

The Uncomfortable Truth of Software Engineering and how to Cure it

Abstract

When Dr. Ivar Jacobson was awarded the Gustaf Dalén Medal in 2003, at the age of 63, you could be forgiven for thinking it would be a fitting tribute to a distinguished career in software engineering; that the next steps would be lazy afternoons playing golf and watching the grandchildren get older.

His achievements had not just been Use Cases, but also the invention of components for Ericsson's AXE system (together with Göran Hemdal), and the Unified Modelling Language (UML, with Grady Booch and James Rumbaugh) and the Rational Unified Process (RUP, with Philippe Kruchten).

But that isn't how Ivar saw things. Rather than looking back at his accomplishments he was looking around at what was happening in software, and what he saw troubled him. Even in 2003 and especially so today, virtually every company is dependent on software and many more companies than you can imagine are in fact software companies. They may think their business is logistics, manufacturing, automotive or finance, but at the core to their success – or failure – is software. The problem was that despite the advances made in software engineering and technology, software was an immature discipline, dominated by behaviors more reminiscent of a craft than the engineering profession many assumed it was.

This realization – that remaining a craft would increasingly become a problem as more and more of the world comes to rely on software for its success – spurred Ivar into action. Rather than settling into retirement, he set about searching for a unifying theory of software that would help elevate software engineering to a fully-fledged engineering discipline. Today, the results of this work are helping organizations rise above the conflict of method wars, tailor development methods without reinventing them and use the power of gameplay to help teams optimize how they work.

In this talk, Ivar will transport you back to 2003 – when Use Cases and Ivar's other "baby" the Rational Unified Process dominated software development – and explain the worrying signs that led him to set out on a different path. He will explain the importance (and challenge) of creating an open standard and introduce you to the results of the work – Essence: a common ground for software engineering.

Although Essence seems straightforward, even obvious, Ivar will show how it is a profoundly different and more powerful way to approach software. One that has the potential to revolutionize the software industry – and given that almost all big companies nowadays are software companies – the impact of this Swedish engineer could yet another time be world changing. This time more dramatic than ever.