Proceedings of the MIT Lean Aerospace Initiative (LAI) Plenary Conference on "Enterprise Transformation: Closing the Gap between Knowledge and Implementation". Cambridge, MD. April 18-19, 2007

Knowledge Integration for Technical Problem Solving in Complex Aerospace Systems Development – The Case of Military Avionics

Marc Haddad, PhD Candidate, MIT
Dr. Kirkor Bozdogan, PhD, MIT

Abstract

This paper proposes a framework for knowledge integration in a complex product development setting, using case studies from large-scale manufacturing enterprises in the US defense aerospace industry. The framework highlights the main channels and mechanisms used most frequently in different technical problem solving situations. We conclude that varying degrees of problem complexity require the establishment of particular knowledge integration channels and the use of appropriate knowledge integration mechanisms for the type of technical problem at hand.

Introduction

Some of the major sources of cost and schedule overruns in new product development are the time and resources employed to troubleshoot unforeseen technical problems at various stages of the development process. Troubleshooting is accomplished by applying the organization's problem-solving capabilities to diagnose and solve the problem (Fujimoto 1999). Organizations employ several formal and informal mechanisms to that end, all of which involve the integration of knowledge from different sources inside and outside the organization (Grant 1996). Problem-solving occurs mostly at the level of the integrated product teams (IPT's) in charge of developing the affected systems and/or subsystems (Browning 1996). The process of integrating knowledge to that end involves several individuals with varied expertise, and with different organizational affiliations (sometimes including the customer, partners and/or suppliers, and other parts of the prime organization). The process consists of transferring specialized knowledge (in the form of tacit knowledge and explicit information) from internal and external sources in the organization's network, combining it with existing knowledge, and applying the new knowledge to solve a specific problem (Grant 1996). It is thus a highly contextual process, and the mechanisms employed vary widely depending on the nature of the problem, the characteristics of the product system in question, and the relationships of the stakeholders involved, among other factors. In the case of complex systems development, troubleshooting becomes a continuous phenomenon known as fire-fighting (Repenning 2001), and knowledge integration becomes increasingly complicated in line with the increasing difficulty of the problems encountered , which is due to the complex interdependencies at the level of the system itself, as well as at the level of the teams and organizations involved in the problem-solving effort (Braha and Bar-Yam 2007). As a result, organizations have a vested interest in evolving their problem-solving capabilities through more efficient and effective integration of their knowledge resources, thus reducing the cost and time to develop complex systems (Takeishi 2002). This paper will identify the most commonly used channels and mechanisms for integrating knowledge in different problem solving contexts with varying degrees of system and organizational complexity, and will present a discussion of the enablers and barriers facing knowledge integration in this context.

Methodology

The research proceeded in two phases: 1) an exploratory phase for building an overview of the knowledge integration channels and mechanisms commonly used in a complex product development context, and 2) a focused phase to investigate design and engineering problems and problem solving approaches at the corresponding stages of the development process (i.e. at the design and engineering - including integration - stages of development). In order to determine the relationship between the problem solving context and the knowledge integration process, the following basic constructs were developed from the first phase of the research and grounded with the literature on knowledge integration, problem solving and complex systems development (Carlile 2002; Moir and Seabridge 2006):

a) Problem and problem solving typology: at the design and integration phase of the development process, the two main types of technical problems encountered most frequently are related to i) system integration and ii) subsystems engineering type issues. The former is the sole responsibility and core competency of the prime organization, while suppliers are tasked with the latter. Thus the technical knowledge being integrated can be classified as system (or architectural) knowledge and subsystem (or component) knowledge, respectively (Aoshima 2002). Problems of either type can be further characterized along two dimensions: i) problems that are highly localized to one subsystem or one part of the overall system, and ii) problems that propagate across multiple subsystems or that affect more than one part of the overall system. In addition, a problem and its corresponding problem solving process can be either new (where the problem is unique and the problem diagnosis and solution must be developed from scratch) or old (where the problem is only partially unique and the problem solving process can benefit from previous knowledge).

b) Problem and problem solving complexity: complexity can be measured as a function of the above constructs in addition to the scale and scope of the problem and problem solving process, where: complexity = f (type, scale, scope). On the system side, scale is a measure of the number of subsystems affected by the problem, and scope refers to the severity of the problem in terms of its disruptiveness to the overall system. On the organizational side, scale is a measure of the number of IPT's engaged in the problem solving process, and scope is a measure of the difficulty of problem solving in terms of the extent of organizational boundaries being spanned.

The grounded theory research approach (Strauss and Corbin 1990) was then used to collect and analyze data about a range of problems using the constructs developed above, in order to build and validate a theoretical framework for knowledge integration in this context.

Case Study: Military Avionics Systems

This paper will present preliminary findings from three case studies involving a prime defense contractor in the US aerospace industry and its two main suppliers of avionics subsystems. The research spans three military aircraft programs where the prime serves as system integrator and where the suppliers are tasked with subsystems engineering for key avionics subsystems across all three platforms. The rationale for choosing the defense aerospace context for this research is that it provides a rich setting for investigating the challenges associated with problem solving and knowledge integration, starting with the classified technology environment preventing open knowledge sharing even within the same organization, and up to the contractual barriers designed to protect core competency at the expense of knowledge transparency across enterprise boundaries. These factors, along with the high level of technical complexity of avionics systems, compound to make the design and development process an ideal research lens for studying the knowledge integration phenomenon in a continuous firefighting context. The research was further focused on three mission-critical avionics subsystems common across all thee aircraft programs, the multi-function radar and the electronics warfare suite provided by the suppliers, and the mission computer developed by the prime. These systems were chosen for their high levels of complexity and their closely linked functionalities within the mission systems suite, which translates into a web of interdependencies in the problem solving and knowledge integration environments surrounding their development. The findings presented in this paper are based on a total of 50 interviews about major technical problems (i.e. "class 1" engineering changes) with individuals from program management, engineering, material and support functions such as knowledge management, at both the prime and supplier organizations.

Proposed Theoretical Framework

The main knowledge interactions associated with technical problem solving in the development of an avionics system are shown in Figure 1. The mapping captures the most common dimensions for knowledge integration in a product development context (Takeishi 2002), namely a) internal knowledge integration and b) integrated problem solving with suppliers. The mapping is based on a project- (or program-) based organizational structure which is considered the most common form adopted in the development of complex systems (Dosi, Hobday et al. 2000). And since the IPT is typically the locus of problem-solving in a new product development context, it is chosen as the unit of analysis for mapping knowledge interactions. In addition, since avionics are layered systems consisting of multiple nested subsystems each made up of a large number of component parts, the interactions at the subsystem and system level are shown separately.

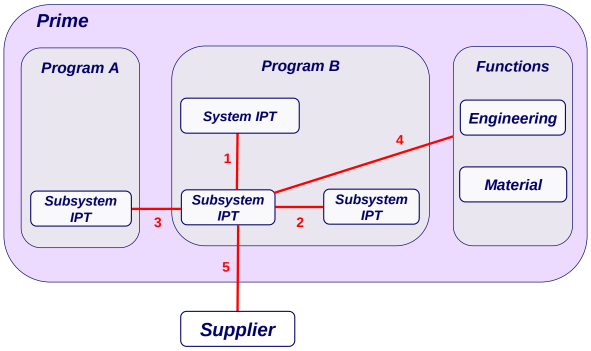

Figure 1: Knowledge Interactions

Only direct knowledge interactions are shown, making up five main channels as follows: Intra-program knowledge integration involving system integration type problems between the overall avionics system IPT and a subsystem IPT are captured along channel #1, whereas channel #2 accounts for internal interactions at the subsystem level. Inter-program knowledge interactions are represented by channel #3 (only the subsystem level is shown for simplicity). While other knowledge interactions are possible between teams in different programs (e.g. at the system level), it is assumed that channel #3 adequately captures the nature of interactions across this boundary. Channel #4 represents the last category of internal links with functional groups such as engineering and material, and channel #5 accounts for external interactions with suppliers. Indirect interactions such as those involving the customer, as well as knowledge integration between suppliers are not shown, since they are not centered on the subsystem IPT chosen as the unit of analysis.

We classify technical knowledge interactions in product development along three dimensions, namely the exchange of i) information, ii) advice and iii) assistance to solve technical problems. Information is codified in documents, tools, processes, and artifacts, while advice and assistance are based on tacit knowledge held by individuals with specialized expertise. Following is the proposed knowledge integration framework at the macro-scale (i.e. at the level of the channels for knowledge integration):

Table 1: Macro KI Framework

Ch.#	**Problem Solving Type**	**Knowledge Integration Type**
1	System integration (avionics subsystem to aircraft)	Information seeking Assistance seeking
2	Subsystem integration (avionics subsystem to subsystem)	Information giving/seeking Assistance giving/seeking
3	System and subsystem integration	Advice seeking Assistance seeking
4	System engineering and integration	Information seeking Advice seeking Assistance seeking
5	Subsystem engineering	Information giving/seeking Advice giving/seeking Assistance giving/seeking

The knowledge integration mechanisms (i.e. the micro-scale of knowledge integration) most commonly used to carry out problem solving along the corresponding channels are presented below:

Table 2: Micro KI Framework

Ch.#	**KI Type**	**KI Mechanisms**
1	a) Information about system requirements b) Assistance with integration problems with other aircraft systems	→ Requirements docs → Chief engineer forum → Special action teams (cross-system)
2	c) Information and assistance with design dependencies between different avionics subsystems	→ Integrated design and management tools → Engineering share sessions → Special action teams (cross-subsystem)
3	d) Advice and assistance with subsystem-specific integration problems (with other avionics subsystems and other aircraft systems)	→ Multi-program lessons learned databases → Multi-program share sessions → Special actions teams (single subsystem), non-advocate reviews
4	e) Information about design standards, processes & procedures f) Advice and Assistance with systems engineering and integration problems	→ Standardized process tools and templates → Formal design reviews → Deployment of tech fellows and subject matter experts to program IPT
5	g) Information about system specifications h) Advice on design processes and procedures i) Assistance with subsystem-specific design problems	→ System requirements and subsystem specification docs → Co-location → Site visits → Multi-organization troubleshooting taskforce

Discussion and Implications

At the macro-level, we found that for highly localized problems, such as a functionality or reliability problem confined to a single subsystem, problem solving occurs mostly along channels 3 and 5. This is because subsystem-specific knowledge resides mostly with the supplier and with IPT's working on similar subsystems in other programs. Whereas for problems that propagate across multiple subsystems, we found that problem solving occurs mostly along channels 1, 2 and 4, with channel #1 being used only when problems involve other aircraft systems outside the avionics suite.

At the micro-level, we found that the use of tacit knowledge mechanisms dominate those that are based on codified information, especially as problem scale increases (e.g. system integration problems and problems affecting multiple subsystems). This is because face-to-face brainstorming is often needed in novel problem situations, or when problem solving involves multiple IPT's. Conversely, one of the most cited shortcomings of knowledge integration in this context is the lack of efficiency and effectiveness of database-type mechanisms, first in terms of the time-consuming and difficult task of searching for useful stored knowledge, and second with respect to the relevance of the stored knowledge that becomes quickly outdated in a fast-clockspeed technology setting as the development of avionics systems. Similarly, and for highly localized problems confined to one subsystem, we found that cross-program knowledge integration (i.e. along channel #3) is a weak link due to the ineffectiveness of lessons-learned databases, and due to the dominance of informal (advice) mechanisms. This was compounded by the fact that knowledge integration along both channels #3 and #5 are hampered by conflicting interests and policy barriers, both internally (e.g. program silos) and externally (contractual and proprietary barriers). This makes localized problems especially hard to solve, which is a counter-intuitive result given that the scale of both the problem itself and the problem solving process is very low. However, in terms of our proposed complexity construct, the implication of this result is that problem-solving scope (i.e. the type of organizational boundaries being crossed – in this case organizational and inter-organizational type boundaries) is a major contributor of problem solving complexity, which is in line with results from research on knowledge networks where network centrality and direct relations are found to reduce the solution search space and increase problem solving efficiency (Hansen 2002). Another implication of this result is that the increasing trend towards more modular system architectures, used to manage design complexity by reducing interdependencies between different parts of the system, can be offset by the difficulty of problem solving across external boundaries. From a product platform perspective, the implication of this finding is that to achieve multi-platform benefits in a non-concurrent development setting as is typical in the defense aerospace context, it is more useful to rotate experienced engineers into new programs than to codify previous knowledge into shared databases. Furthermore, since requirements changes are frequent in a complex development setting, it is more beneficial to carry over previous knowledge early on at the concept definition phase, in order to better inform the requirements definition process and reduce requirements changes along channels #1 and #5.

Future Work

The cost of using specific types of mechanisms will be evaluated in future work in terms of their impact on budget and schedule for equivalent problem solving situations. This will allow a comparison of the efficiency and effectiveness of different problem solving approaches for similar problems.

References

Aoshima, Y. (2002). "Transfer of System Knowledge Across Generations in New Product Development: Empirical Observations from Japanese Automobile Development." Industrial Relations **41**(4): 23.

Braha, D. and Y. Bar-Yam (2007). "The Topology of Large-Scale Engineering Problem-Solving Networks." Physical Review E **69**(1).

Browning, T. R. (1996). Systematic IPT Integration in Lean Development Programs. Engineering Systems Division. Cambridge, MIT.

Carlile, P. R. (2002). "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development." Organization Science **13**(4): 13.

Dosi, G., M. Hobday, et al. (2000). Problem-Solving Behaviors, Organisational Forms and the Complexity of Tasks. Laboratory of Economics and Management Working Paper Series. Pisa, Italy, Sant'Anna School of Advanced Studies**:** 27.

Fujimoto, T. (1999). The Evolution of a

Manufacturing System at Toyota. New York, Oxford University Press.

Grant, R. M. (1996). "Prospering in Dynamically Competitive Environments: Organizational Capability as Knowledge Integration." Organization Science **7**(4): 375-387.

Hansen, M. T. (2002). "Knowledge Networks: Explaining Effective Knowledge Sharing in Multiunit Companies." Organization Science **13**(3): 17.

Moir, I. and A. Seabridge (2006). Military Avionics Systems. Chichester, England John Wiley & Sons.

Repenning, N. (2001). "Understanding fire fighting in new product development." The Journal of Product Innovation Management. **18**(5).

Strauss, A. and J. Corbin (1990). Basics of qualitative research : grounded theory procedures and techniques. Newbury Park, California, Sage Publications.

Takeishi, A. (2002). "Knowledge Partitioning in the Interfirm Division of Labor: The Case of Automotive Product Development." Organization Science **13**(3): 321-338.

Biographies

Marc Haddad is a PhD candidate in the Technology Management and Policy Program of the Engineering Systems Division at MIT, and a graduate research assistant with the Lean Aerospace Initiative (LAI).

Dr. Kirkor Bozdogan, PhD is a principal researcher with the Lean Aerospace Initiative (LAI) at MIT.