

# Deconstruction and Programming

Bidesh Thapaliya

16 Jan, 2014

## Abstract

Deconstruction is a way to read against the text. It looks for ways in which the opposition of the binary is privileged to analyse the contradiction it generates. In most cases, as argued by Derrida, flipping the privileged can show the alternate meaning. A deconstructionist has to continually analyse the oppositions and question the state to challenge the existing meaning. It helps analyse the meaning beyond the text presents and this has lead many disciplines like literature and philosophy, to name a few, use it extensively. However, it might, as well, be very closely related to computer science: especially the art of programming.

Contemporary programming has improved significantly given the advanced machines we have access to. Computers have become a household necessity due to which software business have flourished extensively. Consequently, even the advanced machines have to channel the resources for optimal performance. To achieve this, any programmer deconstruct!

One of the binary opposition for a programmer to analyse is speed vs. space. To optimize the process, if for a program speed is the privileged, a programmer has to flip this binary and see if allowing the code to be space efficient would be better. Second binary opposition that programmers apply is running vs. crashing. An essential part in software development is debugging. What makes the program run is privileged and flipping the binary and analysing what makes it crash is often practiced by programmers. This focuses on finding the bugs and mending them.

Furthermore, meaning is always changing, as dictated by the philosophy of deconstruction. Programmers intend to continually enhance the functionality of the program. This implies that codes are changing too. Therefore, in the art of programing, deconstruction seems essential.