

# UC Berkeley

## Berkeley Scientific Journal

### Title

The Rise and Demise of Software

### Permalink

<https://escholarship.org/uc/item/5k72m9s5>

### Journal

Berkeley Scientific Journal, 24(2)

### ISSN

1097-0967

### Author

Xu, Candy

### Publication Date

2020

### DOI

10.5070/BS3242049342

### Copyright Information

Copyright 2020 by the author(s). All rights reserved unless otherwise indicated. Contact the author(s) for any necessary permissions. Learn more at <https://escholarship.org/terms>

Undergraduate

# The Rise and Demise of Software

BY CANDY XU

The field of information technology bloomed in the late 20th century. With the continuous advancement of hardware during this period, accompanying new software products raced into the market. Many of them, such as Lotus 123 and the Pascal language, became crucial features of their systems and took their respective companies to new heights. However, many of these software products did not last long. Fleeting like sparks, they were quickly replaced by their competitors. Now, all that is left behind are their creators' innovative ideas, ideas that have set up the foundations for our current generation of software products.

## RISE

The development of computer applications revolutionized both technology-based and non-technological industries. "Tech" was no longer a mysterious term used by researchers and computer enthusiasts, but a product accessible for use in consumer's daily lives. Lotus 123, a spreadsheet program developed by Lotus Software, was one such application (Fig. 1).

As the spreadsheet standard in the 1980s, it was widely utilized in various fields including biology and economics.<sup>1</sup> In one scenario, running statistical significance tests across large datasets in biological research was made more efficient by using Lotus 123; in another, multiple reference peak identification methods for certain chemical techniques were developed using the software.<sup>2,3</sup> The Lotus file compatibility structure even became the de facto industry standard for the exchange between database and spreadsheet programs.<sup>1</sup>

Aside from commercial softwares, computer scientists also made user-friendly innovations in programming languages. The Pascal language, first released in 1970, is a programming language that was also highly influential in a number of different industries. Due to its structured design philosophy, Pascal—more specifically, its extension Pascal-FC—became very popular among educators because of its ability to demonstrate running functions concurrently and give students hands-on experience with code.<sup>4</sup> Exciting and unprecedented products like Pascal quickly became extremely widespread.

## DOWNFALL

These applications' prevalence, however, did not last long. Lotus 123's popularity decreased as it was soon replaced by Microsoft Excel in the early '90s. Although there are many hypothesized reasons for Lotus 123's downfall, the singular reason for its demise remains unclear. One hypothesis for Lotus 123's downfall is its failure to match the strong economic modeling features of Microsoft Excel. Excel's Binomial Option Pricing Model, for example, is an extremely powerful tool that streamlines mathematical work that analysts would otherwise do manually, and is a feature that Lotus 123 was unable to effectively mirror.<sup>5</sup> Furthermore, Excel's ability to create

"The Lotus file compatibility structure even became the de facto industry standard for the exchange between database and spreadsheet programs."

modeling tools such as large decision trees, hugely supports the needs of the financial industry, and thus outperforms Lotus 123 in this domain. Aside from the technological side of the story, Excel also had a huge strategic and marketing advantage. Microsoft DOS, later Windows, was the dominant operating system in the market. It was thus much easier for Microsoft to market Office to their existing customer base and to develop applications that aligned better with their own system. What's more, Lotus 123 is a 32-bit program, making it incompatible with 64-bit Windows 10 unless installed in a complex process using a CD, further contributing to its current elimination from the spreadsheet market.<sup>6</sup>

Similarly, Pascal was eclipsed by the C language. The TIOBE index, an indicator for the popularity of programming languages, indicates that Pascal ranked number 5 in popularity in 1985, but had decreased to 229 in 2020, while C remained consistently ranked in the top 2 spots.<sup>7</sup> The differing design philosophies of Pascal and C likely caused the eventual decline in Pascal's popularity. Pascal is more secure than C, making it harder for outsiders to hack into. C, on the other hand, has greater flexibility and can complete more dynamic tasks, thus giving it appeal to a broader audience.<sup>8</sup> Notably, C is also a lower level programming language, meaning that it is better at directly communicating with the hardware of a system. These factors, however, are only some of the reasons for Pascal's downfall.

## LASTING EFFECTS

Despite their decline, applications like Lotus 123 and Pascal remain foundational to future generations of software. Products based on these applications continue to support a variety of subject fields, both in academia and in industry. For instance, Turbo Pascal, released over a decade after Pascal, is a famous system that compiles the Pascal language and provides an integrated development environment to the users (Fig. 2).<sup>9</sup> It directly connects the rather technical programming language to a wide range of users, increasing the accessibility of Pascal to educators. And while it has faced competition from other Pascal compilers, Turbo Pascal stands out due to its low price and short compiling time.<sup>9,10,11</sup> Bruce Webster, famous software engineer and Adjunct Professor at Brigham Young University, also noted

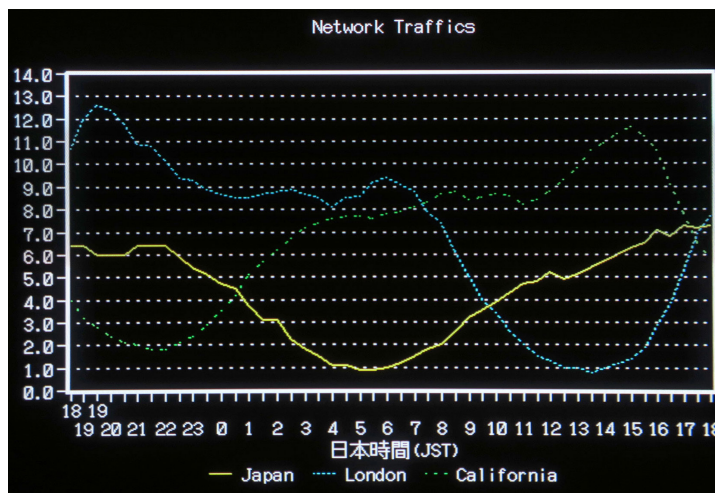


Figure 1: Lotus 123 running on PC-98 DOS in its Japanese version. It displays the network traffic of Japan, London, and California throughout a day.

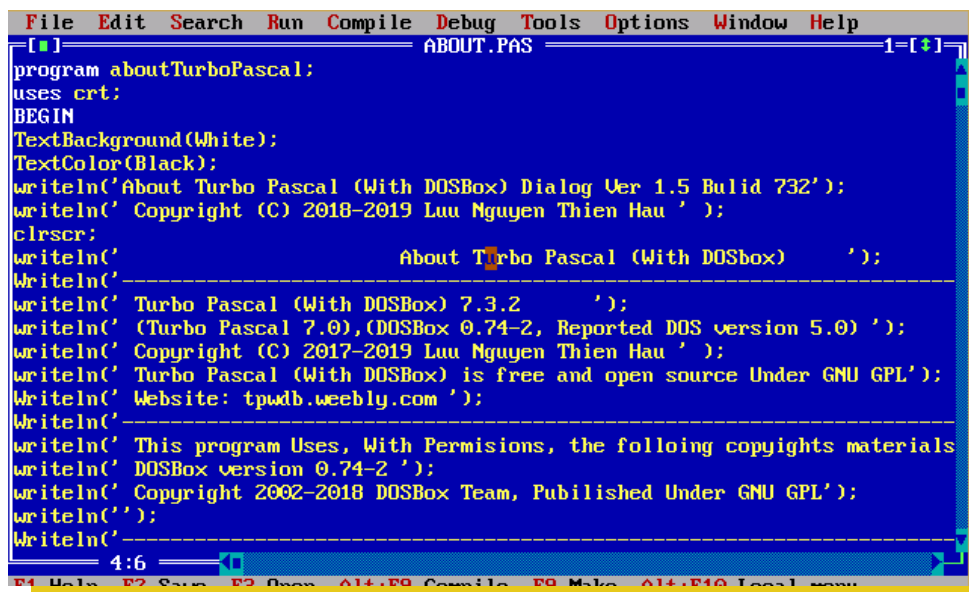
in *Byte Magazine* that Turbo Pascal has a price that is unbelievably low considering its quality.<sup>10</sup> These aggressive pricing and direct marketing strategies further resulted in its widespread use. Newer editions of Turbo Pascal have also been included in packages with other libraries, solidifying its presence in the market. It has not only demonstrated the potential of the Pascal language itself, but also shows a successful product model. Similarly, Lotus 123 set the precedent for spreadsheet softwares in the decades after its demise.

## FUTURE

Developments in both business and scientific fields emerge frequently, with some ideas becoming popular only for short periods of time, and others remaining in the

mainstream for decades. The timing of a product's release and its architecture's flexibility both seem to be important factors in determining how long it lasts in the market. Turbo Pascal was well timed to set up a bridge between Pascal and its growing user base. Although the product lost market share over time, this loss wasn't necessarily due to poor functionality, but rather because of the decrease in popularity of the Pascal language. Lotus 123, when first released, similarly seemed pressed to become the de facto spreadsheet norm. By taking full advantage of the PC's improved memory and display, Lotus 123 could outperform other applications in its field. Furthermore, Lotus Development, the company that owns Lotus 123, sought to eliminate their competition while bringing their own

Figure 2: The Turbo Pascal 7.0 interface running on MS-DOS, an antiquated operating system. It displays some basic information about the software and its copyrights.



“Notably, many popular softwares do not fight alone in their marketing wars.”

product to fame. They purchased VisiCalc (Fig. 3), the first ever computer spreadsheet program and their biggest potential competitor, replacing it with Lotus 123. Unfortunately, since its underlying structure didn't allow it to match enhanced hardwares that other newer applications could fully utilize, the demise of Lotus 123 was inevitable.

Notably, many popular softwares do not fight alone in their marketing wars. They are often accompanied by other applications or features. For example, Microsoft Office overtook the markets of document processing, spreadsheets, and presentations by packaging the three applications together. Here, besides the powerful features of Excel alone, Word and Powerpoint sweetened the deal for consumers trying to choose between different products. As technology advances, developers will be able to embed more complicated features into a single application.

Yet broadly, the contemporary “tech” moment isn't too different from historical ones. With Google offering a new set of collaborative “office” features organized under Google Drive, it's competing with both Microsoft Office and Apple iWork for market share. Google Drive's architecture is fashioned to offer real-time collaboration, a feature harder to achieve with Microsoft Office. For programming languages, Python is the new star. In fact, our very own Berkeley campus has chosen Python as the first language to teach in its introductory Computer Science course, CS 61A. Other applications, such as those based on cloud databases and blockchains, are also

interesting innovations—each of which will likely create new eras in software markets. As we all continue to move from one software to the next, it will be worthwhile to keep our attention on these trends and witness the rise and fall of new technologies before our very eyes.

**Acknowledgements:** I would like to acknowledge George Kopas, IT Executive & UC Berkeley Founder, Coach, and Mentor, for his detailed feedback and review of my article.

## REFERENCES

1. Gandal, N. (1995). Competing compatibility standards and network externalities in the PC software market. *The Review of Economics and Statistics*, 77(4), 599-608. <https://doi.org/10.2307/2109809>
2. Sundqvist, C., & Enkvist, K. (1987). The use of LOTUS 1-2-3 in statistics. *Computers In Biology and Medicine*, 17(6), 395-399. [https://doi.org/10.1016/0010-4825\(87\)90057-6](https://doi.org/10.1016/0010-4825(87)90057-6)
3. Tejada, S. B., & Sigsby Jr., J. E. (1988). Identification of chromatographic peaks using Lotus 1-2-3\*. *Journal of Chromatographic Science*, 26(10), 494-500. <https://doi.org/10.1093/chromsci/26.10.494>
4. Davies, G. L., & Burns, A. (1990). The teaching language Pascal-FC. *The Computer Journal*, 33(2), 147-154. <https://doi.org/10.1093/comjnl/33.2.147>
5. Lee, J. C. (2001). Using Microsoft Excel and decision trees to demonstrate the binomial option pricing model. In Lee, F. C. (Ed.), *Advances in Investment Analysis and Portfolio Management* (1st ed., Vol. 8, pp. 303-328). JAI Press. <https://www.elsevier.com/books/advances-in-investment-analysis-and-portfolio-management/lee/978-0->

6. Coulter, David. (2015, August 4). Lotus 123 Will Install On A Windows 10 (64bit) Computer. *Microsoft Community Forums*. Retrieved April 7, 2020, from [https://answers.microsoft.com/en-us/windows/forum/windows\\_10-other\\_settings/lotus-123-will-install-on-a-windows-10-64bit/1ce639c0-0e7e-4f34-b6d0-bd5768ae1023](https://answers.microsoft.com/en-us/windows/forum/windows_10-other_settings/lotus-123-will-install-on-a-windows-10-64bit/1ce639c0-0e7e-4f34-b6d0-bd5768ae1023)
7. TIOBE Software BV. (2020, April). *TIOBE Index for April 2020*. Retrieved April 7, 2020, from <https://www.tiobe.com/tiobe-index/>
8. Feuer, A. R., & Gehani, N. H. (1982). Comparison of the programming languages C and PASCAL. *ACM Computing Surveys*, 14(1), 73-92. <https://doi.org/10.1145/356869.356872>
9. Turbo Pascal. (2020). *Turbo Pascal Compiler Internals*. Retrieved April 7, 2020, from <http://turbopascal.org/>
10. Webster, B. F. (1985, August). Greetings and Agitations. *Byte*, 10(8), 355-364.
11. Brysbaert, M., Bovens, N., d'Ydewalle, G., & van Calster, J. (1989). Turbo Pascal timing routines for the IBM microcomputer family. *Behavior Research Methods, Instruments, & Computers*, 21(1), 73-83. <https://doi.org/10.3758/BF03203873>

## IMAGE REFERENCES

1. *Banner*: Pexels. (2016, November 23). *Coding Programming CSS* [jpg image]. Pixabay. <https://pixabay.com/photos/coding-programming-css-1853305/>
2. *Figure 1*: Darklanlan. (2013, March 10). *Lotus 1-2-3 on PC-98 DOS chart* [jpg image]. Wikimedia Commons. [https://en.wikipedia.org/wiki/File:Lotus\\_1-2-3\\_on\\_PC-98\\_DOS\\_chart.jpg](https://en.wikipedia.org/wiki/File:Lotus_1-2-3_on_PC-98_DOS_chart.jpg)
3. *Figure 2*: Lưu Nguyễn Thiện Hậu. (2018, December 18). *Turbo Pascal 7.0 Screen* [png image]. Wikimedia Commons. [https://commons.wikimedia.org/wiki/File:Turbo\\_Pascal\\_7.0\\_Screen.png](https://commons.wikimedia.org/wiki/File:Turbo_Pascal_7.0_Screen.png)
4. *Figure 3*: Gortu. (2005, September 1). *Visicalc* [png image]. Wikimedia Commons. <https://commons.wikimedia.org/wiki/File:Visicalc.png>

A	B	C	D
ITEM	NO.	UNIT	COST
MUCK RAKE	4	12.95	556.85
BUZZ CUT	15	6.75	101.25
TOE TONER	250	49.95	12487.50
EYE SNUFF	2	4.95	9.90
SUBTOTAL			13155.50
9.75% TAX			1282.66
TOTAL			14438.16

**Figure 3:** A VisiCalc spreadsheet running on Apple II. It displays a simple price calculation of four goods plus tax.