Open Systems Transcript

As Jaafari (2003) describes, we live in an increasingly complex society, and the nature of modern projects, especially software projects, reflects this complexity. Modern trends lean increasingly towards Open Systems, which typically entails code reuse, shared APIs and increased challenges around legal, ethical, social and professional aspects of a system.

The expectations of stakeholders on a modern SE project manager have not changed, they expect him/her to be able to manage stakeholders and resources, schedule, produce plans and estimates and ensure milestones and deliverables are met. However, the environment in which modern projects are produced has changed significantly. If we recall the first large software projects, such as the System 360 project, estimates could be made based on the lines of code (LOC) produced or predicted; planning models tended to follow the waterfall approach and most of the effort required would be people centric in nature.

In a modern project, typically between 60 - 90% of the code will most probably come from external, third party (often open-source) libraries (Makitalo et al, 2020); that means that traditional LOC or even function point estimation techniques are no longer viable. Even expert judgement techniques are limited because as the authors note "it can be argued that the overall understanding of the reused software has decreased over the years" (Makitalo et al, 2020).

Secondly, many modern systems will rely on external systems via API calls, to perform tasks as basic as address lookup or geolocation to functions as integral as verifying the financial and/or legal standing of a company.

Thirdly, a lot of the testing, building and releasing parts of the SDLC is increasingly automated, meaning that checking, changing and rescheduling is an increasingly technical role.