Epilogue by Brian Randell

The successful completion of the first practical stored-program electronic computers marked “the end of the beginning”, so to speak, of what has been described as “the computer era”. This was not apparent at the time. Nor indeed was the massive extent to which computer technology and the applications of computers were to develop – a process which continues to this day, and still shows no signs of slowing down, let alone coming to an end. However, we can now see that by about 1950 the most fundamental digital computer concepts, in particular those embodied in the term “stored program computer”, had appeared – though of course many further highly significant advances were yet to be made. In these next pages an attempt is made to provide a brief summary of and commentary on these advances.

During the 1950s digital computers progressed from being largely one-of-a-kind developments, used mainly for scientific and engineering calculations by the institutions that had designed or built them. They became commercial products, manufactured in at least modest quantities by a growing number of computer manufacturers. The first commercially manufactured computer was the Ferranti Mark 1, delivered to Manchester University in February 1951. In the United States Univac took an initial market lead, but eventually lost this to IBM who quickly became the dominant computer supplier in many countries.

Computers, though still very large and expensive, began to be put to a variety of commercial, industrial, and military and other government uses by many different organizations. In the scientific arena, their use rapidly overtook that of analogue computers, and made possible far more extensive computations than had ever been feasible even with large batteries of desk calculators, such as the first numerical weather predictions. In the commercial arena they became commonly employed as additions to, or replacements for, large punched card tabulating installations used for major data processing tasks.

At first, virtually all computer development was concentrated in the United States and, to a lesser extent, the UK. But during the 1950s other, mainly European, countries which had not already done so became involved in computer development and application. The total number of computer installations grew to approximately 6000, and the first on-line systems were developed, of which perhaps the most notable was the SAGE air defense system. Ferrite core memories took over from such technologies as electrostatic storage tubes and delay lines, while vacuum tubes were replaced by transistors, so introducing what became known as the second computer “generation”. The term “artificial intelligence” was coined, and the first high level programming languages were introduced, including Fortran and COBOL, but at the end of the 1950s assembly language programming was still predominant. Batch-processing operating systems had come into use, though were principally aimed at maximizing the productivity of large and expensive computers rather than aiding the efforts of individual users.

The next decade saw computer centers set up in many medium- and large-scale scientific, government and business establishments. Additionally, in the mid-1960s, with the advent of the first commercial minicomputer, Digital Equipment Corporation’s PDP-8, computers started to appear in laboratories and similar environments. The first university computing science departments were created, usually as offspring of mathematics or electronic engineering departments, and set about trying to justify their choice of title.

Computing and communications technologies started to come together. For example, the first on-line airline seat reservations system, namely the SABRE system built by IBM for American Airlines, and AT&T’s first electronic switching system were introduced. The ARPANET project, to develop a resource-sharing network, was launched by the US Defense Department’s Advanced Research Projects Agency (ARPA). Integrated circuits, albeit only at the SSI (Small Scale Integration) level, began to become available early in the 1960s, spurred in part by the needs of the US aerospace industry. They largely replaced the use of discrete transistors, and were challenging the dominance of ferrite core memories by the end of the decade. These circuits enabled the development of computers that were vastly more powerful and reliable than their predecessors, and that came to be called the third computer generation.

In those days, shifting to a new computer normally implied having to abandon or rewrite existing applications programs because of hardware incompatibilities. This situation led IBM in 1964 to introduce a range of compatible machines of varying power and capacity, the System/360 series, to replace their previous distinct types of mutually incompatible scientific and commercial computers. This strategy of having a range of compatible machines was soon followed by other manufacturers – in some cases by introducing actual System/360-compatible machines.

The term “software” came into use, though as yet systems software was usually provided “free” with the hardware by the manufacturer, and applications software was normally designed specially for particular clients and particular computers. It was perhaps only when, in 1969, IBM "unbundled" its software by pricing it separately from its hardware that software became a commodity. Memory capacities increased, and the first time-sharing systems were brought into use, starting with MIT's CTSS in 1963. They were largely motivated by a wish to improve programmers' and users' ability to interact with their computers, though batch-processing systems remained the more common.

Increasingly ambitious applications and systems software projects were being undertaken, and organizations found themselves becoming much more dependent on large and complex computer systems than had previously been the case. Although there were some major success stories, one result was a growing concern about software cost and software project schedule over-runs, and about failures, some quite spectacular, to achieve performance and reliability goals. The term "software crisis" was used by some to describe the situation, and "software engineering" to describe the hoped-for solution.

The early 1970s saw the introduction of the first microprocessors, starting with the Intel 4004, which came out of a project to develop a set of chips for an electronic calculator. The limited facilities of the early microprocessors initially caused them to be treated more as complex electronic components than as real programmable computers, and to be regarded with more interest by electronic engineers than computer scientists. Their use as the basis of a complete computer occurred first in the hobby market, but technology improvements soon led to the development of personal computers, such as the Apple II introduced in 1977, which were eminently practical.

Computer networking developed rapidly. Wide area packet-switched networks were introduced by various organizations in a number of countries. At XEROX PARC the Ethernet local area network was developed, in connection with what turned out to be a very influential program of research on distributed systems, personal workstations and human-computer interaction. Ironically, it was to be other companies, including the personal computer manufacturers, who exploited the results of this research most successfully.

The growing availability of on-line storage using large capacity magnetic disks, and the consequent trend for organizations to use their computing centers as central data repositories, led to greatly increased usage of data base systems. These systems, the earliest of which dated back to the mid-1960s, provided means for organizing and protecting such data, and facilitating its use by many different application programs.

The term "supercomputer" came into general use for an increasingly distinct, though evolving, category of computer. Such computers were significantly faster on large mathematical computations than the most powerful standard mainframe computers. One of the most successful such machines was the Cray-1, first announced in 1976, and capable of over a hundred million arithmetic operations per second.

The 1980s saw the personal computer market grow explosively. This was made possible by continuing technological developments, but also was fueled partly by IBM entering the market in 1981 with their PC, and by the rapidly growing strength of the Japanese and other Far-Eastern manufacturers. Somewhat higher performance was provided by personal workstations, which were usually networked together and running the UNIX operating system, though the distinction between personal computers and personal workstations seemed likely to disappear. Near the other end of the market, the decade saw the move towards the use of various forms of parallel processing in order to gain increased performance over and above that provided by technology improvements. Some of these were fairly conventional, others demanded quite novel programming techniques. However the major development was the vastly increased amount of packaged software produced, almost entirely for the more popular types of personal computer, for very sophisticated applications as well as a vast range of computer games.

This development led to the introduction of a myriad of specialist application packages, intended for use by all sorts of organizations and individuals, many of whom regarded their computers not as general purpose computers but as specialist devices – used for example solely for document preparation or standard financial calculations. Indeed many computers were being used quite unknowingly, being embedded into all sorts of devices and machines, such as central heating systems, dishwashers, automobiles and cameras. An interesting analogy can therefore be drawn to the electric motor – originally very large and expensive, used to power complete factories, it has been developed to the point where typical households have no idea how many electric motors they possess. Similarly, they now can no longer accurately count their digital computers.

Recognition of the economic and strategic importance of computing and communications led various countries to undertake major "information technology" initiatives. Japan in particular set up several major projects, including one on so-called "fifth-generation" computers. The term "fifth generation" was intended to dramatize the novelty not of the technology to be employed, as had been the case up to the third generation of computers, but of the architectures to be developed. (There had been no general consensus as to the characteristics of a “fourth generation” of computers, though the term was later applied to certain types of programming language.) The Japenese “fifth-generation” were not intended for conventional computations, but rather for performing logical inferencing and for supporting advanced artificial intelligence applications. However, the various other major collaborative research and development programs that were subsequently launched in the USA and Europe, partly in response to the fifth-generation project, were mostly much more general in their aims and scope.

A totally new major industry has come into existence over the last forty years, one that in many cases the world has become critically dependent on. This industry has grown extremely rapidly, yet steadily, and shows every sign of continuing to do so. Its impact on society, particularly in the developed world, has been immense. This impact is sometimes direct, but often indirect. However, it has rarely been closely in line with the many dramatic predictions, whether positive (relating, for example, to personal domestic robots) or negative (e. g. concerning the advent of a computerized Big Brother society) that have frequently been made.

At least some of these predictions seem to have arisen from a misunderstanding of the nature, and hence likely future path, of the various sorts of development that fuel the growth of the computer industry. The principal factor in this growth has been progress, often funded by the military, in first electronic and then microelectronic (and perhaps in the future opto-electronic) technologies, and well as in various storage technologies. This progress can be measured by ever greater component densities, and hence speed and capacity increases, and cost decreases. It has been a result of, and hence has justified continuing expenditure on, a variety of scientific investigations and engineering developments. The resulting improvements have tended to reinforce each other, and have led to continued exponential growth in all sorts of measures such as main memory size, processor speed, storage access rate. These exponential growth rates have continued for many years, with consequential dramatic effects. (For example, semiconductor memory prices are declining by a factor of ten every five years.) Moreover, there is considerable unanimity among experts that these growth rates will go on for some years to come, and that any fundamental technological limits are a long way from being reached.

The direct consequences on the computer industry of this growth have ranged from enabling ever more powerful computers to be provided, for basically the same cost, to roughly constant levels of computing power to be produced ever more cheaply. An increasingly wide market sector has therefore been opened up within which many subspecies of computer – and indeed subindustries – have been created. Both trends (increased power, and decreased cost) have spurred work on applying computers to an ever wider range of types of application.

The speed with which hardware developments are occurring is often, but misleadingly, contrasted with the more modest but nevertheless very significant rate of improvement that is occurring in software productivity. However it is hardware *technology* which is developing so dramatically. The problems of hardware logic design, and the rate of productivity improvement being achieved, are essentially similar to those in software. This is not surprising because, especially with the advent of Very Large Scale Integration (VLSI), the similarities between software design and hardware logic design are much greater than their differences.

Both types of design center on the problems of mastering logical complexity. Perhaps the most interesting difference, at least conceptually, is that to a first approximation hardware designers, particularly processor designers, are continually designing essentially the same thing, each time from a different starting point as technology improves. In contrast, many software designers start from roughly the same point each time, and design something different as new applications are invented. Nevertheless, both hardware and software design methods are improving, and indeed converging, as more adequate theoretical foundations and effective higher-level notations are created, and new construction methods and tools are developed.

The other main factors in the growth of the computing industry are in fact consequential on the cost reductions that hardware technology improvements have permitted. These have made it practical to apply computers ever more widely, and to involve ever more people in the design of applications and systems software. As a result, particularly in the microcomputer field, there has been enough production of similar systems for the normal processes of competition to work fairly effectively. Thus much of the best microcomputer software is of much higher quality, at least with regard to usability, than most mainframe software. More important, however, with so many more people engaged in the creation of software, there is rapid development of many new and innovative applications.

In contrast, customers' very practical desires for continuity and compatibility are tending to inhibit progress, and are motivating increased interest in standardization, whether de facto (e. g. the Postscript notation for formatted text) or de jure, such as OSI (Open Systems Interconnection) networking protocols. Novel application programs that cannot work on standard hardware or exchange data with relevant existing popular systems are much less likely to be successful than those that can. New hardware architectures typically also have to provide a degree of backward compatibility. For example they might have the ability to support conventional programming languages and a standard operating system, or to connect to networks or devices over standard interfaces. Such characteristics, though aiding the initial exploitation of the new architectures, can at least partly compromise their potential advantages.

Taking all these various factors into account in predicting how the world of computers will change is not easy. In fact it is as hard to predict what the next forty years of computing will bring as it must have been in 1950 to foresee the developments of the past forty years. It is one thing to estimate how processing speeds and costs will change, and perhaps how our ability to design and implement comparatively well-understood applications will improve. It is quite another to predict what new, and perhaps revolutionary, application programs will be thought up (for example, the next decade's equivalent of the spreadsheet program). Equally difficult is the prediction of when and how various existing limits to our knowledge of how to solve various very challenging design problems will be breached, and various long-term goals, in artificial intelligence, for example, achieved. Failure to understand these difficulties has led to some dramatic, and dubious, predictions whose fulfillment will require innovative breakthroughs rather than foreseeable improvements in technology.

Predicting the impact of computer developments on society is even harder. The indirect effects of most radical inventions are more significant than their direct effects. The world of computers will surely continue to be technically highly innovative for years to come. The problem is ensuring that the consequences of all this innovation will be adequately beneficial to mankind, and to mankind as a whole rather than just to a technological elite.