools?

Computer Hardware

You're leading a software-focused team. How can you make hardware considerations a priority for them?

Show more ▼

Others also viewed



How to Write Power Fx in Power Automate Desktop

Virag Shah · 1y



Efficient File Naming: Streamlining Your Digital Organization

Jessica.Genius Butler · 1y



TO
PURCHASE
OR NOT?

To Purchase and Deploy or Not?

Valerie Merrill - FLPI · 10mo



How to document existing processes and redesign new ones

Fredrik Engstrand · 11mo



Splitting ODT Files: A Comprehensive Guide

Shubham Dixit · 5mo



The 6 formats for written work instructions and how to choose the right one.

Brian Wood · 2y

Show more ▾

Explore topics

Sales

Marketing

IT Services

Business Administration

HR Management

Engineering

Soft Skills

See All

© 2025

- Accessibility
- Privacy Policy
- Cookie Policy
- Brand Policy
- Community Guidelines

- About
- User Agreement
- Your California Privacy Choices
- Copyright Policy
- Guest Controls
- Language