

## Please ... . draw me a Software Engineer

Nicolas GUELFY  
University of Luxembourg  
Luxembourg  
nicolas.guelfy@uni.lu

### Abstract

*The objectives of this short paper are to dream about software engineering science and to think about making this dream alive. In my dream, there are no more software engineers but "senseware" engineers that build physical space and time augmentations that adapt to human direct and indirect needs. Making this dream alive could be started by providing research and development directions that might make this dream a reached target. Anyhow, it is an awaken-dream which is made and it is mainly synthesized from the author's experience in research and development of software engineering. Moving from today's software engineering nightmare in which "What You Get Is What You Get" to a dreamed world in which "What I Get Is What I Need" needs to turn science-fiction into "science-vision".*

## 1 Introduction

This paper is a science-fiction to science-vision paper. Science fiction because it presents a futuristic world for the software engineering science and, science-vision because it tries to provide pieces of a puzzle map that would make this futuristic world reachable.

Why doing software engineering science fiction? I think that it might be the only way to solve the software engineering chaos as I see and live it. A world in which the software engineering community, since its birth at the well known 1968's software crisis NATO conference, targets to define theories methods and tools for engineers to help them building less failing IT systems and master their engineering.

Success or failure after 50 years? "Both!" to be objective, "Total failure" to be provocative. Hopefully for the scientist we could be, the subject here is NOT to be objective.

Don't blame software engineers. We are asking them to develop and maintain a safe and well-ordered house with rooms full of fragile and expensive goods; while

walls, floors, furniture are moving and changing their configuration in an unpredicted way at an always growing speed.

As an example, let's consider the current situation concerning artificial intelligence. More and more systems using deep learning technics are currently built but, how many of those theories, method and tools to master their engineering are currently available, efficient and deployed? A realistic approximation of this number with a precision of  $10^0$  would be ... .0.

This paper is about making this change by drawing the future of software engineering as I dream it and thinking of pieces of elements that could help to find a path from now to this dreamed future.

It is proposed to stop chasing the moon and to start building our SE-moon and make our guests come and live there.

For this, I first describe a fictive dream world, then I try to explain how and why this dream came to me in order to prepare for the last part in which I give ideas and visions that might be useful to make this fictional world real.

Even though partly available, proofs and references have deliberately been omitted to provide only harmonious visions from a dreaming mind of an old software engineering researcher.

## 2 Dreaming a Future

*(Tina is in her autonomous vehicle)*

**Tina:** I need to evaluate the applications for our new professor position in security.

**Hal:** Tina, your applicant's evaluation environment is now available and configured.

**John:** Hi, Tina! It's John, your university secretary bot for this session.

**Tina:** First applicant!

**John:** Here it is the application of mister Alan Turing, it contains a virtual presentation and a written document.

*(Tina is now in front of three visual representations, one of John, another of Hal and a last one for the data concerning Alan Turing's documents).*

*(Tina takes time to hear and look at the deliverables)*

**Hal:** Tina, it seems to what I can perceive that you feel uncomfortable with this profile, may I suggest the use of some external service from a partner university where there are specialists in cryptanalysis?

**Tina:** cost?

**Hal:** the 10 best choices range from 0.5 to 2 sensecoins.

**Tina:** is there any available bot for automatic evaluation of applicants for a professor position?

**John:** Tina, we have recently bought one at U.F.O. [University of Future - Oregon -- Ed.], it's Steve and can only evaluate default human resource criteria and not the ones related to research and education.

**Tina:** thanks John, please make him join the session. Hal, please send a quotation request to BBC (i.e. the Bot Building Company) for engineering a HR recruitment bot for academic research and teaching sector.

**Arnold:** Hi, Tina, it's Arnold, your security officer, may I remind you that applications are confidential and thus you can only select those three ones from the list which signed a confidentiality agreement with U.F.O. Furthermore, you'll have to temporarily close this session in 45 minutes as we will enter in a not enough secure environment for this session. You'll be able to continue in 1h45' from now if conditions do not change.

**Hal:** John, I just received the offer from BBC for an Academics HR recruitment bot, it costs 100 sensecoins. We also have to be available to educate this bot for our needs for a period of 150 to 200 hours by recruitment committee members and HR members. It is also included its integration with our U.F.O bots including John.

**John:** Tina, do you want to buy this bot on your department budget, which has a planned residual

balance of 2000 sensecoins, or do you want me to enter the request at central HR service?

**Tina:** Hal and John, I'll decide later, remind me this next week.

**Hal:** Tina, your ideal lunch time can be made during the session closure for security reasons, here are what I think would be the best lunch for you for biological and emotional reasons. They can be available on board or at this stop station.

*(Tina takes a look at the lunch proposals and at the stop station found by HAL).*

**Tina:** I let you choose for me HAL ...

### **3 The Software Engineering dimensions behind the dreamed future**

The science fiction scenario described in the previous section needs now to be analyzed from a software engineering futuristic viewpoint. In this section, it is proposed to provide and motivate the main drivers that are considered to conduct to this dreamed future. The objective is to clarify this targeted future from a software engineering point of view to determine proposals for SE research and development activities. Figure 1, summarizes the main aspects of this viewpoint which are all directly or indirectly illustrated in our SF scenario.

#### **3.1 From computers and programs to augmentations**

If we observe human and computer interactions according to the space and time dimensions, we could say that the current target could be to reduce the human space occupied by computers and the human time dedicated to the control of a computer using software. If we extrapolate, there might be a point in time at which the computer and software will disappear. The human space and time obstructed by computer and programs will become space and time augmentations offered to humans.

Humans will evolve in augmented environments. An augmented environment instance, that can be called "an augmentation", allows interactions with humans on biological and psychological dimensions. The entire known sensing spectrum will be considered on an adaptive individual basis (vision, hearing, touch, smell, taste).

In an augmentation, interactions will range from direct and explicit to indirect and implicit on a context-specific and adaptive basis.

Where are the computers and the software? In the proposed science fiction scenario, they have been replaced by sensitive interactions with bots in a physical location (i.e. the automotive car). We could imagine that such interactions could be accessible at any point in time and space using sound, displays, glasses, holograms, or humanoids ...

An augmentation will be sensible to context and dynamically synthesize augmentation adaptations based on perceived needs.

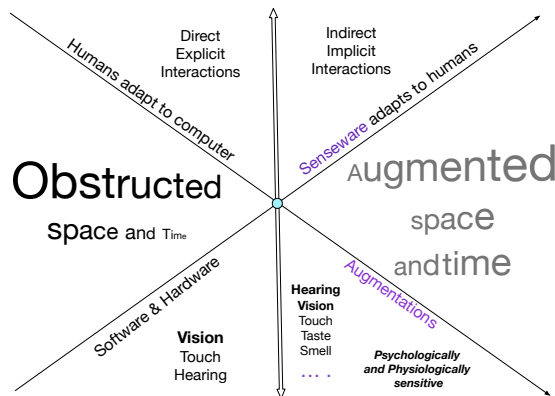


Figure 1.: Dimensions behind the dream

### 3.2 Senseware

Physical exploitation of the hardware layer is foreseen to be made by a sensitive individualized layer: the "senseware". Changes in the physical layer will not be handled by the humans but by the senseware itself.

So, what is a senseware made of? A first approximation based on current approaches could be that a senseware is made of advanced physical or logical machine learning modules. In this case, software engineers would become senseware engineers mastering machine learning theories, methods and tools. As an illustration, our Hal bot could be a machine learning module (engineered by the BBC company that relates pairs of hearable data (i.e. sound representations) and managing at least input and output devices for hearable data interactions (e.g. microphones and speakers). It is deliberately not provided a physical representation of the elements interacting in our scenario to have it as less constrained as possible and illustrate the possible non-differentiability between human and non-human interacting elements.

We already have our "prehistoric" senseware modules. As an example, Apple's Siri or Google Assistant could be seen as an ancestor of our Hal. Figures 2, 3 and 4 show samples of this module in operation, including manual error correction in the third one.

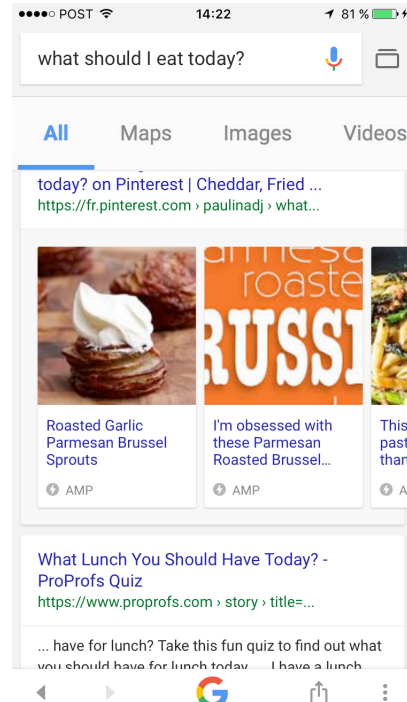


Figure 2.: Current "pre-historic" senseware module

Our dreamed world seems consistent with the one of Google if we believe Jeffrey Dean (leading the Google Brain project) saying that "if Google were created from scratch today, much of it would be learned, not coded. Around 10% of Google's 25,000 developers are proficient in ML; it should be 100%", or Greg Corrado (co-founder of Google's large scale deep neural networks project) speaking about the TPU - Tensor Processing Unit - (a microprocessor chip optimized for training and running machine language models comparable to a Graphics Processing Units) saying "I didn't learn to talk from a linguist, I learned to talk from hearing other people talk".

The future of an advanced TPUs is a senseware. A polymorph layer capable to: interact in a dedicated physical environment, learn from its interactions, handle environment changes. A senseware engineer could then be seen as responsible of engineering (specifying, designing, validating,...) complex advanced machine learning modules. From creation to delivery, this senseware engineer will be responsible of the advanced education of its senseware which would mainly consist in learning activities.

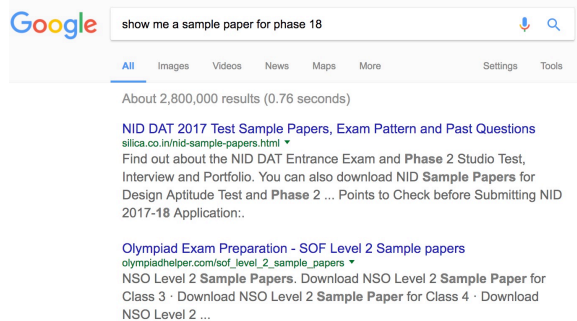


Figure 3.: Current vocal senseware module (a)

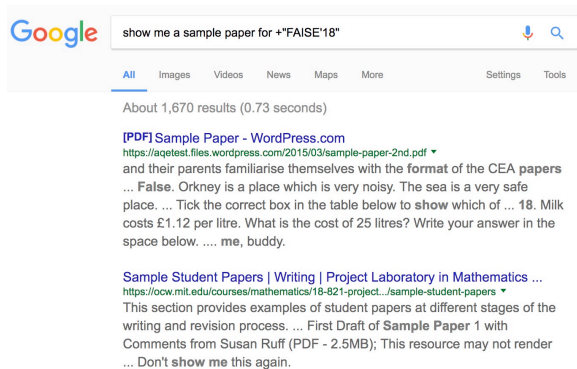


Figure 4.: Current vocal senseware module (b)

### 3.3 Socializing senseware

In this vision about the future of software engineering, we move from a situation in which humans adapt to computers, hardware, and software to a world for which a senseware adapts the environment to the humans' needs.

Perceived nowadays as IT tool, senseware will be perceived as part of the society at all layers: population, culture, material products, social organization, and social institutions.

As such, the engineering of senseware will be constrained by the legal laws applying to the society in which such senseware is supposed to be integrated.

### 3.4 Senseware engineering senseware module

A senseware module capable to create, train and deploy other senseware modules could be named a "senseware engineering senseware module". If totally autonomous w.r.t. interactions with human senseware engineers it could then be the replacement of today's software engineers.

## 4 Drawing the future of software engineering

The first objective of this paper has been reached in describing a fictional world for the future of software engineering.

The second objective is to turn science-fiction into science-vision. As explained in the introduction section, the proposal is to stop acting as usual in the software and hardware development chaos. Instead, it is proposed to fix a target and build a path towards it.

The target proposed is "senseware engineering". Even though senseware is a notion created for this FAISE'18 dream paper, senseware engineering would need to define and apply some principles I believe fundamental for the discipline, and which are partly and abstractly described in this section.

### 4.1 Natural coherence is everywhere

The discretization associated to digitalization made us mapping real world elements to discrete presentations. The history of this mapping made us abstract or forget the natural laws related to those elements. The computer controls the status and evaluation of real world objects based on their abstract discrete representations. As an example, it is possible that the velocity vector angle of a rocket changes from 85° to 345° in a nanosecond or that your weight value, stored in a medical database, changes from 80kg to - 80kg or 800kg.

Software data and behaviors are not implicitly constrained by laws associated to the real-world objects they control and are related to. Standard programming languages do not offer this capability by default. If wished, it must be manually added and used (dynamic constraints validation with compensation using fault tolerance mechanisms).

Programming languages syntax and semantics lack of natural coherence. Let's think about a distance measure based on the geometrical similarity of a program. In Java "x = x + 1" and "x == x + 1" have a high geometrical similarity but a low semantical one.

Machine learning modules are trained to associated objects to similar ones (similarity representing a kind of equality measure between two objects). A code generation machine learning module would thus consider as equivalent the two expressions provided above which are semantically different.

## 4.2 Learning is everything

Learning data is the new gold of the senseware area. It is foreseen that senseware engineers will be responsible of finding, and creating data to feed the advanced machine learning modules.

Deep learning, as a general approach for learning higher-level representations and efficient classification and prediction, has its quality that depends on the learning data and its capability to synthesis good abstract representations. Simple and powerful illustrations can be found in language translation learning modules that are based on sequence representation of natural language sentences.

Learning quality depends on the natural coherence, subtending similarity, that exist in the learning data and its presence in the data space the learning module will encounter.

The future, as seen in this paper, will be made of augmentations made with advanced machine learning modules having continuous learning capabilities. Thus, the capability of humans to "educate" learning modules they interact with is a key success factor.

Senseware engineers will use senseware module to engineer senseware modules. Thus, developing learning and education theories, methods and tools for senseware engineering could be one of the main areas of interest for our community.

## 4.3 Oralisation is the first "senseaction"

It is foreseen that interactions inside augmentations will mainly be based on the human senses. While the weight of each sense in the interaction will strongly vary in time, I envision that they could be ordered by decreasing importance as follow: vision, hearing, touch, smell, taste.

Concerning the human means used to communicate in an augmented environment, called here a "senseaction means", the most important ones in the short term could be oralisation (i.e. speaking) and motion (i.e. moving).

This will apply for senseware engineers using senseware engineering environments (the equivalent of the software engineering environments of today) using oralisation and motion. Thus, to prepare for the future of our software engineer we could invest more resources in fields like vocal programming or vocal interfaces both using advanced learning modules.

## 4.4 Senseware science as a multi-science

The senseware engineer is foreseen, if coherent with this paper's vision, to be at the crossroads of physical, life, human, social, human and formal sciences. Finding, creating learning data and advanced learning modules will need competencies in nearly all those sciences.

The body of knowledge necessary for a senseware engineer will highly differ from the current software engineering body of knowledge.

In order to be prepared for the foreseen senseware engineering we could start an initiative to build a SEEBOK (SEnsware Engineering Body Of Knowledge) learning from our experience in creating and using the current SWEBOK (SoftWare Engineering Body Of Knowledge).

## 5 Experiment and generalize for senseware engineering

Concerning a possible development plan for senseware engineering, a first proposal could be to experiment on senseware engineering.

Based on the visions and directions presented in this paper one of the best candidates could be a Vocal Language Learning Module for Naturally Coherent Interactions with Artificial Intelligence and its application to vocal programming augmented environments.

Based on this first experiment, an outline of a software engineering for advanced learning body of knowledge (why not naming it SEALBOK) could be the basis for further development leading, who knows, to the senseware engineering discipline.

## 6 Conclusion

I guess there is nothing to conclude else than all that dream get out from the maybe insignificant mind of a long time being academics specialized in traditional software engineering theories, methods and tools who opportunely felt a need for a sudden and significant change of its discipline. In short, a software engineering quantum jump!

Might be the starting point of a new drawing, who knows ....