ON INFRASTRUCTURES FOR RESOLVING NOVELTY IN PRODUCT DEVELOPMENT: A VIEW FROM THE FAST PACED WORLD OF IMAGING AND PRINTING

by

Steven M. Gilbert
Bachelor of Mechanical Engineering
Georgia Institute of Technology, 1997

Submitted to the System Design & Management Program in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

at the

Massachusetts Institute of Technology

June 2004

©2004 Steven M. Gilbert. All rights reserved. The author grants to MIT permission to reproduce, and to distribute publicly, paper and electronic copies of this thesis document in whole or in part.

Signature of Author

Steven M. Gilbert
System Design & Management Program

Certified by

Paul Carlile
Thesis Supervisor
Assistant Professor of Organizational Studies Sloan School of Management

Accepted by

Tomas J. Allen
Co-Director, LFM/SDM
Howard W. Johnson Professor of Management

Accepted by

David Simchi-Levi
Co-Director, LFM/SDM
Professor of Engineering Systems
ON INFRASTRUCTURES FOR RESOLVING NOVELTY IN PRODUCT DEVELOPMENT: A VIEW FROM THE FAST PACED WORLD OF IMAGING AND PRINTING

by Steven M. Gilbert

Submitted to the System Design & Management Program in Partial Fulfillment of the Requirements for the Degree of Master of Science in Engineering and Management

Abstract

Change is an absolutely essential component of product development. However, some changes are too difficult to manage. It is contended that the difficult changes stem from the emergence of novelty. The significance of novelty is that it is not immediately apparent and can be overlooked. It is a common element in disruptive technology, knowledge management, and firefighting research. This work examines the effects of emergent novelty in a complex product development system. In order to do this a framework is developed to categorize potential types of novelty that are encountered. In addition, a unique perspective on the concept of organizational capability is introduced. What makes it unique is the idea that organizational capability is composed of the capacity to do work and the ability of actors to use that capacity. “Organizational infrastructure” is used to speak more concretely about organizational capability. These conceptual models are used to analyze the events of three case studies developed from actual projects in the Imaging & Printing division of Hewlett-Packard. Through the case analysis it is shown there is significant pressures to approach development as if all novelty is understood. However, by doing so almost guarantees problems late in development if latent novelty exists. It is speculated that the addition of excess capacity to the organizational infrastructure will allow for greater novelty detection. This in turn should decrease the complications from resolving issues related to the emergence of novelty.

Thesis Advisor: Paul Carlile
Title: Assistant Professor of Organizational Studies Sloan School of Management
# Table of Contents

Abstract........................................................................................................................... 3  
List of Figures ................................................................................................................. 7  
Acknowledgements ...................................................................................................... 8  
1 Introduction...............................................................................................................11  
2 Literature Review ......................................................................................................12  
2.1 Disruptive Technologies ......................................................................................12  
2.2 Core Rigidities .....................................................................................................16  
2.3 3-T Framework ....................................................................................................17  
2.4 The Knowledge Transformation Cycle ...............................................................19  
2.5 Utilization & Firefighting ....................................................................................21  
2.6 Summary – Where This Research Fits In.............................................................25  
3 Conceptual Template ...............................................................................................27  
3.1 Definitions ............................................................................................................27  
3.2 Proposed Hierarchy of Novelty ...........................................................................30  
3.2.1 Level 0 – “No Novelty” ..................................................................................31  
3.2.2 Level 1 – “Expected Novelty” ........................................................................32  
3.2.3 Level 2 – “Latent Novelty” ............................................................................33  
3.2.4 Level 3 – “White Space Novelty” ...................................................................34  
3.3 Novelty & Capability ...........................................................................................34  
3.3.1 Novelty in Practice .........................................................................................34  
3.3.2 Significance of Capability ...............................................................................35  
4 Company Selection and Empirical Approach ..........................................................38  
4.1 Motivation – Theoretical Questions .....................................................................38  
4.2 Approach ...............................................................................................................39  
4.3 Company Selection ...............................................................................................40  
4.4 Case Introduction ..................................................................................................40  
4.4.1 The First Case – “Triode” ..............................................................................41  
4.4.2 The Second Case – “Failure Mode L” (FML): .............................................42  
4.4.3 The Final Case – “DEW” ..............................................................................42  
4.5 Summary ..............................................................................................................42  
5 Cases .........................................................................................................................44  
5.1 Triode ....................................................................................................................44  
5.1.1 What is Triode? .............................................................................................44  
5.1.2 Triode & Program M .....................................................................................44  
5.1.3 Novelty Encountered in Triode ......................................................................46  
5.2 Failure Mode L ......................................................................................................49  
5.2.1 What is Failure Mode L? ................................................................................49  
5.2.2 Background ....................................................................................................49  
5.2.3 Failure Mode L & Program M .........................................................................52
### List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>13</td>
</tr>
<tr>
<td>2-2</td>
<td>15</td>
</tr>
<tr>
<td>2-3</td>
<td>18</td>
</tr>
<tr>
<td>2-4</td>
<td>23</td>
</tr>
<tr>
<td>2-5</td>
<td>24</td>
</tr>
<tr>
<td>3-1</td>
<td>31</td>
</tr>
<tr>
<td>7-1</td>
<td>80</td>
</tr>
<tr>
<td>7-2</td>
<td>84</td>
</tr>
</tbody>
</table>
Acknowledgements

First, I would like to thank and acknowledge the efforts of my Thesis Supervisor, Paul Carlile. He taught my very first class at MIT and immediately sparked my imagination. “Organizational Processes” touched all of the real world challenges that motivated me to return to graduate school. The collaborative environment that Paul fostered was essential to the emergence of a topic and completion of this thesis journey. This has been one of the most challenging and rewarding experiences that I’ve ever had. I thank him for his time, patience, and energy.

It is also important to acknowledge the contributions of Ursula Tuffli. Without her enthusiasm and diligence the collaboration with Hewlett-Packard would not have been possible. Even when her schedule was frantic, Ursula made time for this work. I should also recognize the efforts of Thomas Savage for managing the specifics of the collaboration. I also thank George Custer, Jim Martinez, Paula Kanarek, and Betty Coulman for providing sponsorship. Last but not least, I would like to acknowledge the many people that volunteered time for interviews and data collection.

I also acknowledge the influence of some friends over the years. I recognize Casey Miller for his vision. The spirited conversations we used to have about the dynamics of our jobs were both thought provoking and inspiring. Sam Weinstein and Nick McKenna challenged me intellectually as well as interjected some much-needed fun into the graduate school experience. I also need to thank the members of the 2001 Montana
fishing group. The campfire conversation that I had with Butch Eleam, Kirk Krueger, and my father is one of the main reasons that I went back to school.

I thank my parents, Bill and Joanna, who have supported me and given me every opportunity to succeed in life. My mother taught me about perseverance, dependability, and loyalty. My father has always been an inspiration to exceed beyond your boundaries and take on new challenges. I honor them with this accomplishment, which required all the qualities that they have passed on to me.

Finally, I thank my wife Laura, who has willingly made sacrifices and supported us over the last two years. She manages to love the best and worst that I have to offer. We’ve learned a great deal about each other through this experience. This work would not have been possible without her.
1 INTRODUCTION

Prior research has demonstrated that successful firms can be undone by the inability to evolve their expertise to new market demands. Organizations frequently spend millions on research and development (R&D) in an effort to grow and sustain the business. The goal is to generate competitive advantage through technological innovation and/or application breakthroughs that will change the market. However, the same companies are generally not willing to invest in the continued innovation of their development teams. Rather, it is desired to maintain a repeatable structure and process that can be refined and rendered more efficient. A logical argument can be constructed that the teams tasked with innovation (and the organization that must bring the idea to market and sustain it) will ultimately determine the success of the project.
2 LITERATURE REVIEW

2.1 DISRUPTIVE TECHNOLOGIES

Clay Christensen popularized the concept of disruptive technologies, which is an explanation of why dominant companies have been suddenly overtaken in some industries. A disruptive technology has the following attributes:

- It is less expensive than the market leader
- It performs worse based on traditional measures
- It provides an ancillary benefit to a new segment of customers

The leading company (or incumbent) has knowledge and expertise relevant to the pervading design. Development work centers on sustaining incremental improvement in the standard design, which meets the needs of their core customers. The effect is illustrated in Figure 2-1:
Christensen’s argument is that the rate of sustaining improvements will eventually outpace the needs of an increasing percentage of the marketplace. This creates an opportunity for an inferior technology to enter into the low end of the market. In the majority of cases, the incumbent firm ignores the new entrant because the customers that are lost do not add significant value to the bottom line. As time goes on, the incumbent firm will continue to move “up-market” and focus on high-end customers that still desire the sustaining technological advances. At the same time, the inferior (or disruptive) technology is improving in performance and gathering more market share. Eventually
the disruptive technology will meet the needs of the majority of the market. At this point
the incumbent firm has lost competitive advantage because their product is more
expensive and provides a level of performance that is well beyond the needs of the
majority of the market. History indicates that the once this occurs the dominant firm will
be driven out of the market and in some cases it will fail.

An interesting insight from Christensen's work is that even when the incumbent
firm had expertise in the disruptive technology, they were still unable to compete. In
other words, the disruption was not the result of some technical deficiency of the
incumbent. Rather, Christensen points to the organization as the root cause of failure.
The new technologies don’t initially interest the major customers. The processes,
metrics, and award systems of the organization are all focused on supporting the needs of
these core customers and sustaining the technology. Even when the new technology is
strongly advocated in one part of the organization, it is likely that it will ultimately be
rejected by the whole. The incumbent does not have the capability to see the value of the
new technology. This hides the threat and allows time for the new entrant to improve on
the fringes until it is competitive in the major markets. At that point the majority of
incumbents are overwhelmed without a fight.

James Utterback shows similar results in his research on disruptive technologies.
While his definition of disruption is slightly different, the resulting conclusion is very
similar. Utterback defines a disruption as those technologies that displace the core
competencies of the incumbent. A market leading company that has its core
competencies displaced by a disruptive technology will eventually be overwhelmed. The skills developed by the incumbent no longer have value and they are unable to evolve their expertise to adapt to the new requirements of the market. This effect once again is independent as to whether or not the incumbent had knowledge of the threatening innovation.

Another distinction made by Utterback is that disruption can come from Product or Process Innovation. Utterback presents the following Model for innovation that is shown in Figure 2-2:

**Figure 2-2**

The point is made that companies need to be adept to dealing with the shifts in product and process innovation. As products become standardized, innovation on manufacturing
or delivery methods can lead to success. This is evident in Dell’s recent impact on the PC market. In the case of a new product disrupting an established technology, a company may be faced with moving away from process expertise to developing new design capability. Utterback points to the organization’s ability to make these transitions in core competency as a key to survival.

The work of Christensen and Utterback establishes that even the most successful and well-managed companies are highly dependant on their value delivery to the market. Per Christensen, “the capabilities of most organizations are far more specialized and context-specific than most managers are inclined to believe.” (Christensen, 2000) These companies are capable of sustaining changes in the technology and market. However, the majority of organizations find it difficult to recognize the signs that the trajectory of the market is changing. Disruptive technology research makes the case for building the capacity to detect signals of disruption. Once signals are detected the organization must then respond to adapt their capabilities to the new context.

2.2 CORE RIGIDITIES

Dorothy Leonard-Barton’s work gives further insight on why established organizations have difficulty dealing with change. It expands upon the idea of core competencies and introduces the concept of core rigidities. Given time companies will gain efficiency by developing skills in the activities that are essential to meeting the needs of their core customers. This skill set is derived from gaining efficiency in value delivery and can be described as the company’s core competencies. In a stable market, a
company that focuses on and cultivates its core competencies can be successful. However, Leonard-Barton notes that organizations are often unsuccessful when trying to adapt to a new market or innovate outside of their core competencies. There is a great inertia in the organization to resist change in the processes that have brought success in the past. Leonard-Barton describes the difficulty in modifying core competencies as core rigidities. The effect of rigidity appears to be stronger as the threat of competition increases, which is counterproductive.

Christensen, Utterback, and Leonard-Barton all point to the organization as the root cause when companies fail to adapt to evolving markets. Yet, their explanations are insufficient in so far as they “black box” the idea of instituting change. The research cited further emphasizes the need to understand what capabilities are required to successfully deal with change in a product development organization.

2.3 3-T FRAMEWORK

Paul Carlile has described organizations as a collection of boundaries where knowledge/information must be shared for the team to be effective. The challenges that an organization faces are dynamic. As a result, the nature of boundaries is also in flux, which presents different contexts for the movement of knowledge. The 3-T framework (Carlile, 2004), shown in Figure 2-3, provides a means of discussing the different types of boundaries.
The transient nature of contextual knowledge is captured by “novelty”. As the level of novelty increases between two actors, the movement of knowledge across the boundary changes. The first boundary is Syntactic. At this level, a common syntax exists between actors and the status of the boundary is stable. Therefore, it is sufficient to simply transfer information. The next boundary is Semantic. Interpretations and relevancy of knowledge are different on each side of the boundary. In addition, all of the differences and dependences between the actors are not known. A shared meaning or common syntax must be created to communicate effectively. In other words, the knowledge on each side of the boundary must be translated for it to be relevant for the
actors on each side. The Pragmatic boundary is at the highest level. Novelty has risen to
the point where each actor’s knowledge impedes the other. Change will be required to
create a common set of interests. Differences and dependencies across the boundaries
have negative consequences. Knowledge must be transformed to represent the impact of
novelty. In addition, the consequences for each side need to be understood before
making tradeoffs. Carlile’s 3-T Framework directly relates novelty with increasing
knowledge transfer complexity. It provides a language to describe the tension within a
design group to deal with novelty.

2.4 THE KNOWLEDGE TRANSFORMATION CYCLE

Carlile & Rebentisch expanded on the knowledge transformation cycle presented
by Nonaka & Takeuchi. They extend the idea that the context of knowledge is dynamic
by adding that knowledge is also path dependant. The movement of knowledge is
required to integrate information from outside the organization. When the context of
stored knowledge changes prior to retrieval (i.e. novelty is introduced) the value of that
knowledge declines and can also become harmful. This is analogous to Christensen’s
description of customer focus, Utterback’s core processes, and Leonard-Barton’s core
rigidities. As the cycle of knowledge integration is repeated, it becomes less about
integrating knowledge from multiple stakeholders and more about jointly determining
what and whose knowledge is relevant to the current context. Dependence between
groups implies a difference in their knowledge. An increasing amount of novelty also
causes increasing dependence and difference that constrains knowledge transformation.
In addition, novelty disrupts past relationships formed between groups. Knowledge is captured within a firm through its processes and artifacts. Past successes of transforming knowledge within a firm usually live on in the form of these artifacts. An example could be a Product Development Process (PDP). Artifacts are never just syntactical, but also have political and cultural meaning to the organization. Sentiments such as “This is the way we do things around here” would be an example of that cultural attachment. When under duress, such as being threatened by a competitor, a firm may tend to cling to what got them through in the past. It’s a show of faith that what weathered the storm of prior challenges will work under current conditions. The artifacts present a rigidity trap that can impair change. In summary, stored knowledge is a competitive advantage only if it can be re-used in a way that reduces knowledge retrieval, transfer, and transformation costs.

Carlile further illustrates the link between transaction costs of firm knowledge and competitive advantage. First a firm must be able to transform knowledge across its many boundaries to drive innovation and establish a position of competitive advantage. Once that position is established, companies must progressively render those boundaries efficient through knowledge transfer to maintain and profit from it. In other words, they must be able to cultivate core competencies related to the source of their competitive advantage. This in turn, reduces the transaction costs of knowledge retrieval. However, when new market demands emerge (i.e. novelty), the abilities and interest built up in the efficiencies (core competencies and/or core rigidities) impede the firm from repeating the
cycle of knowledge transformation and repositioning itself. The capability to deal with novelty in development groups is in fact linked to maintaining competitive advantage.

Carlile makes the argument that it is equally (or more) important to develop a disruptive organization rather than a disruptive technology. It is imperative to innovate within teams or an organization will not be able to effectively transfer the best ideas to market. In addition, it is proposed that an organization that is capable of evolving its form and function will be better at generating and sustaining competitive advantage. Technology can eventually be copied while it is incredibly difficult to mold an organization in the shape of another.

2.5 UTILIZATION & FIREFIGHTING

Another challenge that a development organization faces is controlling the phenomenon commonly referred to in management literature as “firefighting”. “In product development, firefighting describes the unplanned allocation of developers and other resources to fix problems discovered late in the development cycle.” (Repenning, Goncalves, Black 2001) Firefighting can lead to costly overruns and schedule slips for a single project. In a multi-project organization the effects can be even more widespread. Repenning, Goncalves, & Black sought to explain why firefighting appears to spread in some organizations. The results of their study concluded that development organizations have a tipping point. It represents “a threshold for problem-solving activity that, when crossed, causes firefighting to spread rapidly from a few isolated projects to the entire development system.” (Repenning, Goncalves, & Black 2001) A system dynamics
model was created of a hypothetical development organization to illustrate the tipping point. The assumptions were that the organization designed products on a two-year cycle. Year one activities include upfront planning while year two activities are dedicated to design execution. A project is started every year, so there are always concurrent planning and execution tasks. It was also assumed that the project in the execution phase would be susceptible to firefighting and would take priority. As a result, firefighting will impede upfront work for the concurrent project. Model results are summarized in the phase plot shown in Figure 2-4. The horizontal axis represents the amount of upfront work completed this year. The vertical axis indicates the amount of upfront work that will be completed the following year. In this case, if 60% of the upfront tasks are completed this year then the system will recover and complete 70% of the upfront tasks next year. However, if only 40% is completed, the system degrades and just over 20% of upfront activities will be completed in the following year. Arrows indicate how the system will move to equilibrium on each side of the tipping point. The location of the tipping point is highly sensitive to the organization’s steady-state utilization of resources. As the utilization increases the tipping point moves further to the right.
There are serious implications to the tipping point discovery. First, firefighting can go from a last resort to standard operating procedure where every project will face a last minute crisis. The existence of a tipping point also indicates that even a temporary increase in resource consumption can permanently degrade the performance of the entire organization. While firefighting is an unavoidable and necessary task in development, this research indicates that it should be kept at a minimum. The costs have the potential of being higher than ever expected.
There is additional research that contends that not only is firefighting expensive; it is also ineffective as a project management tool. Blanchard & Fabrycky make this point in Figure 2-5:

Figure 2-5

The ability for managers to make an impact on the outcome of a project starts out very high and decreases over time. What Blanchard & Fabrycky observed was that management activity was out of phase and high only in the later stages of the project. This suggests that a flurry of late activity (or firefighting efforts) is not as effective as it may seem. Even the most successful firefighters are limited by the constraints developed from the beginning of the project.
2.6 SUMMARY – WHERE THIS RESEARCH FITS IN

Christensen and Utterback make a strategic argument that it is essential for organizations to have the capacity to change. They also point out that the majority of companies that have faced a disruptive technology have been unable to make the necessary conversion. Leonard-Barton provides some explanation of why it is difficult. The core competencies that are the source of past successes present a rigidity trap that can keep the organization from changing. Carlile's 3-T framework burrows deeper into the organization and provides a method for discussing how knowledge is moved within teams. It also incorporates the idea of novelty. The introduction of novelty necessitates change in the organization, but it also impedes the knowledge transfer required to make the conversion. Resolving novelty will assist in minimizing the transactions costs of moving knowledge. The capacity to do this is a key component to creating/sustaining competitive advantage. If novelty is not fully resolved, it can emerge late in development and lead to firefighting. The existence of a tipping point illustrates that the impact of firefighting is not isolated. Rather, it effects the entire organization and in the worst case fires will spread.

Novelty is the common element in the literature reviewed above. It has impacts from individual communication to corporate strategy. This work will further explore the concept of novelty and its dynamic effects on product development in a competitive environment. There are varying degrees of novelty. Dealing with them requires varied application of abilities, processes, objects, and tools. The central question is what does
the capability to identify and resolve novelty look like? It is believed that an organization with an emergent organizational infrastructure (one whose structure and abilities adapt to novelty) will be better equipped to respond to changing market context and to generate a competitive advantage.
3 CONCEPTUAL TEMPLATE

3.1 DEFINITIONS

The following key words are defined below for the purpose of this discussion.

Novelty

A change in a complex system that is different or unique when compared to what is originally known. "Novelty" is deliberately used in place of 'uncertainty'. Unlike uncertainty, that which is novel is not immediately recognized as something unknown for the person experiencing it and can easily be seen as irrelevant.”  (Carlile, 2004)

Signals

Events that act to refute the underlying assumptions of a project and serve as an indicator that novelty is present in the system.

Complex

"Having many interrelated elements and interfaces.” (Crawley, 2003) “The concept of 'complex' is a measure of the number and type of interrelationships between elements in a system. Generally speaking, the more complex a system, the more difficult it is to design, build, and use.” (Maier & Rechtin, 2000)

System

A system is defined as "a set of interrelated elements that perform a function, whose functionality is greater than the sum of the parts”. (Crawley, 2003)
**Complex System**

Complex systems have many levels of elements and sub-elements. They also require a great deal of information to understand, specify, and evolve.

**Dependence**

"It is the quality or state of being influenced or determined by or subject to another." (Merriam-Webster, 2003) Dependence between elements or actors in a system implies that there is a difference in their respective abilities.

**Infrastructure**

It is the arrangement of the elements and sub-elements within a complex system that is applied within a temporal context. Elements include the organizational structure, processes, objects, metrics, etc. In other words, all the components of the organization except for the actors involved. The infrastructure in a given context defines the capacity of the system.

**Capacity**

It is the potential of the infrastructure to do “work”. The analogy in physics would be potential energy. As an example, intra-company e-mail is an infrastructure and it provides the potential (capacity) for electronic communication.

**Ability**

It is the potential of the actors to make use of the infrastructure that is in place. Ability is a generic term that represents how skilled the actors are at utilizing the tools, objects, and structure that are in place. It is composed of training, experience, tacit
knowledge, etc. As with infrastructure, ability is not constant and it changes given the context in which it is applied. To continue the previous example, if an employee is properly trained on the e-mail system and knows who to contact for information then that employee has the potential (ability) to make use of the capacity provided by the infrastructure.

**Capability (Organizational Capability)**

It represents the overall potential of the system in a given context. A development organization (or project team) will have a capability that is defined by the system capacity and the collective abilities of the actors. To make use of the previous example, the capability of the system would be represented by the ability of one actor to effectively represent relevant information to another actor via the e-mail infrastructure provided. A mathematical summary of capability could be interpreted as follows:

\[
X (Sys) = \chi (I)^m \cdot \alpha (A)^n
\]

The Capability (X) of the system is equal to the Capacity (\(\chi\)) of the Infrastructure (or collection of “m” infrastructure) times the Ability (\(\alpha\)) of the actor (or “n” actors) that are accessing the capacity.

**Organizational Infrastructure**

It is the collective infrastructure(s) of the organization in conjunction with the multiple effects of the abilities of all the actors. When an Organizational Infrastructure is applied in a given context, it will have a specific capability.
3.2 PROPOSED HIERARCHY OF NOVELTY

The framework illustrated in Figure 3-1 is presented as a means for categorizing the varying degrees of novelty in development. It will simplify the discussion and allow focus on particularly problematic areas. The framework is analogous to Carlile’s 3-T Model (Carlile, 2004) and proposes that there are four general levels that mark the span of potential novelty in development. An explanation of the four levels and relevant similarities to the 3-T Model is given below.
3.2.1 Level 0 – “No Novelty”

The system (or the relevant knowledge required to understand the system) is unchanged. All dependencies are understood as well as the organizational processes related to the system. In this case, adapting to the “new” system is simply transferring and reusing what has been done before. The 3-T model would call this a syntactic
boundary. All prior processes and knowledge are relevant. Therefore the existing infrastructure, capacity, and ability of the organization are sufficient.

3.2.2 Level 1 – “Expected Novelty”

The system is to be altered and novelty is anticipated or expected. In addition, the organizational structures and processes required to resolve the novelty are in place. However, they may require modification and/or new actors may need to be involved to deal with the change to the system. In the context of the 3-T Model, it would be described as the boundary where the altered system dependency is known. Activities revolve around translating the relevant portion of the old process to fit the context of the new system. In other words, the infrastructure offers sufficient capacity for able actors to explore and understand the novelty.

An example of expected novelty could be the following. The thickness of a key component of a widget is to be reduced by half. Prior processing experience with the component has suggested that improper material handling can lead to system failures. Therefore, it is expected that material-handling issues may come up with the reduced thickness. Resources and material handling expertise are applied to investigate potential problems and develop a solution. In the majority of cases, the resolution of expected novelty will lead to the creation of knowledge that is an extension of what is already known. These incremental advances add strength to the organization’s capability as long as the infrastructure remains constant. If the context is altered, a strong capability can become rigid and difficult to adapt.
3.2.3 Level 2 – “Latent Novelty”

The system is to be altered and potential areas of novelty are not detected or expected to be in other locations. In addition, the current organizational structure and processes required to resolve the latent novelty are not in place. The 3-T Model refers to the embedded nature of knowledge and points out that at the pragmatic level, “prior knowledge can act as a barrier to as well as a source of innovation”. (Carlile, 2004) Processes will have to be modified and/or created in addition to the introduction of new actors to deal with the new state of the system. In other words, the system capability must be altered. The urge to reuse prior knowledge poses a trap when latent novelty is present. What was known about the old system may no longer be relevant. This can lead to the misapplication of resources and late discovery of critical issues of the system (i.e. firefighting). The causes being that no resources were allocated to determine the presence of latent novelty.

Most examples arise from underestimating the technical challenges or dependencies of a change in the system. The prior example used in the description of Level 1 expected novelty is now extended to incorporate Level 2 latent novelty. Assume that the thickness change does in fact lead to system failures and work is focused on seeking root cause in material handling. Time passes and while lots of experiments have been completed, no signal has been found within handling. Late in the project it is discovered that thermal effects in the system are potentially causing the half-thick widget component to fail. Resources are redirected to understand the new failure mode and find
the solution space. It is a significant point that latent novelty is never immediately apparent. However, signals will begin to appear that latent novelty exists during the development process. These signals, in this example, were unseen or considered irrelevant due to the focus on finding the problem in material handling. This is a concern because latent novelty increases the likelihood of firefighting within the project, which is detrimental to meeting cost and schedule targets. Firefighting also acts to impede permanent modification to organization infrastructure (capability) because activities typically return to the beginning state once the crisis is resolved.

3.2.4 Level 3 – “White Space Novelty”

The final case would be a “white space” design where a system that is completely new to the organization is pursued. An important element of this level is that there is no prior system knowledge that can be leveraged. Significant learning and invention will be required in the technology, organization, and design processes. An infrastructure must be created. Few projects fall purely into this category. The design of the first lunar lander module may be one of the best examples.

3.3 NOVELTY & CAPABILITY

3.3.1 Novelty in Practice

The typical progression is to move down the hierarchy from Level 3 to Level 2 to Level 1 and finally to Level 0. Once a change is understood and relevant knowledge is created, novelty is driven from the system and a temporal point of efficiency is achieved. Manufacturing organizations attempt to operate at Level 0 as much as possible given that
it is the most defined and efficient stage. The repeatable nature of manufacturing allows for the build up of an explicit syntax for almost all tasks. Processes such as the Lean Manufacturing and Six Sigma provide a tool for developing the common syntax to operate at Level 0. A development organization would ideally (from the perspective of efficiency) iterate between Level 1 and Level 0. One of the primary intents of a Product Development Process (PDP) is to outline the assumed areas of novelty and prescribe the processes to follow. The PDP is the process infrastructure that facilitates development. However, some changes will invariably force the organization into a Level 2 situation where prior expertise and existing processes are no longer sufficient. This is an unavoidable circumstance of development work as no design space is boundless. However, to generate competitive advantage, a development organization must able to do of the following:

- Minimize the time to identify latent novelty
- Minimize the transaction cost to resolve the novelty
- Internalize the relevant knowledge related to the latent novelty in order to reduce the probability of it being misidentified in the future.

3.3.2 Significance of Capability

As defined earlier, the capability of an organizational infrastructure is not constant and will be dependent on the context in which it’s applied. It is important to understand that novelty changes the context and alters the capability of the system. The ability of an experienced actor and the capacity of a product development infrastructure may be
diminished by the emergence of novelty. Resource allocation is typically completed at the beginning of a project with a static view of system's capability. The people and the processes must have the ability and capacity to detect and interpret signals that latent novelty may exist. Once this is done, the infrastructure can be modified to test and understand where the novelty is coming from (effectively moving from Level 2 to Level 1). Once the capability has been adapted, the novelty should be within the means of the organization and can be resolved with consequences that are understood.

As financial, schedule, and competitive pressures increase, there is a greater incentive to staff and resource projects as if all novelty is expected (i.e. at Level 1). It is difficult to tradeoff the short-term increased cost versus the reduced potential of firefighting later in the project. Even though firefighting ultimately consumes more, the cost is not as salient as that of adding excess capacity at the beginning of the project. In addition, the more complex a system becomes, the less likely it is that any one actor (or group of actors) will understand all the dependencies and subtle relationships within the system. It also increases the sensitivity of the system capability to novelty. This forces groups to be more reliant on prior knowledge as a means of managing the breadth of complexity. In turn this has the effect of increasing the probability of encountering latent novelty (Level 2). Complex systems also present more confounding variables, which make it difficult to extract signals that prior knowledge is no longer relevant.

Dealing with novelty is an essential component to design. It represents significant opportunities for success and failure. Product development organizations should develop
an infrastructure and actors with the capacity and ability to identify latent novelty as quickly as possible. It is also important to recognize that the capability of the system must be evolved to resolve the inherent novelty. In effect, the organization and processes are adapted to meet the challenges of a specific development project rather applying the same structure to every program (i.e. “print the build” rather than “build to print”). Occurrences of latent novelty and the transaction cost of resolving it should reduce. In addition, the infrastructure is updated from project to project. The hypothesis is that emergent development capability will ultimately translate into competitive advantage.
4 COMPANY SELECTION AND EMPIRICAL APPROACH

4.1 MOTIVATION – THEORETICAL QUESTIONS

A product development organization is a fundamental example of a highly complex system. This is a point that is not immediately apparent when looking at an organizational chart. The interdependencies are more clear when viewed from the perspective of the previously proposed equation of organizational capability \((X[Sys]=\chi[I^m]*\alpha[A^n])\). Different groups must communicate across multiple boundaries while knowledge is constantly being changed, translated, and created to understand the system being developed. This research focuses on the characteristics of innovative product development. In particular, focusing on the processes and activities that teams use to explore and manage the emergence of novelty into the design process.

The hypothesis is that firms that can identify novelty at the different levels and apply the right capabilities to resolve it will reap a competitive advantage. Effectively the organization will make better trade-offs among short-term and long-term demands in a highly complex system. Of particular importance is the ability to recognize that latent novelty is in the system early. Not doing so could lead the project team to assume that all novelty is at Level 1. This masks latent novelty (Level 2) and increases the likelihood of late design changes and costly firefighting efforts. The challenge is in defining how to develop and evolve organizational capability to incorporate novelty.
4.2 APPROACH

The impacts of novelty and the stickiness of knowledge are best discussed within the context of actual experiences. Three cases will be introduced as a means of discussing the proposed novelty levels. The basis of which are the performance of development teams within the Imaging and Printing division of Hewlett-Packard. Case topics were selected from a recent development program and illustrate the impacts of novelty under competitive conditions. The topics emerged from weekly discussions with my liaison within the design organization. It was important that these were recent experiences and the effects were still salient to the actors involved. In addition, the program team was engaged and interested in learning more about factors that may have contributed to these issues.

Data collection came from two primary sources. The first was through documents provided by my liaison in the organization. These records provided some documentation of the evolution of the cases during development. The second source was through contact with the actors directly involved in the cases. Face to face interviews were conducted for the bulk of the information collected. Follow-up phone and e-mail correspondence was used to clarify information during case construction.

The cases as a whole will be used to build an empirical argument for the concepts introduced. While an ideal case is not included, the events described demonstrate how teams are exposed to latent novelty and the difficulty it poses in projects with aggressive cost and schedule targets. In addition, the processes they use will be examined to identify
the common patterns and distinguish among factors that allow/prevent the organization to effectively identify and resolve the novelty encountered.

4.3 COMPANY SELECTION

The reasons for selecting the Imaging and Printing division of Hewlett-Packard are threefold. First, the company and division have demonstrated the capability to innovate in the past. The organization arose to market leadership out of substantial change. Second, it is the opinion of the author that the division is in a highly competitive and evolving market. While the division is still the market leader, the nature of competition appears to be changing. Some competitors appear to be emphasizing cost over technology. The author believes that these conditions provide incentives for the division to reduce design cost, shorten time to market, and reach new market segments. These factors act to increase the severity of encountering latent novelty late in the design process. The third and final reason the division was selected was due to the author’s high level of familiarity with the business. As a former employee, I came into this research with an appreciation for the challenges the organizations faces as well as the acumen required to understand the context of development.

4.4 CASE INTRODUCTION

The three cases included here are all taken from a development program that will be referred to as “Program M”. Project teams are focused on developing a product system that delivers value when installed in a larger system. “Product” will be used to describe the subsystem and “system” will be used to describe the complete value delivery
system. Program M is comprised of several product variations. The common element of each case is the general treatment of novelty as Level 1. Problems arise from the emergence of latent novelty (Level 2) inherent in the system. While the cases detail seemingly different issues, it is hoped that this central point will become clear through the discussion.

Portions of the case discussions have been altered, disguised and/or omitted to protect information that has been deemed proprietary. Every attempt was made to maintain the integrity of the case concepts. However, a conscious effort was made to err on the side of conservatism to insure confidentiality.

4.4.1 The First Case—“Triode”:

The first case is the most straightforward and it will be referred to as “Triode”. The organization had a long history of using a type of material (Diode) from the same supplier. A well-developed Diode infrastructure was in place. Program M passed through the first two (out of four) design milestones with the Diode material. Due to a strong business incentive, the program team agreed to develop a relationship with a new supplier whose material is made from a Triode process. Problems arose when trying to replicate the relationship of the old supplier with the new one. The infrastructure could not be copied because a portion of the capacity remained in with the old supplier and the ability of new supplier was lower. Ultimately the project finishes ahead of time and accomplishes its goals. However, the Triode change made an impact on other
components of the product system and contributed to the latent novelty that emerged as "Failure Mode L" (FML).

4.4.2 The Second Case – “Failure Mode L” (FML):

The most severe of the cases is Failure Mode L. FML is a processing defect that first appeared in Program L, the predecessor to Program M. FML was an active issue for Program L that disappeared without resolution. Early in the development of Program M, FML reemerged as an issue. Several items, including Triode, contributed to the severity of FML. The prior experience in Program L was not helpful and may have been a hindrance to resolution. The design organization was slow to react to FML and ended up dealing with the issue late in the project in a firefighting mode.

4.4.3 The Final Case – “DEW”:

The final case is also the most abstract, but should be clearer after covering Triode and FML. It was proposed that DEW, a new value added test step, be added to Program M. Several alternatives were explored, but two emerged as the most viable. One path, “DOVE”, was technically complex and highly favored. The other path, “CODE”, was technically elegant but required new communication infrastructure to be created. Both options were pursued in parallel for the majority of Program M. CODE was finally selected late in the program schedule.

4.5 SUMMARY

The cases will be covered in the manner of telling a story. Context is of great importance and will emerge from the case background and highlighted details.
Conceptual comments will be included to frame the cases and pull them together. The structure of the cases is mostly consistent, however, variations do occur to highlight the relevant issues of each case. Every effort was made to maintain accuracy between what is presented and what actually occurred. However, some editorial assumptions were necessary to resolve conflicting and/or missing information. In addition, each case is abridged to keep the discussion manageable and within the scope of this document.
5 CASES

5.1 TRIODE

5.1.1 What is Triode?

Triode is a combination of material and organizational changes. A compelling business case exists for adding a new supplier for one of the major outsourced components of the product. Three criteria existed for accommodating the new supplier. First, the component material would have to change from Diode to Triode. Second, the organization would have to adapt from an established supplier relationship to a completely new one. Lastly, the project must be completed within the Program M schedule. The clear benefit of completing Triode is the substantial business case. Risks of making the change lie in the disruption to the product(s) and process(es) that are in design for a change that is ultimately optional.

5.1.2 Triode & Program M

Program M was well underway before the Triode change was considered. Two major design milestones (out of four) had been completed with the Diode material as default. The proposal to make the Triode switch was brought to the program team early in Program Year 2. Program M was selected due to its size, budget, and projected volume, which further improved the business case. Discussion was centered on the favorable business case and a feasibility study was approved with little resistance. Results of the feasibility study confirmed initial assumptions and the project was launched.
Since Program M was already beyond the second design milestone, an aggressive schedule was set for the project team. Adding Triode depended on the assumption that any issues that came up could be worked out quickly. The organization had years of experience of purchasing this material and expected to be able to leverage that experience to a new supplier. For the business case to hold it was necessary to assume that the processes required for Triode would be analogous to Diode. It was also assumed that the material used in the feasibility study would be representative of high volume material.

Making the change to Triode was more complicated than anticipated. Managing communication and logistics with the new supplier proved to be the biggest hurdles to implementation. Critical tooling was delayed when an equipment vendor was impacted by an unavoidable act of nature. Aligning schedules was also a challenge. The design team had to deal with the changing demands of the program while remaining on schedule. At the same time, the supplier was committing to aggressively completing tasks that they had never done before. Several issues also came up around specifications. It was discovered that no spec existed for the containment and transport vehicle that interfaces with the material handling devices. In another case, conflicting specs for found for different components. Issues also arose where a specification was not required with the previous supplier, but one needed to be created for the new supplier to have sufficient information.

Through the diligence and determination of the project team, the Triode change was completed ahead of schedule. Issues were pursued aggressively and resources were
used to resolve changes. While all the objectives of the project were completed, the scope of the tasks may have been too locally focused. An example is the link between Triode material and Failure Model L. Triode material that was made from material from two different Tier 2 suppliers (TA & TB) was tested to detect any difference. One of those checks was the contribution to Failure Mode L. The test completed during Triode implementation detected no difference. However, the FML team later discovered a signal between Triode from TA and TB. The focus on meeting schedule targets may have masked signals of novelty that were passed on to other parts of the organization.

5.1.3 Novelty Encountered in Triode

According to observations from interviews, "the project was approached cavalierly" and it was approved with little discussion or resistance. One engineer expressed disbelief that the project had been approved on a feasibility study alone and not a qualification. In other words, only a small amount of the Triode material was tested for one product prior to approval. Program M was composed of a family of products that would all be affected by the Triode change. A full qualification would require a large sample of each product type to be tested. These tests would have required time that was not available if Triode was to be implemented on the Program M schedule.

The Triode project was resourced and scoped as having expected novelty (Level 1). Several factors were in play that supports the classification. First, the business case for making the change was so overwhelmingly positive that it made it difficult to present counter-arguments. The program team had the capability to clearly understand the
explicit syntax of the financial numbers. However, it appears that there was little capacity and/or ability within the team to articulate the technical challenges in a similarly clear fashion. Second, not all groups were represented the day that the Triode proposal was presented to the program team. Members of these groups normally attend, but were coincidentally absent the day of the proposal. While there is no guarantee that the discussion would have been different, these groups at least had the potential of raising concerns. From the perspective of the conceptual models presented, the decision to go forward with Triode was made with missing actors and portions of the portions of the project team (infrastructure). Third, Central Assembly (CA) had just successfully adapted its processes to a material change. It is plausible to say that the program team assumed that another material change was well within CA’s capacity. Finally; it was assumed that the knowledge that was built up from years of experience with the old supplier would be easily transferred.

However, the problems encountered in implementation were signals of latent novelty (Level 2). There was a significant amount of capability that had been built up in the previous supplier relationship. It was not sufficient to just copy the same structure because replacing one of the actors (the old supplier in this case) in the system with a lower ability decreases the overall capability. Time and iteration were required to adapt to the new context and build up the ability of the new supplier.

The team appeared to quickly recognize the signals of latent novelty. Part of this can be attributed to the aggressive schedule and the project leader’s sensitivity to any
issue that was unexpected. As problems arose they discussed within the project team to try and build understanding. If the issue was within their scope they developed a plan to resolve it. If not, they promptly escalated a request for resources. It is a credit to the team that they were able to complete the implementation ahead of schedule and the project appears to be a localized success. However, issues deemed outside of Triode scope (FML for example) were passed on to other parts of the organization.

Conceptually, the benefits of the Triode project were apparent. However, the full effects (costs) of making the transition were not known. The program team did not have the capability to represent the technical impacts as a tradeoff to the understood gains. As a result, latent novelty (Level 2) in the project was disregarded and treated as expected (Level 1). One latent issue that was not resolved was the contribution of Triode to Failure Mode L (FML). The Triode change did not cause FML, but it did exacerbate the difficulties experienced. A signal between Triode Tier 2 suppliers and the occurrence of FML went undiscovered for several months. In addition, as the Triode team was working through logistical challenges and specifications (i.e. making the transition from Level 2 to Level 1), the rest of the program team had scant access to any Triode material. As a result, the discovery of Triode related issues in the overall system was delayed which increases their severity.
5.2 FAILURE MODE L

5.2.1 What is Failure Mode L?

Failure Mode L (FML) is technically defined as the variation of material topography on the finished product. If the variation is high enough, it can interfere with product manufacturing and testing, as well as the maintenance of the product in the completed system. FML interference can be linked to multiple system performance issues and failures. These failures can lead to the following major program problems:

- Testers are unable to assess the quality of a product
- Manufacturing is unable to construct a good product
- The end user is exposed to latent failures in the system

5.2.2 Background

Failure Mode L has been present to some extent since the early days of the division’s products. Over the years, increases in performance and quality requirements as well as pressures to reduce product form factor and cost have increased the system’s sensitivity to topography variation. While FML was known about, it was never explored because it did not make a significant contribution to any failures. This changed during Program L, which was the predecessor to Program M. System level tests were experiencing failures due to product maintenance issues. Investigation uncovered that the maintenance area in the system was being impaired (and in some cases damaged) by high material around the portion of the product to be maintained. Not surprisingly, competitive pressures exacerbated this problem. Maintenance area designs were pushing
performance limitations to cut overall system cost. In addition, Program L products were targeting a broader spectrum of users. The same product would be applied in a high performance (premium) system as well as more economical versions.

It is unknown as to when FML failures were first discovered in Program L. However, it is known that work did not begin to reduce FML in Final Assembly (FA) and Central Assembly (CA) until late in the program schedule. Failure Mode L was one of many issues being dealt with at the time. A FA Process Engineer did a lot of the initial investigation in FML. His efforts were focused on how to effectively measure material variation and to develop a spec to screen out “bad” parts. Several experiments were run trying to detect a signal in FA based on incoming material. No true correlation or causality was ever determined, although a spec was proposed empirically. This was achieved by measuring a sample of parts over a period of four months, but was not tested at the system level. FML was never resolved because it disappeared. Several last minute changes were made to the product and production processes prior to the release milestone. These changes were intended to resolve other issues and apparently some combination of them eliminated FML. Due to a multitude of critical issues and a high utilization of resources, no investigation was completed to find the switch that turned off FML.

Two main artifacts of the troubleshooting work done during Program L were codified. First was an entry in the product Quality Assurance Checklist (QAC) that stated, “Failure Mode L is caused by a defect in [name omitted] process in FA”. This
was a popular hypothesis, but one that was never correlated to root cause. It also implies that the solution space for resolving FML is in the process. The second was a “spec” for maximum material variation in Final Assembly. “Spec” carries a strong meaning. Typically, a FA spec has a capable & calibrated gauge, a requirement for incoming material condition, and a response plan for parts that have been determined to be out-of-spec. The spec proposed in Program L had none of these parameters defined. This is an important point regarding knowledge creation and reuse. The artifacts represent knowledge that was created by the organization. However, the codification of this knowledge was not complete and does not fully represent the context under which the knowledge was created. Therefore, as the actors change the tacit knowledge of the context is no longer captured and the value of knowledge is diminished. It also makes the misapplication of the knowledge more likely. For example, if one takes the Program L QAC entry literally, it will be assumed that controlling the specific FA process will control FML until that information is refuted (a Level 1 novelty assumption). However, if it is also known that the entry in the QAC was only a hypothesis, then it is likely that parallel paths will be pursued to test the dependencies of the new system (a potential Level 2 to Level 1 transition process).

The emergence of Failure Mode L fits the model of latent novelty (Level 2) because it was inherent to the system. It took several years for the design space to shrink to the point where material variation could lead to failures in maintenance. It is appropriate that the organization never investigated FML prior to Program L. However,
it appears that little was learned and captured during those initial troubleshooting efforts. It could be rationalized that there were too many issues and too few people to deal with them prior to program release. Crisis efforts were focused on meeting the releases milestone. However, resources could have been allocated to understanding the increasing interaction (dependencies) between product topography and system maintenance after release. This was not done and the inherent latent novelty was left unresolved. As a result, the potential for FML to reemerge in future programs was created. This is exactly what happened in Program M.

5.2.3 Failure Mode L & Program M

5.2.3.1 Phase 1 - Discovery

As with Program L, it is not known for certain when the first products with FML were discovered. It appears that the first rumblings regarding FML coincided with the transition to a new Central Assembly process. Operators, who had prior experience with the Program L products, were identifying FML on Program M prototypes. They were passing that information back to the FA process-engineering group. There were some scattered general discussions in the process engineering area about FML. A Senior Member of Technical Staff (SMTS) organized an informal meeting with the intent of gathering what was known at the time. The group agreed that it was an issue and a small team was put together to look into FML.

Over the course of several weeks, it does not appear that the team was able gain any traction on the issue. There were several factors that contributed to their inability to
make progress. First, the process area was pursuing the issue but the novelty was at the system level (a higher level of dependence). Without a spec that was verified at the system level it was difficult to define how much material variation could be tolerated. In addition, it was difficult to scope the work required on the process. Maintenance area designers were not engaged as they were pursuing several design variations. The threat of FML was not salient to them because their own design had not stabilized enough for them to appreciate the potential impact. The second major factor was the high workload of the process engineering area. With no external pressure to eliminate topography variation and with other issues looming large, FML slipped to the back burner. The SMTS reflected on the situation by saying; “Operating at a high bandwidth can make you less curious.” After completing the interviews, it appeared that following modification seemed appropriate. Operating at a high bandwidth without the means of understanding a new problem (i.e. a system level spec) makes you less curious.

In the discovery phase, a signal was recognized that the latent novelty had reappeared. That signal was identified at a local level and an attempt was made to seek, understand, and resolve what was causing FML. However, the team did not have access to all the stakeholders and it did not have the motivation to pull in the resources it needed. In this case, an internal or external customer that felt the issue was urgent would supply the motivation. At this point Failure Mode L had presented two distinct signals that were recognized. The first was late in Program L and the second early in Program M. While it was understood that FML posed a threat, it does not appear that anyone was able to
articulate it and represent that information to other parts of the program team. There organizational infrastructure lacked the capability to communicate the issues. As a result, the latent issue is passed over for problems that are more urgent and/or better understood. It is speculated that two types of excess infrastructure capacity could have given the team the capability to understand the novelty. The first would be resources from Maintenance Area Design and FA Process engineering to develop a system level spec. The second could be in the form of an experienced actor with the ability to understand the process and the maintenance area.

5.2.3.2 Phase 2 – Escalation

The next phase of Failure Mode L was driven by the Triode project, which was discussed in the previous case. The lead for Triode implementation was a process engineer in CA as well as a resource of the FML team. Failure Mode L was apparent and potentially more severe in initial builds completed with Triode. It was felt that the schedule for Triode was extremely tight and that troubleshooting of FML could not be added to the scope without compromising objectives. The Triode project owner escalated the issue to his management and to the program team. The response was the formation of a new and more formal FML team that would be led by a Product Engineer.

The team was formed in Spring of Program Year 2 and was composed of Product, CA Process, FA Process, and Systems Interaction Engineering (SIE) resources with an extended team member in knowledgeable in maintenance. Objectives were scoped from a process perspective. They were to determine the acceptable variation in material
height, define a gauge to measure it, and prescribe the process to build within the acceptable range. Milestones were identified and the project was to be completed in four months.

The team “threw everything at it” from a process perspective. Multiple process changes were explored in an effort to find a signal. Early results indicated that some modifications had the potential to be a “fix”. A key assumption made by the team was that if the “spec” for Program L material-height was maintained then FML could be controlled. Process improvements were measured by their ability to pull measurements within spec. A key accomplishment at this time was identifying a gauge that was capable of measuring the variation of material height of manufactured parts. In addition, several factors were ruled out as not having any impact on FML.

Comments have been made that said the team lacked a sense of urgency. Another perspective could be that the team did not have access to all the necessary resources. In the team’s Project Data Sheet (PDS), three major dependencies were listed. The first was material availability to run experiments. Sufficient quantities of Diode were available, but the Triode material was in very short supply as mentioned earlier. As a result, little to no experiments were done with Triode and it was assumed (or hoped) that the information gained from Diode would be applicable. A second dependency was sufficient time on the manufacturing lines to build experiments. This need appears to have been met. The final dependency was stable system test beds, which were not available. As a result, the team depended on visual inspection and measurement to assess
FML. There was a lot of activity focused on FML but the team and its sponsors didn’t recognize that their current trajectory was not going to deliver significant on-time results. This point is troublesome from the perspective that it keeps information regarding the performance of the project localized. On the other hand, the program team may not have been receptive to hearing that a FML solution was not in the near future.

Failure Mode L got increased visibility in the organization at the third major design checkpoint for three reasons. First, product that could be verified as within the Program L spec was failing system level tests. This emphasized that the sensitivity of the system was different than in Program L. Second, this was the first large quantity build with Triode material and results were worse than what had been seen with Diode. The final reason for higher visibility was because it impacted the tests that were required to go through a major checkpoint, which affected the entire development team. One engineer said that, “By August, everyone agreed that FML was a serious issue.” While this appears to be true, the “lack of urgency” around FML persisted. An observer in the technical review checkpoint meeting noted that FML was highlighted as one of the key issues, but no one seemed concerned about it. Along those same lines, the system maintenance area designers did not appear to be adding pressure to the situation. The attitude of the system designers has been described as, “They felt that as long as we had a team working on it...that we would eventually figure it out.” Either they were being patient or were expecting to have to deal with prototypes with FML. Both of the previous explanations are plausible based on observations from the interviews. Since it
was not a surprise, there appeared to be little complaining. In addition, the maintenance area designers were dealing with serious issues of their own. As a result, one of the key stakeholders was playing a passive role at a major boundary.

The start of the Triode project produced another signal that Failure Mode L was a potential issue and that latent novelty (Level 2) was in the system. Again the signal was recognized, but improperly categorized. It seems that the second FML Team was tasked with a project to resolve expected novelty (Level 1) in the process area. This statement is based on the scope being limited to process changes only, which is similar to saying, “we know that it’s a process issue and all we have to do is figure that out”. It was also assumed that the Program L specification would be applicable, but that knowledge was proven to be insufficient after a couple of months of work. No system level testing occurred during that time in an attempt to verify the spec.

Another area of concern is the lack of resources for the FML team. Two of the three major dependencies identified by the team were not met to expectations. That is a problem in itself, but it also key that the team did not escalate any risks to their schedule or results based on the lack of resources. The Triode team did a good job of this. However, the Triode project also starved the FML team of material, which did not escalate. As a result, more time passed without progress against goals. Despite the signals of latent novelty and the problems encountered, the project was still being pursued at Level 1 and not Level 2.
5.2.3.3 Phase 3 – Task Force

By September, pressure to resolve FML was growing. The second FML team had passed its original deadline without resolution. Product testing and assessment were being compromised and the final design milestone was quickly approaching. It was agreed that FML was on the critical path and a Task Force was commissioned to work the issue. Designating a team as a task force communicates to the rest of the organization that their activities are critical and priority has been granted to them by management. As a result, resources are much more readily available than before. A process engineering SMTS and 15-year division veteran was selected to take ownership of the team. It was felt that the issue had escalated to firefighting mode and that the SMTS’ experience would be essential to dealing with the increased pressure and scrutiny.

The task force was still process focused. Due to ever increasing schedule pressure, options for modification to the product design was extremely limited. One engineer commented; “If we knew that process changes wouldn’t work by themselves we could have pursued design changes earlier.” The CA processes had also become much more rigid after the stabilization of a major process conversion and the implementation of Triode. Other issues besides FML were leading to poor product performance, which made it difficult to get feedback on process changes from the system level.

As a response, the team came up with a way to test the system level interaction themselves. Products were soiled and then processed in the maintenance area of the system. After the test it was possible to see what portion of the product that the system
was unable to properly maintain. These results were used to characterize what impact different variations of material lift had on the system. This is a good example of moving up to a higher level and testing the dependencies before diving back down to specific process characterization. It also marks the first action of the FML team to understand the new context (latent novelty) of the system. With the benefit of hindsight, it is can also be said that this experiment could have been completed months earlier. The team was able to propose a specification that had more validity that the one from Program L.

By winter, a spec developed from the maintenance test had been proposed for CA to prevent FML in the final product. A gauge was also in place, but there were several implementation issues. It was estimated that the CA spec was conservative and would fail 6 lots for every 2 failed in FA. The screener in CA was also slow and could not process parts at the required production rate. Program management made the decision not to turn on the screen in CA, as it was not a feasible operating method. The effort was disconnected from implementation because work was dedicated to screening parts and not enough consideration was given to how the process would scale. Final Assembly was tasked with finding a method for resolving for FML.

Another issue that was magnified during the Task Force phase was the capability of CA and FA to communicate effectively. These groups are linked on the organization chart but are in reality quite disconnected. The teams are not co-located and the areas they support are in different buildings. In addition, the interaction between engineers and technicians in CA is different from FA. Technicians have a big role (high ability) in CA
processing and complete the majority of machine related tasks. The engineers in FA have traditionally been more hands-on and personally involved in the machinery. Lastly, few engineers have worked in both CA & FA. Challenges were encountered when attempting to craft experiments to effectively test process changes in both areas. The teams experienced several issues where setup and execution instructions were sufficient in CA, but confusing to FA and vice versa. As a result, these experiments were experiencing delays and in several cases they were not processed correctly. These situations are particularly frustrating for task forces where processing errors add even more pressure and scrutiny. An unfortunate result is that with increasing duress, the teams were more apt to retreat to their respective silos rather that operating at the boundary.

One last observation should be made about the two groups. Utilization of resources was high which contributed to a slow response over the year. Each group agreed that FML was an issue, but no one really wanted more work in their area because it would put everything else at risk. The absence of a system level specification made it even harder to get started. In addition, the high utilization of resources did not allow the teams the capacity to drive the issue on their own. Unconsciously, it seemed that they wanted it to be someone else’s problem. The sentiment that seems to captures the Failure Mode L is “I feel like we could have killed this off a long time ago, but it just didn’t happen.” As with the escalation phase of Failure Mode L, it is felt that excess capacity in the infrastructure could have aided in progress on the issues.
5.2.3.4 Phase 4 – Final Status

It was determined that there was a Failure Mode L signal in incoming Triode material. Two different Tier 2 suppliers provide a raw material of the Triode fabrication process. The result is two types of Triode material, which will be called TA and TB. The Triode implementation team tested TA and TB material and found no difference in the contribution to FML. However, it has been demonstrated over time that instances of FML are much higher in TB than TA. Ceasing the use of TB material improved Final Assembly’s ability to control FML, but does not eliminate the problem.

After a year of work from multiple FML teams, a manageable solution is now in place. A new process was developed in Final Assembly that significantly reduces the material variation that causes Failure Mode L. This new process was installed and initial results are positive. However, the new process reduces the operating margin of other processes on the FA line. As a result, instances of other failure modes have risen. Overall the conditions are workable but could be improved. Efforts to eliminate failures caused by the new process are ongoing.
5.3 DEW

5.3.1 What is DEW?

DEW is a method of compensating for variations in the product fabrication process and product performance. Products are measured and/or tested and then marked as to what characteristics they possess. The system then reads the marking and adapts its performance based on the product(s)' characteristics. Competitive pressures have pushed performance targets much higher. It has also contributed to the use of multiple products per system application. As a result, the sensitivity of the system to product-to-product variation has increased. The impacts were less consistent system output and compromised value delivered to the end-user. Giving the system some information about each product’s characteristics was a way of lowering that sensitivity.

5.3.2 Background

The concept of DEW has been around for about 10 years and was first explored by programs in a different design group than Program M. The other division will be referred to as SIS. Three SIS programs, prior to Program M, explored different DEW techniques and implemented them with varying levels of success.

5.3.3 DEW & Program M

5.3.3.1 Phase 1 – Proposal & Alternative Selection

The proposal to implement a DEW process on Program M came from discussions between the System Designers, Product Designers, and Product Engineering in the spring of Program Year 1. There were differing perspectives as to how important DEW was for the program. One person involved in those discussions said that output accuracy was a
source of complaints and that it was the one of the biggest issues related to the product. This person felt that it was essential to add DEW capability to address the complaints and meet customer expectations. Another individual in the same meeting framed it differently. They referred to DEW as what was viewed as “the easiest area to push for improvement in system output”. In the end it was agreed that a DEW process would be a nice thing to have if possible. A system output improvement project, which included the enabling of DEW, was defined and a Product Engineering manager was tasked with leading the effort. For the record, DEW was listed as a program “want”. In this context, a program “want” is something that should be pursued diligently as if it were on the program schedule. However, a “want” should never take priority over or resources from a program “must”. If the item is capable at the release milestone, it will be included in the product. If not, the “want” will be dropped and product release will continue as scheduled. On the other hand, a program “must” is considered essential and will delay product release if it is not ready.

The DEW team was formed and began work to assess viable alternatives for implementation. Over the course of four months, five options were proposed within the team. Subgroups were tasked with investigating their feasibility. Inputs and ideas were collected from all the major stakeholders within the design organization. In addition, some experiments were conducted on Program L products when possible to gain information on alternatives. Two of the five alternatives that were being explored stood
out from the rest. For the purpose of this discussion the leading alternatives will be referred to as “DOVE” and “CODE”.

DOVE was a technique that would require test tooling to be installed at the end of the Final Assembly line. Product would be tested and the performance characteristics would be measured. That information would then be captured and saved. DOVE seemed to be a strong candidate because a SIS manufacturing line had an operational tool installed. In addition, it was seen the most successful effort of previous SIS programs. While the SIS Product and Process Engineering resources that handled the implementation were not available, a dedicated test tool development team was in place to help. The group that owned the DOVE technology was highly motivated to adapt it to meet Program M’s needs. Having the dedicated test resources in place was viewed as a definite plus. A decision was made during late summer to order long lead-time tooling that would be required for a DOVE on Program M. Investigation into other alternatives was to continue while the tooling was on order.

CODE, the second alternative, was a method of approximating product characteristics based on product metrology collected during fabrication. It was seen as a viable alternative because it was such an elegant solution. No additional tooling or processes would have to be developed or maintained. However, there were significant concerns about being able to manage the data transfer from all of the metrology collection points to the final stage of assembly where products would be marked. A SIS program team had previously proposed doing CODE and that proposal had been rejected.
The challenges of implementing CODE were gaining access to scarce Information Technology (IT) resources and integrating data systems that had no prior interfaces. For CODE to be successful, all of the data systems would have to provide near 100% availability.

By the end of Program Year 1, the DEW team was prepared to select DOVE as the default path forward and sit all other alternatives to the side. However, when this proposal went before management the decision was met with resistance. Management was not prepared to support the decision prior to DOVE being demonstrated on an installed tool. CODE was added back to the program scope. Two teams would pursue parallel paths to present the best case for each alternative. The DOVE tooling was expected to arrive by the beginning of the year. It was hoped that enough information would be present to make a selection by late winter of Program Year 2.

Initial investigation seems to indicate that the program team was approaching DEW as if it had potential of latent novelty (Level 2). Early work was set on exploring what had been done before and testing different alternatives. This work could be used to define the space and narrow down areas of expected novelty (Level 2 to Level 1 transition). However, information came out of the interviews that implied that several assumptions were being made about the alternatives that indicated that the project was operating at Level 1. Although they were never made explicitly, it is contended that the following were the underlying assumptions:
• DOVE had been matured on the SIS program to a level that would be leveragable for Program M

• The dependencies of changing from the context of SIS to Program M were understood

• All the required resources to implement DOVE were accessible to the team

• All the required resources to implement CODE were not fully accessible to the team

From the beginning, DOVE was viewed as the “de facto standard”. DOVE can be considered a superior method over CODE from a technical standpoint because it is measuring the actual product performance and was not using an approximation. However, that was only part of the reason why it was favored. A large portion of it was based on the fact that there was a DOVE installed on a SIS manufacturing line. As one person put it, “there was a definite feeling that it shouldn’t be that difficult because SIS had done it.” In addition, the test tool development group that was a part of the SIS design was committed to developing a Program M solution. They were motivated and spoke highly of the tool’s capability. However, the infrastructure was lacking in that it did not contain the engineers that had implemented the tool on the SIS line. In addition, no activities were planned to test or challenge the validity of the assumptions. It appears that DOVE was unofficially viewed as default once the tooling was ordered. Work continued on other alternatives, but almost as an afterthought. These statements are supported by the team’s willingness to make DOVE default prior to marking a product.
5.3.3.2 Phase 2 – Parallel Paths

It had been estimated that one alternative would be selected by the end of winter of Program Year 2. That selection never occurred because both projects were lagging relative to the program schedule. “Proof of concept” had been demonstrated for the factors used in the approximation for CODE. A lot of work remained, however, to detail implementation of the data management system. The DOVE tool was behind schedule for installation and was not running experiments until the spring. Completing experiments was also a struggle. The tool was not as mature as originally believed and many issues were encountered while integrating the DOVE subsystems. Another set of challenges emerged during integration on the final assembly line. To complicate things further, time for DOVE development and experimentation on the manufacturing line was scarce.

It does not appear that a consensus was ever built for the need for DEW outside of the team. No one seemed to dispute the validity of the concept, but there were a lot of opinions as to whether any of the implementation options would be worth the effort or add value to the customers. An artifact, called “green babies”, was used to sell people on the need for DEW. “Green babies” was a collection of the same output created with variant products. A collection of good output in the center represented products that had been compensated for with DEW. The outer outputs were examples of what could happen without a DEW process. Some felt the bad output in “green babies” were “horrible” and they were convinced that DEW was necessary. Another person said, “I
really couldn’t tell a difference in any of the outputs...and I certainly doubt that most customers would be able to see it.” As pursuit of the parallel paths continued, a fervent debate in the R&D community continued about which method (CODE or DOVE) was better. Concerns were raised that DEW may do more harm than good if the marking was not correct and there was a lot of skepticism regarding the reliability of both options.

In the spring of Program Year 2, the program team leadership acknowledged that both alternatives were not on track. The decision was made to increase the priority of DEW from a “want” to a “must” in an effort to have at least one of the two options ready by the release milestone. It was reasoned that changing the priority would free up required resources as well as increase the focus and urgency of the design teams. The projects continued in parallel and no alternative selection date was scheduled. Work would continue in parallel for almost nine months before a selection would be made.

The DEW project manager, made an interesting comment regarding the differences between SIS and Program M. He said that he pushed for DEW to stay a program “want” for as long as possible. Based on his SIS experience, projects that were labeled as a “must” were routinely cancelled if they were behind schedule late in the program. On the other hand, a “want” was diligently pursued in parallel without the same schedule scrutiny and more often succeeded. He now feels that it was a mistake for him to advocate keeping DEW a “want”. In his opinion, “if the project (in Program M) is not a ‘must’ at the first milestone, then it won’t get done”.

68
It was proposed in the preceding section, that DEW was actually being treated as a Level 1 project even though outward appearances indicated it at Level 2. This position is strengthened by events in the parallel path phase. Both projects (DOVE in particular) struggled to stay on schedule. In addition, there were signals that were emerging that indicated that the original assumptions were not entirely valid.

First, the assumption about the maturity of DOVE did not hold up. The DOVE tool that was in operation on the SIS product line fought with frequent downtime issues. It was described as a “nightmare” by some and “not quite robust” by others. In either case, several iterations were required between tool design, process characterization, and product testing to run experiments. These iterations consumed time that was not available and contributed to schedule slip.

The assumption is at fault in part from the dependence on the test tool development group to represent DOVE maturity. While nothing the tool owners said was incorrect, the context was not appropriate. One person put it best in retrospect by saying; “they see their system boundary only at the tool itself”. In other words, the test tool group alone did not have the knowledge (ability) to represent the integration impacts. From their perspective, a Program M solution was being able to accurately and repeatably measure product performance. However, this is only a portion of what was required. The program needed a DOVE tool that could not only measure the product performance accurately, but also do it at manufacturing speeds with product that had been subjected to several other processes just seconds before. It does not appear that the DEW team tested
the meaning of “maturity” in the planning discussions. As a result, the dependency was embedded in the project and latent novelty (Level 2) emerged as multiple integration issues.

Second, the dependencies of changing from the context of SIS to Program M were not understood. The DOVE was installed into a highly coupled area of the line with processes immediately before and after it, which added more noise compared to the SIS application. The tester was subjected to vibration and thermal effects that had not been experienced before. Program M products also demanded higher accuracy because they could only be marked one time while the SIS products could be reworked if necessary. This meant that the DOVE had to have the same level of availability as the rest of the manufacturing equipment, which was another new requirement. Lastly, it was discovered that some of the subsystems that were necessary to implement the DOVE tool on a Program M manufacturing line were behind schedule and/or controlled by resources external to the team.

Third, the required resources to implement DOVE were not accessible to the team. The scope continued to grow for DOVE as more integration issues were encountered. Some of the scope changes required access to resources outside that were not planned for and out of the control of the team. Unfortunately, access to these resources was limited as they were committed to other priorities. Manufacturing line time, as already mentioned, was also a precious commodity in very limited supply.
It is contended that the events that act to refute the underlying assumptions of the project are signals of latent novelty. Several of these signals emerged during the parallel path phase of DEW. DOVE was dealing with several integration issues and schedule slips. The scope of work required for a fully capable tool was growing. It was also discovered that DOVE was dependant on additional resources, which were not readily available. Even the growing debate in the design community could be viewed as a signal that not everything that was assumed about DEW was completely relevant.

The program team did not have the capability to react to the signals of latent novelty. Rather, efforts were focused on enabling a solution within the program schedule. It may have been more appropriate to revisit the original project scope, objectives, and assumptions before forging ahead. A Level 2 question that could have been asked is, “are we still facing the same problem?” Instead the program team acknowledged the signals by shifting DEW from a program “want” to a “must”. This step is in response to one of the more salient signals that not all the resources are available. However, the rest of the issues seemed to have been unnoticed at the program level. DOVE was still considered to be the front-runner. Groups with Program M, however, were being influenced by the signals. The manufacturing organization, which was wary of DOVE to begin with, was becoming increasingly skeptical. An informal survey of the design organization indicated that DOVE was the technical favorite, but that the majority felt CODE was more feasible for Program M. A DOVE proponent’s response to the survey; “The only way to convince people that it will work is to do it.”
5.3.3.3 Phase 3 – Final Selection

In summer of Program Year 2, an R&D Manager took over the role of DEW project lead. The new leader was tasked with assessing the state of DOVE and CODE and developing a proposal for how to proceed. The CODE project had made progress, but still had significant data management issues that needed to be addressed. The DOVE project had struggled for months but the tooling was finally operational. While some data had been collected about DOVE’s performance, it was very limited. After many reviews with different stakeholders, the proposal ultimately asked to implement both CODE and DOVE on Program M. It was felt that having both would provide a means of check & balance and the best possible system performance for the customer. The management team heard the proposal and they decided that DOVE should no longer be pursued as an alternative. CODE would be the default path and management allocated additional resources to tackle some of the data management challenges. CODE was operational by late fall of Program Year 2 and successfully passed through the manufacturing release milestone. The DOVE tool is still in operation, but is being used only in monitor mode with the intent of collecting data. However, it is getting very little attention due to low priority.

In the end, the official cause of death for DOVE was “schedule risk”. Management felt there were too many outstanding issues to be resolved in the time left before release. The deadline for ordering additional tooling for manufacturing ramp had also passed, which added to schedule woes. Some have speculated that this was not a coincidence.
The schedule woes are the manifestation of the latent novelty that was never directly addressed.

5.3.3.4 Wrap-up

The overall success of DEW depends on the perspective that is taken. It appears that CODE has been successfully implemented and is functional. Opinions differ on its effectiveness. Knowledge about DWE methods has been accumulated by the organization through this effort. However, those gains came at a high cost to Program M from developing a technology “on program”. Multiple resources were dedicated to parallel paths for over a year. Some of the costs were a part of reaching a solution. However, a significant portion of them can be attributed to working under the original unchallenged assumptions even as signals emerged that they may not be valid (Level 1 vs. Level 2). Resources and time consumption were increased to deal with emerging issues. As a result, the project consumed resources that could have been redirected to other efforts.
6 CONCEPTUAL SUMMARY OF CASES

6.1 NOVELTY AT LEVEL 1 VS LEVEL 2

Three common elements can be pulled from the cases. First, novelty was repeatedly assumed to be at Level 1 (expected). When latent novelty was inherent in the system, it manifested as late design changes. Additional resources (people, money, and time) were required to incorporate the necessary changes. Second, making the assumption that all novelty was at Level 1 allowed for the reuse of existing organizational infrastructures. This in effect assumes that sufficient capability exists to deal with potential issues. Competitive pressures to render product development more efficient heavily influence these assumptions. Finally, the teams in Program M lacked the capacity and/or ability to respond to signals of latent novelty. The Level 1 novelty approach was maintained until late in the project. A typical response was to apply resources in the form of increased ability (human resources) and/or schedule slips (time) without modifying the underlying infrastructure. As a result, future organizational capability would not have the benefit of learning from the current context.

6.1.1 Triode

The Triode project contained three key areas where the organizational infrastructures applied resulted in a Level 1 (expected) novelty approach. First was in the approval of the project. The discussion of whether or not to go ahead with Triode implementation was significantly overshadowed by the potential benefits offered by the project. Technical implications were under represented, as the program team did not
have the capability of pulling out these issues. Approving the project to be completed within Program M schedule was significant because it drove the majority of Level 1 activities. Second, copying the infrastructure of the original supplier relationship led to problems with execution and unplanned activities. Holes were found in the old infrastructure that had been filled by the ability of the actors from the original supplier. Time and additional resources were required to add to the infrastructure and to develop the ability of the new supplier. Finally, the technical implications of Triode were not fully resolved during implementation and passed latent novelty in other areas of the organization. While Triode did not cause Failure Mode L, it did exacerbate its magnitude.

6.1.2 Failure Mode L

Failure Mode L is an excellent example of approaching latent novelty (Level 2) as expected (Level 1). There are several observations from the case that support this point. First, the teams tasked with eliminating FML were given a process focus. The organizational infrastructure was applied to find a process solution. However, not enough was known about FML to justify the expected novelty approach. Without a system level specification the infrastructure applied to the problem was incomplete. Despite the effort and ability of the actors involved a solution was not achieved because the capacity to find it was not available. Significant progress was made when the team moved up to the system level to test their understanding of FML (a Level 2 approach). This completed the capability to find a workable process solution. However, the reuse of
process capability to resolve product topography issues may be losing its relevancy. In other words, the organizational infrastructure required to deal with FML may have to change in future programs.

6.1.3 DEW

The sluggish progress of the DEW project stemmed from the central assumptions made at the beginning. It is contended that relying on those assumptions led to the Level 1 approach. In the end, the maturity of DOVE as well as the barriers to implementing CODE was overstated. The organizational infrastructures applied based on the original assumptions did not have the capability to meet all of the DEW objectives. Schedule slips, increased pressure, and more people could not make up for the lack of capability. CODE became a possibility after key elements of the organizational infrastructure (IT resources) were added by management.

6.2 SUMMARY

An interesting point emerged while conducting interviews for the case studies. One manager told me “you can’t plan for everything because it’s not a perfect world.” Her comment is absolutely correct. However, the irony in that statement is that if one approaches all novelty as if it were at Level 1 it is similar to expecting a perfect world. The cases illustrate the costs of attempting to resolve novelty at Level 2 with Level 1 capability. These challenges are non-trivial and extremely difficult to manage in practice.

It is important for product development organizations to enable organizational infrastructures with the capability to deal with novelty. The literature review
demonstrated the link between novelty and strategic organizational issues. It was speculated that product development organizations should take the following steps to generate an advantage in competitive markets. First, the capacity to recognize novelty should be fostered within the development infrastructure. Second, the actors in the organizations should develop the ability to detect signals that latent novelty may be present. Finally, once the emergence of novelty has been recognized, the capability of the development group must be adapted to meet the new context that the novelty represents. The result is a relevant and highly capable organizational infrastructure that is suitable for generating and maintaining competitive advantage in evolving markets.
7 RESULTS, CONCLUSIONS, & RECOMMENDATIONS

7.1 CONTRIBUTIONS OF THIS RESEARCH

7.1.1 Utilization & Firefighting

Repenning, Goncalves, & Black offered three main suggestions to avoid the organizational tipping point. The first is to aggregate resource planning between all projects that have access to the same sources. Crossing over the tipping point is an organizational effect that needs a high level view to be controlled. Second, cancel projects with inadequate concepts as early as possible. Repenning et al call for the resource allocation to be aggressive and stringent on its requirements for project approval. The logic is that a bad project cancelled late has already done its damage. Finally, if a project runs into trouble late in the schedule don’t try to “catch up” until the project plan has been revisited. Before stealing resources from other projects it is important to understand the impact on the organization as a whole to keep firefighting from spreading.

The theme in the recommendations above is to “fire-proof” the organization by doing rigorously doing less. While this addresses the issue of high utilization, it does not answer the question of what to do with the capacity gained by doing less. This thesis suggests that the organization should apply the excess capacity to the early stages of projects. The primary intent is to increase the capability of the organizational infrastructure to identify and resolve latent novelty. In addition, the misapplication of resources at Level 1 (expected novelty) that should be at Level 2 (latent) is minimized.
This in turn should reduce the occurrence of firefighting and move the tipping point in Figure 2.4 further to the left.

**7.1.2 3-T Framework & Knowledge Transformation Cycle**

The hierarchy of novelty used in this work is leveraged off of Carlile’s 3-T Framework. Increasing novelty impedes the movement of knowledge in an organization. Knowledge is also path dependent and adds value based in the context in which it is applied. The emergence of novelty indicates the alteration of context, which affects the relevancy of stored knowledge. Over time the challenge is to determine what knowledge is relevant for the given context.

The concepts of organizational infrastructure and capability were introduced and they describe the potential of the organization to accomplish tasks against objectives. Capability is comprised of the capacity of the infrastructure as well as the ability of the actors in the organization to capitalize on it in a given context. In the end, the effect on organizational infrastructure is analogous to knowledge. Novelty disrupts the context under which the previous capability was developed. Changes in context alter organization’s capacities and abilities. As the product development organization tackles new tasks, it is imperative to determine which organizational infrastructures are relevant to meeting objectives. Figure 3.1 has been updated from the perspective of organizational capability and is shown in Figure 7-1.
At Level 0 the impact of novelty is minimal. As a result there are no changes required in the organizational infrastructure. Novelty starts to play a role at Level 1. Modifications in capacity and ability will be required. However, the changes will sustain what was known about the previous organizational infrastructure. Level 2 marks a significant change in context where capability must be evolved. Capacity and ability are “at stake”. What was previously known impedes resolution. Therefore the
organizational infrastructure must be adapted to have the capability to fit the new context. In the white space of Level 3, no prior organizational infrastructure or capability exist and must be accumulated through experience.

7.1.3 Core Rigidities & Disruptive Technologies

The concepts covered in this research are relevant to the challenges presented by Christensen and Utterback. It is logical to think of “disruptive technologies” placing an organization in a position such that their organizational infrastructure has to be evolved (Level 2). The startling observation in their research is that the majority of leading companies are unable to make the change or even realize that change is required. The emergence of novelty is an essential part of the story. Incumbents disregard market predators because they are unable to understand the changing context. Most do not see what has happened until it is too late. Leonard-Barton labels this effect as a core rigidity trap where the successes of the past act as a barrier to change. The concept of organizational capability, as presented here, provides some insight into the previously “black boxed” view of organizational change.

7.2 CONCLUSIONS

The material presented allows the conclusion to be made that organizational infrastructures that enable the capability to properly identify and resolve novelty have definite value. Novelty is an integral part of product development and will emerge in different forms as contexts change. Competitive pressures provide strong incentives to either disregard novelty or to treat all novelty as within the capability of the existing
organizational infrastructure. However, doing so almost guarantees problems late in development when latent novelty (Level 2) is inherent in the system. The negative effects of latent novelty discovered late in the design are serious and outweigh the benefits of a highly utilized project team.

7.3 RECOMMENDATIONS

"The further we progress in knowledge, the more clearly we can discern the vastness of our ignorance." (Popper, 1994) I feel that the observation from philosopher Karl Popper is appropriate for this discussion. It is like a corollary of Leonard-Barton’s idea of core rigidities in that past successes almost guarantee that the organization will have to change to be successful in the future. An underlying message in this thesis is that product development is more closely linked to what is unknown rather than what is known. Too often organizations focus only on rendering their development process efficient while not allowing the flexibility to change. To restate Popper’s point in the context of this discussion, “the further we progress in organizational capability, the more aware we must be of the necessity to change our organizational infrastructures”. Change, especially when attempted under competitive conditions, is extremely hard. However, little is to be gained and a lot can be lost by an unwillingness to evolve.

It is recommended that the addition of excess capacity in the form of additional actors possessing key “knowledge” and/or shared methods or objects with a greater capacity that allow for greater novelty detection. Of particular importance is the capability to recognize latent novelty. This is a permanent alteration of the organizational
infrastructure as opposed to the late addition of temporary resources, which are typically deployed in firefighting. Excess capacity has similar benefits as that of the chief engineer in the Toyota product development process (Kennedy, 2003). It is a person (or group) that has knowledge of the context of development while not involved directly in execution. This allows them to see the "forest and the trees" and assess the efficacy of development.

The second recommendation is process focused. Key assumptions must be made in the beginning of a development project. It is recommended that the central assumptions be codified explicitly and rigorously challenged for validity. Examples were noted in the cases where initial assumptions went unchallenged and allowed the project to continue working at Level 1 (expected novelty) when it should have moved to Level 2 to deal with latent novelty in the system. Therefore it is important to understand if the assumptions are relevant to the context because they justify the application of organizational infrastructure. It is speculated that treating the assumptions with suspicion will heighten sensitivity to signals of latent novelty. If a key assumption is rejected, the objectives and purpose of the project must be revisited. The relevance of the organizational infrastructure applied must also be assessed and modified if necessary.

The preceding recommendations will require additional resources to be effective. While additional cost may not be desirable at the beginning of the project, it is contended that the savings from avoiding late design changes will outweigh the capacity consumed up front. Substantial research exists that supports this contention. One such example is
from Musa & Ackerman’s research into software development projects. The chart in Figure 7-2 shown below approximates the costs of fixing an error in different phases of development. (Musa & Ackerman, 1989)

**Figure 7-2**

![Chart showing relative cost of correcting an error in different phases of development]

It is contended that detecting and resolving novelty early allows for issues to be fixed earlier in the design. As a result, the costs (resources consumed) of making changes should be relative to the data shown above.

### 7.4 IMPLEMENTATION & FUTURE WORK

There are two mains areas of future work required to implement the recommendations given above. First is the identification of the “key actors” that will serve as the excess capacity for improved novelty detection. If these actors do not exist,
then work should begin to develop individuals that are able to communicate on both sides of major boundaries. Second is to develop the organizational capability to make effective tradeoffs at the beginning of the project. The explicit syntax to understand the additional cost of excess capacity is in place. What is missing is the capability to represent semantic issues such as the potential for latent novelty emergence. The recommendations given in this thesis propose modification to capacity. However, it will take time for the ability of the actors to improve at representing novelty risk.

Andy Grove said, “Only the paranoid survive”. (Grove, 1999) The comment has value in this discussion. No one process will ever be perfect and it is the opinion of the author that management literature spends too much time searching for the infallible solution. Project post-mortem discussions too often shift towards rationalizing why things did not work as planned. The key takeaway from this work is that the capability of organizational infrastructure will vary depending on context. With that point in mind, it is imperative to be paranoid about changes in context (i.e. novelty), which can render capability ineffective. I rephrase Grove’s statement as, “only those with the capability to change will survive.”
8 REFERENCES


