SYM-AM-21-072

EXCERPT FROM THE PROCEEDINGS OF THE EIGHTEENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM

Addressing Software-Based, Platform Interoperability Risks in Defense Systems by Using Distressed Debt Financial Strategies: A Technical Debt Mitigation Concept

May 11–13, 2021

Published: May 10, 2021

Prepared for the Naval Postgraduate School, Monterey, CA 93943.

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF DEFENSE MANAGEMENT
NAVAL POSTGRADUATE SCHOOL

The research presented in this report was supported by the Acquisition Research Program of the Graduate School of Defense Management at the Naval Postgraduate School.

To request defense acquisition research, to become a research sponsor, or to print additional copies of reports, please contact any of the staff listed on the Acquisition Research Program website (www.acquisitionresearch.net).

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF DEFENSE MANAGEMENT
NAVAL POSTGRADUATE SCHOOL

Addressing Software-Based, Platform Interoperability Risks in Defense Systems by Using Distressed Debt Financial Strategies: A Technical Debt Mitigation Concept

Ann Gallenson—is a Lecturer with the Center for Executive Education and the Naval Postgraduate School. She facilitates Navy leaders and their executive teams in devising strategic initiatives to meet changing mission requirements. Her research interests center around intrapreneurship and how new technologies and practices create opportunities for organizational change and transformation. [acgallen@nps.edu]

CAPT Scot Miller—is a Faculty Associate Researcher at the Naval Postgraduate School. He began his active-duty career as a P-3C pilot. He is a proud plank owner of the U.S. Navy's Information Professional Community. [scot.miller@nps.edu]

Susan Higgins—is a Faculty Associate Researcher in Information Sciences at the Naval Postgraduate School, where she has been an advisor to senior military and defense leaders on innovation, culture change, and learning for 20 years. As an active-duty naval officer for the previous 20 years, she helped to transform the Navy's use of technologies in space systems, information sharing, learning, and communications. [shiggins@nps.edu]

Abstract

This concept paper explores an innovative approach to detecting and managing software vulnerabilities in cyber-physical defense systems. Software-based vulnerabilities that hinder or preclude the maintainability and evolvability of combat systems are a pernicious form of technical debt that threaten all cyber-physical systems. The risks associated with technical debt across increasingly interdependent DoD cyber-physical systems will accelerate if left unchecked. Without changes in acquisition and maintenance practices, we can foresee cascading, potentially catastrophic cross-system failures. To illustrate the risk and possible solutions, we focus on the software embedded in combat systems that are subject to ongoing modernization efforts that extend their applicability to evolving operations. Our research revealed that software vulnerabilities in critical combat systems can threaten the reliability and readiness of those systems. These vulnerabilities provide an opportunity for the defense acquisition communities to create a new capability within their organizations, an Acquisition Technical Debt Team (ATDT) to help detect, manage, and mitigate technical debt. We explore risk classification by including interoperability into risk evaluation schemas. We then apply common distressed debt management models to suggest when and how the ATDT might help manage and mitigate technical debt to help rehabilitate an ailing system.

Introduction

Over the past 60 years the Navy's critical combat, command, control, and communications systems have evolved into software-dependent technological ecosystems that combine weapon, communication, and detection systems. These cyber-physical systems rely on the tight coupling of computational and physical processes to create responsive, timely, accurate, and lethal deterrence capabilities. This coupling creates management challenges throughout a system's life cycle. Physical defense systems are designed over multiple years, maintained for decades, and periodically re-engineered to meet evolving operational requirements. Modern software systems rely on adaptive development cycles to meet evolving cybersecurity, technical, operational, and interoperability changes. Legacy software in critical, networked defense systems can pose an increasing risk to the systems and the warfighters who rely on them. Modernization requirements for both the hardware and software often introduce further misalignments and vulnerabilities. Software incompatibilities, diminished systems interoperability, and direct cyberattacks are a few of the risks that system maintainers must

mitigate to ensure combat system readiness. These software vulnerabilities, commonly known as technical debt, arise throughout the life cycle of the system and are mostly due to trade-offs made to meet time, cost, and capability requirements.

Technical debt is a software engineering term that characterizes the design or implementation trade-offs taken to meet short-term business and development requirements that create barriers to future changes, including the maintainability and evolvability of the system; it can make these future changes more costly or impossible (Krutchen, 2019, p. 5). The technical debt landscape depicts the software elements that contribute to evolution and maintenance challenges (see Figure 1). Combat systems interoperability requirements can occur at any time in the software's life cycle and primarily impact the software's internal, invisible, elements. Furthermore, networked warfare systems may be susceptible to software vulnerabilities that occur in one combat component but permeate across the network.

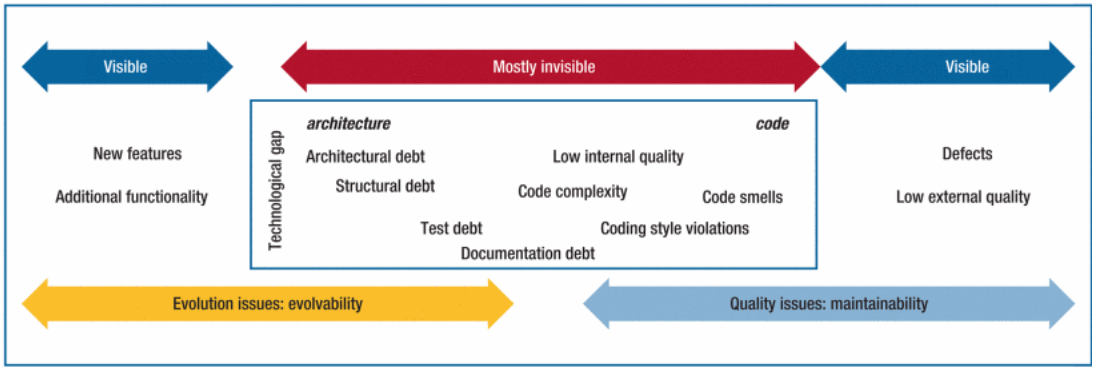

Figure 1: The technical debt landscape depicts the visible and invisible software elements that contribute to evolution and quality challenges.
(Kruchten, 2012, p. 19)

The interdependence of cyber-physical combat systems and their need to function and adapt to changing environmental conditions and operational requirements led us to wonder if the methods used for dealing with technical debt in the private sector were adequate for managing technical debt for weapon systems. Our research surveyed current practices in technical debt detection, classification, and management in the private and public sectors through a literature review. We augmented this overview with discussions with software scientists, combat system engineers, and military leaders regarding current practices and challenges of managing technical debt in combat systems. The research questions were: 1) What methods are being used to detect, assess, monitor, and mitigate technical debt, and realign/re-introduce refactored code into existing systems; and 2) What program management practices can best meet the time and technical demands of maintaining platform reliability while addressing complex, embedded software reliability issues?

This paper is organized into the following sections: 1) a brief review of the impact of poor- quality software and technical debt on the U.S. and world economies; 2) a literature review of technical debt practices and concerns; 3) an overview of the challenges of maintaining weapon systems while managing technical debt; 4) exploration of a technical debt program management using distressed debt practices; and 5) conclusion and next steps.

Technical Debt Impact

The Consortium for Information and Software Quality (CISQ) reported that in 2018 the cost of poor software quality (CPSQ) in the United States was $2.26 trillion and the future (deferred) estimated cost of technical debt was $580 billion. They estimate that $635 billion was spent on legacy systems. The top five global IT failures in the news in 2018 were Wells Fargo Bank, PSA Airlines, Uber Technologies, TSB Bank, and Welsh NHS IT (Krasner, 2019). The CISQ report for 2020, in the midst of a global pandemic, reported the CPSQ in the United States was $2.08 trillion and the future (deferred) estimated cost of technical debt was $1.31 trillion residing in severe defects that need to be addressed in the future. They estimate that $520 billion was spent on legacy systems. The top five global IT failures in the news in 2020 were Home Depot, Iowa Caucus Smartphone app, NASA/Boeing Starliner, Heathrow Airport disruption, and the Google Plus Security glitch. The majority of the costs were aggregated from publicly available source material on the cost of poor software. However, most IT and software organizations do not currently collect cost of software quality data, making these figures an underestimate of the actual costs (Krasner, 2021).

The CISQ 2020 CPSQ report estimated that worldwide cybercrime will cost companies an estimated $6 trillion annually by 2021. They conclude that software vulnerabilities across the IT industries in the United States are creating a fertile environment for cybercrime. The three cost contributors to this exploitable environment are $260 billion spent on unsuccessful projects ($177.5 billion in 2018), $520 billion on legacy system problems ($635 billion in 2018), and $1.56 trillion spent on software failures in operational systems. Given that there could be overlap between legacy problems and operational failures, the report reduces legacy system failures by 50% and concludes with a total estimate of $2.08 trillion in U.S. CPSQ for 2020 (Krasner, 2021, p. 24).

Technical Debt Practices, and Concerns

The results of our research revealed that IT professionals and academics across the public and private sectors are actively working to identify, assess, mitigate, and manage software technical debt. Technical debt vulnerabilities can be introduced at any time in the software life cycle and across the technical debt landscape (see Figure 1). A meta-study of 94 technical debt research studies conducted from 1992 through 2013 identified 10 technical debt types, eight technical debt management activities, and 29 tools for technical debt management (Li, 2015).

The Object Management Group (OMG), an international technology standards consortium, has developed an automated technical debt measurement (ATDM) tool to measure source code reliability, security, performance efficiency, and maintainability and then create a time estimate for remediations of the found weaknesses to help guide corrective actions (OMG, 2017). Early efforts that use machine learning methods to detect technical debt through natural language processing (Rantala, 2020) and forecasting for technical debt evolution through the use of linear Regularization models and non-linear Random Forest regression (Tsoukalas, 2020) are producing encouraging results.

Federal IT professionals report that technical debt directly impacts their mission by limiting the speed with which new functionality can be delivered. The complexity of the code base makes it more difficult and time consuming to modernize and increases the probability that additional technical debt will be injected into software (Curtis, 2018). There is a growing concern that modernizing or developing software capabilities by using machine learning for prediction or other functions, though cost effective, introduces more complicated, hidden, and rapidly accumulating technical debt vulnerabilities into the system. Software engineers at Google, Inc., warn that machine learning integrated into a system's design can result in specific risk factors

that include boundary erosion, entanglement, hidden feedback loops, undeclared consumers, data dependencies, configuration issues, changes in the external world, and system-level anti-patterns (Sculley, 2015).

For the Department of Defense (DoD), technical debt is complicated by a spectrum of requirements including specialized combat systems, extreme and changing environmental conditions, changing adversarial requirements, changing operation plans, longevity expectations, spending requirements, contractual agreements, and legal constraints. Military missions often require an array of commercial off-the-shelf (COTS) components for software and hardware modification. COTS requirements add another layer of complexity to technical debt due to system requirements differing from COTS capabilities. COTS and system mismatches include functional, performance, interoperability, configuration problems across versions, documentation, COTS evolution limits, interface differences, and COTS longevity maps. COTS requirements add additional technical debt detection and sustainment needs (Yang, 2018).

Technical debt can result in vulnerabilities that can be exploited through a cyberattack. Weapon systems are cyber-physical systems of systems that are networked and rely on software integration and interoperability to perform critical functions. As the systems become more software interdependent and complex, the probability of cyberattacks increases. As weapon systems become more lethal, the need for sophisticated and trustworthy cybersecurity systems becomes more imperative (Chaplain, 2018).

Managing Combat Systems and Technical Debt

Our discussions with U.S. Navy leaders, software engineers, weapon systems engineers, and combat system maintainers revealed that they have been managing and working around technical debt issues for decades. Due to the specialized nature of weapons and weapon systems, many of their programs were early adopters of cyber-physical integration and have been in the forefront of developing these capabilities. Continual hardware and software modernization has resulted in complicated middleware processes to ensure interoperability across multiple systems. The need for increased capabilities without modernizing legacy code can create internal complexities and increase the probability of accruing additional technical debt. Sophisticated development and testing capabilities have been devised to maintain reliability and adaptability. Software architecture is a key component to interoperability, usability, and adaptability. Technical debt assessment needs to be broken down by components and interoperability patterns.

Systems routinely undergo risk assessments and are scheduled for maintenance based on a standard risk scale that measures the severity against the likelihood of the risk. Technical debt may accrue because the assessed risk falls below the critical mark. Maintainers need a way to communicate the need for technical debt mitigation to program managers before functional or interoperability risks become critical.

Maintenance and management of complex, integrated, and interoperable weapon systems suffer from technical and organizational constraints. Dealing with complex system interoperability issues across different generations of weapon systems is complicated, time consuming, and expensive. The acquisition requirements for meeting scope, cost, and time parameters are often in conflict with the realities of managing and mitigating legacy system technical debt.

In summary, cyber-physical combat systems represent a special case of technical debt that falls outside of many public and private sector requirements. Their customized builds and configurations, the complexity of the systems, the age of some of their components, their

interoperability requirements, and the criticality of their functions make them more vulnerable to the risks of technical debt. These characteristics also make combat systems good candidates for the creation of a new acquisition capability designed to assist in reducing the defense specific risks associated with technical debt. Processes and practices acquired through this endeavor could save the Navy time and money by minimizing the current and future costs of technical debt while leveraging expertise within the Navy enterprise.

Recommendation: Create a Technical Debt Program Management Capability Using Distressed Debt Practices

Our recommendation argues that mitigating and managing combat systems' technical debt risks during a system's maintenance phase will require specialized and innovative teams within the acquisition community. The goal of an acquisition-based team is to assist combat systems managers and maintainers with the complexities of managing and mitigating technical debt across the component portfolio of their systems. By creating a specialized team that initially focuses on one combat system, program managers and combat systems experts can create and refine sustainable processes for technical debt assessment, reporting, management, mitigation, and reintegration. Ultimately the processes can grow into an acquisition capability that could be applied to systems across the Navy. A technical debt capability in acquisition could help mitigate technical debt at early stages and throughout the life cycle of cyber-physical combat systems.

The U.S. bankruptcy codes and distressed debt management practices present a starting point for developing assessment and mitigation processes. The codes and practices have evolved to evaluate the "type" of debt-related stresses a company is under and methods to best rehabilitate the company or liquidate their assets. The basic goal is to realize that there is value in the larger cyber-physical system that can be rehabilitated if the ailments due to technical debt are mitigated. Creating mitigation strategies based on both the severity of the risk and the interoperability of the system is akin to treating the whole system rather than a single symptom. The initial focus on combat systems includes an inherent portfolio approach that combines integrated and interoperable systems whose dependencies may exhibit technical debt through association. The following is a conceptual framework whose intent is to serve as a starting point for future exploration, creation, and refinement.

Create an Acquisition Technical Debt Team

Create a cross-organizational Acquisition Technical Debt Team (ATDT) with expertise in acquisition processes and an initial focus on a pilot combat system that has been dealing with technical debt due to the interoperability of legacy systems. The team should include members of the combat system's engineering and managerial staff, program managers who have experience with the system, and external experts familiar with the technological and operational needs for the system. The purpose of the team is to develop ways of assessing the health of the combat systems and identifying the processes needed to manage and resolve technical debt issues. The team could initially include external experts versed in assessing, managing, and liquidating distressed debt companies.

Assess and Manage System-Wide Risks Associated with Technical Debt

Given the integration and interoperability requirements of combat systems, we have created a risk cube that places interoperability along the z-axis (see Figure 2). The incorporation of interoperability into the assessment of technical debt risk will help determine the associated risk to external components and associated vulnerabilities that extend across the greater system. The goal of this approach is to create a classification schema, similar to the U.S.

bankruptcy codes, to helps the combat system's resident team determine when to report the risk to the program manager and when to request a transfer of the problem to the ATDT.

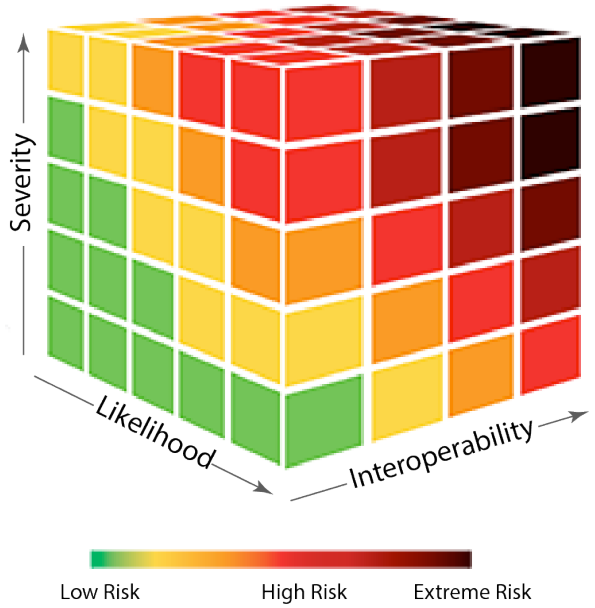

Figure 2: The Inclusion of Interoperability in a System's Risk Assessment Charts the Degree that Technical Debt or Component Failure Effects the Greater System

In practice, the risk cube would be generated using data from the combat system, its repair history, and relevant vendor or industry models. Multiple signals at the ATDT level over a short period of time, operational shifts, fielding of new capabilities for that combat system, etc., could place the system at a level where the ATDT should take over for the program manager until the risks are mitigated.

In Distressed Debt Management (DDM), a distressed business performing "below investment grade" is determined to be either undervalued or about to default on their loans. A company may become distressed because of excessive debt, but other factors that can threaten the company are declining profitability, loss of competitive position, or changing business climate. There are three management strategies that depend on the type of debt the company is dealing with and the time horizon for recovery or liquidation. The three management strategies are as follows (Jain, 2011):

1. **Active Control**—the business process is taken over by the management team with an exit timeline of 2–3 years and a profit target of 15–25% per year.
2. **Active Non-Control**—the management is actively involved with existing management but does not take over the business with an exit strategy of 1–2 years and a profit target of 12–20% per year.
3. **Passive**—the management company invests in the existing company to create financial buffers with an exit strategy of 0.5–1 year, and a profit target of 12–15% per year.

These management strategies can be extrapolated to the type and degree of involvement the ATDT would pursue to manage and mitigate the risks from technical debt on a combat system

and the systems reliant on it. The time horizons and profitability parameters could be used as guides for ATDT involvement and follow-on savings and/or estimated extended use of the combat system.

Using the categorizations from the interoperability risk cube we can classify the type of involvement of the ATDT as follows:

A. **No ATDT Involvement**—if the severity and the likelihood of a vulnerability are high but interoperability is low (HHL), then there is no need to elevate the problem to the ATDT. The risk should be dealt with by the combat systems managers and engineers.

B. **ATDT Passive Involvement**—if the severity, likelihood, and interoperability of a vulnerability are moderate (MMM), the problem should be elevated to the program manager for that combat system, who would monitor the progress and might ask for guidance from the ATDT specialists.

C. **ADTD Active Non-Control**—if the severity is low, the likelihood is moderate, and the interoperability of a vulnerability is high (LMH), the program manager and ATDT should be notified.

D. **ADTD Active/Non-Control** or **Active Control**—if the severity is high, the likelihood is moderate, and the interoperability of the vulnerability is high (HMH) the ATDT might want to monitor the risk, or if there have been several risks reported, they could take over the management and mitigation of the technical debt of the system from the program manager and return the authority when interoperability risks have been mitigated and components replaced or refactored.

E. **ADTD Active Control**—if the severity, the likelihood, and the interoperability of the vulnerability are high (HHH), the ATDT would take over the program management of the system and work in tight coordination with the combat system's management and engineering team. Once the debt has been mitigated and the system has been restabilized (or retired), the management would be returned to the original program management team.

Conclusion and Next Steps

The effects of technical debt on software-based systems are creating a global concern and costing trillions of dollars in recovery and mitigation. The vulnerabilities inherent to complex systems and those reliant on legacy architectures are creating a fertile ground for cyber criminals and other adversaries. The DoD weapon systems are complex cyber-physical systems. These systems have been subject to ongoing requirements to modernize and modify both their physical and software systems to meet changing environmental factors and operational requirements. These modification pressures result in technical debt that their managers and maintainers have been contending with for decades. The risks associated with technical debt across increasingly interdependent DoD cyber-physical systems will accelerate if left unchecked. Without changes in acquisition and maintenance practices we can foresee cascading, potentially catastrophic cross-system failures. We suggest that the acquisition enterprise help manage and mitigate these risks by creating cross-functional teams dedicated to classifying and managing technical debt. We have used distressed debt management practices from the financial sector to seed an acquisition-based conversation.

Suggested next steps:

- Select a combat system that is already engaged in working through technical debt issues to learn existing technical debt evaluation processes and create a risk cube from their data.

- Create a cross-functional team familiar with the combat system and the technical issues that are contributing to vulnerability concerns.
- Evaluate the interoperability risk cube and the technical debt management strategies suggested here to learn the relevance of these models to the combat system's needs.
- Continue this research to refine these concepts with the acquisition community.

References

Curtis, W. (2018, February 23). *Solving the "technical debt" problem*. FCW. https://fcw.com/articles/2018/02/23/comment-technical-debt-curtis.aspx

Chaplain, C. (2018). *Weapon systems cybersecurity: DoD just beginning to grapple with scale of vulnerabilities* (GAO-19-128).

Jain, S. (2011). Investing in distressed debt. *UBS Alternative Investments, June*, *15*.

Krasner, H. (2019). *The cost of poor quality software in the U.S.: A 2018 report* [Technical report]. Consortium for IT Software Quality.

Krasner, H. (2021). *The cost of poor quality software in the U.S.: A 2020 report* [Technical report]. Consortium for IT Software Quality.

Kruchten, P., Nord, R., & Ozkaya, I. (2012). Technical debt: From metaphor to theory and practice. *IEEE Software*, *29*(6), 18–21.

Kruchten, P., Nord, R., & Ozkaya, I. (2019). *Managing technical debt: Reducing friction in software development*. Addison-Wesley Professional.

Li, Z., Avgeriou, P., & Liang, P. (2015). A systematic mapping study on technical debt and its management. *Journal of Systems and Software*, *101*, 193–220.

Object Management Group. (2017, December). *Automated technical debt measure specification* (Version 1.0) [Computer software specifications]. Consortium for IT Software Quality. https://www.omg.org/spec/ATDM

Rantala, L. (2020, October). Towards better technical debt detection with NLP and machine learning methods. In *2020 IEEE/ACM 42nd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)* (pp. 242–245). IEEE.

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., ... Dennison, D. (2015). Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems* (pp. 2503–2511).

Tsoukalas, D., Kehagias, D., Siavvas, M., & Chatzigeorgiou, A. (2020). Technical debt forecasting: An empirical study on open-source repositories. *Journal of Systems and Software, 170,* 110777.

Yang, Y., Wade, J., Alelyani, T., & Stanton, P. (2018). *RT 193: Framework for analyzing versioning and technical debt*. Stevens Institute of Technology.

Acquisition Research Program
Graduate School of Defense Management
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net