

# **SOFTWARE ENGINEERING COLLABORATION – A RESPONSE TO THE EVER INCREASING SPLINTERING OF ENGINEERING SPECIALISMS**

**Peter Howard**

**Technical Lead (Software Engineering)**

**Altran UK**

## **ABSTRACT**

The best way to get custom simulation, modelling, optimisation and design software for engineers written is to have the engineers do it themselves. Discuss.

It's worth discussing because there are many advantages to that approach.

In the 19th century, engineering was developing fast but it was still possible for a genius, like Brunel to design in mechanical, marine, rail and civil engineering disciplines. Today, all these branches of engineering are now so sophisticated that not even Brunel would be silly enough to attempt them all.

In the 21st century, things get done through multi-disciplinary teamwork.

This applies to software engineering. It has reached a level now where, in the hands of professionals, it is able to meet engineers' demands that 25 years ago were not even thought of. People who work in all the other engineering fields spend all their spare energy just keeping up to date with developments in their own area of expertise. And so they cannot be expected to create software to the level that the professionals can – no matter how much fun it may be to try.

In this talk we will give some examples of how multi-disciplinary teamwork has been applied to create cutting edge, specialist engineering tools and demonstrate that it is NOT better to do it yourself rather than work WITH other people. The talk classifies a set of fairly universal problems and weaknesses that occur when engineers "go it alone" – which you will almost certainly recognize even if you haven't formalised their classification before. And it then looks at how a properly crafted collaboration can make it better. We look at the potential for getting more software tools for your money, getting it sooner, making it easier to change and evolve, having more confidence in what it's doing, making it feasible for more people to contribute to it, reducing the amount of effort that must be allocated to looking after it. And many other areas of potential

Of course collaboration like this is not a panacea. So we'll also look at the difficulties, obstacles, challenges, and some lessons learned.

## **SUGGESTED THEMES**

**Themes:** Relevant across themes

**Keywords:** Bespoke Custom Software, Collaboration, Software Engineering