Defining Usability:

Ajanta Sen

(1) The definitional and semantic context(s) of Usability:

(1.1) In its dictionary *meaning:*
Usability comes from the word 'usable' (an adjective) which means: *serviceable*, at one's disposal, available, *fit for use*, functional, practical, *in running order*, ready for use, utilizable, valid, working…….etc.,

(1.2) In its universal *connotation:*
Usability, at its simplest commonplace best, and arising from the above semantics, would represent the factors that determine the *index/coefficient of 'use'* of an object or service.
In other words, the sum of the factors of 'how often', 'when not', 'why', 'how', etc.,

The component definition from 1.1 that should interest us the most is *'fitness for use'.* Why? Because, across the history of the discipline that we call industrial design, the big philosophy that designers and design thinkers have consistently fought for recognition is the idea of 'fitness for use' and restated by design as *'form follows function'*.
The stalwarts: **pre-Industrial Design** - John Ruskin, William Morris, Adolph Loos
Industrial Design (Modern movement) – Walter Gropius, Le Corbusier, Raymond Leowey, Henry Dreyfus, Misha Black, Massimo and Lella Vignelli, Bill Moggridge, others
Industrial Design (post-Modern) – Victor Papanek, Tomas Maldonado, Ettore Sottsas, Jasper Morrison, April Greiman, Shigeo Fukuda, Peter Shire, others

(Bringing us to the definition of usability in industrial design terms)

(1.3) By *definition* in *industrial design* terms:
Usability has come to mean - an understanding of the design trajectory that reflects the raising or lowering of the *barriers of use* of an object/service itself.
Which means factors that inhibit/facilitate the use of an object/service giving us the notion of '*usability barriers'*.
This is akin to the notion of *'entry barriers'* as a concept used by entrepreneurs to assess funds flow into business, or by marketing to assess the prospects of a product in the market.

Mathematically speaking: *barriers of use/entry barriers* are computed in inverse relation to their coefficient value/index of use: the lower the barrier, higher the use or acceptance of object/service, and hence higher the Index/Coefficient of Usability.
In other words, can we map usability barriers as entry barriers to ascertain the extent of validity of a product with the user?

(1.4) By definition in terms of *computing-based products* or products built around a computing environment, and often described under a category, presently called HCI:
Usability relates to products that reside visibly as screens, and ask the following assessment questions - how does the screen work, and how may we optimize its manueverabilites for its users? The users are often secondary consumers – those who service primary ones. E.g., B2C category of users who work with screens as ERP's etc.,

The above definition of a screen-driven computing environment/interface, implies that:

(a) all information is encased within a screen;
(b) the usability of information on the screen needs testing; and most importantly,
(c) that there is already something built for it to be tested, as opposed to starting green field to say it could well be a product that has no screen.

This, by implication, constrains the very scope of usability within the product cycle, which is not just about a product already built (and as mentioned - in this case, the screen).
Instead, a product cycle is made up several stages, usually starting with innovative thinking, leading to concept generation, into iterations and prototyping, and finally into testing. All this is done with the user and its environment weaving in and out of this process.
A screen-based definition of a product would restrict usability in present HCI terms largely to the stage of testing.

(1.5) What then is the larger picture of usability?
To understand the larger picture, we'll need to look at *usability as a system* and not as a stand-alone factor, and one which positions all the relevant, interconnected factors that go to interact with each other within the framework of a coherent single system.
It is this holistic view that will give us a picture of the Usability Ecology

Let's cursorily examine the Usability Ecology in the background of the universal connotations of usability, to understand where HCI is presently positioned in the common psyche of the IT industry

(not computer research). And how its actual evolution across tells a different story.

(2) The Usability Ecology:

In the larger time cycle of the evolution of products, where our primates had started shaping tools from stone to make the first stone axe (on display in South Africa today), usability of objects was meant to contain a meaning/connotation not very different from the generic/broad *connotation of usability as attributed to 'human-computer' by its early researchers in the forties and fifties.*
Both of these definitions - early man-artifact and the man-computer relations of the fifties, sixties and seventies - remain in sharp contrast to the present narrow definition of usability as a screen-based HCI, and the term HCI itself reduced to a special interest group.
This meeting of minds about usability between ancient man and the modern day computer scientist suggests that usability as a construct *must not been seen as being unique to the use of computers* alone.
Which is to say, and as we all know, that there were no computers at the disposal of the early man and yet, usability as a notion had already come into force.
If anything, it was computing that borrowed from already existing man-artifact notions from the past, and not vice versa.

(2.1) In evolution terms, usability derives its principles *not from human interactions with machine*. It derives its premises from *human interactions with information,* and this taking the form of **'man-artifact interface'** because artifacts are a form of information - something, as we all know, has existed for centuries since the birth of the artifact.

This is a view that comes not just from anthropologists studying early man, or design historians concerned with the functionality of a product, but from computer researchers at MIT or at Xerox PARC in the fifties, sixties and the seventies, looking to understand the computing environment by drawing lessons from other disciplines and historical information about the development of man. So that, they are able to make the computing environment go beyond research to become a relevant product of use – something like the artifacts (fire, hoe, smelted iron weapons, etc.,) as thrown up by civilization.

One of Xerox PARC's key researchers, **Douglas Engelbart**, himself inspired by the research of **Norbert Weiner** (cybernetics), **Benjamin Lee Whorf** (mathematician-language structures) and **Alfred Korzybski** (physicist-language studies), called the computer interface to be an artifact that could only work as "a process of information exchange that is *not necessarily unique to humans using computers"* (or even machines).

It is important to take a minute to understand why, in addition to the great knowledge base of these scientists, these three intellectuals had played a seminal role in humanizing the computer environment, and why they were so keen to look at the way human behaviour held lessons for the computing environment.
All three individuals were humanists who, in spite of being scientists and engineers, had given Engelbart to believe that "more engineering was not the dominant need of the world at that time." And all of them considered the "human side" of the technology grossly neglected.

They were convinced that the computing environment had to find familiar and generic human ways to interact with the human. And base it on an understanding of *how humans interact with artifacts (and not how humans would interact with computers).*
This had resulted in Douglas Engelbart's notion of human-computer relationship, and it was his aim to forward what he called, the process of the *Augmentation of Human Intellect.*

So, for Engelbart it was important to see
(2.1.1) man retain his position of intellectual dominance through a constant search for means to *augment human intellect* through the machine, and

(2.1.2) the need to gain insights into this process of information exchange between man and artifact that have resulted in usability situations, and the abject need to go back to this ultimate usability ecology – viz., one *where humans interacted with artifacts.*

Man interacted with artifacts because these were an extension of his physiology.
Raising the question, "how did early man light a fire each time he wanted a fire?"
What was that interface?
Was it a Charlie Chaplin/Gregory Peck/Western Cowboy style of igniting a flame by rubbing a stick/piece of wood against one's boot-soles each time they needed a fire?

Acquiring a fire was a complex act and had to be *simplified through an interface – friction via stone.*

Likewise,
-how did man smelt hot iron to make implements without getting his hands scalded?

-how did man manipulate the rough, uneven earth, to sow the first seeds?
it was *the hoe (giving him his first revolution, the Agrarian Revolution)*

–how did man interface with dense, turbulent wind or water to turn these into energies? through *turbines and blades* that became the windmill;

-how did man use captured and stored-up steam and direct it as a form of energy every time he needed it?
by the principles of the same turbine (that gave him *his second revolution, the Industrial Revolution)*

Engelbart considered the genesis of these activities as due largely to *an information exchange process.* And within which was embedded key information on potential interfaces.
For him it was important to understand:
what are *the insights hidden away in these information exchange processes* that we might today leverage to assist/augment information exchange between human and computer (giving man *his third revolution, the Information Revolution*)?

(2.2) The next key insight that Engelbart took away from this model of man-artifact interfaces was that:
"all information exchanges took place within a larger framework" (where different forces interacted with one another to form a system) and no force could defy was outside the logical framework.

Based on this, Engelbart envisaged the shape of a master framework in the way information would operate with respect to computers, and named it as *"H-LAM/T System",* where:
H=Human, who uses
L=Language, Artifact Methodology, and
/T= in which he is Trained.

This was in 1962 at Xerox PARC.

The question remains:
Why did Engelbart and his team at the Augmentation Centre pin their faith on a systems approach?
In this, they were inspired by *Norbert Weiner*, father of *Cybernetics*, and the most recent advances made in the fifties.
Cybernetics is "the science of communication and control" *(Bootstrapping, Thiery bardini, p.11)* and involves the "technology of the processes of automated command and control, within automated systems for the acquisition and processing of data." (George Ifrah, The Universal History of Computing).
Cybernetics and its systems-approach take their sustenance from a multi disciplinary approach, with inputs from psychology, mathematics, engineering and the social sciences.
Cybernetics also played an important role in the cultural context of the USA in the mid-of the century (20^{th} c) by aiming to solve

the complexities of the prevailing societal problems arising from unbridled growth and use of resources or years prior to that. *(Club of Rome, meadows-Meadows study 'Limits to Growth' etc.,)*

Engelbart's faith in cybernetics and the systems approach is validated by his own use and interpretation of the concept of a system. To establish this, he took the basics of cybernetics from Weiner and went a step further to propose something called *'Bootstrapping',* meant to represent *"an iterative and co-adaptive learning experience"* (Bardini, p.24).
The term 'bootstrap' itself comes from a noun, which in 1913 had meant *"unaided effort"* and a "loop strap sewed at the side or the rear top of a boot to help in pulling it on."
As an adjective in 1926, it had come to refer as being *"designed to function independently of outside direction:* capable of using one internal function or process to control another."
Finally, in 1951, "bootstrap" became a transitive verb meaning *"to promote or develop an initiative and effort with little or no assistance."*
(In computer parlance today "to boot" means: "to load and initialize the operating system on a machine.")
It is easy to see the humanizing relationship that the computer science research community was attempting to build into the computer – driven by the belief that if you made the computing process capable enough of self-organising itself, this could make it easier for the human to enter the process without help from outsiders/programmers. And free it of mechanized controls.

In more technical terms, Engelbart's 'bootstrapping' was inspired by Norbert Weiner's cybernetics notion of *positive feedback into the research process*.
According to Weiner (in his famous publication 'Cybernetics and Society'): "'*Feedback' is the property of being able to adjust future conduct by past performance."*

(2.3) But introducing a systemic framework was not going to be enough. For the Augmentation of Human Intellect to work as a system would essentially require *"matching processes" as a shared mechanism for humans and machines to co-negotiate with the "outer world".*
This would then mark the start of humanizing HCI by looking at it in terms of a broader vision and human reality, and its first attempts at finding the necessary "matching processes"

What would be the best "matching process?"
At that point in time, the computing environment was dominated by a "matching process" called *Artificial Intelligence (AI),* which was predicated on the idea of 'automating' human thought.
AI is the "science and technology of the artificial realization of various functions of human intelligence." (Ifrah, Universal History of Computing).

AI was being researched at MIT's AI Lab as well as at the Lincoln Lab, by J.C.R. Licklider who was looking to work towards the concept of "intelligence amplification" and 'Man-Computer Symbiosis' by building *interactive computing* – a kind of *computing that involved two-way exchanges across the interface between user and machine*. And totally in contrast to the one-way "number-crunching" of the main-frames of the fifties and the sixties.

The AI notions developed by Licklider and aided by Weiner were interesting because they chose *the biological concept of symbiosis* instead of a mechanistic vision of the computer's environment. The inspiration for humanizing the computer was to come from the animal world, through the notion of symbiosis, which according to him was *"cooperative living together in intimate association, of two dissimilar organisms."* Norbert Weiner's cybernetics had already described animals and machines with the same vocabulary.

However, looked at retrospectively, AI hardly merited the term "matching process" meant to facilitate the human-computer dialogue.
Why?
Because, AI - by then firmlyestablished at MIT's AI Lab in the 50's by its founder Marvyn Minski – had begun to consider the human brain to be entirely replicable by the computing environment through the AI process.
Which means, given a set of artificial languages that were built on programming, the machine would interact with the human and accomplish most human mental capabilities.
And *even go a step further to accomplish what "no human brain has ever thought nor processed as data."*
This is where Minsky's AI parted ways with the process of augmentation of human intellect, which unlike AI, had continued to place the human brain at the centre of all activities, not at its periphery.

AI banked on what came to be known as *'indirect manipulation'* because the human could interact with computing only through the help of programmers. There were no available means for the user of the computer to directly change the interfaces, which of course were his only pathway to the computing environment.

Famously, Minski had termed the human brain as being nothing more than *"computers made of meat"* and that it was potentially possible to automate human thought through complex algorithmic behaviour of the machine.

From this astoundingly machine-centric HCI view, Engelbart then made the earth-shattering shift of mindset in the computer research community to what was to become *the first steps towards humanizing the computer.*
Basically, Engelbart wanted to move the idea of HCI and the processes for computers from the domain of a few programmers with deep dependency on artificial languages whose job was to

provide the computer with input, instruct it about what to do with the input, and obtain usable output, creating indirect manipulators, such as robots.
And instead establish a direct path of communication between man and computer. This he called the search for a *'natural language'* that would allow for *'direct manipulation'* as a way for man to interface directly with computers without assistance from programmers.

Engelbart's mentors here were *Benjamin Lee Whorf and Alfred Korzybski*. Between them, they invoked the notions of culture and language, hermeneutics, and of course bootstrapping – all as aids for developing 'natural languages' as the best "matching process" between human and computer:

Essentially, three principles were invoked to find the best "matching process" between human and computer:
(2.3.1) At the highest level of abstraction it was *the principle of bootstrapping,* and surely reflecting the ability of a great mind such as Engelbart's, that was able to see a bootstrapping process inherent in man's earliest form of communication, viz., language. Just as in bootstrapping, human language had the natural propensity to ……

(2.3.2) The principle of the play of *culture in language (Whorf and Korzybski)*

(2.3.3) It also meant invoking another set of principles called *'hermeneutics'*, assisted particularly by the computer research community at MIT.
Hermeneutics allows for…….

(2.4) Engelbart was further assisted in his search for more humanizing ways for the computer to connect with man by *Alan Kay*, who was able to characterize the computer's narrow mechanistic ability to carry out orders and do no more beyond that, as making the computer no more than *strictly a tool in the hands of the human*.

This forced the AI community to further admit that the time had come for AI to recognize that its comparison of the computer with the human brain was premature.
That, for the computer to acquire a higher-level status, it would have to move on to a higher plane of interactivity with the human, viz., to acquire the capabilities of a *medium.*

(Incidentally, in the way that the screen requires to be instructed, not much has changed since in the present-day HCI idioms of approaching the computer, although the computer itself has moved on to being far more sophisticated in its ability to be a medium and not just a tool – but HCI continues to treat it rather primitively as a tool).

To summarize,
in simple terms, for Engelbart, HCI had to be viewed as being driven by a process that "had existed for centuries, ever since humans began using artifacts and executing composite processes."

Essentially Engelbart was keen to clarify that:

(i) "the term "*man-machine interface*" *has been used for some years* to represent *the boundary across which energy is exchanged* between the two domains (viz., man and the complex machine representing the principal artifact with which a human being cooperates

(ii) however, the "*man-artifact interface" has existed for centuries*, ever since humans began using artifacts to carry out complex tasks (lighting fire with stone and friction, tilling the rough earth with a hoe, fashioning tools out of hot, smelting iron, storing energy from external forces through turbines, etc).

(iii) that, this *exchange across the "interface" occurs* when an explicit-human process (EH-P) is coupled to an explicit-artifact process (EA-P)

(iv) that, these *coupled processes are designed for* just this exchange process, & provide a functional match between EH-P and EA-P

(v) that, the elements of EH-P and EA-P and its matching/coupling process have *to be seen as a system* (influence of Norbert Weiner's Cybernetics)

(vi) that, for computing, it was natural language/direct manipulation, not artificial language/indirect manipulation that was going to be the key to matching the two processes (EH-P and EA-P)

(vii) *natural language* as defined to *be rich in both its physiological and social dimensions* and *a language available to all*, and fashioning AI around these (influence of Whorf and Kryzobski's contributions in terms of structuring of languages from the principles of sciences – mathematics and physics, respectively) to develop a alter language for the computers, that would dramatically and forever change the human-computer relationship.

(viii) Engelbart called this powerful coupling mechanism of a natural language for computing a *'meta-tool'*, followed by similar advances made by Alan Kay (Utah and Xerox PARC) – bringing in the question: why was language a "meta-tool"? – because language has always been seen as *a tool to augment the basic human condition*

(ix) Engelbart and Artificial Intelligence (AI) researchers can also be credited to see *the human body as the contact point between human and computer intelligence*, the boundary that separates and joins the two forms of intelligence (with implications for latter day sensory-use in interfaces).

(3) Models of the Usability Ecology:

In other words, the three broad models of Usability across civilizational time may be seen as:
A – Man-Artifact Usability

B – Man-Machine Usability
C – Man/Human-Computer Usability

With HCI pushing itself as a narrowed down sub-ecology of usability within the broader Usability Ecology, and confining itself to the testing environment of a product development cycle (of innovation, iteration, prototyping, testing)

(4) Critiquing the HCI sub-ecology:

HCI, seen as a screen-driven computing environment/interface, throws itself open to the following conditions/implications, that:
(i) all information is encased within a screen;
(ii) the usability of information on the screen needs testing; and obviously, and most importantly, that
(iii) there is already something built for it to be tested (and by implication, constraining the scope of usability within the product cycle of innovation, iteration, testing, largely to testing.

It is this downstream testing exercise that it chooses to call 'usability', and which it has plucked away from its broader universal connotations to narrowly mean the testing of a computer-driven product, that distorts the much broader idea of a Human Computer Interaction – based on Engelbart's Human-Artifact Interaction.
And since what the present-day 'HCI' tests is the screen, it necessarily equates all products as being the screen-driven interfaces

Two major lines of critique arising from this narrowing down of usability to a screen-driven interface:

(a) **Donald Norman** (HCI as upstream vs. downstream) - decrying the idea of using the interface as starting point of product development vs. user environment/user artifacts as the real starting point – in the process, obfuscating the need to pay attention to the ubiquitous nature of the user's environment, the products embedded in their environments, the pervasive nature of the influence of design and not a special interest group nature to design, and design features embedded into everyday products (such as in pens, keys, etc.,) – do we notice any more?

(b) **Brenda Laurel** and **Alan Kay** (computer as tool vs. medium) - decrying the idea of the screen as too narrow a construct to represent the complete environment underlying computing.
For them, a screen-based usability definition was valid while the computer was still in its early stages of evolution, when the screen (under a regime of AI) had the capacity to be a mere tool, requiring complex sets of commands to make it work.
However, with the advent of the PC (and natural language), the computer evolved to to acquire cognitive capabilities and move on to becoming a medium.
Which means that we needed to look at the computer with a new eye as a medium that was capable of 'thinking' = anticipating

solutions = exercising its will and conceptualizing solutions = innovating.
Alan Kay equates this ability of the computer to innovate to be a 'meta-language' – a medium that has its own language.

(5) Moving away from an HCI-Usability driven scenario to an Innovation-driven scenario:

The reason this is possible is because technology has since upgraded the computer's environment to become a medium with its own language at its command, although the dominance of the screen interface has dumbed down the computer's ability from being a medium to acting as a mere tool (that required commands and instructions to function).

A tool cannot be expected to envisage or conceptualise solutions. It can only execute solutions already conceived by someone else. It means having to deal with what already exists. Which is the reverse of innovation.

Hence, the only way to introduce innovation into the computing environment is by moving up the value chain of usability and the product development cycle, where product development starts with envisaging a product not yet conceived of and still awaiting to be envisaged and built. And where a product development cycle starts with out-of-the box thinking towards 'anticipating' fresh, innovative conditions to solve a user need, and conceptualizing a solution not otherwise considered to be possible, something that is totally new.

(to be continued…..)