Abstract

This paper is an outcome of the book project based on the book of Italo Calvino, Six Memos for The Next Millennium. The goal of the paper is to examine the ways in which the memos of Italo Calvino can be applied to the enhancement of software engineering in terms of the team collaboration, product building, and the process itself. The book presents five following memos: Lightness, Quickness, Exactitude, Visibility, and Multiplicity. Possible ideas of contribution from each memo to software engineering will be considered throughout the paper.

Introduction

Organizing a team and developing a software engineering product can be done in an incredibly diverse way. The goal of this paper is to understand the significance of the messages as revealed in the book Six Memos for the Next Millennium by Italo Calvino. The book is divided into separate memos and the purpose of the paper is to draw connections between the takeaways from the memos and how they fit into the software engineering process.These include the use of the Lightness memo in building better teams, the application of the Quickness memo in software development, the Exactitude memo in requirements gathering, the application of the Visibility memo to unambiguous communication, and the Multiplicity memo’s contribution towards the process of software development

The Lightness memo and the software teams

Within the first memo, Italo Calvino discusses the Lightness principle and its significance for the coming millennium. The examples presented by Calvino when discussing the memo present multiple meanings to this word.

For instance, consider the author’s example of the Kafka’s story about a Bucket, whose message is

“In fact, the fuller it is, the less it will be able to fly.

The organization of the software engineering team must be examined in light of the Lightness memo at this point.Taking the concept of Lightness to its simplest, most basic meaning, it is related to the core principle of organizing a team - preventing it from being overburdened.

The success of teams depends on the reduction of areas of focus and the reduction of workloads among members. Teams that are overloaded in any sense are less likely to succeed.

The research conducted on this topic (Fussell et al., 1998) suggests that rather than burdening the team with cognitive overload, intrateam communication should be enhanced to help everyone accomplish their goals.

Moreover, it is crucial to provide members of a team with a moderate level of autonomy or freedom in their actions as they develop their team structure.

“he feels the need to allow atoms to make unpredictable deviations from the straight line, thereby ensuring freedom both to atoms and to human beings.”

In breaking from the straight line that Italo Calvino contemplates, each member of the team is given a chance to be heard, which ultimately leads to open communication and better approaches to completing the work.

A complementary method of implementing this organizational structure is to recognize, appreciate, and admire individuality. This idea is illustrated by the story of poet Guido Cavalcanti, who is not popular among the group of other elegant people, because

“his mysterious philosophy is suspected of impiety”. The poet later leaped over one of the great tombs to get himself rid of the company.

The idea is fundamentally counter to the rigid structure of a team with a strict hierarchy that leaves no room for deviation, which can be compared to a Weight term presented by Calvino.

Nevertheless, it is important to also be aware that too much discretionary power can turn destructive. It is basically the kind of Lightness interpretation that should not be desired. This is because each team member’s roles and responsibilities must be clearly defined and stated. By becoming vague, they can lead to the opposite outcome of that desired.

“Lightness for me goes with precision and determination, not with vagueness...”

This leads to the third important application of Lightness memo, which is in terms of team communication, it is recommended that face-to-face conversation be preferred to online one, since non-verbal communication contributes more to meaning than verbal communication. This idea is communicated by Calvino on the example of Don Quixote scene and the effect of its image on the consciousness of readers.

“Some literary inventions are impressed on our memories by their verbal implications rather than by their actual words.”

To conclude, Calvino opposes the Lightness principle to the Weight. In terms of its application to software engineering, it is recommended that when setting up a team, individual roles and responsibilities must be clearly defined, yet a certain level of flexibility should not be omitted as well.

The Quickness memo and the process of building software

The second memo is presented by Italo Calvino, the Quickness memo. According to Calvino,

“the relativity of time” abounds everywhere. In this book, the author views the horse as

“an emblem of speed, even speed of mind”. At this point of time, it is crucial to consider how quickness is related to the process of building a software

Quickness is illustrated in the relation between physical speed and the speed of mind. According to Italo Calvino, there is something particularly pleasing about the physical speed of the horse:

“vivacity, the energy, the strength, the sheer of life”

A similar concept can be applied to the importance of quickness in the software engineering process. Quickness will promote a dynamic working environment, which will in turn motivate the team to work harder.

As the author does not believe that

“quickness is a value in itself"

, these principles can be applied to making quick yet reasonable decisions when building products

Using Galileo as an example, the quickness is actually compared to racing because

“Discoursing is like coursing".

Therefore, quick decisions should not result in losing the race at the expense of long-term reasoning. Quick decisions, however, cannot be irrational at the same time.

In order to have a successful and fast software building process, one must develop and practice

“Speed of thought".

Furthermore, whenever there is a chance to increase productivity without sacrificing quality, it is better to take advantage of it. Because, as Italo Calvino states,

“the more time we save, the more we can afford to lose".

It is therefore important to optimize the process whenever possible, probably by using appropriate tools and software. One more example of this idea is using the open source code where this opportunity is possible.

In accordance with the research (Wohlin et al., 2021), there are several ways to speed up the process of making decisions and building a product without sacrificing quality. Some of these are

“Sharing experiences and data"..

It relates to the exchange of experiences between teams or organizations in accomplishing the same goal. This methodology has been successful in avoiding mistakes and identifying the best practices as quickly as possible.

In addition, the practice of

“Aligning software architecture and decision-making" involves making sure that decisions are made based on the characteristics and needs of the system’s architecture.

All these factors will provide a risk resolving time in the event of unanticipated issues.

“hurry slowly" is arguably the most important lesson one can learn about software building and teamwork.

Even though the process should be optimized to the maximum extent possible, every step of the process needs to be logically reasonable and practical. Otherwise, quickness may be a hindrance.

As a result, being quick not only enhances and speeds up the software building process, but also reinvigorates the team working on it. Because being quick enables the members to observe the progress in real time, which in turn results in greater vitality and motivation.

**The Exactitude memo and requirements collection.**

For the purpose of successfully completing software, assembling accurate and feasible requirements for the software is a crucial first step. Rules and advice for constructing correct requirements can be derived from the exactitude memo of Italo Calvino’s book.

Calvino begins by introducing a Maat feather to represent the Egyptian concept of exactitude. The feather was used to scale the weight of souls and measure the unit of length. It is a symbol that represents precision both in ancient Egypt and today.

Italo Calvino provides three definitions for the term of Exactitude. These definitions are the following:

“A well-defined and well calculated plan for the work in question"

“An evocation of clear, incisive, memorable visual images"

“A language as precise as possible in both in choice of words and expression of the subtleties of thought and imagination".

Italo Calvino’s reflection on Leopardi’s argument can be traced back to the first definition, in which even the profound and unchanging picture is forged by the smallest details.

It can be traced that Leopardi’s argument of vagueness supports the idea of preciseness, in that the detailed descriptions of each image are what make each image undefined. That is why, planning the requirements demands consideration of the details of each image.

As a decent illustration of exactitude, the author brings the ideas of geometry, symmetry and numbers. Combining the first and second definitions of exactitude with the examples provided, it is possible to derive a conclusion: the exact requirements are measurable requirements. Numbers and exactness should be the forefront of any requirement for the software. This will allow to seek a desired “clear"

larger image.

The third definition of the exactitude has a special and significant role when working on the requirements. This is about

“A the effort made by words".

In Calvino’s view, natural language is more communicative than formal language, yet it also contains more noise in itself. This sort of idea can be applied to requirements development in the following way: precise requirements are better to be constructed by using formal language. Using this method, some secondary noise can be eliminated, making the focus more on the most essential values. The value of language and words should never be dismissed when it comes to the communication of ideas, and especially designing requirements for software development.

The conclusion that is drawn here is that Exactitude memo implies that designing measurable and thus detailed requirements by using a precise and formal language is among the keys for the successful product completion.

The Visibility memo and intra-team communication.

The next memo Italo Calvino discusses is called Visibility. This memo primarily focuses on the importance of imagination. When Dante is meditating and viewing the

“images directly form in his mind",

, he realises that images are actually

“raining down from heaven”.

This is exactly what the author depicts as an imagination and visibility.

The importance of visibility is traced in the words of Dante, where he compares the imagination as something that

“stealing us away from the outer world and carrying us off into an inner one".

There are two different imagination processes, first one starts with the written text and ends up in

“visualized"

image, as when the book is read, and the second one, on the contrary, starts with visual image and ends up in its

“verbal explanation", as with the movies.

The importance of visualizing and imagination is now clear, yet there is

“warning of the danger we run in losing a basic human faculty: the power of bringing visions into focus with our eyes shut".

As such, applying these methods to software engineering can be of an advantage. One proposition of the application is to organize the individual and team brainstorming sessions, where each member can delve deep into his or her inner world and mind.

The visibility memo raises the importance of communication through visual image as

“a ways of attaining this knowledge of the most profound meaning".

A few conclusions regarding software engineering can be drawn here. Visual communication for instance can be a great tool to boost the overall communication and experience of the team.

By communicating ideas in this manner, one get a better perspective on the product and its goals, and team members can clearly convey their ideas and challenges, and everyone else can understand them.The most common example of this happens when developers and product managers in the team speak in different languages, since they both use specific terminology in their expressions.

Although there could be linguistic problems in this particular situation, visual communication could serve as a useful mechanism. While word choices can be subjective and according to the glossology of a specific person, which might make them ambiguous or obscure, images come in place by communicating ideas unequivocally due to their more or less similar interpretation.

**The Multiplicity memo and software building process.**

Italo Calvino begins the discussion of Multiplicity concept with the passage from Gadda, where they share a common interpretation of the world as a

“system of systems, where each system conditions the other and is conditioned by them".

Software building can be interpreted in this manner as well. Multiplicity refers to a multifaceted approach to building software. It means that developing software should be an inherently flexible process. It needs to be capable of reflecting the ever changing values and trends of the current world.

In the larger picture of the world, multiple large systems live in mutual dependency from each other, and this also applies to smaller systems. Thus, when developing software, the changes in neighboring systems should be traced so that the requirements can be reflected and adjusted in accordance with changing values and trends.

Next, according to Musel,

“knowledge is the awareness of incompatibility of two different polarities",

where one is

“exactitude",

and the other one is

“soul, or irrationality".

And the the value that Italo Calvino wants to be

“passed on to the next millennium"

is the combination of these two polarities, making them exist simultaneously. It is possible to apply this conclusion to the software building process as well. The product should be built in accordance with all the mathematical and scientific rules (or technical requirements), yet this product must also maintain a soul. Metaphorically, the soul can be represented as the core idea of the product. It must have some value and impact on those using it.

By pursuing only the ability to produce a perfectly programmed product and ignoring the values and needs this product will possess will eventually turn into something that no one will be eager to use. In the meantime, contemplating solely about the changes and values the product can bring will create nothing but the ideas only.

Thus, the next millennium should utilize the concept of multiplicity to maintain a balance between two.

**Conclusion**.

There are five memos that Italo Calvino has passed on to the next millennium: Lightness, Quickness, Exactitude, Visibility, and Multiplicity. Each of these ideas can be applied to the software engineering arena with a promising benefit. By applying Lightness principles into the team building and management processes, better product can be delivered. Quickness memo from the book can be adopted to optimize the workflow and contribute to the work completion in less period of time. The Exactitude value relates to the precise requirements designing process and will increase the chances of achieving a successful and complete product. Visibility pertains to facilitating comprehensive and unambiguous communication between team members. Last but not the least, the Multiplicity memo emphasizes the importance of being flexible on the way of building a product. Applying all of the advice provided, software engineering can reach new heights.