Chapter 6
Towards a Conceptual Framework for Open Systems Developments

James A. Cowling
School of Systems and Enterprises, Stevens Institute of Technology, USA

Christopher V. Morgan
PA Consulting Group, UK

Robert Cloutier
School of Systems and Enterprises, Stevens Institute of Technology, USA

ABSTRACT

The systems engineering discipline has made great strides in developing a manageable approach to system development. This is predicated on thoroughly articulating the stakeholder requirements. However, in some engineering environments, requirements are changing faster than they can be captured and realized, making this 'traditional' form of systems engineering less tenable. An iterative system refinement approach, characterized by open systems developments, may be a more appropriate and timely response for fast-changing needs. The open systems development approach has been utilized in a number of domains including open source software, Wikipedia®, and open innovation in manufacturing. However, open systems development appears difficult to recreate successfully, and while domain tradecraft advice is often available, no engineering management methodology has emerged to improve the likelihood of success. The authors discuss the essential features of openness in these three domains and use them to propose a conceptual framework for the further exploration of the effect of governance in determining success in such open endeavors. It is the authors' hope that further research to apply this conceptual framework to open source software projects may reveal some rudimentary elements of a management methodology for environments where requirements are highly uncertain, volatile, or 'traditional' systems engineering is otherwise sub-optimal.

DOI: 10.4018/978-1-4666-7230-7.ch006

INTRODUCTION

The systems engineering discipline has made great strides in developing repeatable, measurable, predictable, and controllable processes to aid the management of system development projects. These processes are highly disciplined and structured, and often start with a thorough examination of stakeholder requirements. But, in some environments, these requirements are changing faster than they can be captured; therefore these 'traditional' system-engineering processes can prove to be less useful.

Complex project management tradecraft includes a number of possible solutions to counter this problem. The end deliverable can be broken down into smaller elements and systems, therefore reducing complexity and allowing a quicker solution to the changing needs. Alternatively, a more proven technology can be chosen which is better tested, giving a quicker and de-risked implementation path - once the solution has been designed. Or, a trade off of requirements can be made, promising less to the owner initially, but with successive stages to build functionality. All of these options are valid paths and have their own merits and demerits. In this paper, we are choosing to focus on another path, that of iterative development as a solution to the problem. It is not the only solution, but it is one that has gained a successful track record and is worthy of study. In environments with a high degree of uncertainty, an iterative system-refinement approach, characterized by open systems developments, may allow for a more timely response to changing operational needs.

However, open systems developments appear to be difficult to control and there are wide variations in project success rates. For example, despite increasing popularity and prominent achievements, many open source software projects seem to languish in the SourceForge repository: only a relative few achieve a state of continued useful software production (Madley, G., ed, 2010). These open system developments typically use few of the established systems engineering techniques and, almost by definition, applying such mechanisms is infeasible in the open environment.

Without management and structure, how then can an open system development take advantage of the systems engineering discipline? The work of Checkland and others provides guidance, by defining approaches to complex problems that do not respond to the rigor of traditional systems engineering. Checkland (1994) presents a review of the evolution of system thinking philosophy over the last half century demonstrating that when systems engineering began to be applied to organizational challenges, a new paradigm became necessary. This was one that would better cater for the nature of self-governing organizations (such as firms or, in the context of this paper, open communities). Checkland describes organizations as more than rational goal seeking machines, and that all social groupings take on some characteristics of the tribe (Tönnies, 1955). This provides a richer picture of behaviors and models of collective efforts. Further, the work of March and Simon (1958) establishes that managers are unable to optimize, but instead seek to 'satisfice'; finding solutions that are good enough rather than ideal. These developments were brought into focus and given practical applicability by Checkland's own work on the soft systems methodology, which provided an analytical and predictive basis for understanding complex undertakings, and finding 'actions to improve' their outcomes. Thus, these advances in systems thinking philosophy direct the exploration of open systems development management towards studying governance improvement actions that effect sufficient progress towards success.

The authors believe that if a relationship between the collective decisions made by open projects and their ultimate success or failure could be found it may be possible to develop a useful management methodology. But this presents at least two further questions that need to be addressed before a management method could be