Playing a 3-stringed violin: Innovation via the joint evolution of people, process, and Knowledge Management System

Lynne P. Cooper
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive Pasadena, CA 91109
lynne.p.cooper@jpl.nasa.gov

Abstract

Users continuously evaluate the value and performance of their Knowledge Management Systems (KMS). As suggested by a punctuated socio-technical system process model, today's success can quickly become tomorrow's failure should the KMS fail to meet evolving needs and expectations. The more deeply a tool is embedded in the actual work process, the more vulnerable it is to emergent changes and perturbations. This paper uses the metaphor of a "3-stringed violin" to explore how differing levels of user knowledge about tools and processes can lead to system perturbations and how the active involvement of other actors can dampen the impact of perturbations, i.e., help the system survive the operational equivalent of a broken string. Recommendations suggest ways to increase system resiliency and contribute to incremental innovation.

1. Introduction

Violins have four strings that are tuned to have significant overlap between the notes that can be played on adjacent strings. For a given composition, violinists choose what fingering (which string to play, with which finger) to use to play the required notes. This flexibility allows the musician to exploit the capabilities of the violin (e.g., vibrato, cross-string resonance) to best interpret the music. Over time, preferred fingerings have evolved for classic compositions, and violinists are taught these fingerings as they develop their repertoire.

A large catalogue of classical music has been composed to take advantage of the range, beauty, and expressiveness of the violin, as a four-stringed instrument. In theory, however, if one of the center strings were to break during a performance, the violinist could continue to play the piece by dynamically reworking the fingering. Stories of virtuosos who have performed complex pieces under such circumstances are legendary e.g., [18].

In the same sense that a violin is a tool to make music, a Knowledge Management System (KMS) is a tool to do work. As supported work processes increase in complexity, the tools need to become more sophisticated. For processes where problem interpretations, deliberations, and actions unfold unpredictably and equivocally in interaction with others, known as emergent knowledge processes (EKP) [16], one can argue that it is nearly impossible to design the perfect KMS a priori, i.e., create a "4-string violin" version of a KMS from scratch. It is only through the interaction of the user with the tool to execute the process that the full set of requirements emerge and evolve. Accepting this premise implies that new KMS are likely to become "3-string violins" that require the users to dynamically alter their behavior to accomplish their work.

To continue this metaphor – when faced with playing a piece composed for four strings on an instrument that suddenly has only three working strings, what is a violinist to do? First, the violinist can cancel the performance. Second, she can pause the performance to replace the string, the violin, or the performer. Third, the performance can go on with the violinist compensating for the limitations of the instrument, accepting that the performance may suffer (e.g., timing, tuning, resonance). Fourth, the violinist can switch to an easier piece. And finally, if the unfortunate violinist is part of the string section (vs. a soloist), he can either stop playing or play the piece minus the notes on that string with fellow orchestra members compensating.

Although KMS developers strive for their systems to be perceived so, in general, KMS are not as inherently instrumental to work processes as the violin is to the violinist. When faced with a situation in which a tool does not do what is needed – even if that need has just been discovered – KMS users have options similar to the violinist discussed above. In the worst case, the user cancels the "performance" abandoning both tool and process. More likely, however, the user abandons the KMS and finds a way

to work-around the situation. If enough "strings" break or break too often, even a previously successful KMS risks being abandoned.

No one intentionally sets out to deliver a 3-stringed violin. However, the demands of KMS field operations in all but the most simple of environments eventually will break a string. This paper builds on the experiences gained fielding multiple successful KMS to support emergent knowledge processes, and on lessons learned keeping those systems successful despite the occasional broken string.

This paper presents a practitioner perspective via a descriptive case study [22] of the evolution of a KMS to support peer review of high-risk projects. It starts by presenting an overview of the domain and the KMS. It then proceeds to show how different types of users contribute to product and process innovation via the joint evolution of people (users & developers), process, and product over the course of the project lifecycle. It ends with a discussion of lessons learned to reduce the occurrence and impact of the inevitable broken strings.

2. Domain Description

New product development (NPD) projects are credited with improving performance, increasing innovation, and extending the life of organizations [5]. To ensure project success, it is critical to detect flaws and identify problems early in the development lifecycle. This becomes more difficult as the level of innovation increases, for example, from incremental improvements to next generation improvements, new-to-the-firm, and new-to-the-world projects [Griffin].

Formal review processes are a popular mechanism for applying the combined knowledge and experience of an organization toward evaluating projects [7][20]. In a formal review, people with experience in relevant domains independently assess the project and the product being produced with respect to, for example, quality of the design, potential problems or risk areas, errors and omissions, and sufficiency of resources to complete the project.

For more complex projects, formal review boards can consist of a large number of specialists in order to cover all the necessary disciplines. This requisite variety [3] in review board membership is needed to address detailed technical issues across multiple disciplines, as well as provide an integrated strategic assessment of the project relative to organizational goals [6]. This variety, however, can lead to disjoint or confusing feedback to the project. Firms that can, however, integrate the specialized knowledge (e.g., of a large number of individual reviewers) can gain a competitive advantage [9][13]. Therefore, it is not enough to just assemble a multi-disciplinary review board; an organization must also support the integration of this expertise.

The domain addressed in this paper is the formal peer review process for new product development proposals for a major US national research laboratory. The proposals document the conceptual design and implementation plan for space-based scientific research missions. The proposals are competitively selected and involve a broad spectrum

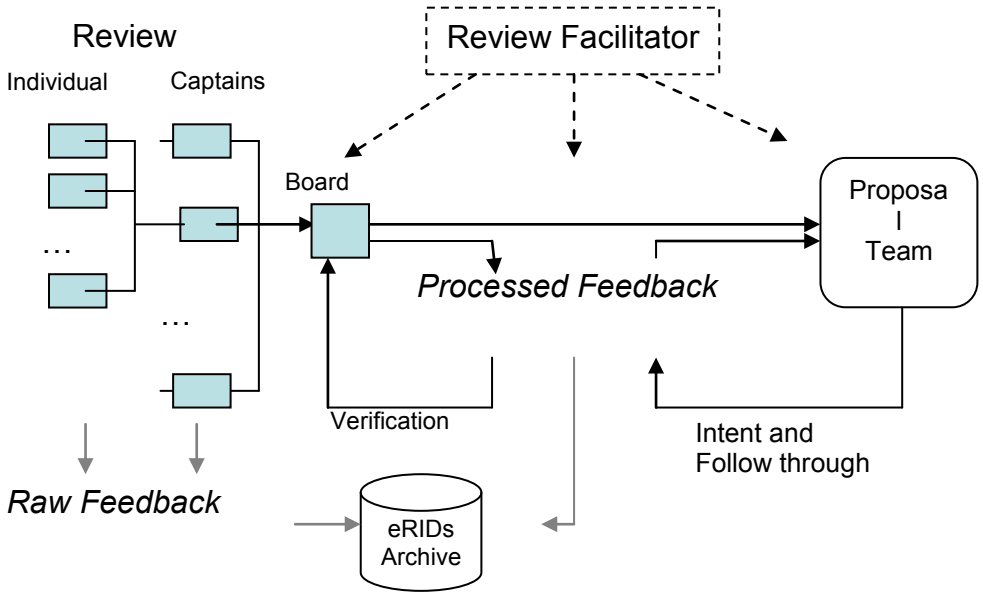

Figure 1. Peer Review Socio-technical System and Process Flow

of science, technology, engineering, and programmatic disciplines. The number of people involved in a formal peer review ranges from less than 10 to over 100, and review cycles tend to last up to two weeks.

This organization's review process begins with the assembly of the review board. The review boards are hierarchically structured and led by an executive from the part of the organization responsible for implementing the project, should the proposal get selected. The next level in the hierarchy consists of domain captains selected for each of the major technical and programmatic areas. The lowest level of the hierarchy consists of individual reviewers selected for either their specialized expertise in a relevant domain or their ability to evaluate the overall system.

Figure 1 presents an overview of the review process. The review process is a combination of individual efforts to generate and quality check feedback, and group sessions to integrate and make sense of this feedback. After receiving the proposal document, each reviewer reads it and generates feedback. This feedback is routed to the respective domain captains who consolidate it, make sure it is clear and consistent, identify major problem themes in their area, and present a list of feedback items to the review board chairs.

The review board chair then evaluates the feedback from across domains, identifies the most critical areas, categorizes the rest of the feedback, and provides specific direction to the project team for required actions. The proposal team members then review this feedback, and decide how they will address the feedback (accept, ask for clarification, or decline to address). The review board meets with the proposal team to address questions and reach agreement on the most critical actions. The proposal team then implements changes and the review board evaluates the final product to verify that all required changes have been addressed. The success of the process depends heavily on the knowledge and judgment of all the participants in both identifying issues and evaluating the severity of these issues [10].

This general process evolved over many years, first as ad hoc comments penciled onto hardcopies of the proposal document, and evolving to the current electronic system for collecting, assessing, prioritizing, packaging, distributing, and tracking feedback. The major process innovation occurred when feedback became more structured through the introduction of the Review Item Discrepancy (RID) form. The RID form required reviewers to provide a title, description of the issue or problem, recommended fix, location in the document, and assessment of project consequences if the issue was not addressed, rather than ad hoc comments, questions, and editorial changes.

After using a paper-based RID system for review of over twenty proposals in a six-month period, management commissioned the development of an electronic system to support the process. Following an accelerated development cycle [4][19] that included detailed requirements definition, implementation, multiple rounds of functional and user testing with prototypes, and beta testing with users, the electronic Review Item Discrepancy system (eRIDs) began operation.

The eRIDs tool is a knowledge management system (KMS) that consists of a web-based interface to an access-controlled database. It has been operational for over two years. The initial operational version consisted of a simple electronic implementation of the previous paper-based process. The system reached full operational capability eight months later and has been in routine operations for nearly eighteen months.

Based on management and user accounts, eRIDs has been highly successful and has contributed significantly to the efficiency of the review process. Recent experience has shown, however, that as the process has continued to evolve, the system has been stressed to keep up. The metaphor of "playing a three-stringed violin" emerged during the latest round of reviews conducted during May/June 2009. During this review cycle, 103 review board members produced over 1700 RIDs during the simultaneous review of four major proposals. While the system provided adequate support for the majority of participants, a small number of special users pushed the performance boundaries of the system, thereby breaking several "strings." The next section describes the system and the evolutionary drivers uncovered by the interaction of process, tool, and these special categories of users.

3. Case Study: Canonical Users

The peer review process can be characterized as a socio-technical system [21]. Following Leavitt's [14] open system model of organizational change, as modified by Lyytinen & Newman [15], the system consists of actors, structure, technology, and tasks, as listed in Table 1.

Per Lyytinen & Newman's [15] Punctuated Socio-technical Process (PSP) model, reaching a stable equilibrium state requires the four components of the system to remain in balance. Organization systems change by adjusting to perturbations; the components (actors, structure, tasks, and technology) need to be

constantly adapted to keep the system in equilibrium [15, p.9]. These changes can be either incremental or punctuated based on the resiliency of the deep structure of the socio-technical system.

Actors
▪ Review board members: Chair(s), domain captains, and individual reviewers ▪ Proposal Team members ▪ Board appointed verifiers ▪ Review Facilitator ▪ eRIDs Development Team
Structure
▪ Knowledge structure: Standardized content requirements encapsulated in RID form, with required fields and some restricted values ▪ Knowledge providers: hierarchical review board, with institutionally defined roles
Technology
▪ eRIDs KMS tool, with web-based interface to access-controlled database
Task (Processes)
▪ Provide concise, clear, consistent, prioritized, and actionable feedback to proposal team following content standards, e.g., one issue per RID ▪ Assess organizational risk and uncertainty ▪ Align organizational members with respect to future implementation ▪ Gain insight into efficiency and effectiveness of the review process

Table 1. Components of the Review Process Socio-technical System

This study focuses on the relationship between technology (the eRIDs KMS tool) and actors (review board members) as moderated by other actors (Review Facilitator, fellow board members). Canonical user types are categorized based on a 2x2 assessment of (1) the user's level of understanding of the tool and (2) the user's level of understanding of the process.

Level of understanding of the tool refers to the ability of the user to access and use the tool's functions. It includes general skills for operating a computer, navigating a website through a standard browser, and logging in via institutionally standard username and password mechanisms. For eRIDs, basic functions include navigating to the desired review and proposal, creating and editing a RID, and submitting a RID to a domain captain. Advanced features consist of combining, routing, prioritizing, and packaging sets of RIDs. Users assessed at a high level of understanding have mastered basic functions and are competent at using advanced features. Users assessed as low ranged from those unable to independently perform basic functions to those unable (or unwilling) to access the tool via their computer.

The primary goal of the review process is to provide feedback to the team in a way that helps the team produce a better proposal and makes inherent risks visible to the team and the organization. *Level of understanding of the process* refers to both how familiar the user is with the overall flow of the review (what steps occur when and by whom) and the ability of the user to align their behavior with the goal of the process.

Users assessed as having low understanding of the process did not know the review process steps and therefore missed deadlines, skipped important steps, provided incomplete information and provided feedback that was inappropriate for the level of review. Users assessed as having high levels of process understanding met deadlines, provided insightful and clear feedback that was actionable by the proposal team, focused their efforts on the most important issues, understood the value and intent of each step in the review process, and often facilitated the process for other users.

Review board members were categorized based on a 2 x 2 structure of high and low levels of tool and process understanding, as shown in Table 2. The following sections discuss each of these user categories and how their behavior perturbed the task, technology, structure and other actors in the review process socio-technical system. The review board members and other actors received process support from the Review Facilitator (RF). A senior engineer who was a member of both the eRIDs development team and the organization responsible for the review process filled the RF role. The RF was responsible for the maintaining the overall flow of the review, system set-up, user training and support, trouble-shooting, and process facilitation/operation during review board meetings. A goal for future review cycles is to reduce the demand on the skills required to perform the RF function. As such, each of the users discussed below present both challenges and opportunities for meeting this goal.

		Understanding of Tool	
		Low	High
Under-standing of Process	High	Abdicated User	Ideal User
	Low	Non-User	Over-Enthusiast

Table 2. Categorization of Review Board Members

3.1 Non-Users

Non-users had low levels of understanding of both the process and the tool, but had domain-specific knowledge important to the goals of the review. Therefore, their participation in the review was needed regardless of their willingness or ability to use the eRIDs KMS. Accommodating non-users required the use of advanced features of the tool, adjustment to the process, and modified responsibilities for other users.

Based on experiences in earlier review cycles, the current version of eRIDs allows for bulk input from an excel spreadsheet template. Spreadsheet entry is the de facto approach for external reviewers who do not have access to eRIDs. It is also used when reviewers have limited internet connectivity (e.g., when they are on travel). The spreadsheet format mirrors the RID structure and uses pick lists to constrain category and consequence inputs.

The system experiences "broken strings" regularly when trying to upload spreadsheet contents provided by non-users. Non-users often override constraints, leave required fields blank, add or subtract columns or introduce formatting beyond the row/column boundaries of the template. Because of these problems, importing external data required additional processing and error checking.

The eRIDs tool has two features that enable review process support to non-users. The first, implemented based on past "broken strings" is to define a proxy user – where an internal user is given proxy to simulate an external reviewer. The second is the ability for the RF to impersonate any other user. Originally implemented to support testing, this feature was reactivated and is now critical to support operations. In addition to allowing the RF to enter RIDs on behalf of another user, it also enables the RF to view the system from that user's perspective, which is helpful when providing remote assistance to a given user and troubleshooting eRIDs versus browser/platform system issues.

Beyond the logistics of entering feedback, non-users also stress the process because additional effort is often required to quality control their inputs. In the most recent review cycle, domain captains were burdened with interpreting, rewriting, re-titling, and otherwise processing a large volume of non-user RIDs. While non-users provided some critical feedback, it came at a high price. Although non-users represented only 14% of the users, they generated 54% of the RIDs that were rejected by the Review Board. In a process with extremely tight time constraints, these users added significant noise to and a lot of extra work for the process.

In addition, the non-user input often disrupted the overall flow of the review by being late and therefore bypassing crucial quality control steps early in the process. In many cases, the non-user RIDs had to be sent back to the domain captains to be re-processed in parallel with later process steps. The domain captain role expanded from simply editing RIDs to co-authoring RIDs. This resulted in a lot of confusion.

The role of the RF also expanded to support non-users. The RF was responsible for developing on-the-fly work-arounds to by-pass system configuration control mechanisms, functionally implement a "reject" feature, and reverse the flow of the RID process for a subset of RIDs while supporting continued execution of the rest of the process. The non-users, via their impact on process and tool, make it unlikely that the RF role can be down-skilled without modifying the eRIDs KMS.

3.2 Over-enthusiast

The second category of user is the over-enthusiast. This type of user is characterized as having low process understanding but high tool understanding. The over-enthusiast takes advantage of advanced features without necessarily understanding the resultant impact to the process.

The most common example of an over-enthusiast is the reviewer that generates a large volume of feedback without regard to the overarching goal of providing the proposal team with a manageable volume of focused feedback. These reviewers often step out of their domain area to address other areas (resulting in redundant feedback) or provide volumes of primarily editorial comments. Because the process errs on the side of completeness (one never knows which reviewer will uncover the fatal flaw), these types of reviewers add noise to the system. Unlike the non-user, however, the structure and discipline of using the eRIDs tool tends to limit the noise.

Another example of an over-enthusiast served as a domain captain. As such, he was responsible for grouping redundancies. The overall system is designed to accommodate redundancies to show that a particular problem was visible from multiple perspectives and was important enough for multiple reviewers to identify it. The eRIDs tool therefore incorporates a facility for grouping referred to as "SuperRIDs" to combine true redundancies. The over-enthusiast, however, used it to group distinct but related feedback, with a goal of reducing the sheer number of RIDs given to the team. By doing this, however, he significantly increased the complexity of the feedback, violating review guidance of "one issue per RID." Having multiple issues per SuperRID

rippled through the system making the SuperRID difficult to prioritize, and making it difficult for the proposal team to disposition and assign responsibility. The "3-stringed" work-arounds involved multiple approaches: delete the SuperRID and re-enter the data for the individual RID (for those that were caught) and having the proposal team manage their RIDs outside the RID tool so they could functionally split up this compound feedback. Additional eRIDs features were eventually implemented to allow SuperRIDs to be broken up back into constituent RIDs.

To address the over-achievers, part of the kickoff meeting held at the start of each new review cycle now incorporates an expanded training section on how to use the tool, how to write a good RID, reasons RIDs get rejected, and the importance of one issue per RID. Reviewers are explicitly instructed not to submit minor editorial RIDs, which they routinely disregard.

Domain captains now receive specialized instruction on how to process the RIDs, based on the needs of the review board. The amount of time they have to perform captain duties has been reduced to limit the time available for over-enthusiast behaviors.

3.3 Abdicated Users

A third category is the "abdicated user," characterized by high process understanding and low tool understanding. These users tend to be upper level managers and executives who provide compact, but extremely important feedback. Abdicated users don't – or won't – use the system, requiring others to take care of the mundane aspects of data entry for them based on statements in meetings, hallway conversations, or emails. This responsibility usually falls to a review board chair, domain captain, or the RF.

Abdicated users have not (yet) placed any driving demands on the tool, process, or other users. They do, however, require other users to have tool and role capabilities similar to those required to support non-users.

Abdicated users often provide an institutional, strategic perspective to the review process. One process change implemented to facilitate communication with these users at the end of the review process was an expansion of disposition category information to highlight those areas where the review board and the proposal team disagree on critical items.

3.4 Ideal Users

The final category is the ideal user, characterized by high levels of both tool and process understanding. These users are sharply focused on the end-goal of the review and use the tool to support meeting that goal in ways that are effective, efficient, and often elegant. One archetype ideal user recently served as a domain captain. She performed all of the required steps in the process independently and without error, and then proceeded to use features of the eRIDs tool to refine and enhance the feedback both individually and as a package. The overall result was a small, compact set of RIDs that were internally consistent, complementary, and simultaneously met the "one issue per RID" standard while also preserving redundancy information. She synthesized the content to make it clear, concise and compelling.

The process impact of this ideal user was firstly verification that this level of performance is possible, and secondly providing an exemplar [11]. Her work showed that the task could be accomplished within the allotted time. This ideal user created a new level of "best practice" that can now be shared with other users in terms of what she did (example RIDs) and how she did it. We are currently assessing the potential for making changes to the tool and process to facilitate adoption of these best practices by other users.

A second type of ideal user represents a pioneer who sees opportunities for expanding the tool to support other processes. This person has boundary-spanning insights, e.g. [1][2] into how to integrate adjacent processes via shared tools or enhanced interfaces.

The pioneer ideal user generally has experience doing multiple different roles within the review process, resulting in both a breadth and depth of insight into the process. During operations, this type of user continually asks, "how can I do ___ with the tool?" The KMS development team often interacts directly with the pioneer ideal user to develop work-arounds that enable him to make the tool do what he wants – or to make use of other tools when the request truly exceeds the adaptability of eRIDs (e.g., exporting RID information into a spreadsheet that can be used to convert RIDs into action items for individuals on the proposal team). These users are energetic at developing new requirements, and are often willing to serve as beta-testers for future releases. Finally, these types of users make strong advocates for the use of the tool with other users and for obtaining resources from management to improve the tool.

3.7 Summary

All users are not created equal. One useful way to evaluate user needs, beyond simply looking at roles, is to evaluate their knowledge relative to the tools they use and processes they perform. While others have addressed the importance of knowledge, skills, and abilities for general models of individual performance, the approach shown here highlights two important knowledge dimensions that result in different stressors and opportunities for operational KMS.

In the next section, I summarize lessons learned and discuss the significance for both research and practice.

4. Discussion

The profiles of the four user types provide insights into how users with differing levels of knowledge can stress a process or tool – or provide opportunities for greater value. KMS developers can contribute to the overall success of their tools by understanding how emergent needs and behavior affect process performance and user experience. The following recommendations suggest ways in which KMS developers can adapt their tools and processes to increase systems robustness against "breaking a string."

4.1 Adapt test features to operation

We discovered that many of the features implemented to make testing easier also contributed to operational flexibility. For example, as discussed previously, the ability for a highly trusted user (the RF) to impersonate other users proved crucial to enabling smooth operations. Particularly for systems where content and function visibility is role- and perhaps person-dependent, it is critical to be able to view the system from that person's perspective. In addition, it is helpful to be able to move easily between roles. One KMS system we've worked on allows the admin-level user to impersonate any other user – but requires the admin to log out and log back in to switch back. Implemented that way because of stringent access control requirements, the additional logout/in cycles inflict an annoying amount of overhead.

Other test features that have migrated to operations include batch input of data, standard queries to export data, and table-based set-up and configuration management functions.

4.2 Support on-the-fly tailoring

When eRIDs development first started, users said that the review board would be specified in advance and remain fixed over the review. In over two years of operation and multiple review cycles, this has never actually happened. While the structure generally remains stable, membership continually changes so it is important to be able to add, move, change the privileges for, and delete users during the course of the review. Similarly, we get routine requests to add options for pick-list fields and handle changes in the name of the proposals.

While eRIDs provides significant tailoring support, safeguards currently exist to prevent modification of picklists for certain fields. The logic behind this restriction is that once the review is in process, changes in options that have already been employed by users could result in corrupting the RID content. Addressing this "3-string" issue, we've made use of keyword fields (we started with one, evolved to three). However, the need for flexibility currently exceeds the ability of the tool. A future version will implement changes that will enable tailoring options post-start of operations and add error-checking to prevent undesired consequences.

4.3 Incorporate ability to handle non-users

When the focus of the process is the exchange of critical knowledge, particularly under serious time constraints, few KMSs have the luxury of ignoring non-users. If the knowledge is critical and feeds into downstream processes, but the user won't use the tool, then the overall socio-technical system must provide a way for that knowledge to be captured. A key operational goal for our KMS has been to accommodate non-users, but also to try to convert them. It is important to realize, however, that non-users suffer from low levels of both process and tool understanding; therefore it is easy for them to confound process and tool issues. For eRIDs, the evolutionary path generally involves first increasing process knowledge and then increasing tool knowledge because it is easier to get users to adopt a tool when they understand the process it facilitates.

4.4 Err on the side of broader participation

One benefit to the review process as supported by eRIDs is that the process makes it easy for individuals to contribute across disciplines. Research into psychological safety indicates that team members are less likely to "speak up" at meetings if

they perceive a personal risk in doing so [8]. By removing the participant from a "meeting" situation, users perceive less risk speaking up about areas outside their normal comfort zone.

Conversely, feedback from reviewers outside of their assigned domain can be much noisier. The compromise solution which encourages speaking up while minimizing extraneous noise has been to allow users to provide feedback in any area (tool flexibility) while requiring that such feedback to be filtered through their domain captains (tool restriction) who can then re-route the feedback as appropriate (process flexibility). The tool therefore restricts the actions of one type of user, but the process is flexible and allows for that feedback to be submitted through an alternate path after additional review (tool-based restriction but process flexibility.

Supporting a diverse user base can be challenging, but the different perspectives associated with these users can quickly shine a spotlight on a problem or opportunity within the tool or process. For example, providing support to external review participants who could not be given access to the tool due to security restrictions resulted in refinement of the batch input and output tool functions and required modification of the review process itself.

4.5 Actively seek out exemplars

Exemplars contribute in two ways – by growing best practices and by providing clear and compelling examples [11]. No matter how well crafted the developer makes their illustrations and examples, these rarely match the richness and power of content generated by insightful users addressing real issues during an actual review. Real content, even when sanitized, is powerful and speaks well to other users.

Exemplars also enable the system help and training to move from a prescriptive mode (the user *should do* this) to a descriptive mode (this user was able to *create this by doing that*). We have found that users respond much more positively to descriptive recommendations than prescriptive ones.

It often takes effort to find exemplars – akin to hunting for truffles. One has to poke into the details of the review to find strong examples and the users who have created them. These users are comfortable with the system and the process and therefore don't generally require help. Therefore, they tend to be less visible than users requiring greater levels of support.

4.6 Partner with Pioneers

Individuals routinely mediate the interface between one process and another. Where one draws the location of a boundary can have significant impact on performance of both processes and the tools that support them. Pioneer type users work those boundaries and develop special insights into ways in which the larger context can be integrated.

These users routinely break strings because of their boundary spanning roles. If the needs of pioneers are not addressed by the KMS, they can quickly become agents for change. The nature of that change can be either positive (evolution of requirements) or negative (abandoning the KMS for a different tool or approach). It is therefore critical to partner with the pioneer users for the continued success of the KMS.

4.7 Provide tool and process training wheels

Finally, until the socio-technical system reaches equilibrium, it is helpful to provide the equivalent of training wheels to facilitate the joint evolution of tool and process. We accomplished this for eRIDs by filling the Review Facilitator role with a person who has depth of experience in the process, deep understanding of the tool and how it works, and a vested interest in the success of both the tool and the process.

Given an intelligent and entrepreneurial user base, the eRIDs users can be counted on to develop effective work-arounds to any problems encountered during operations. Relying solely on the users however, exposes those users to many annoyances. Over time, those annoyances can erode confidence and satisfaction with the tool.

The types of skills needed to perform the RF role on the latest round of reviews ranged from the mundane (operating the system during review board meetings; uploading batched input from external users), to the social (distilling board member discussion to capture content in RIDs), to the entrepreneurial (devising processes on the fly, adapting fields intended for one purpose to support a new one, circumventing configuration control mechanisms). Many strings broke during the review and it required a depth and breadth of process and tool understanding to keep the orchestra playing.

One challenge for the future of eRIDs is to ease the knowledge and skill requirements for the RF role to increase the pool of people capable of providing that support. Based on recent experiences, a number of tool and process changes are required.

5. Future Research

This practitioner case study describes a successful KMS for an emergent knowledge process, and demonstrates how different levels of user process and tool knowledge can impact overall performance. Using the metaphor of a "3-stringed violin" this case suggests that KMS development should incorporate features and practices that support operational flexibility and address the time between when a new operational need is discovered and when a new capability to meet that need is delivered. It is during this "3-stringed" time that KMS are vulnerable to losing user support.

The general applicability of these recommendations, however, is limited by the specific domain, type of process, types of users, and the emergent behavior of the system that may differ significantly for other KMS [17][22]. This paper, therefore, identifies several avenues for future investigation by KMS researchers. Research questions exist for each of the recommendations in Section 4, for example:

- What types of operational flexibility are required for KMS (beyond those of general Information Systems)? What mechanisms contribute to flexibility? How does one trade flexibility against other system goals?
- What are the ethical and practical issues of trading capability and access against trust?
- What are the threats to the integrity of the KMS and its content that occur when on-the-fly tailoring occurs? What types of safeguards should be used under what conditions?
- How can one measure the impact of non-users on KMS performance? What factors facilitate or hinder the evolution of non-users into other user categories.
- How can KMS development account for the impact of different user perspectives on KMS performance? What factors determine the optimal level of user diversity to support reaching steady state? to encourage innovation?
- What constitutes exemplary performance? Are there common characteristics shared by exemplars? How can organizations encourage the development of these characteristics?
- How do user attributes (e.g., skill, experience) affect preferences for descriptive vs. prescriptive tool and process information?
- How important are "training wheels" to overall KMS success? Do different types of KMS require different types of training wheels? What factors impact the process of removing the training wheels?
- How can the PSP Model [15] be applied to provide greater insight into KMS (vs. general information system) evolution?

As KMS become more embedded in and attempt to address the needs of emergent knowledge processes, both the research and practice communities will need to address the issues raised above. This paper was intended to shed light on an area of practical importance, and contribute to the continuing evolution of KMS research by identifying practice-driven questions.

The PSP model [15] offers a mechanism for understanding how and why innovation occurs in socio-technical systems. Incremental innovation occurs when the components of the system can be adjusted to restore equilibrium. Punctuated innovation occurs when perturbations exceed the ability of the system to absorb them. The "breaking of a string" metaphor equates to a perturbation that occurs specifically when a gap develops between a user and the ability of the tool (the KMS) to support that user's tasks. The resiliency of the system to provide work-arounds to temporarily bridge these gaps will significantly impact user acceptance and perceptions of performance.

KMS resiliency, therefore, is a system characteristic that could regulate the intensity of innovation required to support emergent requirements. The ability of users to "make it work" within the confines of the existing system, rather than requiring major perturbations to process, task, user roles, and technology should contribute to less disruptive system changes. Thus, we need a better understanding of how to build resiliency into information systems in general, and KMS in specific.

This paper offers several practical suggestions on how to improve KMS resiliency based on experiences with the eRIDs KMS. How these suggestions translate to other domains – or if they do indeed result in the desired ability to preserve KMS success and socio-technical system balance – are open questions.

6. Acknowledgements

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The author would like to express her gratitude to the eRIDs implementation team: Tu-Anh Phan, Cara Cheung, Boris Oks, and Eric Rameriz.

7. References

[1] Allen, T. J. *Managing the flow of technology: Technology transfer and the dissemination of technological information within the R&D organization*. Cambridge, MA: MIT Press, 1977.

[2] Ancona D. G., & Caldwell D. F. "Bridging the boundary: External activity and performance in organizational teams." *Administrative Science Quarterly, 37*(4), 1992, 634-665.

[3] W.R. Ashby. *An Introduction to Cybernetics*. London: Chapman & Hall, 1956.

[4] Boehm, B. "A spiral model of software development and enhancement." *ACM Sigsoft Software Engineering Notes, 11*(4), 1986, pp. 14-24.

[5] S. L. Brown, K. M. Eisenhardt. "Product development: Past research, present findings, and future directions." *Academy of Management Review. 20*(2), 1995, pp. 343-378.

[6] Cooper, L.P. The Value of Weak vs. Strong Ties between Individuals and Projects for New Product Review. *Proceedings of Hawaii International Conference on System Sciences – 41*. Waikoloa, HI, 2007.

[7] J.D. Frame, *The New Project Management: Tools for an Age of Rapid Change*. Jossey-Bass: San Franscisco, 1994.

[8] Denert, J. R., and Edmondson, A. C. "Everyday failures in organizational learning: Explaining the high threshold for speaking up at work." *Harvard Business School Working paper, 06-024*, 2005.

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

[10] K.R. Hammond, L.O.Harvey, R. Hastie. "Making Better Use of Scientific Knowledge: Separating Truth from Justice." In T. Connolly, H.R. Arkes, and K.R.Hammond (Eds.), *Judgment and Decision Making*. Cambridge University Press: New York, 2000.

[11] Gilbert, Thomas F. *Human Competence: Engineering Worthy Performance, Tribute Edition*. John Wiley & Sons/Pfeiffer: San Francisco, 2007.

[12] Griffin, A. "Product development cycle time for business-to-business products." *Industrial Marketing Mangement, 31*, 2002, p. 291-304.

[13] Kogut, B. and Zander, U. "Knowledge of the firm, combinative capabilities, and the replication of technology." *Organization Science. 3*(3), 1992, pp. 383-397.

[14] Leavitt, H.J., "Applied organization change in industry: Structural, technical and human approaches." In: *New Perspectives in Organizational Research*. Chichester: Wiley, pp. 55-71.

[15] Lyytinen, K. and Newman, M. "Punctuated equilibrium, process models and information system development and change: Towards a Socio-technical process analysis." Case Western Reserve University, USA. *Sprouts: Working Papers on Information Systems, 6*(1), 2006. http://sprouts.aisnet.org/6-1.

[16] Markus, M.L., Gasser, L. and Majchrzak, A. "A design theory for systems that support emergent knowledge processes." *MIS Quarterly. 26*(3), 2002, pp. 179-212

[17] Maxwell, J. A. *Qualitative Research Design: An Interactive Approach*. Thousand Oaks, CA: Sage; 1996. (Applied Social Research Methods Series; 41).

[18] Mikkelson, B. and Mikkelson, D. P., "Three Strings and You're Outre" http://www.snopes.com/music/artists/perlman.asp. Accessed 14 June 2009.

[19] Mumford, E. "Participative systems design-structure and method." *Systems Objectives Solutions, 1*(1) 1981, pp. 5-19.

[20] Sater-Black, K. and Iversen, N. "How to conduct a design review." *Mechanical Engineering – CIME, 116*(3), 1994, pp. 89-92.

[21] Trist E. and Bamforth, K. "Some social and psychological consequences of the longwall method of coal getting." *Human Relations, 4*, 1951, pp.3-38.

[22] Yin, R. K. *Case Study Research: Design and Methods*. 2nd ed. Newbury Park: Sage; 1994.