SCIENCE-TECHNOLOGY-PRODUCT: A DYNAMIC TRIAD

Mihaela MALIȚA[1], **Gheorghe M. ȘTEFAN**[2]

gmstefan@comcast.net

ABSTRACT

Science evolves under the sign of Logos. Technology is under the reign of Pathos. The production of an artifact should follow the Ethos. The deviations from a well-tempered evolution of the science-technology-product triad are being instantiated in the history of the world we live in. The non-correlations that appear in the interaction of these three components make the show of our world. We will present two case studies concerning concepts, technologies and products which suffered a historically tensioned evolution. They are related to the domains of scientific computing and artificial intelligence.

KEYWORDS: Computer science, artificial intelligence, mathematics, history of science, philosophy of science.

1. Introduction: Science-Technology-Product Triad

We are considering our world in a totally different way that other living forms. As human beings we are acting in three domains very well defined: we promote the **scientific knowledge**, the development of **technology** and the production of **goods**. Their interaction gives the specific flavor of humanity. Man *unveils* the mysteries of the existence, *invents* new ways of modifying the living space and promotes new ways for *building* its shelter.

The dynamic of the evolutions in science, technology and production of useful goods has unforeseeable twists. The human realm has been developed like a scene where these three actors are playing their parts. There are three independent processes, but with spectacular intersections and convergences. We can ask ourselves: what rules govern their game? A general answer is impossible to find. The following two case studies that we will present will just unveil the spectacular interrelations between science, technology and production.

These three vectors, that concur in defining the evolution of our world (they are not the only ones), correspond to distinct mental components. Human mind can be described as being under the domination, more or less equilibrated, of *ethos*, *pathos* and *logos*.

Science is dominated by logos, but it would be unfair not to recognize the importance of *pathos* (the *will* that underlines action and the *intuition* that guides us or the *imagination* that helps us make choices) or *ethos* (by the way it is guided by the intentionality of our enquiring mind).

Technological inventiveness is ruled by *pathos*, but *logos* and *ethos* are there to guarantee its internal coherence and integrity. Frequently inventions appear in a chaotic manner (and only later other inventions compensate the initial errors[3]) and gaps or they did not consider moral issues (see the case of the atomic bomb in WWII).

Here is the question: what out of what we can design has to be produced and distributed? Only ***ethos*** can answer this question. It is up to ethos to explain the mechanism of the meaning of what is good and what is bad. Ethos come to place when *logos* cannot offer a solution (the deaths cannot be counted) or passions are too enflamed.

We will see from the two case studies how the history of a case can take some unexpected paths where production can decide where science was not „vigilant enough" and how parallel

[1] Associate Professor, Saint Anselm College, Department of Computer Science, Goffstown, New Hampshire, USA.
[2] Professor, eng. Polytechnic University of Bucharest, corresponding member of Romanian Academy.
[3] The parallel computing industrial product designed in the 60's is an example not very well understood by most of us.

histories can connect after long independent evolutions, how technology in conjunction with production can make justice to some old forgotten concepts.

2. Infinitesimal calculus: lights and shadows

The birth of a new conceptual framework can sometimes push in the shadow other concepts that after a while can revive when the technological context changes. Science and technology have this spectacular dance in which they stimulate and, also inhibit one another. Such an example is offered by the computing the classical functions *sin x, log n, arctg y,* a^b ..., using numerical methods or differential calculus. The invention of the differential calculus has pushed away extremely efficient numerical methods that are now unburied when the modern computers needed more precision that was offered by the differential calculus approach.

2.1. Before Leibniz and Newton

Trigonometric and exponential functions have always been very useful, even if they have been used for more or less esoteric purposes. Many algorithms where replaced by more efficient ones. As an example, let's take the square root algorithm known also in Pythagoras' time, called the *stadium algorithm* (see Figure 1). In Antiquity, the stadiums were open on one side for letting the carriages to enter for horse racing. This shape is used to describe the algorithm of finding the square root of a number, which is a sort of division using an evolving divider, starting with the smallest even integer and continuing with even numbers as long as it is possible. The following relation stands:

$$2 \times (1 + 2 + \cdots + (n-1)) + n = n^2$$

We can call pseudo-division this division scheme. The algorithm is extremely efficient (precise and fast).

Similar techniques were developed for more complex operations than the previous ones. Using the operations of *pseudo-division* and *pseudo-multiplication* are described for the first time in 1624 by Henry Briggs (1561-1630) in *Arithmetica Logarithmica*.

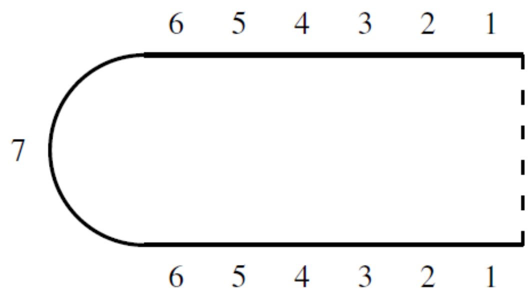


Figure 1 The stadium algorithm for computing the square root

The numerical methods described by Brigs used a very ingenious way to modify the representation of numbers in non-standard number representations, but useful for performing in an efficient way the operations. The algorithm uses a pseudo-division followed by a pseudo-multiplication for computing faster and more precise the logarithm. We start from the possibility that any addition of a binary number can be converted into a multiplication, that is:

$$\sum_{i=0}^{n} x_i \, 2^{-i} \quad \rightarrow \quad \prod_{i=0}^{n-1} (1 + 2^{-1})^{p_i}$$

In these conditions if the binary number $X > 1$ is represented as:

$$X = \prod_{i=0}^{n-1} (1 + 2^{-1})^{p_i}$$

its logarithm can be computed using the formula, after finding the binary coefficients p_i

$$\log(X) = \sum_{i=0}^{n-1} p_i \; log(1 + 2^{-i})$$

In order to compute its coefficients, a pseudo-division will be used (see Figure 2). Once the binary coefficients found, a pseudo-multiplication is applied for computing the logarithm, as shown by the previous formula. The consequence of the two pseudo operations is presented in the following sequence of operations (the first column describes the detailed action of the pseudo-division and the second column describes the effect of the pseudo-multiplication):

$$\begin{array}{ll}
X = (1 + p_0 \times 1/1) & log(X) = p_0 \times log(1 + 1/1) \\
\quad \times (1 + p_1 \times 1/2) & \quad + p_1 \times log(1 + 1/2) \\
\quad \times (1 + p_2 \times 1/4) & \quad + p_2 \times log(1 + 1/4) \\
\quad \times (1 + p_3 \times 1/8) & \quad + p_3 \times log(1 + 1/8) \\
\quad \times (1 + p_4 \times 1/16) & \quad + p_4 \times log(1 + 1/16) \\
\quad \times (1 + p_5 \times 1/32) & \quad + p_5 \times log(1 + 1/32) \\
\quad \times (1 + p_6 \times 1/64) & \quad + p_6 \times log(1 + 1/64) \\
\quad \times (1 + p_7 \times 1/128) & \quad + p_7 \times log(1 + 1/128) \\
\quad \times \; \ldots & \quad + \; \ldots
\end{array}$$

The conversion of X into a multiplication format supposes only shifting operations, addition and comparisons (see Figure 2), and the final computation needs only a selective addition of some constants like $log(1 + 1/2^i)$, dependent on the coefficients p_i.

```
procedure convert(X)
        v = 1;
        for (i = 0; i ≤ n; i = i + 1)
                if (v + v × 2^i ≤ X)
                        then    v = v + v × 2^i ;
                                p_i = 1;
                        else    p_i = 0;
end convert
```

Figure 2

The simplicity of the algorithms made possible for the middle age monks to compute the logarithm tables and to find the tables of the trigonometric functions so much needed in the finding astrological charts.

2.2. After the invention of the Infinitesimal calculus

The infinitesimal calculus permits that for an infinitely differentiable function, *f*, we can write its Taylor series

$$f(a) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n$$

The expansion in series, this invention by Leibniz and Newton (listed here in alphabetical order) as part of the *"technology of mathematics"*, made possible an easy way to compute the values of multiple differentiable functions. Let's take the example:

$$ln(1 + x) = x - x^2/2 + x^3/3 - x^4/4 + \ldots$$
$$e^x = 1 + x + x^2/2! + x^3/3! + \ldots$$

The algorithmic simplicity was a huge advantage in such a way that initially the precision was neglected, to be reconsidered only in our computation age. For obtaining high precisions it is necessary to consider a very big number of terms and the computation must be made using long numerical representations. There are cases when we need hundreds of terms and a double number of digits of the result.

2.3. The numerical calculators

Scientific computation requires the use of trigonometric, exponential functions and other as well. The expansion in series method uses algorithms that consume a lot of memory space and far more execution time than those for basic arithmetic operations.

Members of the academic community where surprised when in 1968 Hewlett-Packard launched on the market the desktop calculator HP9100A. The motive was among other that the logarithm operation with 10 digits precision was realized using only a representation of 12 digits and its execution time was equal with twice the time for a division.

Long time forgotten algorithms preserved in Briggs' book have been unburied[4]. A commercial product realized in a totally new context succeeded to revive old numerical technologies which were lost in oblivion due to the advancement of science in the spectacular and efficient technological outburst of the XVII-XIX centuries.

3. Artificial intelligence: eternal aspiration and promise

Computers are smart machines used to take decisions and communicate. They are based on logical calculus not in arithmetic. We use the term *intelligence* in a common sense because there is no general accepted definition. In this sense, Artificial Intelligence (AI) can be considered the generic function of a computing system. We argue this by the functional challenges raised by the computer science in the last decade. For example: the knowledge organized in Data Centers or the autonomous car.

When and how human intelligence starts to evolve in AI under the name of computer (name that we consider not the best choice)? We shall present three, more or less independent, historical evolving paths. There is an evolution of **technologies**, a **conceptual** evolution and a new evolution of the **products**.

3.1. The evolution of concepts

Using the term computer could be confusing if we try to really understand its meaning. The conceptual evolution which the computer is based on is not related to arithmetic. Logic and the philosophy of natural language are the foundations of the computer, object, nowadays, used mostly in decision making and communication. Due to these two main purposes, we can track its existence starting with Antiquity.

3.1.1. Paradoxes in antiquity

[4] J. E. Volder, "The Cordic trigonometric computing technique," *IRE Transactions on Electronic Computers*, vol. 8, no. 3, pp. 330–334, 1959.
J. E. Meggitt, "Pseudo-Division and Pseudo Multiplication Processes", *IBM JOURNAL* APRIL 1962, pp 210-226

Epimenides of Knossos (7th or 6th century BC), is a half legendary Cretan figure, which in a moment of anger said: *All Cretans are liars!* It is the first occurrence of a statement in natural language that its truth value cannot be determined. There will follow two millennia and a half where the human mind would try to understand the relation with itself. Clarifying this confrontation with itself, in 1931 through Kurt Gödel's (1906-1978) mathematical explanations the fundamental of computation is born.

In the centuries that follow in the Greek world there are numerous attempts to understand the behaviors of the human mind when related to its self. Zenon of Elea (~490-~430) shows the danger of accepting the concept of infinite when explaining real facts. Zenon's warnings did not stop the conceptual evolution of infinity, but this did not mean that it did not have a very twisted road (see previous case).

We had to accommodate, from the oldest times, with the fact that using some limit concepts as: true, false, zero, infinite, is extremely useful, but in the same time somehow dangerous if we are not aware of their limited applicability.

3.1.2. *Middle age aspirations*

Ramon Llull the Catalan (1232–1315), in the space of the miraculously coexisting of the three Abrahamian religions, starts from the idea that thinking is the result of the art of mechanically combining formal concepts. The device he conceives is the first trying to achieve a mental behavior in a technical object.

Llull prefigures an intelligence that operates in a device imagined by our brain and produced using the technological state of the moment. Llull's device was destined to support the conceptual steps of the human mind. It is intelligent tools which support the whole complexity of human behavior.

3.1.3. The Complexity of the Modernity

Athanasius Kircher[5] (1602-1680) continues Ramon Llull's investigations with a set of more diverse symbols (Llull is using only words in Latin). In 1669 he publishes *Ars Magna Sciendi, Sive Combinatoria* which contains a "new and universal" version of Llull's method accompanied by a numerical evaluation of the covered complexity.

The first paper published by Gottfried Wilhelm von Leibniz (1646-1716) in 1666 was *Dissertatio De Arte Combinatoria*, also inspired by Lllull's Ars Magna.

Leibniz is the first to bring up to our attention the relation between the complexity and the utility of the formal approach. In *Discourse on Metaphysics* (1686), Leibniz speaks about the divinity that simultaneously maximizes the diversity and minimizes the conceptual complexity. If a law is too complex then Leibniz thought it's not a law but a description *tale quale*. The difference between a successful theory and a useless one is given by the evaluation of its complexity regarding the reality that it refers.[6]

Leibniz, since its published paper from 1666, compares the logic reasoning with a mechanism that can reduce the rational approach to a computing procedure that can be finalized in the construction of a device that can handle this superior computation.

[5] http://history-computer.com/Dreamers/Kircher.html

[6] Gregory Chaitin: *Epistemology as Information Theory: From Leibniz to Ω*. At: https://www.cs.auckland.ac.nz/~chaitin/ecap.html

Also, Leibniz speaks for the first time about the reduction of complexity by changing the representation of the basic data types used in computation. He recommends using the binary representation in his works *De progressione Dyadica*, (1679) and *Explication de l'Arithmetique Binaire*, 1703.

The more advanced formalization of knowledge questions more and more the coherence of the theory. The coherence by itself but not the coherence with the reality has to be provided in the first place. As an example, the apparition of the non-Euclidean geometry forced mathematicians to assure the internal non-contradiction of their theories. Consistence with reality is now a second concern. Thus, at the end of the 20th century, concern for the internal coherence permits the re-apparition of the paradoxes. The decision of the truth problem becomes essential.

3.1.4. The Decision Problem in the present

The decisive step in the conceptual trajectory of AI is done by David Hilbert (1862-1943) when he stated the decision problem (in a Paris conference in 1900 there appear incipient formulations, but the decisive form is presented under „Hilbert's program" in the 1920s). What is the mechanism in a formal theory to decide if a sentence is true? The same problem in another way was stated by Leibniz when he wanted to figure out a mechanism for establishing the truth.

In 1931, the ***logician*** Kurt Gödel proves the impossibility of Hilbert's program. This is the most important negative result in the history of mathematics. Epimenides' dilemma what provoked us centuries ago is reopened and explained. A fundamental limit was needed to surpass a border which looked intangible: the efficient exteriorization of human thoughts.

Only 5 years after, in 1936, ***four mathematicians*** – Alonzo Church (1903-1995), Emil Post (1897-1954), Stephen Kleene (1909-1994) and Alan Turing (1912-1954) – integrate Gödel's result in different mathematical theories, but theoretically equivalent, in which they state the theoretical limits of computation. Alan Turing is detached himself due to the immediate consequences of the way he tackled this problem (there is still controversy regarding the size of the contribution of his impact[7], due to insufficient study of the history of computation). The theoretical model of Universal Turing Machine (UTM) is at the foundation of the *mono-core* computation which dominated the first stage of computer science.

The UTM model supposes three distinct entities: the physical structure – the data – the program. The program describes in an explicit way how the physical structure should act on the data.

The researchers in the theory of computation have realized from the beginning that here are problems for which the description of the machine's behavior is very difficult or impossible. Thus, in 1943 Warren McCulloch (1898-1969) and Walter Pitts (1923-1969) propose the neural network (NN) model. The "programming" of these NN is done by a training process. The result is expressed as bi-dimensional arrays with a non-intelligible content, because the function is obtained through the self-organizing process of training.

3.1.5. Artificial Intelligence Programs

Artificial Intelligence (AI) started as a distinct branch of computer science after the famous conference in 1956 at Dartmouth College, NH. Participants were among others, John McCarthy (1927-2011), Marvin Minsky (1927-2016), Allen Newell (1927-1992) and Herbert Simon (1916-

[7] Thomas Haigh Actually: "Turing Did Not Invent the Computer" în *Communications of the ACM*, Vol. 57 No. 1, 2014, pp 36-41. At: http://cacm.acm.org/magazines/2014/1/170862-actually-turing-did-not-invent-the-computer/abstract

2001). In this domain, we can work with rules explicitly stated, only if a reasonable complex representation of the reality can be conceived. This was the case of the AI applications in Lisp and Prolog. If this explicit approach cannot be applied, then we can try an implicit one: the training of a neural network. If in the training process of the neural networks the problem does not converge toward a solution, then the problem has maximum complexity, it is called **trans-computable** and we are confronted with a fundamental limit issue. The last theoretical results in AI are centered on applications as *Machine Learning* based on using *Deep Neural Network*, Convolution Neural Network or *Binarized Neural Network* (an optimized form which substitutes arithmetic operations with logical operations).

3.2. The Evolution of Technologies

Externalizing[8] in artifacts his multiple behaviors, the human being succeeded to avoid a specialization that would have limited his evolution. Thus, besides building his ability to travel fast on earth and water, lifting heavy objects, and many other things, man succeeds at one point to start externalizing his intelligence.

3.2.1. Mythic Aspirations

In the mythic space occur for the first time the aspirations who led the world forward. Externalizing human's complex behaviors, smartly coordinated, we discover for the first time in the Western civilizations in the myth of Talos/Talon[9], the bronze humanoid build by Hefaistos for defending the Crete island. This creature had all the features of a modern robot.

Pandora was made out of clay and came to life through Zeus' will.

Pygmalion received as a gift from Aphrodite life for his statue with whom he fell in love.

Daedalus used mercury to make his statues speak[10].

The historian Polybios (~201-~120) talks about Apega, the steel virgin, the invention of the tyrant Nabis of Sparta (207-192) which was used to collect taxes from bad payers.

3.2.2. Ancient Implementations

Heron of Alexandria (~10-70 AD) describes in his book ***Automata*** (preserved in an Arab version) a collection of automatons (*thaumata*) used for generating miracles in temples. He also emphasizes a source of energy for the different devices that „move by themselves”: the ***steam***. Heron's Eoliple is the first device that converts caloric energy into motion.

3.2.3. Middle Age Attempts

The Benedictine monk, Gerbert d'Aurillac (~945/950 – 1003), became Pope Silivestre II in (999-1003), is the inventor of the pendulum used for the first horologe in 994/996. In his time in Rheims he made a *horologium arte mechanica compositum* and another one for a monastery in

[8] André Leroi-Gourhan din: *Le Geste et la Parole*, vol.1. : *Technique et langage*, vol. 2. :*Mémoire et les Rythmes* Paris, Albin Michel, 1964-1965.
[9] http://www.wondersandmarvels.com/2012/03/the-worlds-first-robot-talos.html
[10] http://www.theoi.com/Ther/Automotones.html

Magdeburg for a better knowledge of star movement[11]. He is also credited for introducing the decimal notation, instead of the roman traditional notation. The mechanical clock can be regarded as the first computing device. It had the function of counting a sequential process (the pendulum's *tic-tac*) or the function of the division of the frequency of a periodical process. The decimal notation taken from the Arab culture was essential in building the first physical mechanical computational device: the clock. It would have been impossible to do the same thing using roman notation.

We cannot fail to emphasize the synchronicity between the event of the adoption of the decimal notation in the conceptual space and the technical innovation of the pendulum. This conjunction allowed the occurrence of a new technical object: the horologe. Not by coincidence Gerbert has been intellectually trained in Catalonia, a region where Christianity, Islam and Judaism coexisted peacefully.

3.2.4. Modernity's Achievements

The astronomical clock on Prague's City Hall, build in XV century, can be considered a specialized computing machine. In three centuries, the mechanical technology realized with the help of the gears has been developed, fact that facilitated spectacular devices for public's delight[12]. With this achievement, started in 1410 and finalized in1490, we are now only one step ahead the invention of the ”computer clock”. We are talking about the machine described by Wilhelm Schickard (1592-1635) in two letters in 1623 and 1624 addressed to Johannes Kepler (1571-1630), which he called „arithmeticum organum” and considered it useful for computing astronomical tables. (Is it by chance that in 1624 Henry Briggs published his paper mentioned before?)

Blaise Pascal (1623-1662), motivated by another domain of application – tax collection that was his father responsibility – builds in 1642 (he was 19 years old) a counting machine. He used the same principles used for building the mechanical clock.

After more than 30 years, Gottfried Wilhelm von Leibniz makes a step forward designing a machine, *Instrumentum Arithmeticum,* that executes also multiplications[13]. He presented his invention as a scientific result at the Royal Society in London (1673 and 676) and at the Science Academy in Paris (1675).

In 1801, Joseph Marie Jacquard (1752-1834) programmed binary an automatic weaving machine. The presence or absence of a hole in card made the difference between 0 and 1. Thus, the automatic behavior could be programmed with a binary sequence.

Charles Babbage (1791-1871), starting in 1837 and Ada Lovelace (1815-1852), between 1842-1843, together for the first time come up with the concept of programmed computer. The Analytical Engine would be the starting point of the computer Mark I developed at Harvard University.

George Boole (1815-1864) offers an algebraic form to the Aristotelian logic in *Mathematical Analysis of Logic* (1847) and in *An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities* (1854). This re-formalization, in a new more suitable shape from an operational point of view, facilitates the foundation of the computational devices, the same way the Arab decimal representation, replacing the roman numerical representation, made possible the apparition of the technology for the mechanical clocks.

[11] Stephen C. McCluskey: *Astronomies and Cultures in Early Medieval Europe,* Cambridge University Press, 2000.

[12] Modernity starts, after some scholars, with the fall of Constantinople (1453), the printing press (~1455), or the discovery of America (1492).

[13] http://history-computer.com/MechanicalCalculators/Pioneers/Lebniz.html

Herman Hollerith (1860-1929) invents in 1896 a mechanical calculator, based on punched cards, used in 1890 for United Stated census.

Claude Shannon (1916-2001), in 1937, finished his master thesis, considered the most famous master thesis in history, in which he proves the possibility of transforming Boolean logic in physical structures (circuits with electromagnetic relays): "A Symbolic Analysis of Relay and Switching Circuits", was published in *Transactions of the American Institute of Electrical Engineers (1938).*

3.2.5. Contemporary technologies

John Vincent Atanasoff (1903-1995) and Clifford Berry (1918-1963) in 1939 at the Iowa State College build the first presentation of a prototype of a calculator, realized with electronical technologies, for solving linear equations. *Atanasoff Berry Computer, ABC,* was a project interrupted by the beginning of WWII.

War on the other side, was an accelerator factor for all projects involved in military activities. Two examples are especially significant because they pushed the first timid steps toward the conceptualization of technological artifacts. The concept of an abstract ***model of a computing machine*** is due to the following technological engineering objects.

Harvard University, together with IBM, starting from Howard Aiken's (1900-1973) project, produces in 1944 the mechanical calculation machine Automatic Sequence Controlled Calculator known under the name of Harvard Mark I. The computer Mark I *"brought Babbage's principles of the Analytical Engine almost to full realization, while adding important new features."*[14]

The computer called ENIAC (Electronic Numerical Integrator And Computer) designed and build between 1943 and 1946 by the engineers John Mauchly (1907-1980) and J. Presper Eckert (1919-1995) at the University of Pennsylvania was considered until 1971 the first computer in history (in 1971 a judge decided this is not true). It is the most important result for the theoretical and technological consequences that it generated.

Mark I and ENIAC are the foundations of what we call today the ***Harvard architecture*** and the ***von Neumann architecture***[15]. The word *architecture* is somehow used in an un proper way (it was introduced in the 60's at IBM). More correctly would be to say *abstract model.*

The abstract model (the architecture) became an „interface" between *hardware and software* and the fast development of the technology in what we call today the technology of the computation. Hardware evolved from electromagnetic relays, to electronic tubes, transistors, integrated circuits with an exponential growth in time. The development of the informational structures, the software, happened on several levels. One level was the programming languages (Fortran (1954-55), Lisp (1957-58), Algol (1958), Cobol (1959), Snobol (1962), Pascal (1970), C (1972), C++ (1983), Java (1995), ...) the other level was of the operating systems and not to be neglected an amazing evolution of the user interface with the computer (punch cards, printer etc. to the *touch screen*).

Computers with transistors (1952-53) represent a technological leap which increased the reliability and the speed of the computers and decreased the size and energy needed for computation.

[14] "IBM's ASCC introduction 2". Retrieved 14 December 2013.

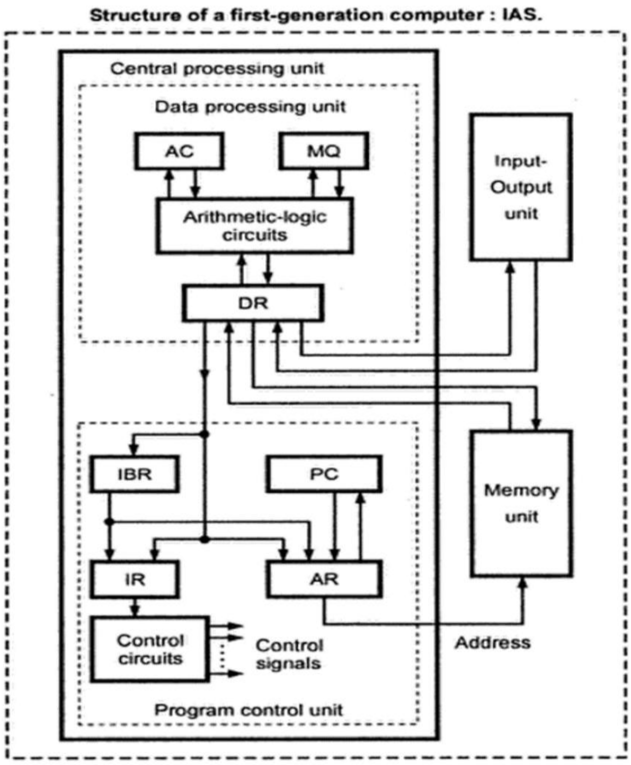
[15] Based on the report *First Draft of a Report on the EDVAC* edited by John von Neumann (1903-1957) in 1945. Based on the discussions of von Neuman at the Moore School of Electrical Engineering of Pennsylvania University with the authors of the ENIAC project.

The computers build with integrated circuits (1963) continue the same evolution.
The microprocessor (1971) opens the new era of embedded computation.

3.3. The evolution of the products

The conceptual and technological evolution determines, at one point, a mature process that provides the market with commercial products. Technologies leave the labs of the research institutes or universities and start their unpredictable trajectories on the market. The computing systems are no exception. Initially the market was formed by governmental organizations and universities and started to extend toward the consumer market.

3.3.1. The Computer



Register	Abbreviation
Accumulator	AC
Multiplier Quotient	MQ
Data Register	DR
Program Counter	PC
Instruction Buffer Register	IBR
Address Register	AR
Instruction Register	IR

Figure 3.

The IBM 701 series launched in 1953 starts the computer market. This prototype was the computer IAS (Institute of Advanced Studies) conceived by von Neumann and built between 1946-1951[16]. The structure of this machine can be considered as canonical (see Figure 3[17]) for what it was and what still is the *mono-core* computer.

[16] John von Neumann's major contribution consisted in his persuasion in the decision o build the computer. His colleagues, Albert Einstein, Kurt Gödel, Robert Oppenheimer, did not consider that a computer can help them in the research. His most important ally was a member of the Board of Trustee, a millionaire with powerful financial arguments toward this decision.
[17] D. A.Godse; A. P.Godse (2010), *Computer Organization*. Technical Publications. pp. 3–9.

An important step in computer science that accelerated the development of the computer was the concept of „architecture”. The architecture, as an interface between hardware and software, allows the independent development of the two domains of computing. The term was used for the first time in an internal IBM report in 1959 and is made public in 1962, where it is defined as:

> *Computer architecture, like other architecture, is the art of determining the needs of the user of a structure and then designing to meet those needs as effectively as possible within economic and technological constraints.* [18]

The computer architecture in its original meaning of the word is what the user really needs to know about the computer. It is a functional definition which allows a diversity of actual solutions. The most important effect was the stable context in which software was developed. Any new hardware realization had to preserve the functional definition of the architecture for allowing the compatibility with programs already written. You could add new features to the architecture but you had to preserve the old ones.

The new product, the computer, could be promoted now on a bigger market by instantiating it with more performance due to the protection of the functional architectural scheme. The corporatist criterion can be easier applied to extend the markets and increase the profit. The first forms of positive feedback appear destabilizing the basic classical mechanisms of market regulation through stabilizing negative feedbacks[19].

3.3.2. Parallel computers

The next and new significant moment in the products’ evolution on the informational technical market is when the corporations take control. The parallel computer stats as a product that does not have in its background a natural sequential evolution similar with the one that lead to the invention of the mono core computer: (1) the theoretical computational model (Turing & Co.), (2) the abstract machine model (Harvard and von Neumann), (3) the production in series, (4) the architectural approach which allowed an accelerated dynamic of products in series[20].

Parallel computer appeared from an ad-hoc model proposed in 1962 by Burroughs Corporation: the computer D825 with 4 processors and 15 modules of memory interconnected by a *crossbar switch*. The conceptual confusion persists until today. In 2010, David Patterson (b. 1947) wrote:

> *“... the semiconductor industry threw the equivalent of a Hail Mary pass when it switched from making microprocessors run faster to putting more of them on a chip – doing so without any clear notion of how such devices would in general be programmed”*[21]

3.3.3. The Personal Computer & www

The acceleration produced by the architectural approach and the evolution of the technology (the transition from tubes to transistors followed by integrated circuits) allowed the most spectacular leap in the short history of computers: the ***personal*** **computer (**1980) and the invention and the implementation of the ***internet*** (1980-1991).

[18] Werner Buchholz (ed.): Planning a Computer System: Project Stretch, McGraw-Hill Book Company, inc., 1962.
[19] See the works of William Brian Arthur (b. 1946).
[20] Gheorghe M. Stefan, Mihaela Malita: "*Can* One-Chip Parallel Computing *Be Liberated From* Ad Hoc Solutions? *A* Computation Model Based Approach *and Its* Implementation", *18th International Conference on Ciruits, Systems, Communications and Computers (CSCC 2014)*, Santorini Island, Greece, July 17-21, 2014, 582-597.
[21] D. Patterson: ``The trouble with multi-core", *Spectrum, IEEE*, 47(7):28–32, 2010.

The computer faces two simultaneously tendencies: it is personal but it is integrated in a unique system. It becomes a tool for communication and interconnections in the first place. The computational function becomes an instrument for required market demands or imposed by the market. The corporatist control of the evolution becomes absolute. *Marketing* – the path from technology to product – becomes the main mechanism in which the computer science evolves. The theoreticians come only with a second look in which they only explain and do not impose new trends.

3.3.4. Functional Electronics

Electronics is the technological base for computing. A boomerang effect has happened when, at the same time with the development of the microprocessor technology (beginning with 1970), the function of the electronic layers starts to be determined by the pair ***circuit & information***. Electronics becomes functional electronics due to the fact that simple structures with a greater and greater number of components can obtain a complex functionality only through the information *embedded in the form of* programs (explicit or not explicitly, under the form of some matrices of weights obtained through the training of some neural networks).

There is a very interesting effect through which a technology (electronics) is fundamentally reconsidered from the point of view of its influence in another technology (computation). This phenomenon occurs with the apparition and development of the microprocessor at the beginning of the 70s[22]. The concept of functional electronics, introduced by Mihai Drăgănescu in 1978, is the synthesis of physical circuit structures with the informational structure of the programs.

We have two modalities two incorporate emergent intelligence in the exponential growing physical structures offered by the technologies of functional electronics: *programs*, in parallel computing systems or weights matrices in deep neural networks.

3.3.5. Smart phone/the tablet

The functional explosion allowed by the functional electronics approach, generates the second spectacular leap, after the PC: the *smart phone* a construct that starts from a mobile phone (product of functional electronics) endowed with an operating system and a computer and a smart terminal.

The transition was produced at the end of the last century of the second millennium but the explosion took place in the first decades of the next millennium.

The last product is the *tablet as a main device anyone can connect to the internet with numerous functionalities*. The tablet becomes a personal assistant, more and more vital in our modern world. We shall call it **iPAL** (Intelligent Personal Assistant Linker).

3.3.6. Data centers

The explosion that we call today *data center* was produced with the occasion of the *dot-com bubble* between 1997 and 2000. Then there were developed *Internet Data Centers*. Later, *Cloud Data Centers* appeared. More recently, the distinction is no more significant, we talk about *Data Centers*. These are huge computer farms with essential components as power supply systems, with cooling, security etc. From a computational point of view, we talk about a very big number of

[22] In 1971 Intel produces the first family of circuits with which they can build a 4 bits processor, and in 1972 the first 8-bit processor is made, Intel 8008.

interconnected computing nodes through *Ethernet*, each node being endowed with conventional computational resources (PU) and local acceleration units (GPU or MIC).

Once connected, all these Data Centers will make up the unique global computer. It looks like Thomas John Watson Sr. (1874-1956), CEO al IBM, in the 1940s, exaggerated says that it will be not necessary more than five computers for the needs of the whole earth (*"I think there is a world market for maybe five computers"*). We actually tend to one. A single intelligent machine, distributed on the surface of the earth, reminds us of the intelligent planet from the science fiction movie Solaris by Stanislaw Lem/Andrei Tarkovsky.

4. Conclusions

The three-dimensional space where man is building its universe very tight is dominated by actions that arise too close around its coordinate axis. Our actions rarely go far away from those axes, where science, technology and production can concur in a convergent way toward the human prosperity.

We identified three histories of the computer that still evolve independently too, each near its own specific axis and this represents an obstacle in the harmonious development of man's world. Independent processes, generated distortional evolutions. We can expect harmonious evolutions only when the points of functioning of our histories will be in the center of the three-dimensional space (logos, pathos, and ethos) assuring in this way an integrative unifying approach.

References

1. Buchholz, Werner (ed.): Planning a Computer System: Project Stretch, McGraw-Hill Book Company, inc., 1962.
2. Chaitin, Gregory. *Epistemology as Information Theory: From Leibniz to Ω*. At: https://www.cs.auckland.ac.nz/~chaitin/ecap.html
3. Godse, D. A.; A. P.Godse (2010), *Computer Organization*. Technical Publications. pp. 3–9.
4. Haigh, Thomas. "Actually, Turing Did Not Invent the Computer" în *Communications of the ACM*, Vol. 57 No. 1, 2014, pp 36-41. At: http://cacm.acm.org/magazines/2014/1/170862-actually-turing-did-not-invent-the-computer/abstract
5. http://history-computer.com/Dreamers/Kircher.html
6. http://www.wondersandmarvels.com/2012/03/the-worlds-first-robot-talos.html
7. http://www.theoi.com/Ther/Automotones.html
8. http://history-computer.com/MechanicalCalculators/Pioneers/Lebniz.html
9. Leroi-Gourhan, André. *Le Geste et la Parole*, vol. 1. (Technique et langage), vol. 2. (Mémoire et Rythmes), Paris, Albin Michel, 1964-1965.
10. McCluskey, Stephen C. *Astronomies and Cultures in Early Medieval Europe*, Cambridge University Press, 2000.
11. Meggitt, J. E. "Pseudo-Division and Pseudo Multiplication Processes", *IBM JOURNAL* April 1962, pp. 210-226.
12. Patterson, D. "The trouble with multi-core", *Spectrum, IEEE*, 47(7):28–32, 2010.
13. Gheorghe M. Stefan, Gheorghe M., Mihaela Malita. "Can One-Chip Parallel Computing Be Liberated From Ad Hoc Solutions? A Computation Model Based Approach and Its Implementation", *18th International Conference on Ciruits, Systems, Communications and Computers (CSCC 2014)*, Santorini Island, Greece, July 17-21, 2014, pp. 582-597.

14. Volder, J. E. “The Cordic trigonometric computing technique,” *IRE Transactions on Electronic Computers*, vol. 8, no. 3, pp. 330–334, 1959.