1. What were the primary contributions of the paper as the author sees it?
2. What were the main contributions of the paper as you (the reader) see it?
3. How does this work move the research forward (or how does the work apply to you)?
4. How was the work validated?
5. How could this research be extended?
6. How could this research be applied in practice?

The author believes that in order to ease the process of software engineering it is necessary to address the problems inherent in the nature of software. He calls these problems its *essence* comprising of complexity of software systems, the need for developers to conform to rules, desire for change over time and invisibility in terms of design. However, according to the author the past breakthroughs have mostly been made in solving *accidental* difficulties i.e. problems faced in software production. According to the author, real progress will be made if energy is focused on solving the problems of *essence* rather than *production*. The author lists down past breakthroughs like high level languages etc and ongoing research in solving *accidental* problems but feels that they will have limited benefit. He lists potential solutions to *essence* problems that could improve on the quality of software delivered.

According to me, at the time of writing this paper the author did a forward thinking analysis of where software engineering research was heading towards. He pointed out that the SE community was heading towards solving *accidental* problems that will have limited impact but if they shifted their focus towards problems of *essence* they could find the silver bullet that will have long lasting improvements to the problems faced in software engineering.

This article moved the research forward by providing the research community by concretely defining the high impact areas of SE research and the progress made thus far in achieving them. It laid down the basis of agile methodology, building general purpose software and the need for solution architects to improve the design phase of the software development process.

Brooks claim that software development wont become better unless problems of essence are solved is also reflected in Wirth’s law that states software is getting slower more rapidly than hardware becomes faster. However, by using this research SE researchers could acts as a guide to the nature of problems that will have a high impact in improving the quality of software delivered.

The author tries to raise awareness of the fact that in order to make the process of software development easier and more effective there is a need to develop solutions that tackle the nature of software that make the software more hard. We focus on the so that the products developed are not flawed. The a