

DISSERTATION

APPLICATIONS OF SYSTEMS ENGINEERING TO DISTRIBUTED-LEDGER-BASED
CYBERSECURITY MANAGEMENT

Submitted by

Kent D. Lambert

Department of Systems Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Engineering

Colorado State University

Fort Collins, Colorado

Spring 2026

Doctoral Committee:

Advisor: Tom Bradley

John Michael Borky

Steve Simske

Indrajit Ray

John Santiago

Copyright by Kent Douglas Lambert 2026

All Rights Reserved

ABSTRACT

APPLICATIONS OF SYSTEMS ENGINEERING TO DISTRIBUTED-LEDGER-BASED CYBERSECURITY MANAGEMENT

This research identifies a set of practical solutions to the uncontrolled growth of malicious cross-industry cybersecurity threats. Given the presence of the behaviors and negative effects of cyber threats, the research seeks to understand the effects of three eco-system-wide strategies for cyber defense: Defense-in-Depth (DiD), Zero-Trust Architectures (ZTA), and secure cryptographic key provisioning. These strategies are developed using the BlockFrame, Inc., Eco-Secure Provisioning™ (ESPTM) Framework, blockchain-assisted digital logistics management and governance, and the application of Emergence Theory and Uniformity. The research addresses five specific questions to organize the approach, with practical translations of this research to selected industrial use cases.

ACKNOWLEDGEMENTS

I would like to express my appreciation to my Dissertation Committee for their time, commitment, and guidance to help to add robustness to my work, including:

My dissertation advisor and Chair of the Colorado State University Department of Systems Engineering, Dr. Tom Bradley,

Dr. John Michael Borky, formerly of the Colorado State University Department of Systems Engineering, and the former Director of the United States Air Force Rome Laboratory, whose life-long work to “systematize” Model-Based Systems Engineering and the Model-Based Systems Architecture Process was invaluable to my studies,

Dr. Steve Simske, of the Colorado State University Department of Systems Engineering, for his contributions to my better understanding of cybersecurity,

Dr. Indrajit Ray, of the Colorado State University Department of Electrical Engineering,

Dr. John Santiago, my Practicum Advisor with Industry, formerly of the United States Air Force, and of the Departments of Systems Engineering and Electrical Engineering of the United States Air Force Academy and Colorado Technical University.

Mr. and Mrs. David and Gail Liniger, the benefactors of the Liniger Honor, Service, and Commitment Scholarship, which greatly assisted me in the resourcing of my studies and research.

I would also like to extend my thanks to a few special colleagues:

Dr. Ronald Segal (Maj Gen, USAF, Ret.), my classmate at the USAF Academy, an exemplary leader in the Air Force and NASA, a pioneering force in systems engineering at Colorado State University, and the primary person to motivate me to pursue studies in systems engineering,

Dr. Larry Rainey, my good friend and fellow airman at the United States Air Force Space Command Space Analysis Center, who has been a national champion to advance systems-of-systems thinking, Emergence Theory studies, and many critical national security systems,

Dr. Kristin Giammarco, of the Systems Engineering Department of the United States Navy Naval Postgraduate School, a mentor and fellow-explorer in model-based examinations of Emergence Theory.

Above all, I would like to acknowledge my friend and business partner, Dr. Christopher Gorog, CEO of BlockFrame, Inc., who is perhaps the most creative and brilliant expert and inventor of digital supply chain security technologies, blockchain technologies, cryptographic key delivery, and digital privacy processes in the industry.

Kent Lambert

TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGEMENTS.....	iii
LIST OF KEYWORDS	viii
1.0. Introduction.....	1
1.1. Objective.....	1
1.2. The Cyber Threat Environment	2
1.3. Systems Engineering Background	5
1.4. Cybersecurity Background.....	6
1.5. BlockFrame, Inc., Approach to Current Cybersecurity Gaps	7
1.6. Emergence Theory Background	8
2.0. Proposed Research	11
2.1. Overview.....	11
2.2. Methodology	11
2.3. Modeling Tools.....	11
2.4. Supporting Data	13
2.5. Existing Literature	13
2.6. Funding and Support.....	14
3.0. Research Questions and Tasks.....	15
3.1. Research Question 1 (RQ1)	15
3.2. Research Question 2 (RQ2)	16
3.3. Research Question 3 (RQ3)	17
3.4. Research Question 4 (RQ4)	19
3.5. Research Question 5 (RQ5)	20
4.0. Translational Practicum Context.....	21
4.1. Introduction to Colorado State University Systems Engineering Department	22
4.2. Introduction to BlockFrame, Inc.....	24
4.3. Business Management and Commercial Considerations.....	26
5.0. Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments.....	27
5.1. Introduction.....	29
5.2. Defining Cybersecurity Threats as Emergent Behaviors.....	30
5.3. Controlling Negative Emergence of Malicious Cyber Activity	33
5.4. Ecosystem Solutions	37
5.5. Modeling Examples of Positive and Negative Emergence in Cybersecurity Environments	45
5.6. Quantifying and Prioritizing Cybersecurity Responses.....	55
5.7. Applying Monterey Phoenix to Qualitative Cybersecurity Logic	57

5.8. Summary	58
5.9. Context of Emergence and Complexity Applied to BlockFrame® Design and Development	59
5.10. Conclusions for Research Question 1	64
5.11. References for Chapter 5	65
6.0. Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) to support the design of cybersecurity systems	67
6.1. Practical Applications of MBSE and MBSAP Research	67
6.2. System-of-Systems Research applied to BlockFrame® Distributed-Ledger-Based Logistics and Cybersecurity Management	70
6.2.1. Systems Description.....	72
6.2.2. Operational Viewpoint.....	76
6.2.3. Logical/Functional Viewpoint	88
6.2.4. Physical Viewpoint	100
6.3. Additional Research Applying SysML Modeling Tools	151
6.4. Philos™ Neutral Governance Architecture	152
6.5. Conclusions for Research Question 2	153
7.0. Applications of Cryptologic Token Provisioning and Distributed Ledger Governance for Secure and Trusted Commercialization.	154
7.1. BlockFrame® Functional Architecture.....	156
7.2. Philos™ Marketplace Operations	164
7.3. Neutral Governance Actions.....	175
7.4. Synthesis	194
8.0. Research in Professional and Public Policy Applications	196
8.1. Prior Articles for Context.....	199
8.2. Assessing Cybersecurity Resilience of Distributed Ledger Technology in Energy Sector Using the MITRE ATT&CK® ICS Framework.....	201
8.3. Assessment of the Distributed Ledger Technology for Energy Sector Industrial and Operational Applications Using the MITRE ATT&CK® ICS Matrix.....	211
8.4. Digital Privacy: Technical, Legal, and Legislative Aspects	279
8.5. IEEE Digital Privacy Model©	304
8.6. Bridging the Technologist/ Policymaker Divide in Digital Privacy	307
8.7. Briefing to the Colorado County Clerks and Records Association and Colorado Department of State	311
8.8. IEEE Region 5 Training.....	312
8.9. Participation to revise the IEEE-USA Digital Privacy Policy Statement	313
9.0. Representative Practical Use Cases	314
9.1. Case 1. Distributed Energy Resources.....	317
9.2. Case 2. Military Supply Chain Use Case.....	319
9.3. Case 3. Remote Education Trust and Security.....	325
10.0. Research Contributions	327

10.1. Systematic Research Process	327
11.0. Conclusions.....	330
Appendix A: Monterey Phoenix Script Chain Example.....	332
Appendix B: Philos™ Neutral Governance Architecture Overview and Summary Information.....	337
Appendix C: Modeling Examples using SysML, MBSE, and MBSAP	349
Appendix D: Risk Analyses of Selected Functions of the Philos™ Blockchain/ Distributed Ledger and Ecosystem	413
Glossary	448
Bibliography	452

LIST OF KEYWORDS

Authentication
Blockchain
BlockFrame, Inc.
Colorado State Government
Colorado State University
Cryptography
Cybersecurity
Dassault Cameo Systems Modeler
Defense-in-Depth
Digital Privacy
Digital Privacy Initiative
Digital Privacy Model
Digital Signatures
Digital Supply Chain Logistics
Emergence Theory
Emergent Behavior
Institute of Electrical and Electronic Engineers (IEEE)
Model-Based Systems Architecture Process (MBSAP)
Model-Based Systems Engineering (MBSE)
Monterey Phoenix/Gryphon Model
Naval Postgraduate School (NPS)
Reliability
Security Architecture
Security Metrics
Security Tools
System-of-Systems
Systems
Systems Modeling Language (SysML)
Trust Chains
Uniformity
Zero-Trust Architectures

1.0. Introduction

1.1. Objective

This research proposes to identify a set of practical solutions to the observable but uncontrolled growth of malicious cross-industry cybersecurity threats. It is assumed that severe uncontrolled cyber threats already exist that cause security, privacy, and economic damage to governments, private businesses, and individuals. Given the presence of these behaviors and their negative effects, research will present control options through three eco-system-wide strategies: Defense-in-Depth (DiD), Zero-Trust Architectures (ZTA)¹, and secure cryptographic key provisioning. These strategies are developed using blockchain-assisted digital logistics management and governance, a Model-Based Systems Engineering (MBSE) approach supported by the Model-Based Systems Architecture Process (MBSAP), and the application of Emergence Theory. The BlockFrame, Inc., Eco-Secure Provisioning™ (ESP™) Framework, facilitates secure digital supply chains, cryptographic key provisioning, and compatible Blockchains/Distributed Ledgers (B/DL) being used in novel patent-pending² designs to demonstrate synergistic solutions through indefinitely-scalable, cross-industry system-of-systems applications using modular, loosely-coupled service interfaces, multi-factor identification, and applied uniformity. Research of these novel system-of-systems (SoS) concepts and architectures can be of benefit to systems engineers, designers, project managers, and policy makers seeking eco-system-wide cybersecurity, and organization-wide digital logistics control solutions. Chapter 3 introduces five specific questions that organize the proposed approach to current and future research in the areas: how Emergence Theory can be applied to complex cybersecurity environments, how Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) can be used to support the design of cybersecurity systems, how Eco-Secure Provisioning™ of cryptologic secrets, with distributed ledger-assisted management, can provide new commercial services, better cybersecurity, and increased trust, how the research can support public policies for cybersecurity, digital supply chains, and digital privacy, and how the research can be applied to selected industrial use cases.

¹ S. W. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero Trust Architecture,” *NIST*, Aug. 2020, Accessed: Jul. 19, 2022, [Online]. Available: <https://www.nist.gov/publications/zero-trust-architecture>.

² “Patents Filed by BlockFrame, Inc. U.S. Patent Application No. 17/488,529; No. 17/488,562; No. 17/488,589; No. 17/488,655; Patent Cooperative Treaty (PCT) Application No. PCT/US21/52839; No. PCT/US21/52854; No. PCT/US21/52865; No. PCT/US21/52872.” United States Patent and Trademark Office., Oct. 01, 2021. The most current information is available at <https://www.blockframetech.com/patents/>.

1.2. The Cyber Threat Environment

Cyber threats are ubiquitous in modern computers, networks, and Internet-of-Things (IoT) devices, including systems that users might perceive to be secure. Cyber-attacks might manifest themselves through continuous attacks, directed surges, swarms of attacks, hidden infiltrations that could be released at pre-scheduled times, or through attacks that continue to act clandestinely over protracted periods of time. Cyber-attack effects could manifest themselves as immediate noticeable destruction, denial of services, public or private extortion, unnoticeable long-term infiltration, the theft of money, the exportation of data, or by the hostile capture of computers or networks to act for the interests of attackers³. Cyber-attacks on computer networks and cyber-physical systems have emerged as primary national and international security concerns. According to the National Cyber Security Centre of the United Kingdom, the loss of sensitive, private data, and revenue by governments, businesses, and individuals takes place through a wide variety of disruptions from ransom payments, loss of access to data, and damage to computers and software⁴. Organized crime and state-supported actors are the leading offenders, and public-sector targets are the victims of more than half of recorded incidents⁵. According to the United States Cyberspace Solarium Commission,

“The reality is that we are dangerously insecure in cyber. Your entire life—your paycheck, your health care, your electricity—increasingly relies on networks of digital devices that store, process, and analyze data. These networks are vulnerable, if not already compromised. Hundreds of billions of dollars have been lost to state-sponsored intellectual property theft and cyber espionage. A major cyber-attack on the nation’s critical infrastructure and economic system would create chaos and lasting damage exceeding that wreaked by fires in California, floods in the Midwest, and hurricanes in the Southeast⁶.”

³ Y. Diogenes and D. E. Ozkaya, “Cybersecurity – Attack and Defense Strategies - Second Edition,” *Packt*, Dec. 2019.

<https://www.packtpub.com/product/cybersecurity-attack-and-defense-strategies-second-edition/9781838827793> (accessed Jul. 19, 2022).

⁴ “The cyber threat to UK business 2017-2018 report.” <https://www.ncsc.gov.uk/information/the-cyber-threat-to-uk-business-2017-2018-report> (accessed Jun. 22, 2022).

⁵ “Verizon 2019-data-breach-investigations-report.pdf.” Accessed: Jun. 22, 2022. [Online]. Available: <https://www.verizon.com/business/resources/reports/2019-data-breach-investigations-report.pdf>.

⁶ “Cyberspace Solarium Commission.” <https://www.solarium.gov/> (accessed Jul. 05, 2022).

In early 2018, the United States Federal Bureau of Investigation and the National Security Agency released a rare joint report directly accusing the Russian government of active cyber-attacks targeting components of the United States energy and other critical civilian infrastructure⁷. The sobering reality of the magnitude of the cybersecurity threat to United States federal networks, critical infrastructure⁸, and bulk power distribution systems⁹ has prompted active research and development to counter strategic threats from cyber-attacks. Beyond the threats to business and government infrastructure, mass breaches of personal and financial data have become a regular occurrence. In 2015, the United States Office of Personnel Management (OPM) reported a breach that resulted in the loss of personal information from over 21 million government employees¹⁰. In 2018, the General Services Administration (GSA) announced that they had discovered a breach of the System for Award Management (SAM) portal, which affected 70,000 government contractors. In 2019, breaches of the Facebook data system may have compromised the personal data of 540 million users¹¹.

The complexity of the system-of-systems (SoS) architectures of modern data enterprises contribute to their own vulnerabilities. According to Bayuk, J., and Horowitz, B. 2011¹², “Many would argue this defense-in-depth pattern should not be applied to the eCommerce environment because the sheer magnitude of identity theft cases shows this pattern to be a failure.” Anecdotally, other (unnamed) frustrated pundits might just conclude that privacy is already dead, so why fight it, or why worry about it? While defense-in-depth, a layered pattern of firewalls and other cybersecurity tools and methods, should not be abandoned, Bayuk and Horowitz’ comments raise a clear question whether that strategy by itself is adequate when attackers are often successful in penetrating inside “hardened” security layers. Another set of techniques for more robust cybersecurity, commonly called “zero trust,”

⁷ “Russian Government Cyber Activity Targeting Energy and Other Critical Infrastructure Sectors | CISA.” <https://www.cisa.gov/uscrt/ncas/alerts/TA18-074A> (accessed Jun. 22, 2022).

⁸ “Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure – The White House.” <https://trumpwhitehouse.archives.gov/presidential-actions/presidential-executive-order-strengthening-cybersecurity-federal-networks-critical-infrastructure/> (accessed Jun. 22, 2022).

⁹ “Executive Order on Securing the United States Bulk-Power System.” May 01, 2020. Accessed: May 06, 2020. [Online]. Available: <https://www.whitehouse.gov/presidential-actions/executive-order-securing-united-states-bulk-power-system/>.

¹⁰ “Cybersecurity Incidents,” *U.S. Office of Personnel Management*. <https://www.opm.gov/cybersecurity/cybersecurity-incidents/> (accessed Jun. 22, 2022).

¹¹ K. Utermohlen, “Facebook Data Breach 2019: 540 Million Users’ Records Exposed Facebook Data Breach 2019: 540 Million Users’ Records Exposed,” *InvestorPlace*, Apr. 04, 2019. <https://investorplace.com/2019/04/facebook-data-breach-2019/> (accessed Jun. 22, 2022).

¹² J. L. Bayuk and B. M. Horowitz, “An architectural systems engineering methodology for addressing cyber security,” *Systems Engineering*, vol. 14, no. 3, pp. 294–304, 2011, doi: 10.1002/sys.20182.

focuses on detecting and stopping cyber penetrations inside and between the defense-in-depth layers by assuming that no person, machine, or software is “trustworthy” until they are proved to be so¹³.

While eCommerce and IoT devices are only part of highly complex government and business infrastructures and supply chains, they represent an explosive growth sector upon which people, businesses and governments have already become reliant. Without adequate protections they are likely to continue to be vulnerable to compromise¹⁴. Current public concerns and legal actions about mass breaches of personal and financial data through businesses and governments, and concerns about the loss of personal privacy including Constitutionally protected civil rights against unwarranted government intrusion have become topics of everyday discussion. The highly visible denial-of-service attack on the Colonial Pipeline¹⁵, recurrent threats to federal government networks, and other elements of United States critical infrastructure led President Trump to issue an Executive Order to address those collective problems¹⁶. The Russian invasion of Ukraine in 2022 also elevated cyber warfare on both sides, with many examples of how cyber has been integrated into more conventional kinetic warfare¹⁷. Direct cyber-attacks by Russia on the Ukraine power grid in 2015 foreshadowed similar attempts in 2022¹⁸¹⁹. The United States involvement in the conflict also raised concerns that Russian could escalate attacks directed toward US critical power infrastructure²⁰. These trends continue to demonstrate the dangers of cyber threats, and the uncertainty as to whether they can be contained by current methods.

¹³ S. W. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero Trust Architecture,” *NIST*, Aug. 2020, Accessed: Jul. 19, 2022. [Online]. Available: <https://www.nist.gov/publications/zero-trust-architecture>.

¹⁴ L. Columbus, “10 Charts That Will Challenge Your Perspective of IoT’s Growth,” *Forbes*. <https://www.forbes.com/sites/louiscolombus/2018/06/06/10-charts-that-will-challenge-your-perspective-of-iots-growth/> (accessed Jun. 22, 2022).

¹⁵ “Colonial Pipeline Cyber Incident,” *Energy.gov*. <https://www.energy.gov/ceser/colonial-pipeline-cyber-incident> (accessed Nov. 27, 2021).

¹⁶ “Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure – The White House.” <https://trumpwhitehouse.archives.gov/presidential-actions/presidential-executive-order-strengthening-cybersecurity-federal-networks-critical-infrastructure/> (accessed Jun. 22, 2022).

¹⁷ “2022 Ukraine cyberattacks,” *Wikipedia*. Jun. 09, 2022. Accessed: Jul. 19, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=2022_Ukraine_cyberattacks&oldid=1092281378.

¹⁸ “Analysis of the Cyber Attack on the Ukrainian Power Grid - TLP: White March 18, 2016 - SANS ICS,” Mar. 18, 2016. <https://www.readkong.com/page/analysis-of-the-cyber-attack-on-the-ukrainian-power-grid-6826988> (accessed Jul. 19, 2022).

¹⁹ Gabor, “Ukraine Power Grid Cyberattacks,” *Rainbow and Unicorn*, May 17, 2022. <https://blog.gaborszathmari.me/ukraine-power-grid-cyberattacks/> (accessed Jul. 19, 2022).

²⁰ J. Runyon, “Power grid faces elevated Russian cyber threats during war in Ukraine,” *POWERGRID International*, Mar. 31, 2022. <https://www.power-grid.com/news/power-grid-faces-elevated-russian-cyber-threats-during-war-in-ukraine/> (accessed Jul. 19, 2022).

1.3. Systems Engineering Background

Prior to the release of *Effective Model-Based Systems Engineering*²¹, there were few if any comprehensive systems engineering textbooks that specifically emphasize the consideration and design of cyber security architectures within a model-based systems engineering (MBSE) framework and using the Model-Based System Architecture Process (MBSAP). Practical applications of MBSE and MBSAP to complex cybersecurity systems need to be thoroughly researched and tested with actual use cases. Use case needs described by Colorado Senate Bill SB18-086 are a fertile starting set for that research, as well as several similar personal, corporate, and/or military systems outside of state governments²². Applying MBSE and MBSAP to help design the Philos™ and ESPT™ Enterprise architectures and then applying those enterprises to the stated cybersecurity gaps will be a subject of investigation for this research. Intrinsic to this research is the idea that cybersecurity can be understood in the systems domain as an emergent property of systems. To date, Emergence Theory as a framework to understand cybersecurity as an emergent property has not been well developed.

²¹ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

²² Lambert, Kent D., Angela Williams, Joann Ginal, and Bob Rankin. 2018. "Colorado Senate Bill SB-086, 'Cyber Coding Cryptology for State Records.'" Colorado General Assembly. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.

1.4. Cybersecurity Background

Recognizing that defense-in-depth tools need to be supplemented by a “zero-trust” security strategy, Colorado Senate Bill SB18-086²³ put a process into statute to improve the security of state records, across a variety of anticipated use cases. The bill explicitly directs the state to study the use of blockchain/distributed ledger technologies for those applications. The questions at hand are whether cyber-security systems based on both defense-in-depth and zero-trust can be successfully applied to government, business, and personal privacy use cases, and whether systems engineering methods can effectively contribute to that process. According to systems engineering and cyber security experts, engineered architectures should include good systems engineering architecture modeling practices, as well as defense-in-depth, and zero-trust cybersecurity principles^{24,25}. Those should include open systems architectures, secure, effective, delivery of secret cypher codes, and sound cryptology practices.

On the basis of the above reflections on the state of the field, we can understand that there is a need for systems engineering architectural processes that can successfully assist in improving the security of highly complex global government, business, and logistics systems from cyber-threats and attacks.

²³ Lambert, Kent D., Angela Williams, Joann Ginal, and Bob Rankin. 2018. “Colorado Senate Bill SB-086, ‘Cyber Coding Cryptology for State Records.’” Colorado General Assembly. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.

²⁴ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

²⁵ Borky, John Michael, “Cybersecurity for Systems Engineers.” Colorado State University, Jan. 16, 2018.

1.5. BlockFrame, Inc., Approach to Current Cybersecurity Gaps

Currently, there are no known systems engineering-based cybersecurity architectures that have successfully applied blockchain/distributed ledger technologies on a sufficient scale to protect sensitive state and national records and supply chains. This research seeks to study the application and exercise of a new set of applications using the “Philos™” blockchain/distributed ledger (BDL) enterprise, in conjunction with an “Eco-Secure Provisioning” (ESP)™ supply chain security enterprise. Those are being developed by BlockFrame, Inc., and Block Chain Development Community (BCDC) volunteers to address these problems²⁶. Philos™ was initially planned to meet needs defined in Colorado Senate Bill SB18-086²⁷. Philos™ is a cryptographically intensive, globally scalable enterprise, based on a series of encapsulated common components connecting across the enterprise architecture. The common units have specialized cryptology (cypher) generation and storage nodes, secure data and cypher delivery, and a neutral global governance structure similar to the Domain Name System (DNS)²⁸ of the Internet. The ESP™ logistics security enterprise will use a cloud-based data architecture to connect to participating objects through embedded encapsulated components. The merger of these two enterprises will increase trust and lower risk among all participating users and will include robust and verifiable cyber-security designs for data confidentiality, integrity, and availability, zero-trust, data immutability, and data non-repudiability²⁹³⁰. They will also have capabilities designed to provide data proof-of-origin (POO), to provide rapid recovery for system faults and data losses, the ability to cryptographically protect objects within the Internet of Things (IoT), to transmit and secure data elements within supply chains, to secure public documents, to secure power grid data and operations, and to secure personal privacy information (PPI). Sustainability and business life cycle costs and considerations will also be researched³¹.

²⁶ C. Gorog, “Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management.” Unpublished, 2019.

²⁷ Lambert, Kent D., Angela Williams, Joann Ginal, and Bob Rankin. 2018. “Colorado Senate Bill SB-086, ‘Cyber Coding Cryptology for State Records.’” Colorado General Assembly. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.

²⁸ “Domain Name System,” *Wikipedia*. Jun. 12, 2022. Accessed: Jun. 22, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Domain_Name_System&oldid=1092781489.

²⁹ C. Gorog and T. E. Boulton, “Solving Global Cybersecurity Problems by Connecting Trust Using Blockchain,” in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1425–1432. doi: 10.1109/Cybermatics_2018.2018.00243.

³⁰ C. Gorog and Boulton defines specific measures to quantify trust that are used in the Philos™ distributed ledger. As all transaction contracts are conducted and come into consensus, trust factors are measured as part of the consensus algorithm to help determine whether participants will accept or reject the consensus by voting with Byzantine Fault Tolerance and allow contracts to be accepted for entry on the distributed ledger.

³¹ K. Lambert and V. Bishop, “Engineering the Philos Blockchain System; Engr 569 Semester Project.” Colorado State University, 2019.

1.6. Emergence Theory Background

The Law of Requisite Variety, first formulated by Ross Ashby, advanced a largely accepted concept that the variety in complexity in any system can only be controlled or regulated by a system with an even higher degree of variety or complexity, thus "only variety can destroy variety"³². Several research and philosophical works have addressed that concept³³³⁴³⁵. Mittal and Rainey proposed that emergence is manifested in complex systems due to a lack of understanding of the systems but becomes even more problematic in systems of systems with geographical displacement, and a lack of adequate computational and systems engineering approaches. They conclude that, "it is difficult to capture the complete holistic behavior of a complex system," and with better understanding, move systems with negative emergence closer to states that can be understood and controlled within the scope of existing knowledge³⁶, that is, to better deal with Requisite Variety.

Research concerning Emergence Theory and its application to cyber-security will be presented in the proposed dissertation, largely based on his chapter, "Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments," in the book, "Emergent Behavior in System of Systems Engineering"³⁷, which was published in 2022. While that work has not yet been released, it may be available as an appendix to the dissertation. Some strategies to control negative emergence, like defense-in-depth and zero trust have already been mentioned. With the application of BlockFrame® technologies, those strategies are employed but are not sufficient in themselves to describe cybersecurity as it is applied throughout the complex BlockFrame® system of systems ecosystem.

BlockFrame® has used the concept and term "uniformity" to better describe fundamental concepts in BlockFrame® design for digital supply chain and cybersecurity applications. Uniformity as used by BlockFrame® does not yet have a rigorous academic definition but might be best described as an evolution of the concepts of defense-in-depth, zero trust, standardization, and the digital management and enforcement of system rules. In

³² "Law of Requisite Variety Definition." http://pespmc1.vub.ac.be/ASC/LAW_VARIE.html (accessed Jul. 21, 2022).

³³ "The Law of Requisite Variety," Aug. 31, 2001. <http://pespmc1.vub.ac.be/REQVAR.html> (accessed Jul. 21, 2022).

³⁴ "Requisite Variety - an overview | ScienceDirect Topics." <https://www.sciencedirect.com/topics/computer-science/requisite-variety> (accessed Jul. 21, 2022).

³⁵ K. Kijima, "Generalization of Law of Requisite Variety," in *Systems Research I: Essays in Honor of Yasuhiko Takahara on Systems Theory and Modeling*, K. Kijima, J. Iijima, R. Sato, H. Deguchi, and B. Nakano, Eds. Singapore: Springer Nature, 2022, pp. 27–46. doi: 10.1007/978-981-16-9937-5_2.

³⁶ S. Mittal and L. Rainey, "Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering," Jul. 2015.

³⁷ K. Lambert, "Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments," in *Emergent Behavior in System of Systems Engineering*, Pre-Release., CRC Press, 2022.

practice, uniformity across a system of systems can help apply consistent practices and standards in logistics, identification, traceability, trust, accountability, and balancing privacy with visibility³⁸. BlockFrame® has made analogies that similar insertions of uniform technologies into several industries have had profound and lasting effects, such as the process of the fractional distillation of crude oil, invented by chemist Samuel Andrews, allowed the separation of “standardized” gasoline, which in turn facilitated John Rockefeller’s ability to commoditize it through the Standard Oil Company. Other “discarded” products eventually served as heating oil, jet fuel, and plastics, and evolved their own associated industries. During WWI, military necessity led to the nationalization of railroads, including centralized standardization of locomotives, railcars, routes, and other changes for efficiency³⁹⁴⁰. The revolutionary uniformity of the assembly line process developed under Henry Ford was applied to aircraft manufacturing during World War II to the point that Ford produced one finished B-24 bomber every hour, helping win World War II⁴¹. Fairly simple ideas like uniform traffic lines, signs, and signals helped control the chaos of motor vehicle traffic but were not fully implemented for almost 50 years after the invention of automobiles. BlockFrame® technologies, especially the Eco-Secure Provisioning™ (ESP™) device, are designed with the similar aspiration to enable uniformity within digital supply chains and cybersecurity systems. Able to provide uniform cryptographic keys to common components throughout complex systems of systems, ESP™ devices can act very much as described in Zeigler⁴² and Mittal and Rainey defined as “Emergence Behavior Observers (EBO),” that can “observe system state transitions and are tuned to catch segments potentially of interest and to record snapshots into memory. A snapshot is defined as an information set comprising components, and their states, taken at specified moments for specified durations. The concept of Experimental Frame (EF) is a general one which separates the model from the conditions under which it is observed and executed and can include the more specific EBO concept⁴³.” As these concepts are fundamental to modeling and controlling emergence, it is a happy coincidence that the BlockFrame® ecosystem is almost an exactly parallel realization of those theoretical frameworks from Emergence

³⁸ C. Gorog and K. Lambert, “BlockFrame, Inc., Technology -- Use Cases for Blockchain Technologies,” Jul. 2021.

³⁹ “United States Railroad Administration,” *Wikipedia*. Jan. 21, 2022. Accessed: Jul. 22, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=United_States_Railroad_Administration&oldid=1067020171.

⁴⁰ The fact that the United States Railroad Administration faced strong opposition due to government legal overreach and was quickly dismantled, is not germane to the observation that several forms of uniformity were technologically useful and were continued by other means such as the Interstate Commerce Commission.

⁴¹ “How Ford’s Willow Run Assembly Plant Helped Win World War II.” Assembly. Jan. 3, 2019.

<https://www.assemblymag.com/articles/94614-how-fords-willow-run-assembly-plant-helped-win-world-war-ii> (accessed Jul. 22, 2022).

⁴² B. P. Zeigler, “DEVs-Based Modeling and Simulation Framework for Emergence in Systems of Systems,” in *Engineering Emergence, A Modeling and Simulation Approach*, First. CRC Press. 2019. p. 548.

⁴³ S. Mittal and L. Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering,” Jul. 2015.

Theory, but with important improvements. The ESP™ can not only observe but can also control systems and deliver cryptographically secure products and services. The Philos™ distributed ledger can not only “record snapshots,” but is designed to store massive amounts of data immutably, with non-repudiable data, certified proof of origin, and with a capability for global capacities and scope. Defense in depth and Zero Trust are also actualized through ESP™ devices within distributed common components with a unique topography, metaphorically called a layered “Russian Doll” architecture. That is, the BlockFrame® System Container that encapsulates each ESP™ device can be provisioned in states in various layers, one “inside” another, and can enforce the uniformity of each device at each level. In summary, the implications of this are that the problems of Requisite Variety may be contained by Emergence Theory design, which may be operationally actualized by BlockFrame® designs and technologies.

2.0. Proposed Research

2.1. Overview

The proposed research included faculty-guided research and documentation to include studies of existing literature, systems modeling, and immersion in industrial research, development, test and evaluation, production, financial and business development, and interface with academic and professional organizations such as the Institute for Electrical and Electronic Engineers (IEEE).

2.2. Methodology

The emphasis of this research follows the Colorado State University Doctor of Engineering in Systems Engineering (DESYSE) Program philosophy to study systems engineering topics within formal graduate-level academic settings, to conduct research within practical research settings with industry outside of strictly academic settings (practicum), and to report that research using robust faculty-guided academic products (a doctoral dissertation and other products). The bulk of this effort is derived through the proposed positions and activities with BlockFrame, Inc., the Block Chain Development Community (BCDC), the International Alliance of Trust Chains (IATC), the Institute for Electrical and Electronic Engineers (IEEE), the National Council of State Legislatures (NCSL), Colorado State University, the University of Colorado Colorado Springs, the Colorado Office of Information Technology, and other corporate partners and advisors.

2.3. Modeling Tools

As a key part of this research, modeling tools were studied and used to demonstrate and analyze characteristics of specific systems.

2.3.1. Dassault Cameo Systems Modeler. Cameo Systems Modeler is used at Colorado State University Department of Systems Engineering to assist in the study of the Systems Modeling Language (SysML), Model-Based Systems Engineering (MBSE), Model-Based System Architecture Process (MBSAP), and architecting of complex systems⁴⁴.

2.3.2 NPS/NSA Monterey Phoenix/Gryphon (MP/G). Monterey Phoenix is a systems-modeling software jointly managed by the US Naval Postgraduate School (NPS) and the National Security Agency

⁴⁴ “Cameo Systems Modeler - CATIA - Dassault Systèmes®.” Extensive model information is available at <https://www.3ds.com/products-services/catia/products/no-magic/cameo-systems-modeler/> (accessed Jul. 21, 2022).

(NSA) and is gaining exposure and acceptance within Department of Defense modeling communities. It is used to interactively visualize and study systems behaviors with subject matter experts to help validate preferred sequences, and to discover and avoid the effects of negative emergence. Monterey Phoenix is an interactive, web and cloud-based model. Gryphon is a Linux variant of Monterey Phoenix able to be used on local platforms. MP/G is designed and used as a minimalist model that is not intended to model or simulate events or behaviors at great levels of detail, such as all the engineering processes within systems, but is more useful to visualize an exhaustive number of high-level behavioral outcomes for examination by analysts, engineers, and stakeholders to qualitatively verify and validate those behaviors. While some simulation is possible, a relatively small number of iterations (or “Scope”) are typically used to allow visualization for the user, which allows users to iteratively test relatively rapid changes to the modeled behaviors while still providing a scope-complete set of behaviors. This MP design and operation are based on the Small Scope Hypothesis⁴⁵, which assumes that “most flaws in models can be demonstrated on small counterexamples”⁴⁶. The relative simplicity of this approach is intended to discover “unknown unknowns” through rapid visualizations that can be communicated with verification and validation or stakeholder groups to discover design flaws or opportunities for process improvement^{47,48}. MP/G can be used as a commonly accepted approach for basic emergence research. While it is not intended to provide a comprehensive MBSE model at the physical level, it is a useful behavioral analysis tool that if used consistently will provide ontologies and processes that can be transferable and understandable within diverse academic and topical settings.

⁴⁵ D. Jackson, “Software Abstractions: Logic, Language, and Analysis.” MIT Press: Cambridge, MA, USA, 2012.

⁴⁶ K. Giammarco and M. Auguston, “Monterey Phoenix Tutorial,” May 2019.

⁴⁷ M. Auguston, “Monterey Phoenix, or how to make software architecture executable,” Oct. 2009.

⁴⁸ M. Auguston, “Software Architecture Built from Behavior Models,” *ACM SIGSOFT Software Engineering Notes*, vol. 34, no. 5, 2009.

2.4. Supporting Data

Because the expected performance of the BlockFrame® ecosystem design to rapidly handle massive amounts of encrypted data worldwide, data performance metrics are very important to the overall system of systems. Typically, developers have been using SQL and SQL Light data protocols, but those might change or expand in the future. For this research, the systems engineering focus was almost exclusively on the Operational and Logical/Functional Viewpoints⁴⁹, and detailed data issues will typically only be addressed by exception. For example, aggregated metric-level data of overall system performance is typically used to monitor and govern high-level system performance in near-real-time, across perhaps hundreds or thousands of different global business customers. That sort of aggregated data might be researched, but for privacy and proprietary concerns, detailed data will not be discussed or revealed.

2.5. Existing Literature

The proposed work will afford unique opportunities to curate important academic, industrial, and professional sources from recent books, papers, articles, and journals, including for Google Scholar, IEEE, and other sources. Sources will include leading-edge research on cybersecurity, blockchain development, cryptologic applications, digital supply chain development and applications, and others within the research goals. Patented technologies from BlockFrame, Inc., and its research, development, and production processes and documentation are a primary source of information when releasable without compromising private or trade secret data⁵⁰.

⁴⁹ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

⁵⁰ “Patents Filed by BlockFrame, Inc. U.S. Patent Application No. 17/488,529; No. 17/488,562; No. 17/488,589; No. 17/488,655; Patent Cooperative Treaty (PCT) Application No. PCT/US21/52839; No. PCT/US21/52854; No. PCT/US21/52865; No. PCT/US21/52872.” United States Patent and Trademark Office., Oct. 01, 2021. The most current information is available at <https://www.blockframetech.com/patents/>.

2.6. Funding and Support

Some resources for BlockFrame[®] systems that have at least indirectly contributed to this research included funding from the State of Colorado, the University of Colorado, grants from government and professional organizations like IEEE, state scholarships through the University of Colorado Colorado Springs, and support from the Colorado State University. Intellectual property owned by BlockFrame[®] is a valuable facilitator to research but was accessed at no cost. Some State appropriated funds, public and private research grants, and scholarships for developer salaries, were used for BlockFrame[®] research and development. Federal COVID recover loans and R&D tax credits helped make early funding more sustainable. As of this writing, new resources for BlockFrame[®] R&D are expected through commercial distributed energy resource (DER) providers, the US Department of Energy, the National Renewable Energy Laboratory (NREL), and the Pacific Northwest National Laboratory (PNNL) for funding, testing space, and cooperative testing platforms, possibly with access to a large number of vendor products. Mr. and Mrs. David and Gail Liniger provided funding for the Liniger Honor, Service, and Commitment Scholarship, of which I was a recipient, and am very grateful. All these resources provided positive opportunities and support for this research.

3.0. Research Questions and Tasks

This research addresses five questions with specific tasks to examine within each question. These questions will form a practical framework for research work within selected academic interests and industrial sectors, as well as the basis for reporting that work in the form of a Doctor of Engineering dissertation. These efforts will attempt to balance philosophical and academic rigor, as well as practical applications with industry, to advance the field of knowledge, but will also form a systematic process to examine practical processes, tools, and applications that will be useful to improve the performance of systems and cybersecurity resilience.

3.1. Research Question 1 (RQ1)

How does Emergence Theory provide a useful framework and ontology to describe options to better control complex, chaotic cybersecurity environments that are dominated by negative emergent behaviors? How can modeling and analysis of emergence help demonstrate, test, and compare useful cyber defensive concepts?

3.1.1. Task 1 (RQ1-T1). Develop a “real world” cybersecurity scenario (MP/G modeling uses the term “Schema”) using the NPS/NSA Monterey Phoenix/Gryphon tool, in partnership with NPS. This task seeks to leverage existing software and novel concepts and ontologies from Emergence Theory research to model and study the behavior of highly complex systems with negative emergence. This task will develop a “**T1 Cybersecurity Schema**” and apply those concepts and definitions to simulate behaviors in an “out-of-control” cybersecurity environment, as an example of negative emergent behavior. The model will be extended to positive behaviors to show how negative emergent behaviors can be mitigated.

3.1.2. Task 2 (RQ1-T2). Publish the T1 Cybersecurity Schema with other studies of emergent behaviors in other environments and Schemas. These studies were accomplished in parallel but independently, with 12 other researchers researching and modeling other Schemas in other environments with emergent behaviors.

3.1.3. Task 3 (RQ1-T3). Deliver the proposed Monterey Phoenix/Gryphon model scenario instance (“Schema”) to NPS and archive the Schema as a documented and standardized reference model. The modeling process and methodology will be reviewed with Dr. Kristen Giammarco of the Systems Engineering Department of the Naval Postgraduate School, Monterey, California. This Schema was integrated into an NPS/NSA MP/G catalog of Schema use cases for teaching users the model, methodologies, and applied Emergence Theory concepts.

3.2. Research Question 2 (RQ2)

How can Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) be used to support the design of cybersecurity systems? While systems engineering and modeling have been applied to design cybersecurity components and behaviors, industry approaches have generally not been consistent and effective in mitigating expanding cybersecurity threats. This research question will be answered by constructing exemplary model-based architectures to identify, design, and satisfy operational and cybersecurity requirements, develop, secure, and operate (DEVSECOPS), verify, and validate (V&V), test and evaluate (T&E), support systems operations, and evaluating non-performance requirements such as system costs and sustainability.

3.2.1. Task 1 (RQ2-T1). Design systems engineering models of cyber-physical products for supply chain security using digital cybersecurity systems. The NPS/NSA Monterey Phoenix/Gryphon Model and the Dassault Cameo Systems Modeler will be used to model selected elements and behaviors of the BlockFrame® system-of-systems, which is comprised of four independent systems working together or independently: the BlockFrame Eco-secure Provisioning System (ESP)™, the Philos™ blockchain/distributed ledger, the BlockFrame® Industry Utility Registrar™ (IUR), and an independent neutral blockchain governance authority (the IATC).

3.2.2. Task 2 (RQ2-T2). Use Model-based systems engineering (MBSE) to describe and evaluate the quality attributes of BlockFrame® systems to operate within hostile cyber-attack environments with high negative emergence. While industries contracting with the federal government have been tasked to improve their cybersecurity surety by a series of mandatory regulatory schemes, major security attacks and breaches have continued to occur. In this task, BlockFrame® designs will be modeled to assess capabilities to meet requirements beyond regulatory compliance to include the application and management of security using advanced cryptology methods, uniform security controls of hardware, software, and people, and using distributed ledgers to record and enforce that uniformity.

3.2.3. Task 3 (RQ2-T3). Research BlockFrame® product capabilities and publish findings as part of a cybersecurity textbook, "Applied Blockchain Technology for Security and Privacy." This task describes a translational research publication process whereby findings of the practical modeling exercises of Tasks 1 and 2 will be published as sections of a publicly available and peer-reviewed textbook. Appropriate documentation of these tasks will help communicate the technological concepts with industrial users and customers, train new users and developers, and advance new academic studies.

3.3. Research Question 3 (RQ3)

How can Eco-Secure Provisioning™ (ESP) of cryptologic non-fungible utility tokens (NFTs), in conjunction with blockchain/distributed ledger-assisted management and governance provide secure commercial services, improve cybersecurity, and to enforce trust in systems of systems (SoS)? For this research, system of systems terminology will refer to both the internal independent BlockFrame® systems, as well as the external systems used by business-to-business (B2B) clients through common components, interfaces, and Service Level Agreements (SLAs). BlockFrame® systems will provide many different services for B2B clients, depending on their own requirements and business plans. The BlockFrame® guiding Mission, Vision, and Values can be summarized as: “BlockFrame® uses blockchain to organize cybersecurity worldwide – achieving trust and uniformity for secure supply chain,” “Provides logistics support to create a secure ecosystem for managing IoT devices using blockchain and distributed ledger technologies,” and “Individualized Privacy, Proof of Origin on data, Zero Trust Networking, Security Defense-in-Depth, Multi-Factor Identification, Accountable Governance”⁵¹. This RQ investigates these key aspects of BlockFrame® to understand how they are applied to practical and applied cybersecure transactions and services.

3.3.1. Task 1 (RQ3-T1). Model the internal functional behaviors of the BlockFrame® ESP™ and the Philos™ blockchain/distributed ledger to process cryptographic key management and transactions using utility NFTs, processes to allow global scalability, and externally facing monetary interfaces to support commercial purchases for secure transactions. The Philos™ distribute ledger uses a “green” design with no energy mining for “proof of stake.” It creates no monetary value as a cryptocurrency, but it will permanently record transactions immutably and non-repudiably. This task will expand upon models developed in RQ1 to describe the system behaviors and economic processes that explain and support the overall business model.

3.3.2. Task 2 (RQ3-T2). Model the behaviors the BlockFrame® system-of-systems to progressively increase trust factors in the complex, cross-industry global Philos Marketplace™. This task describes an algorithmic approach to monitor, govern, and differentiate trust scores from all participating users, and to incentivize or dis-incentivize users through the decisions of their peers while in competition to add blocks to the Philos™ blockchain. A dashboard will use statistical surveillance to constantly monitor the relative behaviors of each real-

⁵¹ “BlockFrame – BlockFrame Inc.” <https://blockframetech.com/> (accessed Jul. 20, 2022).

world user of the Philos™ blockchain, the statistical trends of their behaviors, and overall increases or decreases in trust scores over time. This work will result in an experimental assessment of the effectiveness of the ESP/Philos system design.

3.3.3. Task 3 (RQ3-T3). Monitor and control the Philos™ B/DL process through a neutral governance organization that can set operational parameters for policies to control the trust algorithm. Collect, act on, and report data over time to evaluate the overall effectiveness and trust levels of system-of-systems through the International Alliance of Trust Chains (IATC).

3.4. Research Question 4 (RQ4)

What are the implications of this research for public policies and industrial practice for cybersecurity, digital supply chains, and digital privacy?

3.4.1. Task 1 (RQ4-T1). Research Digital Privacy through participation in and support of the IEEE Digital Privacy Initiative. This task provides research and development on digital privacy methods and policies to the IEEE Digital Privacy Initiative, as a means of translating the findings from the other research questions to the community of practice for Blockchain-enabled cybersecurity applications.

3.4.2. Task 2 (RQ4-T2). Initiate and chair a joint working group and a cooperative partnership between IEEE Digital Privacy Initiative and the National Council of State Legislatures (NCSL) Cybersecurity and Privacy Task Force, and/or other state-level legislative groups, to provide mutually beneficial support for state and interstate cybersecurity and digital privacy issues. Examples of this cooperation could include technical support from IEEE members to NCSL on cybersecurity and digital privacy, providing papers, presentations, and/or model legislation on topics of common interest. This task fills a gap in the translation of technical engineering knowledge to public policy and legislative requirements.

3.4.3. Task 3 (RQ4-T3). Participate in and support IEEE Standards Working Group “Task Force 1”, to contribute to cybersecurity standards research of the DoD and Mitre Att8ck™ analysis effort and Cybersecurity Maturity Model Compliance (CMMC). This task is expected to result in the second of three IEEE white papers to define a useful framework for blockchain technologies (BCTs) to support energy use cases, identify cyber tactics that might attack BCT components, and suggest mitigation steps to improve cybersecurity resilience.

3.4.4. Task 4 (RQ4-T4). Publish and/or contribute to IEEE/NCSL articles on cybersecurity, blockchain/distributed ledger, digital privacy, and/or digital supply chain issues. This task creates environments to communicate policy issues among engineering, policy, and legislative communities of practice for applied cybersecurity, blockchain, digital privacy and supply chain and issues.

3.5. Research Question 5 (RQ5)

What are examples of practical applications of this research for selected industrial use cases examples? When fully ready for operational applications, BlockFrame® technologies are able to operate globally across different industries as a system of systems using common components within each system.

3.5.1. Task 1 (RQ5-T1). Apply and evaluate appropriate BlockFrame® reference models to selected industrial use cases, primarily in the prioritized industries of: distributed transactive energy, education, medical devices, and cybersecurity/privacy applications. The MP/G and Cameo models can describe useful behavioral and architectural models of each selected industry use case, will help verify and validate inter-system designs and processes, and will help educate and train B2B users of new capabilities. Applications to “real world” companies, like remote learning educational systems, transactive energy systems, and cybersecurity for medical devices will be developed as commercial applications as each of those domains develop.

3.5.2. Task 2 (RQ5-T2). Compare the energy consumption characteristics of the “green” Carbon-Neutral Philos™ blockchain/distributed ledger design to other blockchain designs⁵². Document and publish environmental, economic benefits, and cost savings of using the Philos™ design compared to electrical-energy mining blockchain alternatives. The use of a reliable digital ledger with no proof-of-stake mining should be highly useful for governments and customers concerned about wasting costly energy.

3.5.3. Task 3 (RQ5-T3). Apply financial analysis when necessary to research cost comparisons, financial risk management, resilience, analysis of capitalization, grants, investor marketing, return on investment, and lessons learned. Various forms of quantitative business analysis will be used to calculate performance and non-performance system metrics for RQ5 and other research as needed.

⁵² C. Gorog, P. Russell, and T. E. Boulton, “Carbon Neutral Distributed Ledger,” New Orleans, LA, Apr. 2022.

4.0. Translational Practicum Context

Several of the research questions and tasks for this dissertation are based on translational objectives to allow academic knowledge to inform and affect industry practices, and, vice versa, for new knowledge and practices in industry to inform and affect academia. Within the Department of Systems Engineering at Colorado State University, those ideas are among the highlighted objectives for the degree of Doctor of Engineering in Systems Engineering (SE DEng), in contrast to the degree of Doctor of Philosophy in Systems Engineering (SE PhD) which is focused on preparation for pedagogical and teaching-focused objectives. As an important part of the Doctor of Engineering track, special emphasis is placed on practical applied research within the context of an industrial and business setting. For that practicum phase, much of the practical research has been done with BlockFrame, Inc., professional organizations, and several diverse activities that the company conducts. Like systems engineering itself, both doctoral degrees might be correctly described as inherently multi-disciplinary, but with visions of different student academic outcomes.

An authoritative definition of “translational” in scientific fields is, “of or relating to the transfer of scientific knowledge into practical applications”⁵³. One aspirational goal of the Doctor of Engineering program should be to “reverse engineer” that idea by examining the *practical applications*, and then to translate those into *scientific knowledge*. This doctoral practicum and dissertation focus on observing new advanced technologies and processes being developed by the BlockFrame® Corporation and other partners in the areas of digital supply chain management, cybersecurity, and digital privacy. Those capabilities are examined with the Model-Based Systems Engineering (MBSE) and the Model-Based Systems Architecture Process (MBSAP) in mind to help satisfy industry requirements, to help interact with subject matter experts from industry, academia, government, and professional organizations, and to design new functional capabilities to meet those requirements. Articles, white papers, models, projects, and this dissertation have helped articulate the actionable systems engineering principles that have been used to research, document, and advance the state of the art and the body of knowledge in these areas.

⁵³ “Translational — definition, examples, related words and more at Wordnik,” *Wordnik.com*. <https://www.wordnik.com/words/translational> (accessed Jul. 22, 2022).

4.1. Introduction to Colorado State University Systems Engineering Department⁵⁴

Colorado State University Systems Engineering Department (CSU SE) offers two doctoral degrees in Systems Engineering: the Doctor of Philosophy in Systems Engineering (SE PhD), and the Doctor of Engineering in Systems Engineering (SE DEng). Both degrees have equivalent expectations for students to develop their SE mastery, work with expert faculty, perform novel research, author peer-reviewed publications, publish a dissertation, and contribute to the system engineering body of knowledge. The difference between the two degrees is their emphases and audiences in terms of the research process and products.

The Doctor of Philosophy in Systems Engineering⁵⁵ (SE PhD) is a doctoral degree that seeks to emphasize research activities that use both “basic” and “applied” research methods. Basic SE research seeks to build, test, and validate the fundamental and theoretical foundations of systems engineering. Applied research seeks to generate new knowledge applied to the domain of the systems engineering lifecycle processes. At CSU, applicants to the SE PhD program are able to design a research program that is responsive to their interests, expertise, and industrial contexts. They work continuously with a faculty advisor to define, refine, and complete their research program, but our program is able to support a very wide variety of students’ SE research projects because of the breadth of expertise that CSU has available in SE core and affiliate faculty.

The Doctor of Engineering in Systems Engineering⁵⁶ (SE DEng) is a doctoral degree that seeks to emphasize research activities that use both “applied” and “translational” research methods. Where applied research is defined above, translational research activities are concerned with the investigation and development of new knowledge in the context of the Systems Engineering practiced. The DEng degree program seeks to work in collaboration with industry and other SE enterprises to learn about and define industry-relevant research problems, to study how SE is practiced in industry, and to develop novel and improved practices for SE that can be directly translated into industry. The DEng degree asks students to find a “practicum sponsor” from their industry who will help to advise them throughout the doctoral degree, and CSU then appoints that person to our department and doctoral advising

⁵⁴ This section contains inputs from Dr. Thomas Bradley, Chairman, Systems Engineering Department, Colorado State University.

⁵⁵ <https://www.engr.colostate.edu/se/phd/>

⁵⁶ <https://www.engr.colostate.edu/se/wp-content/uploads/2021/06/Doctor-of-Engineering-Flyer.pdf>

committee. In the DEng., the University and industry work together to develop SE expertise, thought leadership, and competitive innovation to improve the state of the practice of SE.

4.2. Introduction to BlockFrame, Inc.

BlockFrame, Inc.⁵⁷, is a Colorado C-Corporation that identifies its primary functions as secure digital supply chain processes and the secure delivery of secret cryptologic keys. To accomplish that, BlockFrame® conducts extensive novel research and development to build sustainable architectures and services that can be commercialized across many industrial verticals. Those characteristics will be described in greater detail in subsequent chapters, but as an example of research with industry that can be translated with academic systems engineering disciplines, it is an excellent environment for research. In order to meet corporate objectives, several unique operational and functional capabilities of BlockFrame® technologies should be mentioned up-front to avoid confusion with similar systems with dissimilar functionalities. In many cases, these capabilities had to be invented because existing blockchain systems are not capable of performing the desired functions. This is a partial list of some key BlockFrame® design requirements:

- The BlockFrame® system-of-systems must be indefinitely scalable to serve very large markets and industries worldwide.
- The BlockFrame® Blockchain/Distributed Ledger (B/DL) system does not use “energy mining” as is common in other blockchains.
- The BlockFrame® distributed ledger is not a cryptocurrency.
- Instead of transferring monetary value, as might be common for a cryptocurrency, the value proposition of the system-of-systems is designed and optimized to increase security, trust, and privacy among its participating clients using voluntary and self-enforcing trust algorithms.
- When properly implemented, BlockFrame® technologies are designed to enforce security and trust for hardware, software, and human components and behaviors from their original points of origin.

Much of the body of knowledge used by BlockFrame® technologies is based on prior research, development, and implementation by Dr. Christopher Gorog, the BlockFrame® CEO, CTO, and inventor, and work in his doctoral dissertation⁵⁸. This dissertation attempts to build upon and expand upon that body of knowledge. Much of Dr

⁵⁷ <https://www.blockframetech.com/>; accessed Sept 1, 2024.

⁵⁸ C. Gorog, A Sustainable Framework for Distributed Ledger, <https://www.proquest.com/docview/2901290591?pq-origsite=gscholar&fromopenview=true&sourcetype=Dissertations%20%20Theses>; accessed Sept 1, 2024.

Gorog's dissertation was conducted to prove the feasibility of the massive scope and indefinite expandability of BlockFrame® technologies, which are assumed for the purposes of this dissertation.

4.3. Business Management and Commercial Considerations

As a commercial business entity, much of the practicum research conducted with BlockFrame[®] was done within a business context. Several course requirements for the Colorado State University Doctor of Engineering in Systems Engineering degree included business courses that were not exclusively within the traditional scope of the systems engineering discipline. Throughout the course of this practicum and dissertation research, I also served within BlockFrame, Inc., as Chief of Operations, and as an officer on the Board of Directors. This provided additional valuable “translational” experience for research with industry in the areas of finance, banking, corporate law, patent law, and several other areas of applied business operations that might not normally gain exposure within a purely academic systems engineering program. The success and intrinsic value of the BlockFrame[®] company relies largely on the development, financing, and protection of its intellectual property. I have been directly involved in each of those business areas, including obtaining four fully secured United States and over twelve international patents, which are essential to the success of both the corporation and protected services to customers worldwide⁵⁹. For the purposes of this research, discussion of some proprietary details is avoided or simplified to avoid disclosure of sensitive information or trade secrets, but general non-proprietary information is included.

⁵⁹ All current BlockFrame[®] patents may be accessed through Google Patents at <https://patents.google.com/patent/WO2022072617A1/en?assignee=blockframe&oq=blockframe>

5.0. Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments

This chapter addresses and satisfies Research Question 1 (RQ1) by applying Emergence Theory concepts, vocabulary, and a unified community research approach to an emergent cybersecurity use case. Restated, Research Question 1 (RQ1) is: “How does Emergence Theory provide a useful framework and ontology to describe options to better control complex, chaotic cybersecurity environments that are dominated by negative emergent behaviors? How can modeling and analysis of emergence help demonstrate, test, and compare useful cyber defensive concepts?”

The results of research on RQ1 demonstrate an approach to modeling emergent secure and insecure behaviors, and methods to contain or control negative emergence in cybersecurity applications. The results of this task will be a compilation of the proposed Schema to allow an examination of the similarities and differences of their findings in the book, “Emergent Behavior in System of Systems Engineering, Real World Applications⁶⁰.” The NPS/NSA MP/G was used as a common model by all 12 researchers to use a common ontology of Emergence Theory, and to relate its common terms to uniquely different environments and behaviors (Schema). The 12 researchers did not have access to each other’s work products during their research. The research was delivered to the book publisher, Chemical Rubber Company (CRC) Press, and met their publishing standards, under the guidance of the editor, Dr. Larry Rainey, formerly of the United States Air Force Institute of Technology and United States Air Force Space Command. As Dr. Rainey’s third book in a trilogy about systems engineering within systems of systems and Emergence Theory, this was part of a novel and consistent process to organize and advance the study and applications of Emergence Theory, and in this case for cybersecurity in particular.⁶¹

⁶⁰ K. Lambert, “Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments,” in *Emergent Behavior in System of Systems Engineering*, Pre-Release., CRC Press, 2022.

⁶¹ All content in this Chapter 5 was originally published as Chapter 5 of Emergent Behavior in Systems of Systems Engineering – Real-World Applications. It is reproduced here as faithfully and verbatim as possible. For ease of reading, the pagination of this chapter has been changed to be consistent with pagination of this dissertation. The References section of Chapter 5 has not been changed to align with other citations or references within this dissertation. They are listed parenthetically at the end of Chapter 5, as they were cited in the original document.

Introductory comments about this Chapter 5 by Dr Larry Rainey, Editor, are included in Emergent Behavior in Systems of Systems Engineering – Real-World Applications⁶²], Chapter 1, pp. 7-8.⁶³

⁶² K. Lambert, “Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments,” in *Emergent Behavior in System of Systems Engineering*, Pre-Release., CRC Press, 2022.

⁶³ “Chapter 5 is authored by Kent Lambert, Colonel USAF (Ret), Dr Thomas Bradley, and Mr. Christopher Gorog, and is entitled, “Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments.” The authors address the presence of these emergent behaviors and their negative effects, we propose controls through three ecosystem-wide strategies, Defense-in-Depth (Did), Zero-Trust Architectures (ZTA), and secure cryptographic key provisioning. This novel solution recommends enhancing those strategies with blockchain-assisted logistics management and a Model-Based Systems Engineering (MBSE) approach, supported by the Model-Based Systems Architecture Process (MBSAP). The BlockFrame, Inc., Eco-Secure Provisioning™ (ESP™) Framework, which contains secure cryptographic key provisioning, and the Philos Blockchain/Distributed Ledger™ (Philos/BDL™) demonstrate synergistic solutions through indefinitely scalable, cross-industry SoS applications using modular, loosely coupled service interfaces, multifactor identification, and applied uniformity. Project managers, designers, and systems engineers seeking ecosystem-wide solutions for logistics control and organization-wide cybersecurity can benefit from these novel SoS architectures.”

5.1. Introduction

Cyberattacks on computer networks and cyber-physical systems have emerged as primary national and international security concerns. According to the National Cyber Security Centre of the United Kingdom, the loss of sensitive, private data, and revenue by governments, businesses, and individuals take place through a wide variety of disruptions from ransom payments, loss of access to data, and damage to computers and software (National Cyber Security Centre 2019). Organized crime and state-supported actors are the leading offenders, and public-sector targets are the victims of more than half of recorded incidents (Verizon 2019). According to the United States Cyberspace Solarium Commission, “The reality is that we are dangerously insecure in cyber. Your entire life—your paycheck, your health care, your electricity—increasingly relies on networks of digital devices that store, process, and analyze data. These networks are vulnerable, if not already compromised. Hundreds of billions of dollars have been lost to state-sponsored intellectual property theft and cyber espionage. A major cyberattack on the nation’s critical infrastructure and economic system would create chaos and lasting damage exceeding that wreaked by fires in California, floods in the Midwest, and hurricanes in the Southeast” (“Cyberspace Solarium Commission,” n.d.). In early 2018, the United States Federal Bureau of Investigation and the National Security Agency released a rare joint report directly accusing the Russian government of active cyberattacks targeting components of the United States energy and other civilian critical infrastructure (FBI/NSA 2018). The sobering reality of the magnitude of the cybersecurity threat to United States federal networks, critical infrastructure (D. Trump 2017), and bulk power distribution systems (Donald Trump 2020) has prompted active research and development to counter strategic threats from cyberattacks.

Beyond the threats to business and government infrastructure, mass breaches of personal and financial data from businesses and governments have become a regular occurrence. In 2015, the United States Office of Personnel Management (OPM) reported a breach that resulted in the loss of personal information from over 21 million government employees (Office of Personnel Management Security Resource Center, n.d.). In 2018, the General Services Administration (GSA) announced that they had discovered a breach of the System for Award Management (SAM) portal, which affected 70,000 government contractors. In 2019, breaches of the Facebook data system may have compromised the personal data of 540 million users (Utermohlen 2019). The complexity of the system-of-systems (SoS) architectures of modern data enterprises contributes to their own vulnerabilities.

5.2. Defining Cybersecurity Threats as Emergent Behaviors

Rainey and Tolk (2015) provide definitions for four levels of Emergence Complexity in complex systems: Simple Emergence, Weak Emergence, Strong Emergence, and Spooky Emergence. These definitions are useful to help characterize the high degree of Emergence Complexity from cybersecurity threats. Figure 5.1 (Mittal and Rainey 2015) graphically displays those concepts as an Emergence Complexity Cone. Increasing complexity is shown on the y-axis; the Cone is divided into deterministic and stochastic search-space domains on the x-axis; and the Cone volume depicts variety. The Cone perimeter forms a Constraint Boundary, while the Boundary of Knowledge is shown as a cylinder that exists between the field or variety between the constraints. The Boundary of Knowledge Cylinder around simple and weak emergence in the deterministic domain signifies that ample knowledge is available for subject matter experts (SME) to develop abstractions using known techniques like cybernetics, systems theory, control theory, or network theory.

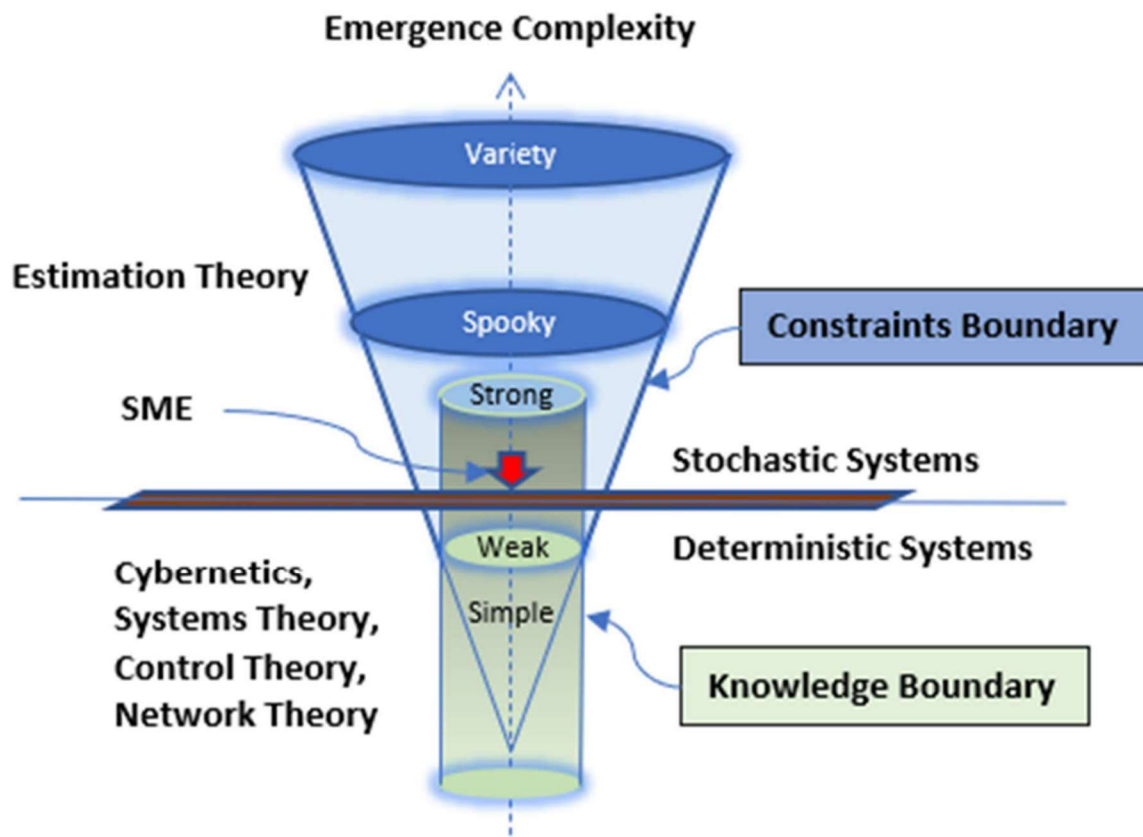


Figure 5.1. Mittal and Rainey Emergence Complexity Cone

The rapid and seemingly unchecked expansions of cybersecurity threats indicate that they exhibit behavior consistent with Strong or Spooky Emergence, with characteristics of unpredictable or stochastic complexity. The

general failure to successfully predict and contain the emergence of cybersecurity threats to a deterministic domain using the best-known technologies, models, and control theories helps validate that categorization. The presence of unpredictability and uncontrollability is also consistent with Mittal and Rainey's observation that, "when the knowledge itself does not exist to identify the variety and constraints [of the system], the behavior is classified spooky" (Mittal and Rainey 2015). Borky and Bradley make very similar observations about complexity within a systems engineering context: "We use the terms, 'complex' and 'complexity' in the sense of describing something that cannot be adequately dealt with using only simple, intuitive tools and methods;" and "Complex systems routinely exhibit behaviors that are difficult to predict or even to understand" (Borky and Bradley 2019)⁶⁴. What is at issue for Emergent Complex system behaviors is whether or not they can be understood and/or controlled. If systems exceed the Boundaries of Knowledge of SME, those system behaviors are axiomatically less understandable and predictable. As system behaviors exceed the Boundaries of Knowledge and approach the Constraint Boundaries, they should demonstrate higher levels of Emergence Complexity, become more stochastic, and therefore less controllable. Just as the levels of Emergence Complexity can provide at least a qualitative description of the growth of those behaviors, they should also provide a qualitative framework for judging whether those levels of Emergence Complexity are increasing or are being reduced. This chapter seeks to transfer these concepts to cybersecurity threats. By classifying a cybersecurity threat as a level of emergence, we can better understand the characteristics of cybersecure systems and better design and measure their controls, constraints, and environments to control emergent threats and improve cybersecurity performance by mitigating threat effects.

Mittal and Rainey also provide useful definitions of "positive," "negative," and "neutral" emergence: Emergence is positive when it fulfills the System-of-Systems' (SoS) purposes. Negative emergence does not fulfill the purposes of the SoS. Neutral emergence is neither positive nor negative. The purpose of attempting to control emergence is to keep systems and their components within optimal or healthy performance ranges, assuming that controls exist that are able to affect emergence either positively or negatively.

Cybersecurity threats exhibit "Spooky Negative Emergence" characteristics, i.e., emergence that does not fulfill SoS purposes, and are not currently subject to effective control mechanisms. One of the primary goals of Emergence Theory is to identify highly emergent systems, and then to apply control mechanisms to move from

⁶⁴ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

stochastic boundaries back to deterministic boundaries where SMEs are knowledgeable and might be able to apply known controls.

The nature of negative emergent behaviors that we now call malicious threats or advanced persistent threats (APT) did not exist before the Internet. Therefore, at the time of the invention of the Internet and during its early history, security controls were largely not considered. The value of rapid Internet expansion for economic growth and rapid inexpensive free expression is undeniable, but it also left many users reliant on legacy systems with huge vulnerabilities. According to the President's Commission on Enhancing National Cybersecurity:

No one—not even the visionaries who created the Internet a half century ago—could have imagined the extent to which digital connectivity would spur innovation, increase economic prosperity, and empower populations across the globe. Indeed, the Internet's origins in the defense community are today almost an afterthought, as its explosive growth has given it a dramatically different shape. Its creators could not have dreamed of the way and the extent to which our national and global economies have thrived, how innovations have been enabled, and how our population has been empowered by our digital connectivity. With these benefits and transformational changes have come costs and challenges. The interconnectedness and openness that the Internet, digital networks, and devices allow have also made securing our cyber landscape a task of unparalleled difficulty.

Commission on Enhancing National Cybersecurity (2016)

Worldwide Internet growth without planned or embedded cybersecurity might have been the first cause for the negative emergent behaviors because it was open to malicious exploitation. Additional negative emergent behaviors then grew from the growth of the threats themselves. Layered Defense-in-Depth (DiD) and Zero Trust Architectures (ZTA) are concerted reactions to cybersecurity threats, have certainly helped, and have not been totally successful in enforcing cybersecurity or trust, especially in legacy systems that were not originally designed for security. Ideally, all new systems should be designed with integrated cybersecurity components and processes, but technologies are also needed with “backward compatibility” to protect legacy systems until they can be replaced. The Eco-Secure Provisioning (ESP)[™] Framework and the Philos[™] blockchain can help reverse those negative emergent behaviors, and MBSE and MBSAP should be able to design, model, simulate, and measure their positive effects.

5.3. Controlling Negative Emergence of Malicious Cyber Activity

The negative effects of an initial lack of designed-in cybersecurity, compounded by emergent cybersecurity threats, have prompted transnational and cross-industrial responses. The recommended strategies to counter emergent cybersecurity threats are layered DiD, ZTA, and secure cryptographic key provisioning. DiD is defined as “deployment of diverse and mutually supportive security controls at various points in the architecture to create concentric shells of protection that an attacker must defeat to reach protected assets” (Borky and Bradley 2019). ZTA may be defined as “a strategy that extends and refines DiD architectures with fine-grained segmentation, strict access control, strict privilege management, and advanced protective devices, all based on a concept that no person and no resource is trusted without adequate verification.” Within DiD and ZTAs, the BlockFrame (ESP)[™] Framework and the Philos[™] blockchain can provide several other techniques to enhance DiD and Zero Trust, including the Proof-of-Origin (POO) for data elements, uniformity of network segments in the event of loss of wide-area communications, multifactor identification (MFI) of components within Zero-Trust Networks to assure all remote components are verified, strengthening Transport Security Layers (TLS) by assuring the trust of certificates, and management and verification of software patches and version updates. ESP[™] can also provide secure cryptographic key provisioning for even higher degrees of trust and security, particularly for distributed Internet-of-Things (IoT) devices.

DiD was a relatively simple, widely practiced, and initially successful architecture strategy to develop and secure cyber systems. In practice, quantitative studies, like the Verizon, “2019 Data Breach Investigations Report,” now show that as a control mechanism for the emergent growth of cybersecurity threats and their risks, DiD alone has proved inadequate to secure cyber systems across government, industries, and private sectors (Verizon 2019). John Kindervag, Principal Analyst at Forrester Research, Inc., created the ZTA (Pratt 2018) in 2010, which was subsequently commercialized by Forrester. Zero Trust recognized that DiD alone did not adequately stop the exploitation of systems that had already been penetrated, insider threats, threats from social engineering, and other situations where trust might have been assumed, but was demonstrated not to be trustworthy. ZTA emphasizes the importance of partitioning the interior of vulnerable systems, the authentication and authorization of personnel, and other security practices under the assumptions that no personnel, hardware, software, or practices should be trusted until it is confirmed and verified as trustworthy (Palo Alto Networks n.d.). According to Borky (Borky and Herber

2020)⁶⁵, DiD is still necessary but is no longer sufficient against APT. That is because cyberattackers are either already inside most systems, possibly resourced by hostile national governments, or because a ZT strategy would assume that they are. When ZTA complement DiD with state-of-the-art products within robust and resilient cyber designs, they represent the best solutions available today to APTs.

Although the combination of DiD and ZTAs are generally considered to be the best available cybersecurity design strategies for systems architectures, in practice the complexity of the cybersecurity threat environment, the complexity of the defensive DiD and ZTA architectures, and the known continuing damage from penetrations, data theft, ransomware attacks, and denial-of-service attacks continue to exist as negatively emergent phenomena. If Zero Trust assumes that systems have already been compromised, then one essential requirement for ZTAs should be the capability to reset or recover from a breach or compromise and to return to a trusted state. The huge proliferation of available security products still might conflict with one another, might not provide total security for all the gaps among the different products, or might not take a systematic and uniform approach to identifying, correcting, and monitoring trust levels throughout networks or enterprises (Gorog and Lambert 2020). Those “unknown” gaps, overlaps, and vulnerabilities still exist and demonstrate that current defenses cannot be trusted without more support from automated enforcement and governance, and better methods to reset or recover to a state of trust following compromises. If any ZTA cannot reset or recover, that should be identified as a systemic weakness that should be mitigated. New technical methods are being developed to help mitigate those trust problems, to enforce trust, and to recover trusted-state conditions from any compromise. Those approaches include secure cryptological key provisioning, blockchain-assisted logistics management, and the use of the improved architecting and design frameworks implicitly available through Model-Based Systems Engineering (MBSE) and supported by the Model-Based Systems Architecture Process (MBSAP). Agile Systems Engineering (ASE) architectural approaches, automated development and testing through Continuous Integration/Continuous Delivery (CI/CD), Test-Driven Development (TDD), and integration with rapid and scalable trust chain support are facilitating those developments.

A new set of applications are currently being developed and tested by BlockFrame, Inc., including the “Eco-Secure ProvisioningTM” (ESPTM) supply chain security enterprise, secure cryptological key provisioning, and the Philos Blockchain/Distributed LedgerTM Enterprise (Gorog 2019). The unique BlockFrame[®] inventions and intellectual

⁶⁵ Borky, John M., and Daniel R. Herber. 2020. *Engr 667: Advanced Model-Based Systems Engineering; Session 13: Cybersecurity of System Architecture (Lecture Slides)*. Fort Collins, Colorado: Colorado State University.

property explained in this chapter are protected by pending United States and international patents (“Patents Filed by BlockFrame, Inc. U.S. Patent Application No. 17/488,529; No. 17/488,562; No. 17/488,589; No. 17/488,655; Patent Cooperative Treaty (PCT) Application No. PCT/US21/52839; No. PCT/US21/52854; No. PCT/US21/52865; No. PCT/US21/52872.” 2021). The ESP™ logistics security enterprise will use a cloud-based data architecture to connect participating objects through embedded, encapsulated, loosely coupled, and modular components. ESP™ secure cryptographic key provisioning has been successfully implemented and demonstrated for the United States Navy (Gorog and Lambert 2020). Philos™, a community-based open architecture blockchain, was initially planned to meet needs defined in Colorado Senate Bill SB18–086 (Lambert et al. 2018). The Philos™ distributed ledger system is designed as a cryptology-intensive, globally scalable enterprise, based on a series of encapsulated common units to connect components across its architecture. The common units have specialized cryptographic generation and storage nodes, secure data, cypher delivery, and a global neutral governance structure, similar to the Domain Name System (DNS) of the Internet (“Domain Name System” n.d.). The successful application of blockchain logistics management using the ESP™ and Philos™ will increase trust and lower risk among all participating users and will include robust and verifiable cyber-security designs for data confidentiality, integrity, and availability (CIA), zero-trust, data immutability, and data non-repudiability.

According to Gorog and Boulton (2019), “In a distributed ecosystem of devices, complete positive identification would require having undeniable POO on each data item from every device.” ESP™ secure cryptographic key provisioning and Philos™ are all specifically designed to provide both hardware and data MFI and POO. Those built-in capabilities will also provide for rapid resets and recoveries for system faults and data losses back to a state of trust, and the capabilities to prove the identity of objects within the IoT through MFI, to secure data elements within supply chains, to secure public documents, to secure power grid data and operations, and to secure personal privacy information (PPI). They are also designed as open systems architectures, for the secure and effective delivery of secret cypher codes, and sound cryptographic practices. In particular, extensive externally managed (“out-of-band”) MFI of actors, hardware, software, or almost anything of unproven trust is a key capability for successful integration and uniformity. MFI of system components, authentication of personal identities, and processes to include authorizations and “need to know” factors for individual users using Attribute-Based Access Control (ABAC) (Priebe 2007) or Risk-Adaptive Access Control (RAAdAC) (McGraw 2014) can be managed by the Enterprises by proving user identities, certificates, passwords, user roles, and user locations.

Unlike other blockchain approaches, Philos™ is optimized to maximize five categories of trust, automate that process, and use dashboards to help monitor and control trust enterprise levels. The foundational cybersecurity goals of CIA are automatically monitored and maximized through the five categories: Suppression Trust, Validation Trust, Reliability Trust, Refutation Trust, and Deprivation Trust, as defined by Gorog and Boulton in 2019 (Gorog and Boulton 2019). The uniformity of trust methods across the enterprises should help enforce the uniformity of ZT applications throughout and provide an underlying support structure to support effective security. In addition, Philos™ also has a unique combination of designed-in capabilities: indefinite scalability, neutral governance, auditability, no mining or wasted energy, end-of-life planning, minimal transaction size, offline storage, individual data sales, and rapid parallel transactions.

MBSE and MBSAP are designed to deal with complex systems and can use ASE as an effective approach to dealing with dynamic changes. According to Borky and Herber (Borky and Herber 2020)⁶⁶, “An MBSE foundation is indispensable if ASE is to be consistent, measurable, high quality, and efficient.” To deal with methods to control Negative Emergence Complexity, measurability is particularly important to be able to monitor whether controls are effective.

MBSE and MBSAP are therefore not only highly supportive and synergistic with architectures based on the ESP™, secure cryptographic key provisioning, and Philos™ products but appear to be the best way to determine when the total benefits of these three applications are greater than any of them working individually.

⁶⁶ Borky, John M., and Daniel R. Herber. 2020. *Engr 667: Advanced Model-Based Systems Engineering; Session 13: Cybersecurity of System Architecture (Lecture Slides)*. Fort Collins, Colorado: Colorado State University.

5.4. Ecosystem Solutions

The proposed solutions to reduce behaviors of negative emergent complexity from malicious cybersecurity threats through embedded cybersecurity design, effective uniformity of defensive methods using BlockFrame®'s novel ESP™, secure cryptographic key provisioning, and the Philos™ blockchain Marketplace. These three products, along with effective systems engineering, will improve control structures and support DiD and ZTA. The novel BlockFrame® approach to systems engineering-based cybersecurity architectures is leading the marketplace with the first-of-a-kind third-generation Philos™ blockchain/distributed ledger concept for advanced logistics using secure cryptograph key provisioning for the IOT. The Philos™ blockchain is being designed using BlockFrame®-created inventions for the unique security management capabilities shown in Table 5.1 (Gorog and Lambert 2021). Many of the capabilities shown in Table 5.1 are also necessary to realize the system applications necessary to provide many cybersecurity services, IoT security, and network security capabilities discussed in this chapter. The fact that all three products are designed with indefinite global scalability in mind also makes them uniquely expandable. Risk identification, risk analysis, and risk reduction are integral to the systems engineering goals of measuring and reducing the financial costs of losses to cyberattacks. In other words, they will help produce an ecosystem-wide environment of positive emergence to control the recognized negative emergence of current cybersecurity threats. Positive emergence can be expected through the uniformity of actions and products designed to improve distributed system behaviors in the areas of balancing privacy and visibility, accountability, secure logistics, identification, traceability, and trust.

These ecosystem-wide technologies support several positive emergent behaviors and use cases, including the capabilities to protect sensitive government records, manage supply chains, and reduce the risks of many cybersecurity threats by using MFI to improve visibility, and to identify and prove trusted participants, components, and software. The solutions support the capability to implement practical applications of MBSE and MBSAP to complex cybersecurity systems and many use cases. Robust systems engineering will identify, replicate, and reuse successful design patterns and reference architectures to help quickly proliferate designs to broader sets of applications. BlockFrame, Inc., has unique subject matter expertise in these areas to provide consulting services and support to companies and government entities through systems engineering and risk management for their individual use cases and requirements.

One of many practical use cases to describe the application of these BlockFrame® technologies is to secure and manage energy grids for robust cybersecurity and management of transactive energy (Gorog and Lambert 2021). It is well known that modern energy grids are highly complex and high-risk targets for cyberattacks and ransom attacks of national concern. As observed in the vulnerability of the Colonial Pipeline (“Colonial Pipeline Cyber Incident” n.d.), executive orders from both President Trump (Donald Trump 2020) and President Biden (“Executive Order on Improving the Nation’s Cybersecurity” 2021), joint reports by the FBI and the National Security Agency (FBI/NSA 2018), and many other examples, United States critical energy infrastructure is known to be under threat of foreign attack, and in many cases has already been penetrated. In 2021, the US Government more specifically identified cybersecurity shortfalls due to the lack of attention to Zero Trust capabilities (“Federal Agencies Face New Zero-Trust Cybersecurity Requirements | CSO Online” n.d.) and planned to start a new office to advance the use of Zero Trust concepts (“Pentagon to Launch Zero Trust Cyber Office in December” n.d.). Even without complex analysis or modeling, these events represent strongly negative emergence from cybersecurity threats, and government reactions represent actions they hope will provide some degree of positive emergence to mitigate those threats. Unfortunately, the rapidity of penetration and destruction from cyberattacks does not yet inspire confidence that defenses are effective now or will be in the foreseeable future.

Table 5.1
Third-Generation Distributed Ledger

Generations of Blockchains and Capabilities	First Generation			Second Generation			Third Generation
	Bitcoin	Crypto Currency	Stable Coins	Ethereum	Hyper-Ledger	Stellar	Philos
No third party needed for transaction	X	X	X	X	X	X	X
Content publicly verifiable	X	X	X	X	X	X	X
Volatile speculation	X	X		X			
Publicly accessible	X	X	X	X			X
Cost effective			X			X	X
Operation of smart contracts	Smart operations and application			X	X	X	X
Control and vetting participants					X	X	X
Minimize data usage						X	X
Expandable to large scale					X	X	X
Separation of multiple industries	Governance cross-industry support and organizational management controls						X
Tracing trust of individuals							X
Separation of private data							X
Control of multiple jurisdictions							X
Licensing of transaction contracts							X
Economic indicator controls							X
Indefinite modular scalability							X
Zero-trust ledgers							X
No wasted energy consensus							X

The x's show the capabilities of each generation of blockchain

Technically, modern energy grids have highly complex routing and financial considerations with associated microgrids, net-metering requirements for alternative energy sharing with individual subscribers, requirements to maintain adequate baseloads during periods of low availability and rapid switching of alternative energy sources, and

political requirements for non-fossil fuel, non-nuclear, and non-hydro portfolios. These complex requirements are unprecedented with previous generations of power systems and suggest that unprecedented levels of smart digital support, cyber defense, and supervisory control and data acquisition (SCADA) capabilities will be required. All of these complex factors combine to create an unprecedented potential for negative emergence across the entire energy industry (Trump, 2020).

BlockFrame®'s solutions for transactive energy use cases are representative of capabilities that can be applied to many complex systems with high levels of negative emergence. BlockFrame® ESP™ devices can be used in an unlimited number of nodes and devices to collect system data and, in conjunction with blockchain management, immutably and non-repudiably record and manage that data by producing transaction contracts using non-fungible tokens (NFTs) over a wide variety of human, hardware, or software actors and events. ESP™ devices can also be used to establish zero-trust rules and securely deliver and provision software and software updates at each node of interest in complex energy grids of producer, distribution, or consumer delivery points. Figures 5.2–5.5 show a decision path for a notional energy company with distributed energy resources (DERs) to deal with transactive energy decisions in a hostile cybersecurity environment. We assume that Third-Generation BlockFrame® technologies and procedures are available to address qualitative management issues and needed capabilities:

Capability 1: Provision robust cybersecurity for DERs using NFTs

Capability 2: Collect and manage immutable forensic proof of all selected energy transaction data from DERs.

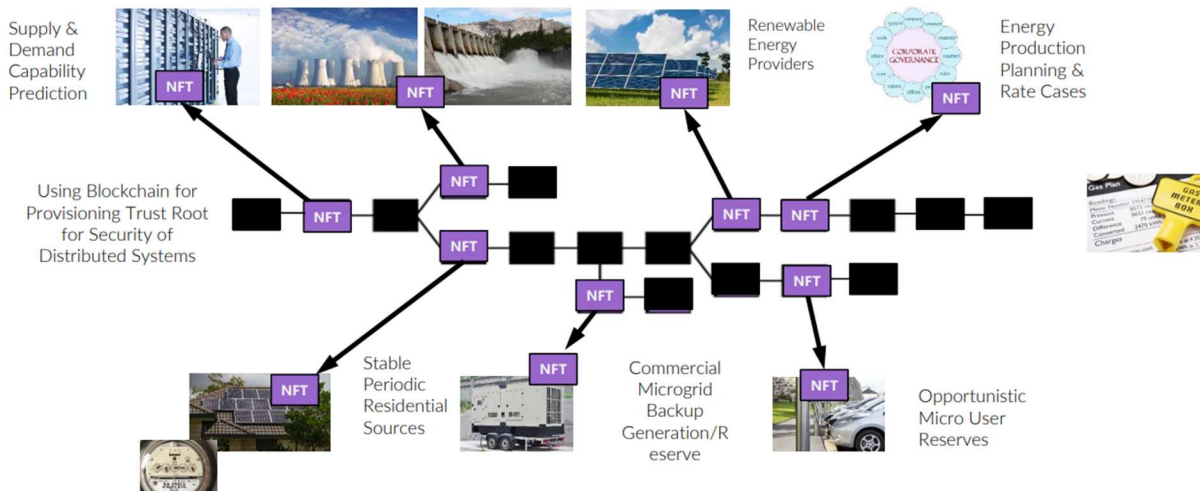
Capability 3: Scale practical blockchain management for an entire energy grid.

Each of these capabilities are currently needed to manage future transactive energy resources but are currently not available. They also address current issues that represent negative emergence. Creating these capabilities to mitigate current problems can also represent a process of positive emergence.

To address Capability 1, BlockFrame proposes using NFT technologies to provide robust cybersecurity capabilities to secure DERs. Figure 5.2 (Gorog and Lambert 2021) shows that the same NFT transaction contract tokenization can be used throughout energy enterprises, thereby reducing the risks of negative emergence. This NFT process through BlockFrame® ESP™ devices is approximately analogous to the definitions and functional behaviors of Emergent Behavior Observers (EBOs), as described in (Mittal and Rainey 2015) and (Zeigler 2019), with the

addition of other security, encryption, provisioning, and recovery functions that allow behaviors that might be better described as Emergent Behavior Controllers (EBCs).

Blockchain for Distributed Energy Resources Security

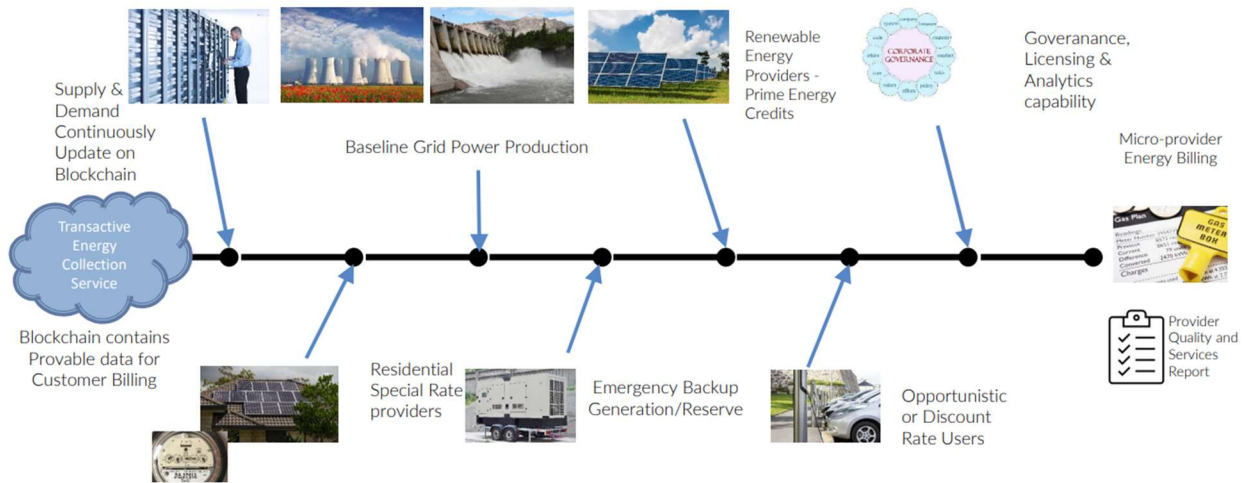


Capability 1 Solved: Non-Fungible Token Provisioning of Security

Figure 5.2. Using NFTs to provision security

The challenge of proving valid energy flow, and therefore valid energy transactions, is shown in Figure 5.3 (Gorog and Lambert 2021). Both security and management can be extended to provide provable, reliable, and trusted management of financial data when delivered from various nodes of DERs, including energy exchanges, energy credits, and interactive consumer participation. This would typically involve cyber-physical ESP™ devices to establish hardened zero-trust processes and then establish highly encrypted asynchronous communications through secure cloud services among all devices and their networks. Collected data would then be managed by the secure blockchain system, so all energy flow data and their associated costs would be indelibly recorded on the blockchain for accurate proof of valid transactions. Transaction contracts would also be non-repudiable as their processes would record all necessary data to prove their validity.

Unchangeable, provable blockchain data of each transaction



Capability 2 Solved: High data quality for direct customer billing

Figure 5.3. Collecting immutable and non-repudiable data for customer billing

High levels of security and provable data for customer billing are essential for the sustainability of viable energy systems. However, highly distributed systems must also be scalable to the appropriate size, locations, and the volume of data collected by modern transactional energy grids. Previous tests and studies of the scalability of blockchains (Gorog and Boulton 2019) show that unless the blockchain is capable of storing massive volumes of transaction data, it is impractical to store those data on distributed servers and still maintain the performance requirements of the blockchain itself. Those tests and operational requirements led to a bottom-up development of the Philos™ blockchain, which is based on low-cost, energy-efficient, and indefinitely scalable management of blockchain operations. The third-generation blockchain capabilities shown in Figure 5.4 (Gorog and Lambert 2021) demonstrate BlockFrame®'s approach to solving those scalability problems, which are necessary for practical automated management of secure and effective financial management of complex distributed power grids.

Applying indefinity – scalable

3rd generation distributed ledgers to DREs

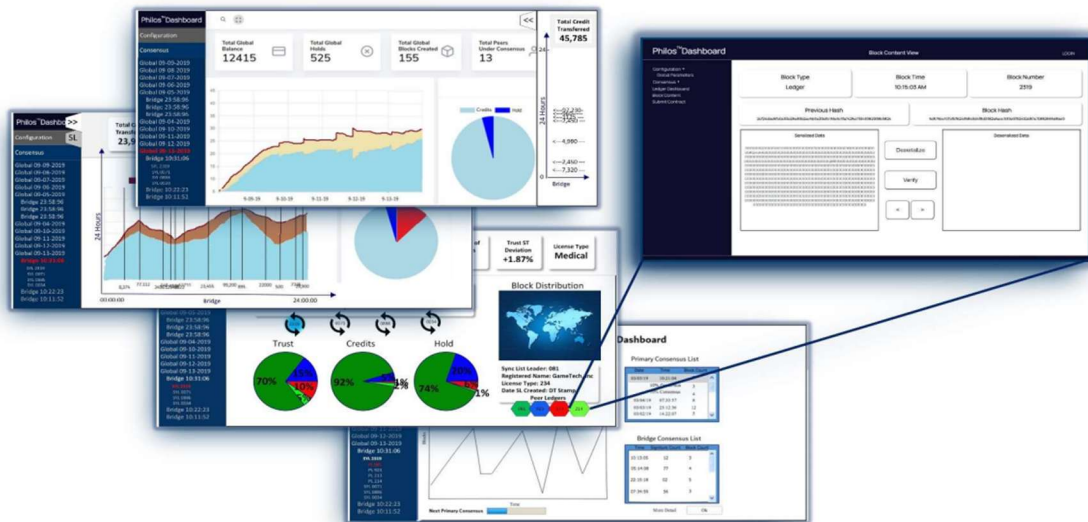
Generations of BlockChain and Capabilities	First Generation			Second Generation			Third Generation
	Bitcoin	Crypto-currency	Stable Coins	Ethereum	Hyper-ledger	Stellar	Philos
No 3rd Party needed for Transaction	x	x	x	x	x	x	x
Content Publicly Verifiable	x	x	x	x	x	x	x
Volitile Speculation	x	x		x			
Publicly Accessible	x	x	x	x			x
Cost Effective			x			x	x
Operation of Smart Contracts	Smart Operations & Application			x	x	x	x
Control and Vetting Participants					x	x	x
Minimize Data Usage						x	x
Expandable To Large Scale					x	x	x
Separation of Multiple Industries	Governance, Cross industry Support, & Organizational Management Controls						x
Tracing Trust of Individuals							x
Separation of Private Data							x
Control of Multiple Jurisdictions							x
Licensing of Transaction Contracts							x
Economic Indicator Controls							x
Indefinite Modular Scalability							x
Zero-Trust Ledgers							x
No Wasted Energy Consensus							x

Capability 3 Solved: Transactions on scale required for energy tracking

Figure 5.4. Applying indefinitely scalable third-generation distributed ledgers to manage DRE

Once established, networks using BlockFrame® technologies provide new management capabilities for energy providers, as represented in Figure 5.5 (Gorog and Lambert 2021). As energy producers must plan for co-mingling every “color” of power—and the corresponding “color” of its monetary value—from traditional and alternate forms of energy, from consumer-generated suppliers, and from other producers—they must be able to prove those details to government regulators, holders of energy credits, and consumers. With cryptographically secure collection devices, networks, and blockchains used for management and storage, these problems can be solved even in highly complex systems.

Philos™ Visibility Dashboards for Energy Delivery Usage and Security Management



BlockFrame® patent pending third generation distributed Ledger

Figure 5.5. Applying indefinitely scalable third-generation distributed ledgers to manage DRE

Because the Philos™ blockchain does not require expensive “data mining,” it maintains very low energy-use and very low costs to operate. The combination of inexpensive NFT-based transactions, blockchain-based security management, and the ability to scale to widely distributed energy grids can streamline automated management, provide essential services at low costs, and save money for energy suppliers. Some of the resulting benefits of these new approaches are:

- Apply blockchain and NFT technology for IoT security
- Carbon-neutral blockchain without wasted energy consumption
- Security and privacy are integrated industry wide
- NFT technology is used for individual monetization of every transaction
- Blockchain saves money for utilities and consumers
- Immutable and non-repudiable transaction contracts are used for payment records
- Pre-purchase consumption as a service capability
- Automated billing using utility tokens
- NFTs to track types and categories of energy use credits
- Peer-to-peer trading among co-ops
- Track rate changes and DER energy transaction history
- Use ESP™ proof-of-origin capabilities to verify communications from DERs

- Create geographical maps of grid events and DER origins
- Create a view to see DER transactions vs. energy rate changes
- Quality and effectiveness reports, and customer trends and patterns
- Reports and accurate billing for consumers
- Trend analyses
- Accurate rate cases
- Forensic usage tracking
- Accurately track and manage carbon credits
- Easy tracking of reserve access
- Tracking of emergency power shedding

5.5. Modeling Examples of Positive and Negative Emergence in Cybersecurity Environments

MBSE analysis and tools can be effectively applied to model the effects of negative emergence from cybersecurity threats and controls to produce positive emergence. Many benefits are derived through these approaches, including:

- Visualizations of SoS architectures, and their subsystems and components
- Definition of specific functional behaviors
- Verification of behaviors with engineers, programmers, and SMEs
- Validation of functional processes by designers, stakeholders, investors, and SMEs
- Replications of reference architectures for reuse and more rapid development
- Discovery of system flaws and vulnerabilities
- Identification of points and processes of risk, calculation of risks, and risk mitigation

For this chapter, the United States Naval Postgraduate School (NPS) Monterey Phoenix (MP) tool MP-Gryphon (Auguston 2009a b) is used to demonstrate examples of negative and positive emergent behaviors in a selected cybersecurity framework. The use of MP can also be compared to other approaches in this book. Several assumptions will be made for the purposes of the scenarios (Schema) being modeled:

- Monterey Phoenix is designed and used as a minimalist model that is not intended to model events or behaviors at refined levels of detail, such as all engineering processes within systems of systems, but is more useful to display an exhaustive number of high-level behavioral outcomes for examination by analysts, engineers, and stakeholders to qualitatively verify and validate those behaviors.
- While some “simulation” is possible, a relatively small number of iterations (or “Scope”) are usually used to allow visualization for the user, which allows users to make relatively rapid changes to the modeled behaviors while still providing a scope-complete set of behaviors. This MP design and operation are based on the Small Scope Hypothesis (Jackson 2012), which assumes that “most flaws in models can be demonstrated on small counterexamples” (Giammarco and Auguston 2019). The relative simplicity of this approach is intended to discover “unknown unknowns” through rapid visualizations that can be communicated with verification and validation groups to discover design flaws or opportunities for process improvement.
- MP can be used as a commonly accepted approach for basic emergence research. The software is jointly maintained by NPS and the National Security Agency (NSA) and is gaining exposure and acceptance within Department of Defense modeling communities. While it is not intended to provide a comprehensive MBSE model, it is a useful behavioral analysis tool that if used consistently will provide ontologies and processes that can be transferable and understandable within diverse academic and topical settings.

- Emergence theory is inherently about chaotic environments where there may be little information about actual causes and effects. As such, providing high levels of resources in attempts to model high levels of detail can be both futile and expensive because the subject is known to be chaotic outside the realm of methods to control it, or even to understand it in detail. Because of the relatively simplified Monterey Phoenix focus on logical events and behaviors, it is more of a “what if” approach to validate functions at a relatively high level to allow analysts and stakeholders to visualize high-level trends and effects, and to examine and “think through” alternatives fairly rapidly. Typically, MP will be used to examine trees of logical alternatives, but once defined, the model will automatically and exhaustively (up to the Scope set by the user) produce event trees (traces) that can be examined for unexpected outcomes. In short, the MP approach to examine emergence is to logically discover and help SMEs think through disconnects that might cause unexpected or strong emergence, and attempt to mitigate negative emergence.
- This chapter will assume that a very simplified representation of a generic blockchain process can be used as a surrogate for more advanced systems that might be vulnerable to cyberattacks. This might be only one of many such analyses for complex systems like transactive energy enterprises. For analysis purposes, we will examine hypothetical situations with high risk and low levels of cyber security to help a method to identify subjective levels of negative and positive emergence and risk. This approach does not attempt to replicate the high levels of capability in advanced blockchains like Philos™ but is purposely focused to examine cyber risks as negative emergence and the need for advanced cyber protections (as positive emergence controls). More detailed models of actual blockchains are not included for several reasons, including possible exposure of proprietary processes, unintended exposure of possible security flaws, or possible liabilities of unfair comparisons with actual products.
- Monterey Phoenix has some capabilities for data generation and analysis, and those are in the process of being expanded. For this chapter, MP will not be used for parametric sensitivity analysis. Selected MP behavioral diagrams (traces) will be examined, but hypothetical compound probability analysis and parametric sensitivities will be calculated and displayed graphically using Microsoft Excel for convenience.

Monterey Phoenix v4 Gryphon GUI v1.0.0 (MP-Gryphon) was used to produce behavioral diagrams (traces) to model a very generic blockchain process to demonstrate normal blockchain operations when effective cybersecurity is assumed to be in place, and then to exhaustively demonstrate events where attacks might occur if defenses fail. At Scope = 1, one trace was generated, representing one pass through the blockchain build cycle without successful threats. That example process is shown in Figure 5.6. In simple terms, the process starts by collecting data that users want to add to and store on the blockchain. Cooperative methods merge data from each user and combine those data with data from many other users, sometimes with high levels of separation of distance, function, management, jurisdiction, and security. At the end of the process, all the data from the many users will merge (through consensus decisions by the users) into a new block which is added to all the previous historical blocks in the blockchain.

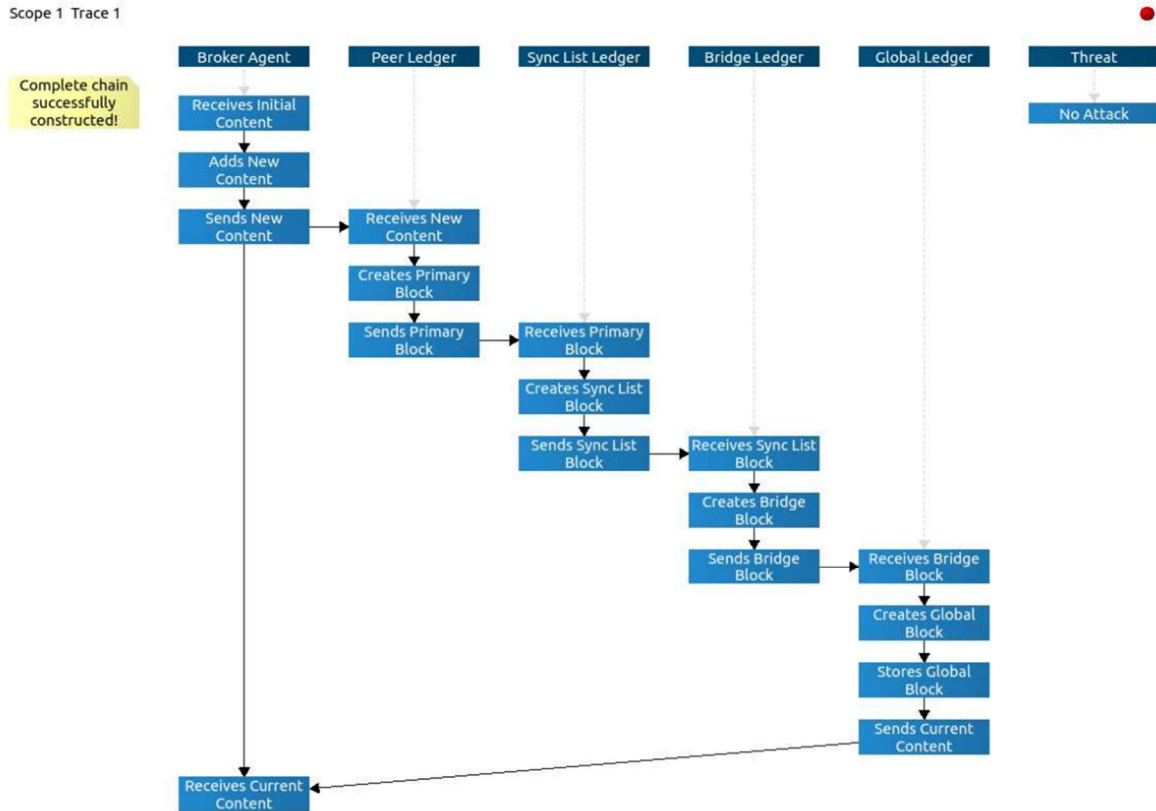


Figure 5.6. Successful build of blockchain on the first cycle with no successful cyberattacks

At Scope = 2, Monterey Phoenix automatically generated 16 traces, representing two blockchain build cycles. Of the 16 traces, 2 represented that the blockchain was successfully constructed, and 14 represented successful cyberattacks if defenses failed at particular events. Figure 5.7 represents two passes through the blockchain build cycle without successful threats, while Figure 5.8 represents a failure of cyber defenses while a Sync List Ledger is attempting to build a new Sync List Block during the second cycle of a blockchain build process.

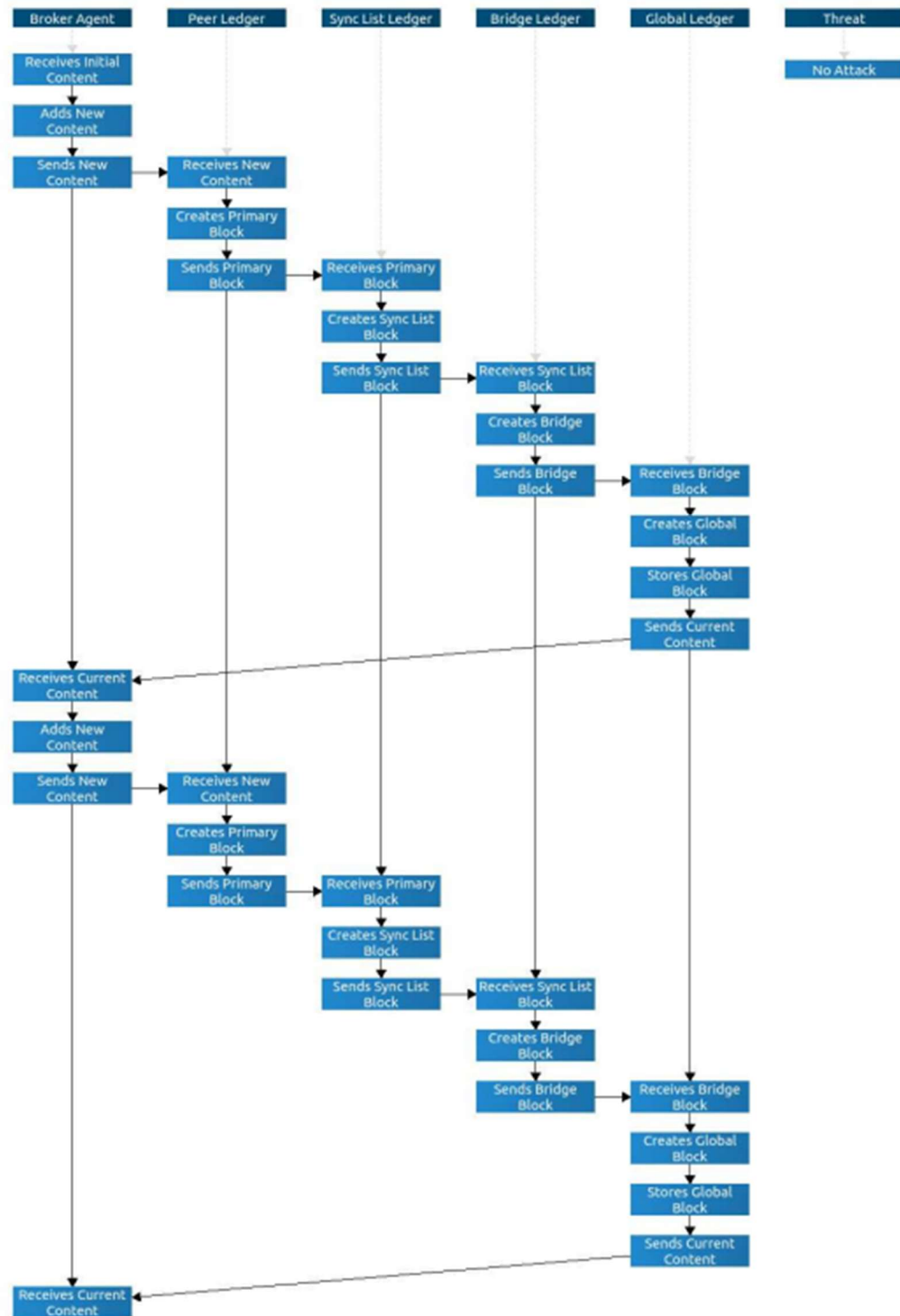
Complete chain
successfully
constructed!

Figure 5.7. Successful build of blockchain through the second cycle with no successful cyberattacks

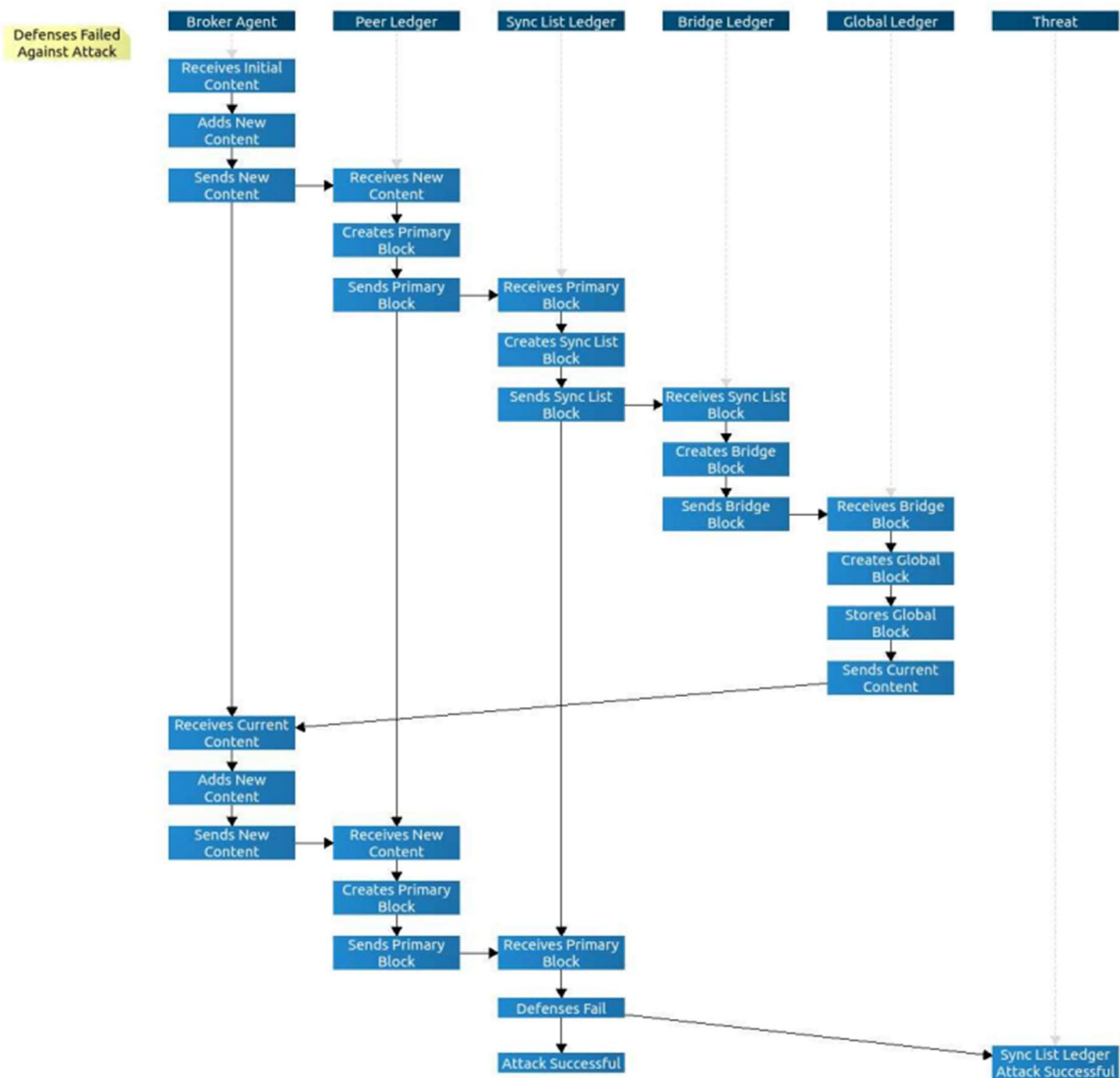


Figure 5.8. Failure to build blockchain on the second cycle due to successful cyberattack on sync list ledger

At Scope = 3, 44 traces were generated, representing three blockchain build cycles. Of the 44 traces, 3 represented the blockchain as being successfully constructed, and 41 represented successful cyberattacks if defenses failed at particular events. Figure 5.9 represents three passes through the blockchain build cycle without successful threats, while Figure 5.10 represents a failure of cyber defenses while a Broker Agent is attempting to Add Content to New Block.

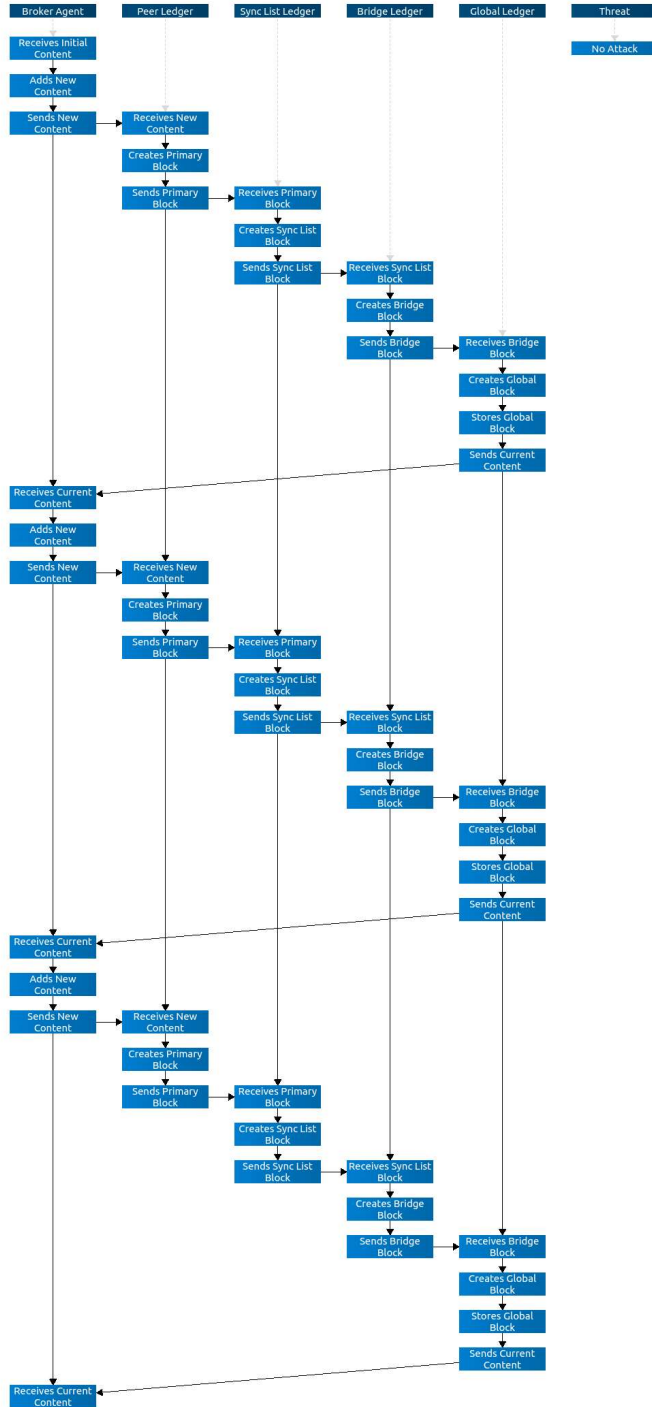
Complete chain
successfully
constructed!

Figure 5.9. Successful build of blockchain through the third cycle with no successful cyberattacks

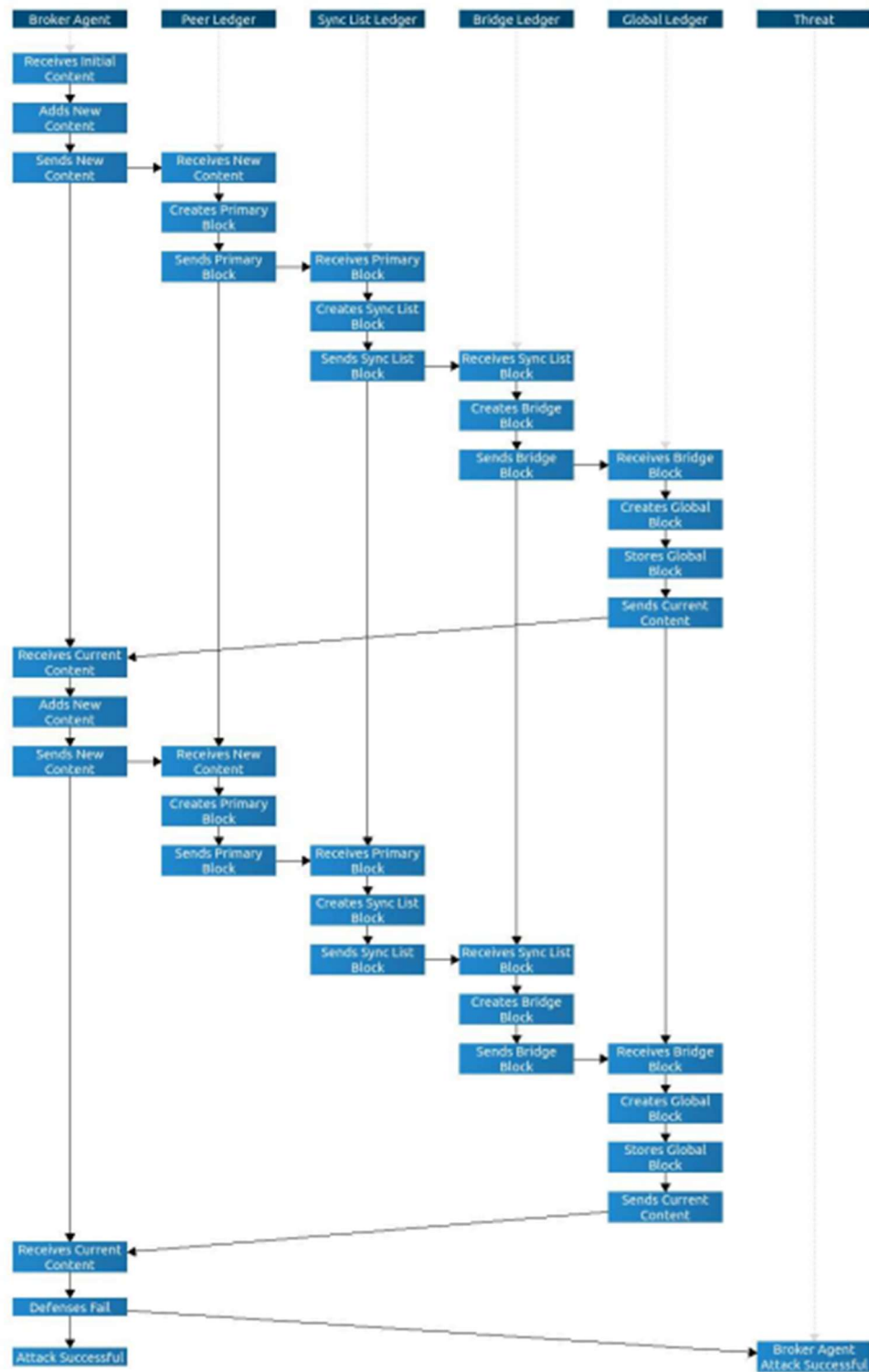
Defenses Failed
Against Attack

Figure 5.10. Failure to build blockchain on the third cycle due to successful cyberattack on broker agent

In all, Monterey Phoenix may generate hundreds or thousands of event traces representing different behaviors, depending on the logical variations of interest to analysts, but the concept of the Small Scope Hypothesis suggests that examination of that many alternatives is probably past a point of diminishing returns and could be a waste of time for knowledgeable analysts to discover whether or not significant negative or positive emergence is observable. When trying to discover and mitigate negative emergence in complex chaotic systems, this model-assisted analytic process could be iterative to discover and correct a variety of factors like unexpected design flaws, external attacks, unexpected results from a validation process, or any “unknown unknowns” that might produce unexpected behaviors or negative emergence. If unexpected or undesirable behaviors are discovered, designers could attempt to mitigate them or move to model other subsystems or components of interest—and to continue the iterative process as they desire.

Just such an example occurred when an early version of this MP model of this Schema showed a Broker Agent being attacked during its construction of the initial block—shown in Figure 5.11. Instead of halting the construction, the logical model of the “expected” behavior sent corrupted data on to the next step in the blockchain process. While this was discovered to be due to a logical error in the initial modeling process of the sequence of behaviors, it could represent the many kinds of errors that sometimes do occur in real-life design due to unintentional programming errors, inexperienced programmers writing poor quality “spaghetti code,” misunderstandings between development teams, the acceptance of unvetted off-the-shelf code, or errors in development/security/operational (DEVSECOPS) design itself. Paraphrasing Murphy’s Law, “If it could happen, it will happen!” In this case, the impact of that little error could have been the disastrous compromise of the core function of any blockchain, the perfect proof of mathematical security. Verification of behaviors like this is exactly how Monterey Phoenix itself identified unexpected emergence while writing this chapter, exposed that behavior, and allowed it to be corrected.

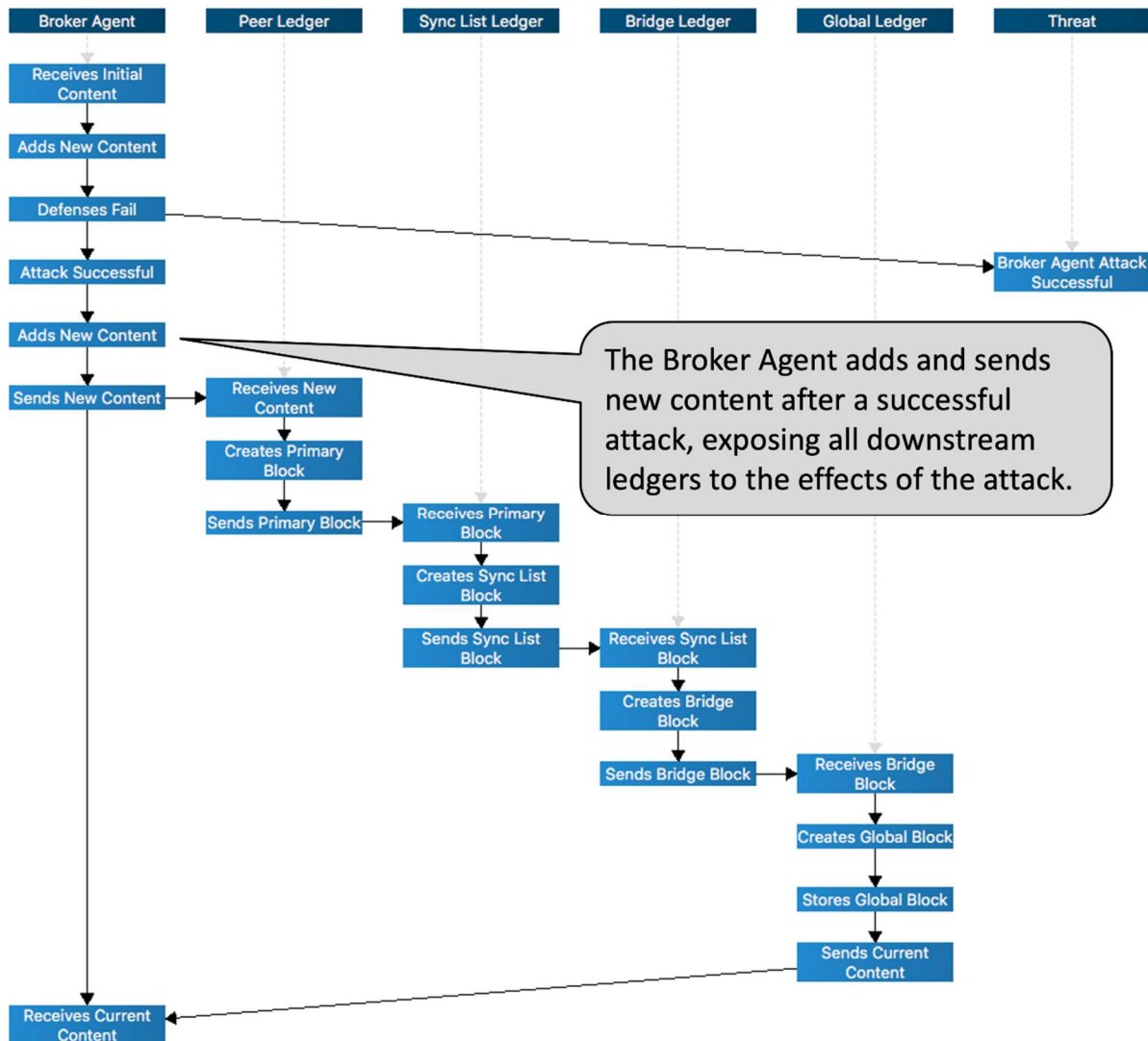


Figure 5.11. Successful cyberattack on broker agent where MP diagram discovered emergent behavior in system design

Many other processes could be modeled for the same purpose to identify emergence, but because of the multilayer architecture of blockchains, this Schema was selected to show how a system of systems will have various layers of attack surfaces that are vulnerable over time. Successful mitigation of risk must consider an adequate sampling of potential risk events over time and introduce effective procedures that will lower risks to a manageable level. When cyber security risks (as forms of negative emergence) are reduced to the point where they are understandable and controllable, emergence could be considered relatively more positive. The scenario of an attack on a Broker Agent at Figures 5.10 or 5.11 may be of particular interest to analysts as a likely point of attack because it might be the most exposed to external users through application programming interfaces (APIs) and involves

more human factors that are open to compromise or error. Although other points of attack are possible, they are probably less likely because they are not as exposed and are totally automatic machine-to-machine interfaces, and the data will be encrypted throughout the block-building process if a particular blockchain (like Philos™) is designed to do that. Most blockchains are not designed to encrypt data in transit but may leave source data exposed throughout their life cycle. As shown in Table 5.1, many of those shortfalls are recognized and mitigated through the design of the Philos™ Blockchain. In the Monterey Phoenix example and Schema and scenarios shown here, we assume that all Philos™ capabilities could be included in the model and that “near perfect security” could be attainable. In this simple example, if analysts are satisfied that the threats shown in Figures 5.10 and 5.11 are in fact more interesting than other traces, they have the option of studying them in more depth, expanding the simulation beyond two or three iterations to many iterations, testing new behaviors like new cybersecurity capabilities, or conducting a variety of other logical tests.

5.6. Quantifying and Prioritizing Cybersecurity Responses

Once a particular attack point is identified and its characteristics and environment are at least partially known, analysts may use simple methods to quantify and compare the relative effects of interest. In the example in Figure 5.10, if a relatively high potential point of vulnerability is discovered, analysts might want to measure the potential risk of that vulnerability. In a highly chaotic environment, attempting to simulate complex cyberattacks in detail might not be worth the effort if simple calculations are able to answer the question of whether defenses are robust enough to contain or mitigate the complexity and costs attackers might incur, or whether users think that their defenses are “robust enough” to mitigate their risk to a manageable level. Referring back to Figure 5.1, that might be described as moving from out-of-control “strong” or “spooky” negative emergence to a point where emergence complexity can be controlled within the “knowledge boundary” where known methods of mitigation can be used.

If quantitative probabilities of attack success can be estimated, based on engineering, historical, or even anecdotal examples, the magnitude of a probability of survival of the target of the attack could be calculated using the formula:

$$P_{\text{survival}} = (1 - P_{\text{kill}})^N \quad (5.1)$$

where the survival of any system or component is simply $1 - P_{\text{kill}}$ for each attack. As a simple example, if the probability of “kill” from a single attack attempt is 0.05, the probability of “survival” from that single attack is $(1 - 0.05)$ or 95%. However, if multiple attacks were attempted at the same probability of kill, the probability of the target surviving may rapidly diminish. At $n = 10$ attacks, or $n = 100$ attacks, the probability of the target surviving falls to about 0.60 and 0.006, respectively.

For a cybersecurity example, a “kill” could be from many mechanisms, including denial of service, physical destruction, theft, or a successful ransom of data. Some “kills” might also be undetectable to users, which is the basis of many reports of cyberattacks that may conclude that the target was under attack for 9 months and never knew it. Whatever the mechanism, SMEs might have to make some “well-informed guesses” as to the potential relative risk of attacks and quantify the relative vulnerabilities of system components over time. In an environment where cyberattackers have diverse tools and methods to attack over time, they also force defenders into requirements to maintain almost perfect defenses in order to sustain themselves over a long period of time. However, if defenders can identify their most likely relative vulnerable points of procedures, they should also be able to make their defenses

more robust over time. For example, if a defender can apply techniques to lower the attacker's single attack probability of kill to 0.00001, even after 100 attacks, the probability that the target survives is 0.99900.

5.7. Applying Monterey Phoenix to Qualitative Cybersecurity Logic

Although attaining a perfect defense is theoretically impossible, a robust unified zero trust defense should strive to attain a low enough risk level over time as a quality requirement for all system components, which could deter attackers from expending their time and cost to attack in the first place. The most important consideration is that risk might not have to be reduced to zero if defenses are adequate to make attacks too expensive or too vulnerable to counterattacks. Admittedly, those decision factors are somewhat subjective and qualitative, but they might be sufficient with decision-support models like Monterey Phoenix to help defenders examine how much effort is enough to defeat the expenditures and threats made by their attackers. The use of cryptographic provisioning devices like the ESP™ and blockchain systems like Philos™ that are designed to monitor and improve the quality of trust factors used by individual users can regulate that quality by peer pressure from other users through incentives for quality compliance over time.

In combination with modeling behaviors within system architectures, Monterey Phoenix could also be used to model behaviors of defenders to compare cybersecurity threats and defensive strategies over time. Without robust defenses, victims of cyberattacks could experience the effects of attacks over extended periods of time, might not know they are being attacked, and suffer effects that include data breaches, loss of customer data, penetration into core company functions, disruption or permanent destruction of data or operations, monetary losses, or ransom attacks. Using models generated from Monterey Phoenix to examine and think through those problems should help identify and control many destructive and urgent problems first, which might not be practical even with larger and more expensive approaches.

5.8. Summary

Additional context for this Chapter 5 was included in Emergent Behavior in Systems of Systems Engineering – Real-World Applications (citation), by Dr Larry Rainey, Editor, and O. Thomas Holland, Georgia Tech Research Institute, as included Chapter 11, “Lessons Learned and the Proposed Way-Ahead,” p. 253:

- *Cybersecurity threats can be characterized with the terminology of complex negative emergent behaviors that are difficult to control using existing processes and technologies.*
- *BlockFrame, Inc., has demonstrated unique processes and technologies that address current cybersecurity vulnerabilities using a system of systems suite to provide defense-in-depth and zero-trust architectures using novel technologies, including cryptologic provisioning devices, highly scalable registries, and indefinitely scalable blockchains/distributed ledgers to manage trust and security through visible dashboards and controls.*
- *Model-based systems engineering can be used to characterize and quantify the effects of both negative and positive emergent behaviors in the cybersecurity environment. The Naval Postgraduate School Monterey Phoenix Model can be used to describe high-level functional behaviors of normal operations, operations under attack, and processes to mitigate cyber threats.*

5.9. Context of Emergence and Complexity Applied to BlockFrame[®] Design and Development

Dynamic and Non-linear Complexity Concepts

The purposes for the ESP[™] and/or the Philos[™] blockchain/distributed ledger Enterprises are largely to counter or control the effects of negative emergent behaviors in a large variety of use cases that have been identified both by the State of Colorado⁶⁷, and the Office of the Secretary of Defense through the Cybersecurity Maturity Model Compliance (CMMC)⁶⁸. The predecessor to the CMMC was a DoD compliance mandate since September 2017, but from informal discussions with The Mitre Corporation, no company to-date has ever demonstrated full compliance. The CMMC was released in January 2020, became a “mandatory” compliance regime the day that it was issued, and is comparatively more demanding than its predecessor regulations. For the purposes of this research, it is assumed that CMMC⁶⁹ is a good surrogate for those federally mandated cybersecurity requirements. For this research, details of every individual CMMC Requirement are described in Appendix C. It is important to recognize that many specific problems with cybersecurity and trust within digital records and supply chains are well-known and recognized as highly complex.

The current digital environment is observed to be highly complex for the following reasons:

- Sheer size: due to the sheer size of transactions within the environment of the Internet and World Wide Web, threats to cybersecurity and supply chains are well-documented.
- Relationships: financial and product delivery are ubiquitous to customers worldwide.
- Heterogeneity: businesses and products cover every type of industry and financial organization, across continents, legal jurisdictions, and all varieties of software.
- Nonlinearities: exponential growth and economic and financial reliance have made the security of the digital environment complex, indispensable, and vulnerable.

⁶⁷ Lambert et al., “Colorado Senate Bill SB-086, ‘Cyber Coding Cryptology for State Records.’”

⁶⁸ Carnegie Mellon University and the Johns Hopkins University Applied Physics Laboratory, LLC, “‘Cybersecurity Maturity Model Certification’ Report for the Office of the Secretary of Defense.”

⁶⁹ CMMC is primarily designed for compliance to unclassified but sensitive government requirements and is therefore more appropriate for unclassified academic research. The Mitre Att&ck Industrial Control System (ICS) Matrix is a more rigorous compliance regime but is very similar in structure to CMMC. That regime is prominently mentioned in Chapter 8 within IEEE papers as it has more applicability to government or classified environments and use cases. Due to its size and level of detail, further discussion of CMMC requirements is presented in Appendix C.

- Human variables: threats from malevolent individual and state actors have produced a constant dynamic spiral of pre-meditated threats, counter-threats, breaches, and extortion.
- Knowledge limitations: The best subject matter experts (SMEs) in cybersecurity and supply chains have been constantly hired to improve security in the current environment, but major “unanticipated” hacks have continued daily⁷⁰.
- Exogenous factors: Threats cross all international, legal and political boundaries, and target all social groups.

Mittal and Rainey⁷¹ describe emergence and emergent behaviors. There, the World Wide Web is specifically described as a “widespread complex system in everyday use,” as stated above, which has brought “widespread attention” to “emergence... for the past two decades.” Rainey defines complexity as, “a characteristic of all modern systems which means the number of multi-level interactions among the components or elements of [the] system.”

From these foundations, we assume that current cybersecurity and global supply chain environments should be considered as having “emergent behaviors” that need to be controlled or contained. Saurabh and Rainey provide several definitions and rationale of why emergence is important. To summarize this report, Mittal’s Figure 4, from Mittal and Rainey⁷², shown in this chapter as Figure 5.12, illustrates part of the problem. “Deterministic” systems are considered as “simple” or “weak” levels of complexity that might be addressed by “traditional” engineering means like control theory, operations research, network theory, etc. Even “strong” emergence could be addressed at least partially with subject matter experts. When complexity exceeds those levels, into the “spooky levels,” highly stochastic levels of complexity, begin to exceed the abilities to control its effects. For the sake of discussion, we could acknowledge that uncontrolled emergence could have good “positive emergence” properties (like constantly growing computer capabilities at lower costs), or they could have bad “negative emergence” properties (like the rapid growth of cyber ransomware that we mentioned above). Once “negative” emergence is discovered, the typical goal

⁷⁰ In Colorado, a direct attack occurred through Epiq one of the foundational tools of the criminal justice system, the e-Discovery system. See Ready, “Epiq’s Ransomware Response Is by The Books, But Dangers Still Lurk,” and Lambert and Gerou, “Colorado Senate Bill SB14-190. ‘Statewide Discovery Sharing System’.”

⁷¹ Mittal and Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering. SCSC’15, Summer Simulation Multi-Conference, Chicago, IL.”

⁷² Mittal, Saurabh, and Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering. SCSC’15, Summer Simulation Multi-Conference, Chicago, IL.”

is to identify “stochastic” behaviors, look for opportunities and capabilities to make them more “deterministic,” and thereby control or reduce the risks of the System.

The original language of Mittal and Rainey’s diagram is useful to understand Figure 5.12: “Figure 4. Emergence Complexity Cone linking emergent behavior taxonomy in increasing complexity on the y-axis, division of Cone into stochastic and deterministic search-spaces on the x-axis, with Cone volume depicting the variety. Cone perimeter as constraints, and the knowledge boundary as a cylinder that addresses variety and constraints. Knowledge cylinder around simple and weak emergence in the deterministic domain signifies ample knowledge available to develop abstractions”⁷³.

Relative to Figure 5.12, this research asserts that: cybersecurity and supply chain security currently lie within the “spooky” or highly stochastic level of complexity. The question is whether systems engineering can be used to discover “stochastic” properties and design enough solutions to convert the highly stochastic “negative emergence” effects into more predictable and controllable deterministic properties that are sufficient to bring that negative emergence under control.

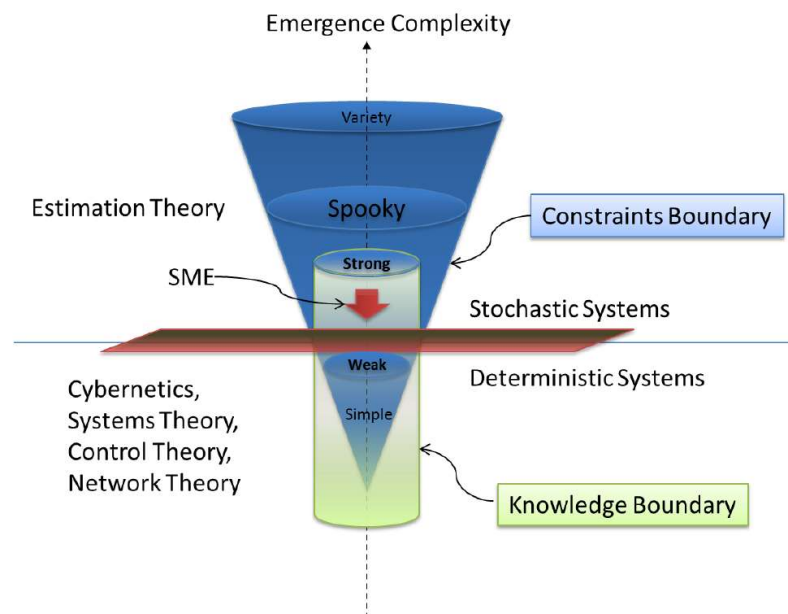


Figure 4. Emergence Complexity Cone linking emergent behavior taxonomy in increasing complexity on the y-axis, division of Cone into stochastic and deterministic search-spaces on the x-axis, with Cone volume depicting the variety, Cone perimeter as constraints, and the knowledge boundary as a cylinder that addresses the variety and constraints. Knowledge cylinder around simple and weak emergence in the deterministic domain signifies ample knowledge available to develop abstractions.

Figure 5.12. Mittal/Rainey’s Figure 4⁷⁴.

⁷³ Mittal, Saurabh, and Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering. SCSC’15, Summer Simulation Multi-Conference, Chicago, IL.”

We assume that cybersecurity and supply chain security already lie within the “spooky” or highly stochastic level of complexity. The question is whether systems engineering can be used to discover “stochastic” properties and design enough solutions to convert the highly stochastic “negative emergence” effects into more predictable and controllable deterministic properties that are sufficient to bring that negative emergence under control.

For the immediate applications of ESP™ and Philos™, that question translates into whether they can satisfy the Requirements that are necessary for their Stakeholders. After Requirements are identified, the next logical step is to expand that Requirements set to include all the Requirements of CMMC to determine whether those Requirements can all be satisfied by the ESP™/Philos™ design. The methodology selected to do that is to create a Satisfy Requirement Matrix, and to exhaustively map every identified Requirement, and every CMMC Requirement into the ESP™/Philos™ capabilities matrix to see if all Requirements can be satisfied. If all capabilities can be shown to satisfy the CMMC Requirements criteria, they could also address many of the negative complexity and emergence issues. This methodology could be a significant step toward satisfying federal and DoD cybersecurity concerns and possibly set a pattern for other Systems to demonstrate compliance.

Due to the large amount of complexity of the BlockFrame® enterprises, some very detailed examples of a SysML Satisfy Requirements Matrix are included in Appendix C. An important caveat is that the Requirements shown in Appendix C as being satisfied are based solely upon my personal subjective judgment of what the ESP™/Philos™ Enterprises *should* do, and have *not* yet been validated by engineers, programmers, and physical test and evaluation. In the future, this pattern should be used as a template for development engineers to conduct actual operational tests and evaluations in the future to prove and validate performance, preferably within simulated and operational hostile cyber-threat environments.

Due to the large size and layout of the MBSE tool requirements matrix, three tables are included in Appendix C. Table C.6 shows a list of previously defined design Requirements. Table C.7 shows a summary list of all the CMMC Requirements that were added. Table C.8 is a fragment of an example strip of screenshots of the ESP™/Philos™ Satisfy Requirement Matrix, showing ESP™/Philos™ Structural elements along the top, and Requirements along the left axis.

⁹⁸ Mittal, Saurabh, and Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering. SCSC’15, Summer Simulation Multi-Conference, Chicago, IL.”

Within the Satisfy Requirements Matrix, it is validated that every Requirement, including CMMC compliance Requirements, can be mapped to Enterprise capabilities, and to the ESP™/Philos™ capabilities that should satisfy each requirement. The Satisfy Requirement Matrix is also an immediately useful design tool to match each design team to the CMMC and other Requirements they would be expected to satisfy and provide objective goals to prove those levels of satisfaction have been realized.

5.10. Conclusions for Research Question 1

Upon completion of this research, I am prepared to answer Research Question 1 and its Tasks. I conclude that Emergence Theory does provide a useful framework and ontology to describe options to better control complex, chaotic cybersecurity environments that are dominated by negative emergent behaviors. Using modeling and analysis of emergence and models like Monterey Phoenix can help demonstrate, test, and compare useful cyber defensive concepts.

This chapter addresses the concepts of negative and positive emergence with practical examples of digital operations within hostile cybersecurity environments. Cybersecurity has been shown to be highly chaotic and represents cases of negative emergence that impede normal and expected operations beyond the point of subject matter experts' abilities to apply known controls. BlockFrame, Inc., has invented and designed technologies and procedures that are specifically intended to help control the negative emergence of malicious cyber threats and to add controls that can result in positive emergence. These processes are first explained theoretically and then demonstrated in one practical example of the requirements of a notional energy company. The usefulness of MBSE is demonstrated by examples using the Naval Postgraduate School Monterey Phoenix tool to show normal operations of generic blockchain design behaviors and then compares several possible points of attack by generic cyber attack behaviors. Examples of relatively simple scenarios are compared using the Small Scope Hypothesis to demonstrate how behavioral modeling can be useful to support decisions to think through needed cyber defenses and capabilities, as well as assisting in model verification and validation and the discovery of unexpected emergent behaviors.

5.11. References for Chapter 5

- Auguston, Mikhail. 2009a. "Software Architecture Built from Behavior Models." *ACM SIGSOFT Software Engineering Notes* 34 (5): 1–15.
- Auguston, Mikhail. 2009b. "Monterey Phoenix, or How to Make Software Architecture Executable." Presented at the OOPSLA '09/Onward Conference, OOPSLA Companion, pp. 1031–1038, October.
- Borky, John M., and Thomas H. Bradley. 2019. *Effective Model-Based Systems Engineering*. 1st ed. Cham: Springer.
- Borky, John M., and Daniel R. Herber. 2020. *Engr 667: Advanced Model-Based Systems Engineering; Session 13: Cybersecurity of System Architecture (Lecture Slides)*. Fort Collins, Colorado: Colorado State University.
- "Colonial Pipeline Cyber Incident." n.d. Energy.Gov. Accessed November 27, 2021. <https://www.energy.gov/ceser/colonial-pipeline-cyber-incident>.
- Commission on Enhancing National Cybersecurity. 2016. "Report on Security and Growing the Digital Economy."
- "Cyberspace Solarium Commission." n.d. United States of America. <https://www.solarium.gov/>.
- "Domain Name System." n.d. Wikipedia. Accessed May 18, 2020. https://en.wikipedia.org/wiki/Domain_Name_System.
- "Executive Order on Improving the Nation's Cybersecurity." 2021. The White House. May 12, 2021. <https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>.
- FBI/NSA. 2018. "Alert (TA18–074A) Russian Government Cyber Activity Targeting Energy and Other Critical Infrastructure Sectors." <https://www.us-cert.gov/ncas/alerts/TA18-074A>.
- "Federal Agencies Face New Zero-Trust Cybersecurity Requirements | CSO Online." n.d. Accessed November 27, 2021. <https://www.csoonline.com.cdn.ampproject.org/c/s/www.csoonline.com/article/3632878/federal-agencies-face-new-zero-trust-cybersecurity-requirements.amp.html>.
- Giammarco, Kristin, and Mikhail Auguston. 2019. "Monterey Phoenix Tutorial." May.
- Gorog, Christopher. 2019. "Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management." Unpublished Draft.
- Gorog, Christopher, and Terrance E. Boulton. 2019. "Solving Global Cybersecurity Problems by Connecting Trust Using Blockchain." University of Colorado -- Colorado Springs.
- Gorog, Christopher, and Kent Lambert. 2020. "National Security Products, Cryptology, Trust, and Blockchain Frameworks to Secure Supply Chains and Operations."
- Gorog, Christopher, and Kent Lambert. 2021. "BlockFrame, Inc., Technology, Blockchain for Transactive Energy Solutions." July.
- Jackson, Daniel. 2012. *Software Abstractions: Logic, Language, and Analysis*. Cambridge, MA: MIT Press.
- Lambert, Kent D., Angela Williams, Joann Ginal, and Bob Rankin. 2018. "Colorado Senate Bill SB-086, 'Cyber Coding Cryptology for State Records.'" Colorado General Assembly. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.
- McGraw, Robert. 2014. "Risk Adaptable Access Control (RADAC)." <https://csrc.nist.gov/csrc/media/events/privilege-management-workshop/documents/radac-paper0001.pdf>.
- Mittal, Saurabh, and Larry Rainey. 2015. "Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering." In *Proceedings of the Conference on Summer Computer Simulation*. Chicago, IL: Society of Computer Simulation International.
- National Cyber Security Centre. 2019. "The Cyber Threat to UK Business 2017–2018 Report." United Kingdom. <https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf>.

- Office of Personnel Management Security Resource Center. n.d. "Cybersecurity Incidents." Office of Personnel Management. <https://www.opm.gov/cybersecurity/cybersecurity-incidents/>.
- Palo Alto Networks. n.d. "What Is Zero Trust?" Accessed May 18, 2020. <https://www.paloaltonetworks.com/cyberpedia/what-is-a-zero-trust-architecture>.
- "Patents Filed by BlockFrame, Inc. U.S. Patent Application No. 17/488,529; No. 17/488,562; No. 17/488,589; No. 17/488,655; Patent Cooperative Treaty (PCT) Application No. PCT/US21/52839; No. PCT/US21/52854; No. PCT/US21/52865; No. PCT/US21/52872." 2021. United States Patent and Trademark Office.
- "Pentagon to Launch Zero Trust Cyber Office in December." n.d. Accessed November 27, 2021. <https://www.securitymagazine.com/articles/96504-pentagon-to-launch-zero-trust-cyber-office-in-december>.
- Pratt, Mary. 2018. "What Is Zero Trust? A Model for More Effective Security." CSO United States. <https://www.csoonline.com/article/3247848/what-is-zero-trust-a-model-for-more-effective-security.html>.
- Priebe, Torsten, Wolfgang Dobmeier, Christian Schläger, and Nora Kamprath. 2007. "Supporting Attribute-Based Access Control in Authorization and Authentication Infrastructures with Ontologies." *Journal of Software: JSW* 2 (1): 27–38.
- Rainey, Larry B., and Andreas Tolk. 2015. *Modeling and Simulation Support for System of Systems Engineering Applications*. Hoboken, NJ: John Wiley and Sons.
- Trump, Donald. 2017. "Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure." The White House. <https://www.whitehouse.gov/presidential-actions/presidential-executive-order-strengthening-cybersecurity-federal-networks-critical-infrastructure/>.
- Trump, Donald. 2020. "Executive Order on Securing the United States Bulk-Power System." White House. <https://www.whitehouse.gov/presidential-actions/executive-order-securing-united-states-bulk-power-system/>.
- Utermohlen, Karl. 2019. "Facebook Data Breach 2019: 540 Million Users' Records Exposed." InvestorPlace. <https://investorplace.com/2019/04/facebook-data-breach-2019/>.
- Verizon. 2019. "2019 Data Breach Investigations Report." <https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf>.
- Zeigler, Bernard P. 2019. "DEVS-Based Modeling and Simulation Framework for Emergence in Systems of Systems." In Larry Rainey (ed.), *Engineering Emergence, A Modeling and Simulation Approach*. Boca Raton: CRC Press.

6.0. Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) to support the design of cybersecurity systems

This Chapter 6.0 addresses and satisfies Research Question 2 (RQ2) by demonstrating how Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) can be applied to the design of the BlockFrame® system-of-systems for supply chain and cybersecurity applications. Restated, Research Question 2 (RQ2) is: “How can Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) be used to support the design of cybersecurity systems?”

6.1. Practical Applications of MBSE and MBSAP Research

Research Question 2 (RQ2) introduces practical techniques to model the various requirements and operational and logical/functional design issues of BlockFrame® technologies. Tasks compare and contrast BlockFrame designs with other designs in the supply chain and cybersecurity industries for publication, patenting, and translation to commercial applications. To apply research as a practicum with industry, the systems involved will be those developed with BlockFrame, Inc., working within an operational security framework, to deliver digital supply chain, blockchain, and cryptologic security services that are managed through the Philos™ distributed ledger when appropriate, and will assume a hostile cybersecurity environment. These products were planned to have been supplemented by sections of a training and academic textbook on cybersecurity concepts and methods, but it was more practical to modify that task to provide video presentations instead, which has been accomplished.

A System of Systems (SoS) can be defined as the integration of a set of independently developed and operated systems with the goal of creating different, usually greater, capabilities than participants can achieve individually⁷⁵. When appropriate, cybersecurity design requirements considered and referenced the Mitre ATT&CK® framework for Industrial Control Systems⁷⁶, the Department of Defense Cybersecurity Maturity Model Compliance (CMMC)⁷⁷, the General Services Administration Federal Risk and Authorization Management Program (FedRAMP)⁷⁸, the National Institute of Standards and Technology (NIST) NIST SP 800-171 Protecting Controlled

⁷⁵ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

⁷⁶ “Mitre ATT&CK for Industrial Control Systems.” https://collaborate.mitre.org/attackics/index.php/Main_Page (accessed Jul. 21, 2022).

⁷⁷ “OUSD A&S - Cybersecurity Maturity Model Certification (CMMC).” <https://www.acq.osd.mil/cmmc/model.html> (accessed Jul. 21, 2022).

⁷⁸ “FedRAMP.” <https://www.gsa.gov/technology/government-it-initiatives/fedramp> (accessed Jul. 21, 2022).

Unclassified Information⁷⁹, NIST 800-207 Zero Trust⁸⁰, the Department of Energy Federal Energy Regulatory Commission (FERC) Order 2222⁸¹, and other FERC Distributed Energy Resource (DER) design requirements. Those requirements have been allocated to models of the BlockFrame[®] systems that satisfy those requirements and other design requirements, following the structures presented in Borky and Bradley⁸², and with the use of the Dassault Cameo Systems Modeler. Design requirements will also assume the goals of defense-in-depth and zero trust must be satisfied through uniform common components and behaviors. The MP/G Model was also used to verify and validate selected system behaviors. Collectively, BlockFrame[®] systems are modeled to result in uniform mitigation of negative emergent cybersecurity threats. When appropriate, models also include functions and behaviors of the IATC, the independent organization that provides neutral governance and oversight.

Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) can be applied and used in cybersecurity systems design.

First, BlockFrame[®] designs were the primary focus of MBSE and MBSAP modeling projects, including the application of SysML, and other business processes and systems engineering analyses.⁸³ All primary BlockFrame[®] system and component architectures were modeled at the operational and logical/functional levels, to help identify and satisfy operational and cybersecurity requirements, systems behaviors, and non-performance requirements such as systems costs and sustainability. In some cases, the speed of the agile development of BlockFrame[®]'s DEVSECOPS and business cases have outpaced the tasks of completing comprehensive formal models of every system and subsystem. Therefore, this research and actual BlockFrame[®] development have often been done concurrently.

Second, a practicum activity in partnership with industry focused on development, security, and operations

⁷⁹ "NIST Publishes SP 800-171 Revision 2: Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations," *NIST*, Feb. 2020, Accessed: Jul. 21, 2022. [Online]. Available: <https://www.nist.gov/news-events/news/2020/02/nist-publishes-sp-800-171-revision-2-protecting-controlled-unclassified>

⁸⁰ S. W. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," *NIST*, Aug. 2020, Accessed: Jul. 19, 2022. [Online]. Available: <https://www.nist.gov/publications/zero-trust-architecture>.

⁸¹ "FERC Order No. 2222: Fact Sheet | Federal Energy Regulatory Commission." <https://ferc.gov/media/ferc-order-no-2222-fact-sheet> (accessed Jul. 21, 2022).

⁸² Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

⁸³ Research to support RQ-2 was accomplished in part during CSU classes Engr 567 in Fall 2019 and Engr 667 in Spring 2020. The final projects for each of those classes were highly structured and were used to research and demonstrate the application of MBSE and MBSAP to specific topics. That research will not be included here in its entirety, but the structure of this chapter will closely follow the structure of the textbook used in those classes. Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

of products within BlockFrame, Inc. The BlockFrame[®] business model, customer demand, and resourcing drives many of the design and production requirements and limitations. As described in Chapter 4, one of the goals of this research is translational: to apply academic principles in systems engineering to industrial examples, and to reapply that industrial experience back to the academic body of knowledge. The industrial environment used for this research is the BlockFrame[®] company and its operations are the environment and focus of all subsequent chapters.

Third, many new opportunities for MBSE and MBSAP have been applied concurrently with follow-on modeling efforts. Several high-priority use cases have emerged, including digital privacy, digital supply chains, novel third-generation blockchain/distributed ledger (BDL) technology, cybersecurity, the management of transactive energy among distributed energy resources (DERs), novel high-efficiency energy production products, and medical devices. The research question itself can be answered that MBSE and MBSAP have been demonstrated for many effective cybersecurity design contributions and operations.

Chapter 5 extensively documented research using MBSE demonstrations of practical cybersecurity using the Monterey Phoenix/Gryphon model. That chapter focused on the principles of controlling negative emergence, but also contributed to answering Research Question 2, Task 2: “Use Model-based systems engineering (MBSE) to describe and evaluate the quality attributes of BlockFrame[®] systems to operate within hostile cyber-attack environments with high negative emergence.” This Chapter 6 will more extensively use MBSE and MBSAP principles from Borky and Bradley, 2019⁸⁴. Research with the applied use of the SysML model was also completed, but since it was largely redundant, it is included in Appendix B as proof of additional work.

⁸⁴ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*, 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

6.2. System-of-Systems Research applied to BlockFrame[®] Distributed-Ledger-Based Logistics and Cybersecurity Management

In keeping with the SOS philosophy, the ESP[™] and Philos[™] architectures allow for those two enterprises to work together, although both can also be used independently for other specific use cases. Both enterprises follow a modular design to anticipate expandability. They are currently in prototype development, component testing, and demonstration phases, and are moving into production design. The goal of this research is to study and apply systems engineering principles to both enterprises using Model-Based Systems Engineering (MBSE), and the Model-Based Systems Architecture Process (MBSAP), primarily as presented in Borky, J.M., and Bradley, T. *Effective Model-Based Systems Engineering*.⁸⁵

The Eco-Secure Provisioning System, ESP[™] logistics enterprise, is a cloud-based data architecture connected to participating objects through embedded encapsulated components. These systems have capabilities designed to provide Proof-of-Origin (POO) of data, provide rapid recovery from system faults to secure objects within the Internet of Things (IoT), to secure elements and data within supply chains, to secure power grid data and operations, and to secure personal privacy information (PPI). ESP[™] is already being used for some logistics operations, but the additional use of Philos[™] will mitigate observed problems with other commercially available BDLs. It leverages the Philos[™] blockchain/distributed ledger for connection, security, and trust functions. Philos[™] is designed as a publicly available, open architecture blockchain, initially planned to meet needs defined in Colorado SB18-086⁸⁶ to secure public documents and serve a wide variety of other anticipated state government use cases. Philos[™] is not a cyber-currency-based system, nor based on crypto-currency languages. Philos[™] is a cryptological-intensive and indefinitely scalable system-of-systems based on encapsulated common units connecting its enterprise architecture. It also has specialized cryptology generation and storage nodes, secure data and cypher code delivery capabilities, and accesses a dedicated neutral global governance structure like the Domain Name System (DNS) of the Internet. The finished enterprise is intended to increase trust and lower risk among the systems of all participating users, including robust and verifiable cyber-security designs for data confidentiality, integrity, and availability,

⁸⁵ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

⁸⁶ Lambert, K. Williams, A. Ginal, J. Rankin, B. Colorado Senate Bill SB08-086, *Cyber Coding Cryptology for State Records*. State of Colorado. <https://leg.colorado.gov/bills/sb18-086>. Accessed 2019.)

defense-in-depth (DiD), and zero-trust (ZT) to defend against network infiltration by hostile external or internal actors, and to provide data immutability and transaction non-repudiability.

6.2.1. Systems Description

Operational considerations

Arizona State University and the United States Navy are stakeholders and prime partners with BlockFrame, Inc., for research and development of the ESP™ Enterprise. The State of Colorado, the Colorado Office of Information Technology (OIT), and the Colorado Information Security Office (CISO) are primary stakeholders for the use of the Philos™ blockchain/distributed ledger. OIT identified 71 possible Philos™ use cases. Colorado State University, Pueblo, has been involved in early blockchain prototype testing and demonstrations. The University of Colorado Colorado Springs (UCCS) has provided resource, technical, and community coordination and assistance for Philos™ development. The Neutral Blockchain Governance Consortium is a cooperative project of Colorado OIT, UCCS, and BlockFrame, Inc. Other government, military, energy, and private organizations are anticipated to be users once ESP™ and Philos™ products go into production. The Colorado State University Department of Systems Engineering has offered mentorship in MBSE and MBSAP skills for this project.

Many operational functionalities have already been realized, but constructing better MBSE models and working operational models should greatly help stakeholders and users to visualize, construct, and operationalize the enterprise ecosystems and reference architectures in the future. The primary long-range goals of this project are to discover and improve areas where lack of information could cause inefficiency and frustration by providing more details, insights, and guidance. Prospective future users and use cases need to be better identified, quantified, and analyzed within capabilities datasets, which will be evolutionary as users commit to using the enterprises and begin working relationships within the ecosystem.

Several ESP™ and Philos™ key performance capabilities and other characteristics are of great interest to the potential user communities. Several stand out as highly needed. Authentication and authorization of white-listed users is important to ensure information is not lost in transit or storage. Immutability and non-repudiability of those data also assure the integrity and security of all transactions. That can be done using Philos™ without the common use and costs of wasteful “crypto mining” that is a great environmental and financial concern to users. Indefinite expandability for applications to very large numbers of networked objects, and to a very wide variety of independent uses and industries are also attractive. Among all those considerations, some examples of the key performance considerations would be the security and privacy of data, and the speed of delivery and computations. On the security side, being able to secure trusted data, users, communications nodes, having data with Proof-of-Origin,

having the capability to reliably reconstruct network operations and data after system faults is highly desired. High-level modeling of fielded user applications might be best understood and applied through loosely coupled and modular information service architecture models to present only the key characteristics and interfaces in which the users have the most interest.

Logical/functional considerations

Planned workloads and work breakdown structures have to date been somewhat imbalanced due to initial priorities of demonstrating chain code capabilities and due to shortages of resources. Due to some initial funding, that tempo is now increasing, and should accelerate in the future. MBSE-assisted design and the MBSAP process enables visualization and communication of all ESP™ and Philos™ architecture designs, to manage the integration with other elements of the blockchain enterprise, as well as other hardware and virtual components. Modeling the initial logical/functional system, enterprise components, and behaviors, should also help better quantify and visualize life cycle considerations of production, deployment, maintenance, and the resources required⁸⁷. That should help planning and preparation for better milestone reviews, strategic decision making, and management of personnel, revenue, purchasing, and finances.

A major consideration and challenge in architecting for the Logical/Functional Viewpoint is the “system-of-enterprises” context inherent with the ESP™ Enterprise and the Philos™ blockchain Enterprise. As a typical characteristic of any system-of-systems, both enterprises are based on nodes of common components – which could be considered “systems” themselves – that inherently help tie the enterprises together into secure and trustworthy ecosystems. The Philos™ Enterprise and the ESP™ Enterprise could either work independently or could combine to serve different applications. This research considers both to be modellable as enterprise architectures. They also both support a modular approach which should enable indefinite expandability with the modular addition of more routers and DNS servers.

Physical considerations

Once the enterprises are ready, production, quantities, schedules, deliveries, physical integration will become primary considerations. Current plans could in some cases involve third-party deliveries and customer-integration of hardware, and “App Store”-like delivery of software – all of which could benefit from MBSE service

⁸⁷ Gorog, Chris. *Blockchain Distributed Ledger for State Governance*. Prepared for discussions with the State of Colorado. 2019. Unpublished draft.

modeling and visualizations. Different overlays of the operational, logical/functional, and physical viewpoints could be very valuable to help explain and document details for developers, users, and stakeholders.

Among important physical considerations, system elements and the enterprise ecosystems include both time-sensitive (real-time) and non-time-sensitive behaviors. One consideration is to design, deliver, and maintain cypher codes in such a way as to assure that they can “roll” very quickly and randomly to prevent hackers from seizing trusted code sequences. This is part of the inherent design, and improves the current state-of-the-art. In some cases, codes may only be exposed for very short time intervals and then can immediately be erased and new codes generated either automatically or on-command. Typically, those processes can be totally automated with no human intervention. Applying MBSAP state machines models, along with quantification of specific performance thresholds or goals can help specify operational capabilities and standards for users.

Another major timing consideration relates to the consensus process that governs the blockchain itself. Because the nodes of the Philos™ blockchain elements are stored in widely distributed locations and require high-speed communications and processing speeds, the governance methodologies must also have both high security and high speed. There must be a trade-off between user service priorities and the total speed of the network to support the creation of new blockchain blocks in a “fair” and rapid manner. While there could be some human-in-the-loop controls of governance policy in some circumstances, most of the consensus computations and communications will be automated according to defined algorithms, statistical distributions, or other possible methods in accordance with defined governance policies. Discovering the most optimum and/or practical algorithms has not been settled yet but could be an element of future research that considers both theoretical distributions and empirical data. The algorithms themselves could also be dynamic depending on “game theory” behaviors of different users of the system (actors in the model) and possibly based on incentives for “good behaviors” to better incentivize trust and security among systems within the Philos™ Enterprise over time, to maintain the economic viability of the enterprise, and to make services affordable. Another consideration is that the Philos™ Enterprise is designed to increase the levels of trust (or an “absence of risk”) that should measurably increase over time. There could be a variety of methods to governing user applications, but it is possible that they could combine into “credit score”-like trust variables, or other objective functions. Research and testing of several candidate methods might be warranted in the future.

Research Scope

This research's first purpose is as a learning tool to study SysML, MBSE, and MBSAP processes, and to apply them to practical industrial examples. Different stakeholders and users will have different service interests for ESP™ and Philos™ for their own use cases. The focus of this research is on the architectural designs for ESP™ and Philos™ as enterprises, not to engineer or modify each users' individual application or use case. Many users might use ESP™ and/or Philos™ as self-contained "services," and build upon the frameworks these projects provide to enable their own use cases. This project is not intended to design system architectures or models for every application or use case, but future research would likely study some as examples for integration, testing, verification, and validation.

Although this project involves many practical applications for academic purposes, its scope is not intended to model or design the entire ESP™ and/or Philos™ enterprises. The following sections of this report present a range of examples to reinforce specific academic concepts, not to represent, verify, or validate a comprehensive architecture of either enterprise. It might be considered the first iteration of a spiral or Agile approach for future architectural efforts.

6.2.2. Operational Viewpoint

Functional/Nonfunctional Requirements

Table 6.1 lists exemplary high-level functional and non-functional requirements for the combined ESP™ and/or Philos™ Enterprises. Table 6.1 might benefit from a more understandable and clustered format when more detailed levels are examined, but that will be deferred for future research⁸⁸ and other working documents were used for initial Operational Viewpoint (OV) descriptions.

Table 6.1. Example High-Level Functional and Non-Functional Requirements

Table 1. Table for Functional and Non-Functional Requirements with Basic Definitions	
BlockChain Cybersecurity Logistics Organization for Globally Scalable Security Management (1)	
Requirements Area	High-Level Requirements
R1.1 Provide an Eco-Secure Provisioning Framework for IoT (Project Domain Nickname ESP)	R1.1.1 The system shall provide uniformity features for supply and trust chains R1.1.2 The system shall be scalable to many use cases in large marketplaces across any industry R1.1.3 The system shall be able to connect to individual IoT devices through embedded logistics devices R1.1.4 The system shall program mass distributed IoT devices in network R1.1.5 The system shall provide cryptologically-secure protection for IoT devices R1.1.6 The system shall provide cryptologically-secure protection for system components R1.1.7 The system shall provide security management and control services at the organization/business level R1.1.8 The system shall provide global services across jurisdictional boundaries R1.1.9 The system shall provide verification and validation of user/actor/device identities R1.1.10 The system shall operationally increase measurable trust levels among users (and reduce risk) over time R1.1.11 The system shall use open architectures of modular components R1.1.12 The system shall design components to accept expandable technologies R1.1.13 The system shall maintain and provide standardized libraries for embedded logistics devices R1.1.14 The system shall have the capability to prove the origin of data -- Proof-of-Origin POO R1.1.15 The system shall coordinate, manage, and synchronize all operational ecosystem comms for Zero Trust R1.1.16 The system shall have the ability to wrap legacy equipment by tools/APIs to support cyber security methods R1.1.17 The system shall manage the use and delivery of cryptographic tools and secret codes R1.1.18 The system shall assure the security of the handling and delivery of products R1.1.19 The system shall conduct capabilities and security testing and evaluation R1.1.20 The system shall (after initial stability) be economically self-sustaining by user fees R1.1.21 The system shall provide education and training to developers and users R1.1.22 The system shall manage cryptographic key management throughout the IoT life cycle R1.1.23 The system shall detect, authenticate, and update any IoT device upon linking to the trust chain R1.1.24 The system shall manage the secure distribution and installation of software to users R1.1.25 The system shall facilitate media control or digital rights management (DRM) R1.1.26 The system shall manage "New Supply Chain Industry Creation" initiatives and modifications
R1.2 Manage Records, Documents, and Transactions (Project Domain Nickname Philos)	R1.2.1 The system shall provide immutable records of data transactions R1.2.2 The system shall provide privacy of individual records R1.2.3 The system shall minimize the use of energy for management R1.2.4 The system shall transact at speeds to support large-scale operations R1.2.5 The system shall incentivize cooperative distributed sharing of managed data R1.2.6 The system shall limit the growth of managed data to a controllable size R1.2.7 The system shall support both continuous and non-continuous operations R1.2.8 The system shall manage data phase-out and end-of-life R1.2.9 The system shall support managed data blocks of different sizes R1.2.10 The system shall balance transparency and visibility with privacy R1.2.11 The system shall manage transactions to allow indefinite number of large-scale smart contracts R1.2.12 The system shall use management subcomponents that allow indefinite expandability and operational consensus of trust R1.2.13 The system shall have an open architecture to allow developers and general public to make their own contracts R1.2.14 The system shall have the capability to add and remove users and organizations R1.2.15 The system shall have usable controls -- like web interfaces, mobile apps, Zero-Trust R1.2.16 The system shall manage management subcomponents to provide service to multiple organizations and industry types R1.2.17 The system shall provide end-to-end transaction security for participating transaction users on supply chains
R1.3 Industry Utility Registrar	R1.3.1 The system shall use expandable industry identification registers of users/devices on multiple networks
R1.4 Manage Neutral Governance	R1.4.1 The system shall use neutral governance infrastructure to manage and optimize the (DNS-like) transaction engine R1.4.2 The system shall use a consortium of common components for all indefinitely scalable governance
R1.5 Quality Attributes (QAts) (Non-Functional Requirements) (2)	R1.5.1 The system can use an open architecture to facilitate integration and evolution R1.5.2 The system can use modular architecture to facilitate integration, evolution, and updates R1.5.3 The system can be operationally suitable with external systems and applications R1.5.4 The system can be verified, validated, and accredited to collect, process, and disseminate sensitive data R1.5.5 The system can be procured in sufficient numbers and kept effective and available over its lifetime within available budgets and resources
(1) Gorog, C. 2019. BlockChain Cybersecurity Logistics Organization for Globally Scalable Security Management. Unpublished Draft.	
(2) Borky, J. & Bradley, T. 2013. Effective Model-Based Systems Engineering. P. 605.	

⁸⁸ Gorog, C. 2019. *Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management*. Unpublished Draft.

Structural Perspective

When fully operational, the combined ESP™ Philos™ Enterprises could be massive in scope. From the Operational Viewpoint (OV), I have defined the overall “system” of this project and report as the ESP™ Enterprise and the Philos™ Enterprise working together, with the Industry Utility Registrar™ (IUR™), and Neutral Governance as the four system Domains.⁸⁹

In plain language, the ESP™ Enterprise Domain is intended to secure, monitor, and manage supply chain operations through secure, encrypted embedded devices. The ESP™ is designed to use a blockchain for many functionalities but does not specifically have to use the Philos™ blockchain.

The Philos™ blockchain/distributed ledger (BDL) Enterprise Domain is intended to manage the “trust chains” to provide high levels of trust and cybersecurity, to reduce risk. For convenience, we assume that for this that Philos™ will be used due to its potential superiority over other BDLs. The Philos™ Enterprise (or “ecosystem,” or “marketplace”), does not have to operate exclusively with ESP™, but could perform many functions independently for other participants and use cases. Hence, they have different levels of mutual composition and dependency.

The IUR Domain and Neutral Governance Domain could be relatively less extensive in size and complexity but are recognized as separate Domains because they are key elements of the system architecture and will act as autonomous “third-party” or “honest broker” entities to help manage potential conflicts of interest.

Figure 6.1 is a Block Definition Diagram (bdd) showing relatively simple high-level OV structures. Figure 6.2 is an Internal Block Diagram (ibd) of system interactions. Very extensive communications, data, interfaces, software, and services are described in more detail in the Logical/Functional (LV) and Physical (PV) Viewpoints. Interactions are also introduced with several stakeholder organizations, displayed as actors, that are part of the OV development process.

⁸⁹ For a taxonomy like this “system of Enterprises,” combining two Enterprises with two other systems creates odd relationships and complicates common organizational terminology levels for “Enterprise, Nodes, Functional Domains, Systems, Subsystems, Modules, and Components.” (Borky. 2019. P. 16.) For this example, the term “Domains” is functionally the best term of convenience, but that could make them higher in the hierarchy than the Nodes. In fact, there will be thousands of “Nodes” below that the “Domain” level for both Enterprises. Paradoxically, the informational “Nodes” may physically fit inside common components (in the system-of-systems terminology) called “system containers,” which in turn could be considered networked common components of the Enterprises. The reader should understand that the nature of this “System” makes consistent hierarchy and taxonomy definitions difficult.

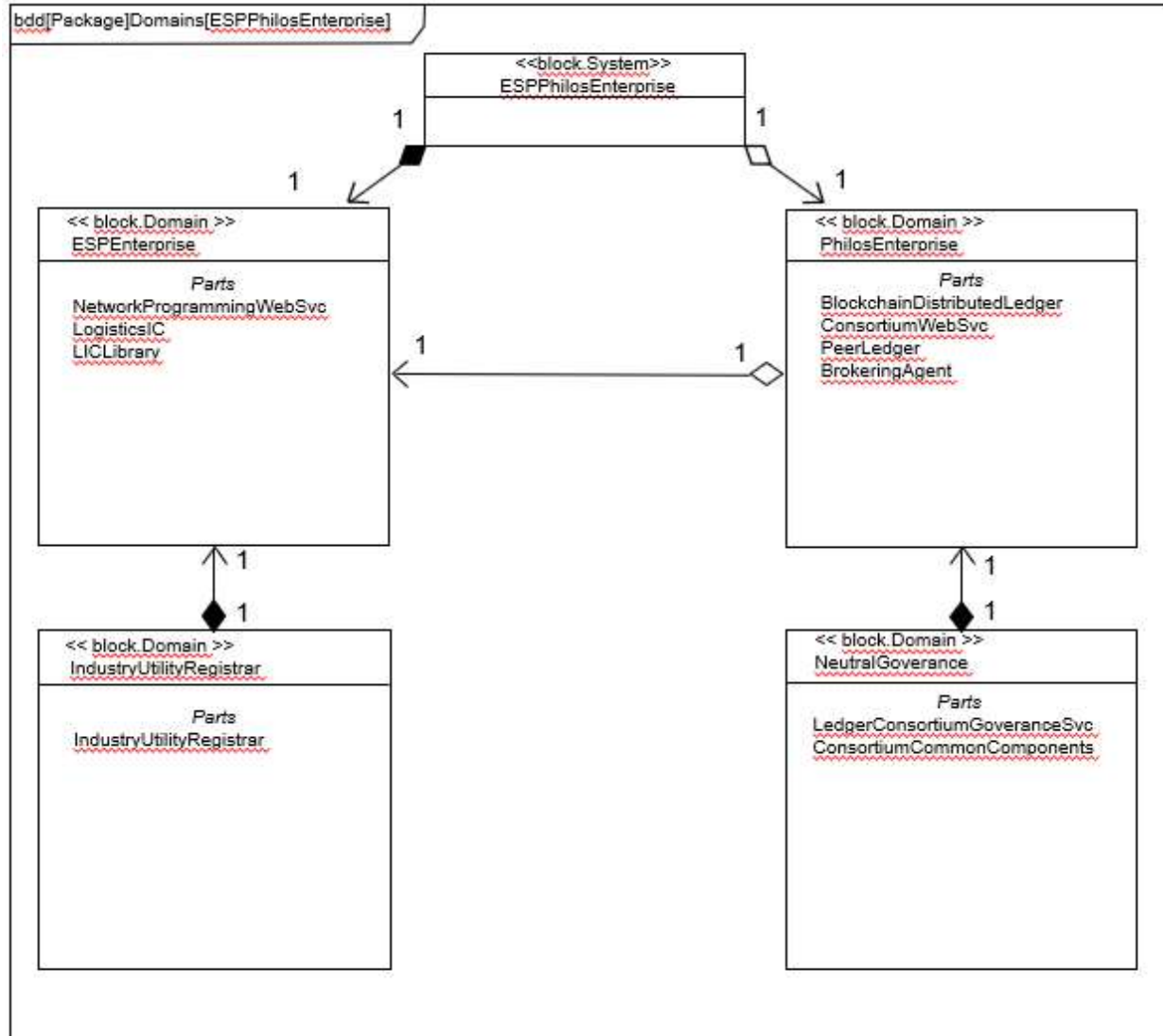


Figure 6.1. Block Definition Diagram (bdd) of the Combined ESP™ and Philos™ Enterprises with Associated IUR™ and Neutral Governance Systems

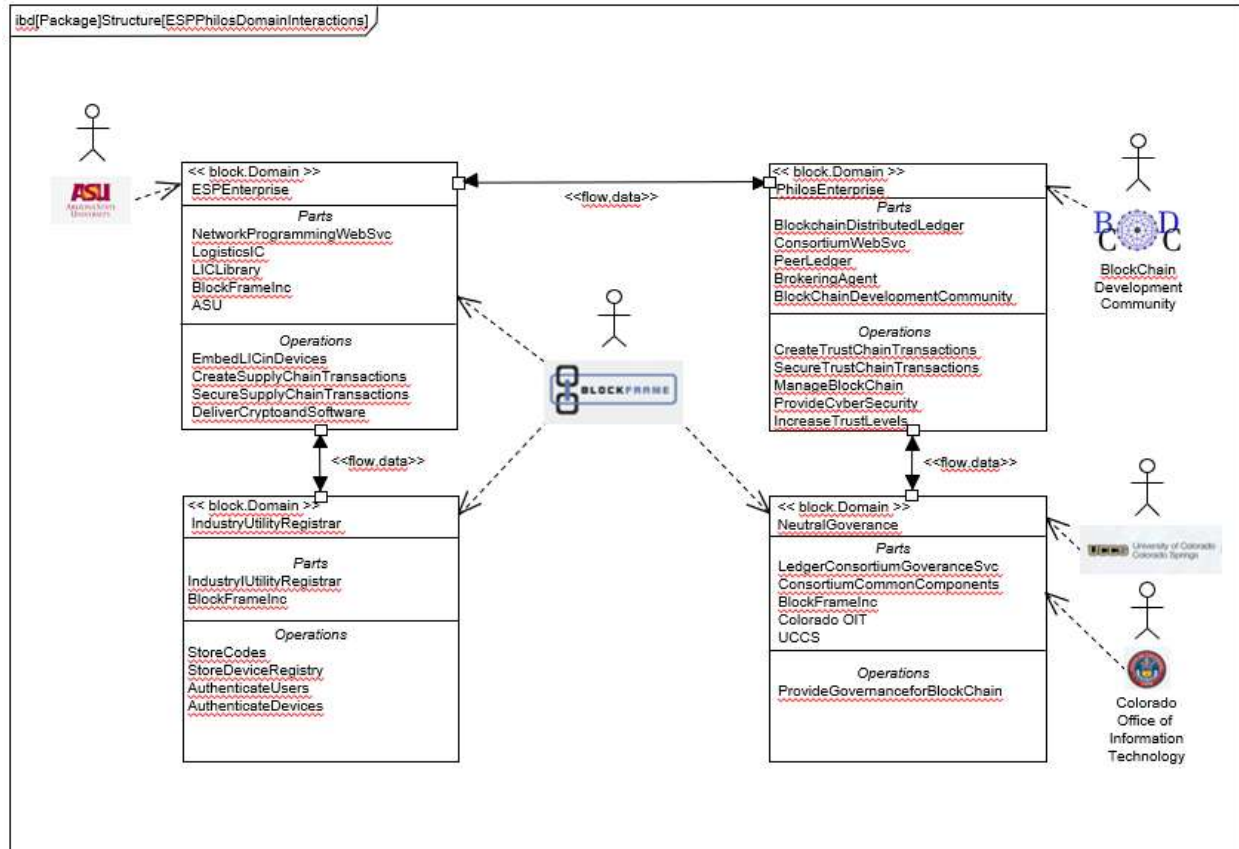


Figure 6.2. Internal Block Diagram (ibd) of the Combined ESPTM and PhilosTM Enterprises with Associated IURTM and Neutral Governance Systems

Behavioral Perspective

Figure 6.3 shows some representative examples of how the ESP™ and Philos™ Enterprises might support four Use Cases. In the first, the Colorado Department of Regulatory Agencies (DORA) could use Philos™ to receive and secure medical licenses for the benefit of doctors. In the second, the Colorado Department of Revenue could use Philos™ to receive and secure driver registration information for the benefit of drivers. In the third, the Colorado Department of State could use data received from the Department of Revenue to register voters, for their benefit, and securely store that data on the Philos™ BDL. In the last example, an energy company could monitor and collect energy use data from a distributed utility supply chain using ESP™. ESP™ could then, in turn, <<include>> the Philos™ BDL as a distributed utility trust chain to store the energy use data for each customer with immutably and non-repudiation.

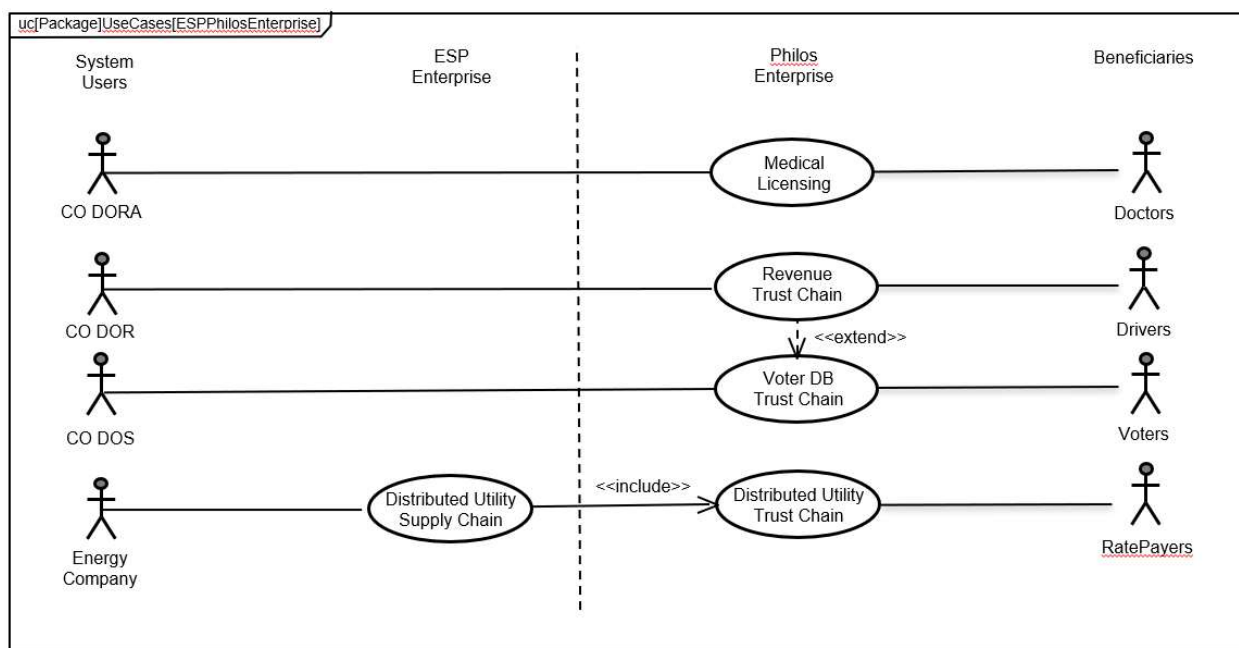


Figure 6.3. Example Use Cases of the ESP™ and Philos™ Enterprises

Figure 6.4 shows an Activity Diagram of an end-to-end process of a major mission thread spanning multiple Domains and Actors using the ESP™ and Philos™ Enterprises. When ESP™ and Philos™ Enterprises operate together, that activity can involve all four domains. In modern smart grids, legal requirements demand that energy companies monitor net flows, so they can prove metering to-and-from the grids. Figure 6.4 shows how metering data may be collected ESP™, validated by the IUR™, reported, validated, and securely stored by the Philos™ BDL and the Neutral Governance System. The company may then use the validated flow data to accurately bill its

customers. That process should create positive legal proof of the flow transactions, and from “trust” processes the data should be immutable and non-repudiable.

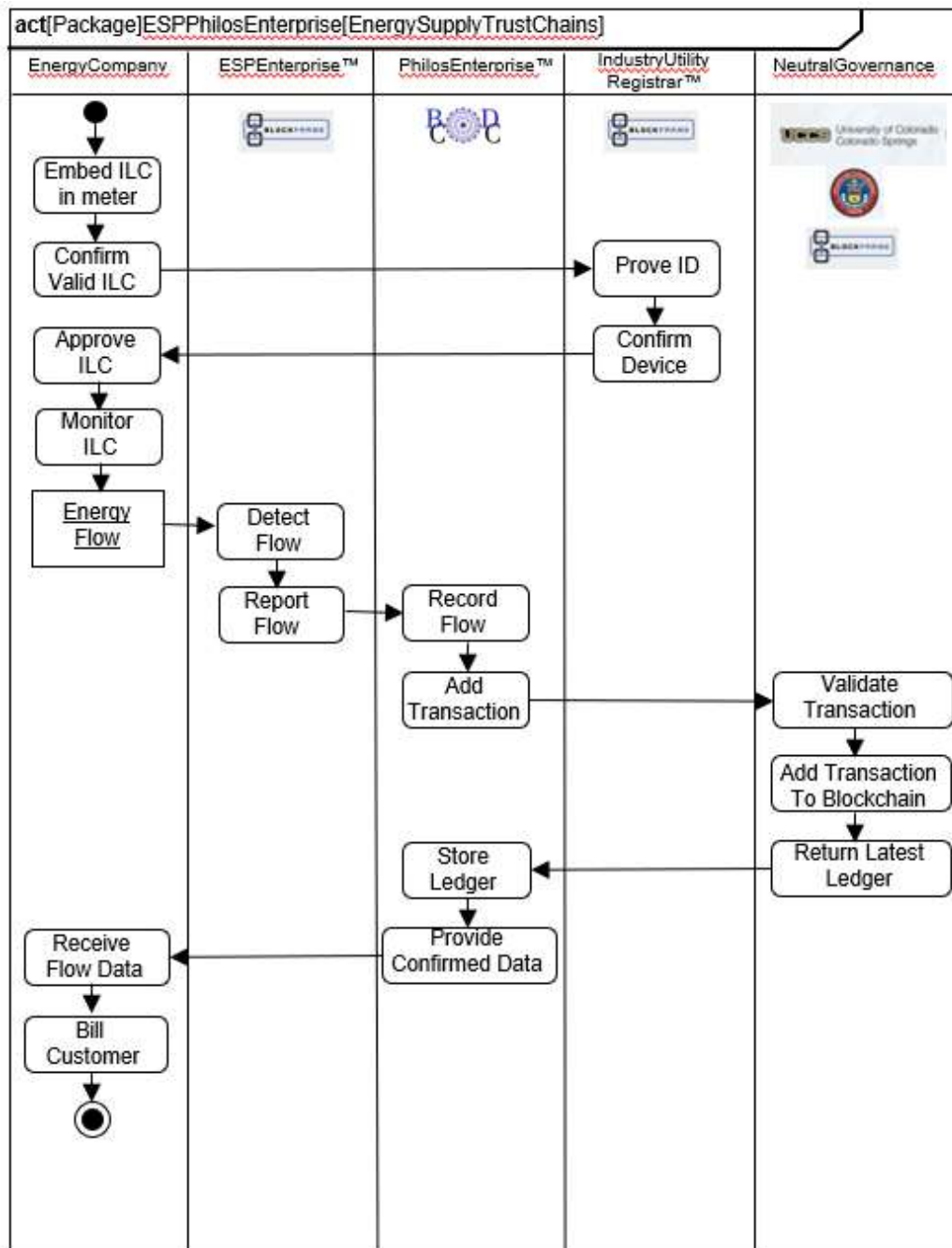


Figure 6.4. Example Activity Diagram to Provide Proof of Energy Company Transactions

Data Perspective

The data perspective for the Operational Viewpoint (OV) is represented by the Conceptual Data Model (CDM) for the Philos™ BDL in Figure 6.5. As an OV of the CDM, administrative data are not included due to their size. The CDM for the Eco-Secure Provisioning™ (ESP™) Framework would mostly be mutually exclusive of Philos™ and will not be shown in this example. The representations of the BlockchainBlock and the ConsensusBlock may include very large numbers of data sets that will interactively update themselves through blockchain operations to the most recent data instantiation. That representation has not been fully demonstrated here, but the intent is to create blockchain blocks in PeerLedgers, and for the Neutral Governance System to retransmit them back to all other Peer Ledgers in the Enterprise.⁹⁰ For the purpose of demonstrating the CDM for the OV, neither Figure 6.6 nor the data tables in Appendix C have been fully validated and translating them into SysML will not be attempted at this time. Data showing the “global variables” that might appear in the Philos™ CDM FoundationDataClass Blocks are displayed in Appendix A due to the size of the source document. They are currently in JSON Language but will be translated into another language in the future.⁹¹

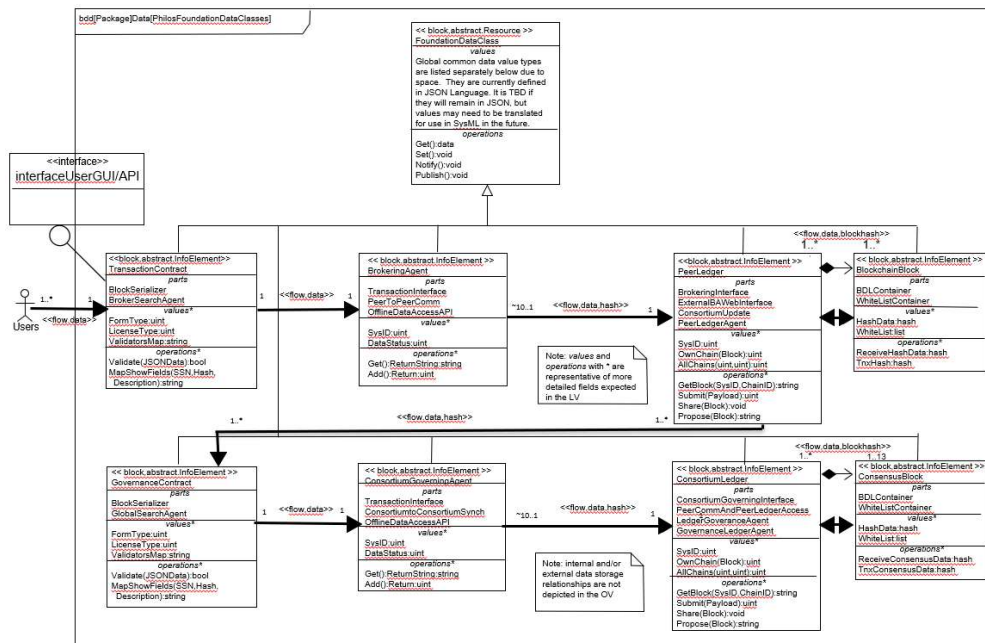


Figure 6.5. Conceptual Data Model for the Philos™ Enterprise

⁹⁰ Gorog C, et, al. *Software Design Document Blockchain Synchronization Consensus V3*. Source Document Proprietary. 2019. Also, some content is from Gorog C. “Personal discussions with Lambert K.” Unpublished. 2019.

⁹¹ Moser, A. *Conversations and Correspondence with Kent Lambert*. BlockFrame, Inc. Unpublished. Sept 29, 2019.

Services Perspective

A combined ESP™ Enterprise and Philos™ Enterprise Domain Diagram in Figure 6.6 shows the Domains, and for convenience shows the composition and <<Usage>> relationships among the four Domains. It also identifies a Services taxonomy within the Domain specifications. To summarize the intent:

- The ESP™ Enterprise Domain *MAY* be composed of the Philos™ Enterprise Domain, or not, or the Philos™ Enterprise Domain may function independently, but the ESP™ Enterprise Domain *MAY USE* the services of the Philos™ Enterprise Domain. The ESP™ Enterprise Domain *WILL* provide services to other Domains.
- The ESP™ Enterprise Domain *SHALL* be composed of the Industry Utility Registrar Domain (IUR™) and *SHALL* also *USE* the services of the Industry Utility Registrar (IUR™) Domain.
- The Philos™ Enterprise Domain *SHALL* be composed of the Neutral Governance Domain and *SHALL* also *USE* the services of the Neutral Governance Domain. The Philos™ Enterprise Domain *WILL* provide services to users.

An example of a layered Service-Oriented Architecture (SOA) within the Philos™ Enterprise is shown in Figure 6.7. (Borky, 2019. P. 186.⁹²). In summary, every Domain as currently defined would be likely to use Services at every layer as shown in Figures 6.6 and 6.7.

⁹² Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

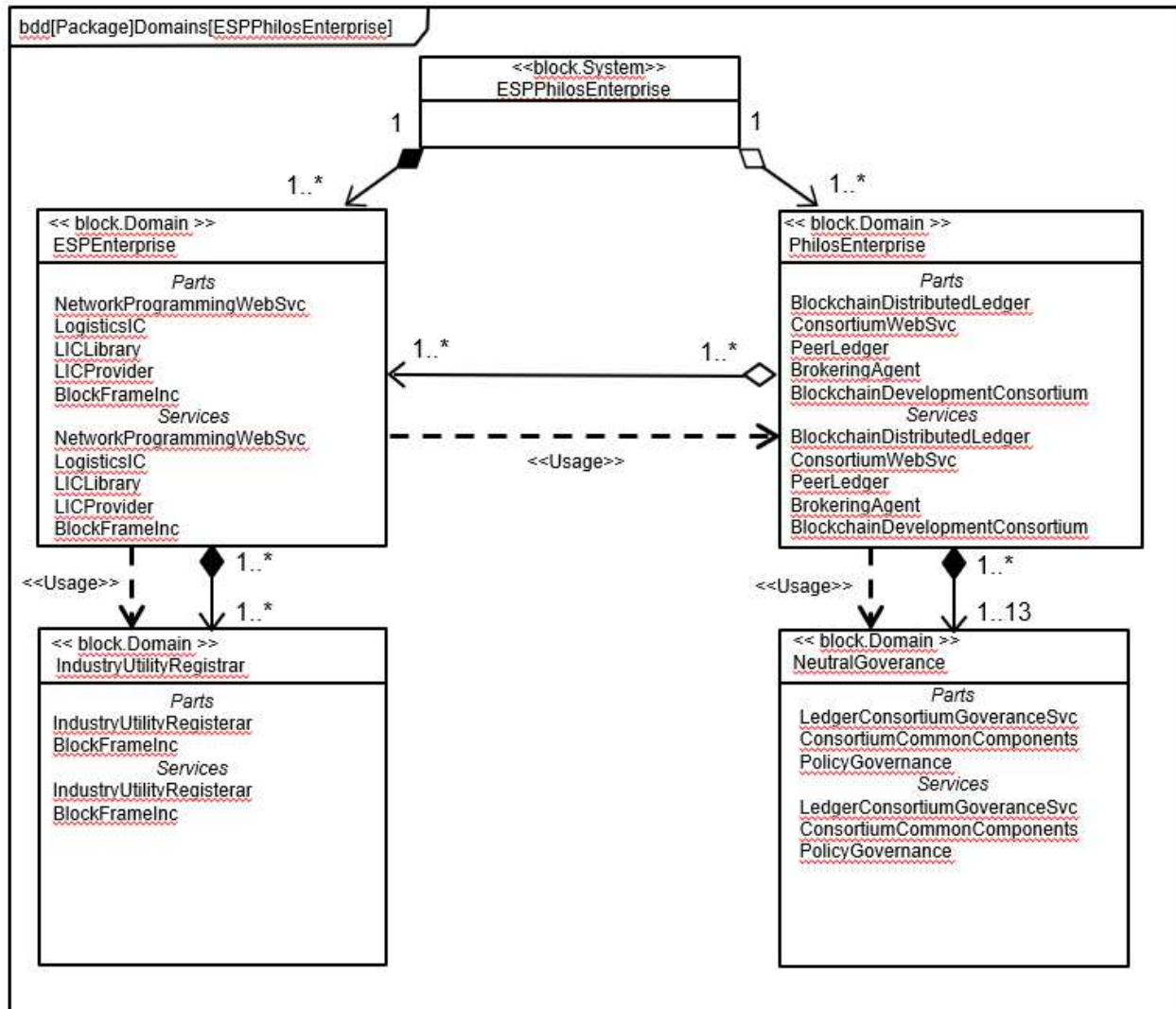


Figure 6.6. Domain Diagram for the Philos™ Enterprise, Displaying Services

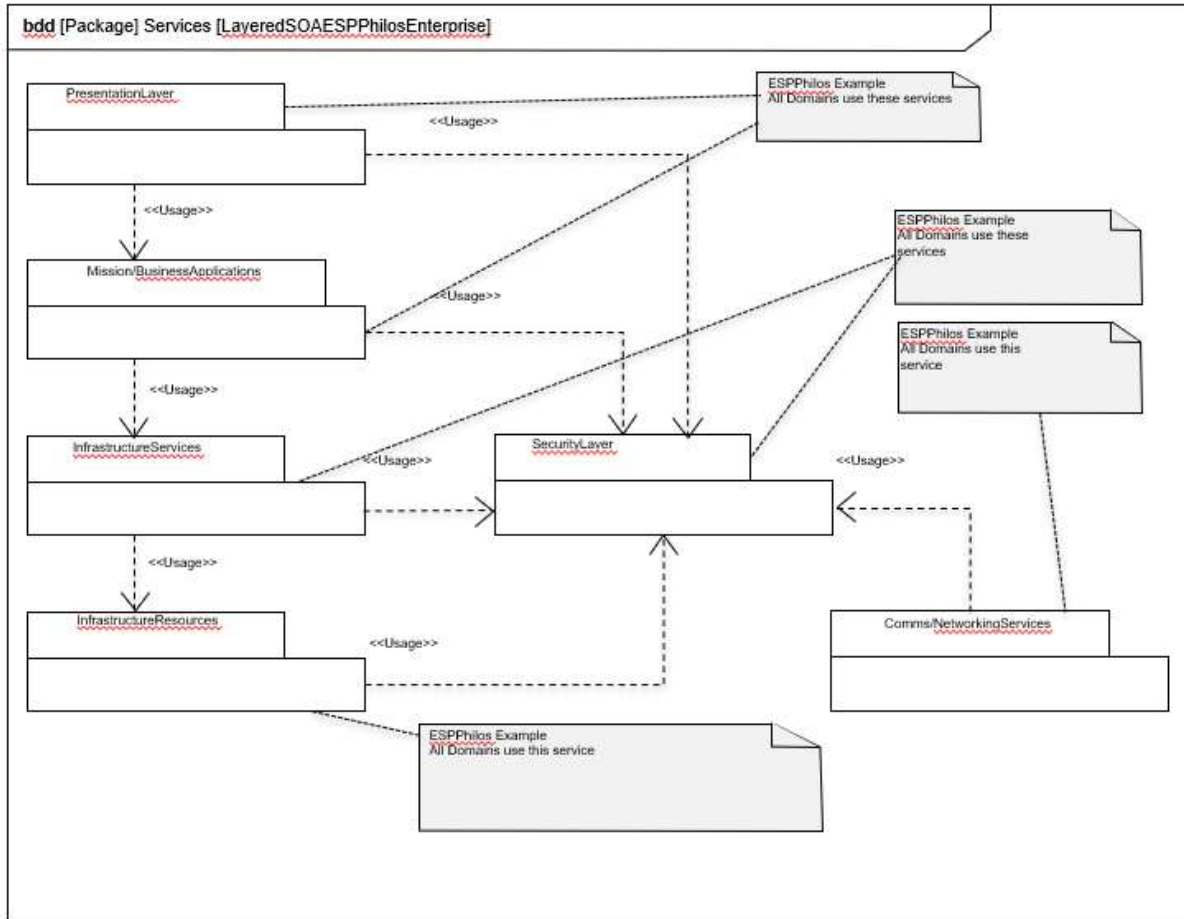


Figure 6.7. Example of Philos™ Layered SOA Structure with Dependencies

Contextual Perspective

A prime driver of enterprise design of the Philos™ Blockchain is Colorado Senate Bill SB08-086.⁹³ The Short Bill Summary is:

Office of information technology - information security - cyber coding cryptology - state records - institutions of higher education - appropriation. The chief information security officer in the governor's office of information technology (OIT), the director of OIT, the department of state, and the executive director of the department of regulatory agencies are required to take certain actions to protect state records containing trusted sensitive and confidential information from criminal, unauthorized, or inadvertent manipulation or theft.

More detailed contextual information may be found in the text of SB18-086, and its associated Fiscal Note, but will not be repeated here verbatim due to size constraints.

Some Nonfunctional Requirements and Quality Attributes applicable to the OV include:

R1.5.1 The system can use an open architecture to facilitate integration and evolution

R1.5.2 The system can use modular architecture to facilitate integration, evolution, and updates

R1.5.3 The system can be operationally suitable with external systems and applications

R1.5.4 The system can be verified, validated, and accredited to collect, process, and disseminate sensitive data

R1.5.5 The system can be procured in sufficient numbers and kept effective and available over its lifetime within available budgets and resources

Some Customer-Specified Artifacts include ongoing guidance from the State of Colorado, Office of Information Technology, and the Chief Information Security Office, which by law are responsible for information security, and adoption and compliance with SB18-086. Acceptance of any proposed blockchain solutions will probably be through the competitive state contract process. Some grants have been available for research and development of the ESP™ and Philos™ Enterprises.

Architecture Policies and Mandates

SB18-086 refers to a suggested framework of national standards and best practices that should be considered:

⁹³ Lambert, K. Williams, A. Ginal, J. Rankin, B. Colorado Senate Bill SB08-086, *Cyber Coding Cryptology for State Records*. State of Colorado. <https://leg.colorado.gov/bills/sb18-086>. Accessed 2019.

(j) ENCOURAGING COORDINATION WITH THE UNITED STATES DEPARTMENT OF COMMERCE AND THE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGIES TO DEVELOP THE CAPABILITY TO ACT AS A COLORADO IN-STATE CENTER OF EXCELLENCE ON CYBERSECURITY ADVICE AND NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGIES STANDARDS;

(k) STUDYING EFFORTS TO PROTECT PRIVACY OF PERSONAL IDENTIFYING INFORMATION MAINTAINED WITHIN DISTRIBUTED LEDGER PROGRAMS, ENSURING THAT PROGRAMS MAKE ALL ATTEMPTS TO FOLLOW BEST PRACTICES FOR PRIVACY, AND PROVIDING ADVICE TO ALL PROGRAM STAKEHOLDERS ON THE REQUIREMENT TO MAINTAIN PRIVACY IN ACCORDANCE WITH REQUIRED REGULATORY BODIES AND GOVERNING STANDARDS...

6.2.3. Logical/Functional Viewpoint

Structural Perspective

As an exercise to illustrate the Structural Perspective of the Logical/Functional Viewpoint, Figure 6.8 decomposes the Neutral Governance Domain, showing compositions, parts, values, operations, and constraints. Figure 6.9 shows high-level data flows and external connections. The purpose of the Neutral Governance Domain is to provide a neutral third-party Service for the Philos™ blockchain to select competitive “side-blocks” from Peer Ledgers throughout the entire Philos™ Enterprise as candidates to be added to the formation of the next block in the blockchain. The Consortium Common Components are common units connecting peer ledgers and consortium ledgers within the Philos™ Enterprise. An algorithm within the Ledger Consortium Governance Service will analyze the descriptive data of each candidate block and score them with an algorithm. Research is under way, but future options could be considered whether the scoring algorithm will be statistical – for example, an exponential or Weibull function, perhaps considering the ability to represent the historical “memory” of data that represent the behaviors of the user base – or whether a type of state machine can be used to observe various states within the Enterprise, and incentivize or disincentivize the competitive blocks from various users based on their behaviors. For example, compliance with security or risk-reduction goals, financial participation, computer processing participation, storage space contributions, or several other possible factors, that would result in “value” to the Enterprise/Eco-System as a whole. In turn, “good actors” using the blockchain could be rewarded by more-rapid processing speed, and/or more discounted access to blockchain storage. Whatever methods are ultimately chosen will have to consider a balance of overall system processing speed versus system control and security requirements. The Policy Governance factors will be decided by the Policy Governance sub-Domain, as a neutral third-party organization, comprised through a contract of memorandum of understanding with the State of Colorado Office of Information Technology, the University of Colorado Colorado Springs (UCCS), and the prime developer, BlockFrame, Inc.

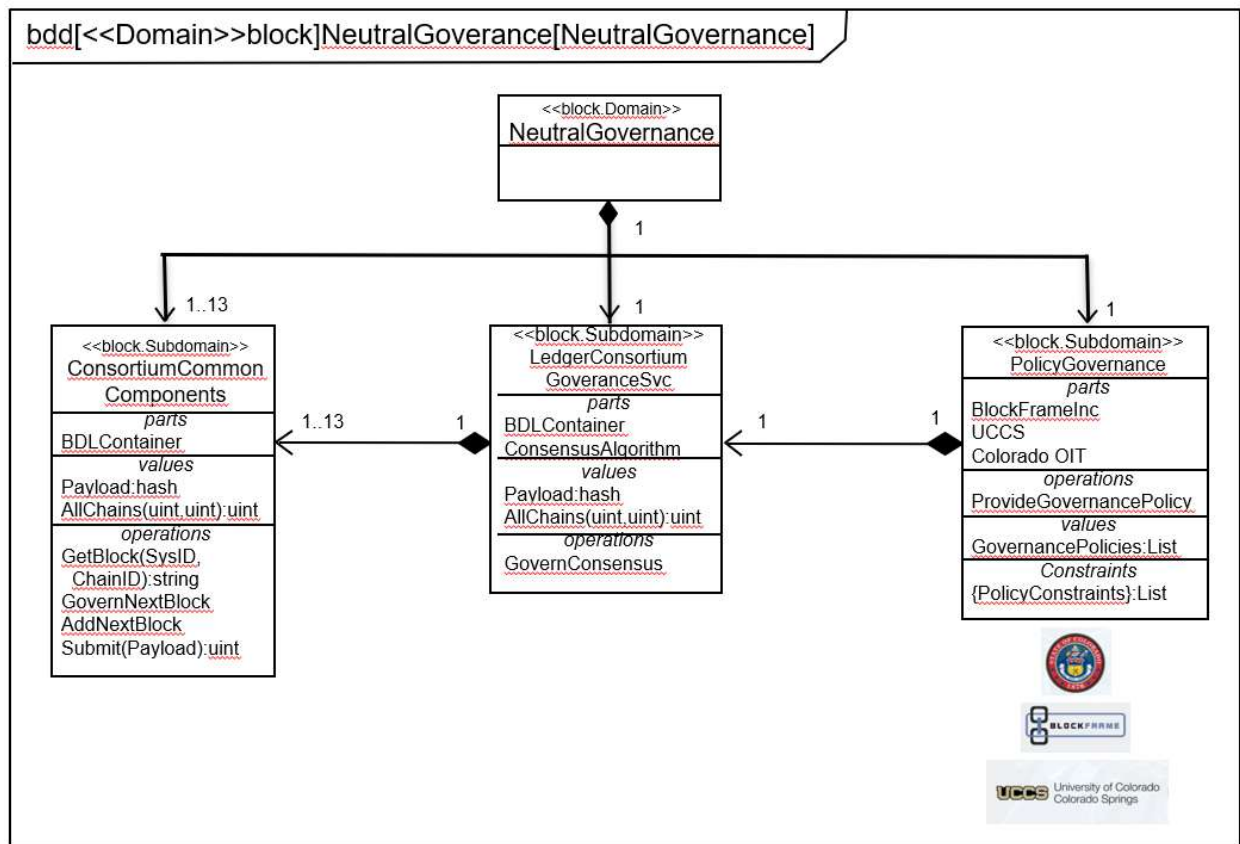


Figure 6.8. Neutral Governance Domain with Parts, Values, Operations, and Constraints

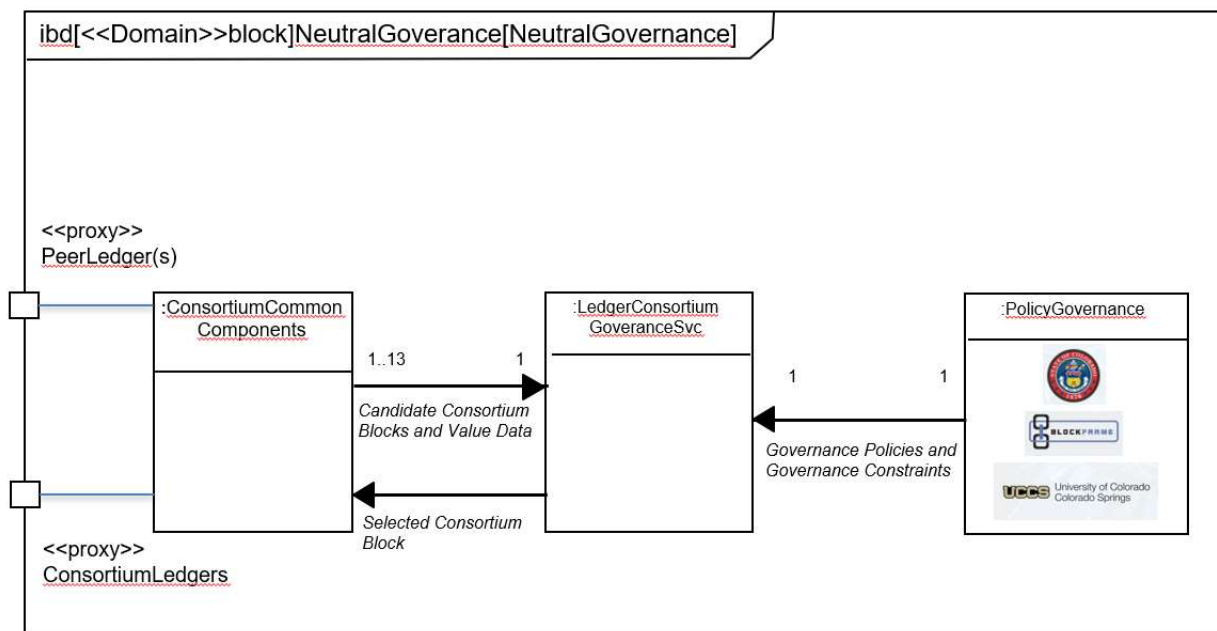


Figure 6.9. Neutral Governance Internal Block Diagram

Behavioral Perspective

The sequence diagram shown in Figure 6.10 represents an example of non-stateful behavior of a group of Blocks using a Sequence Diagram, include timing intervals for several parts of the behavior, using a snapshot of interactions within the Philos™ Blockchain Consensus process. A much higher number of Brokering Agents, Peer Ledgers, and Consortium Ledgers would typically be expected than those shown. A sequence diagram of the actual Enterprise would be proportionally very large but would probably scale up using elements similar to those shown. In the Philos™ Blockchain/Distributed Ledger Enterprise, the timing throughput should be manageable since many of the processing and calculative tasks can be done in parallel within the distributed system, but actual test and evaluation in a representative up-scaled system should be accomplished to confirm that. Key capabilities, requirements, and goals for timing would include the timing for scoring and selecting algorithmic methods to select the “next block,” and the timing necessary to synchronize the blockchain with blocks within the Enterprise.⁹⁴ With new requests for new blocks in the blockchain, the consensus process is competitive, and the “looping” nature of the sequence diagram implies that the most competitive requests could continue to be up-selected, while “less-deserving” users might have to wait longer. The nature of the “incentives” and/or “disincentives” for users are set by the controlling algorithms and the policies of the Neutral Governance Domain.

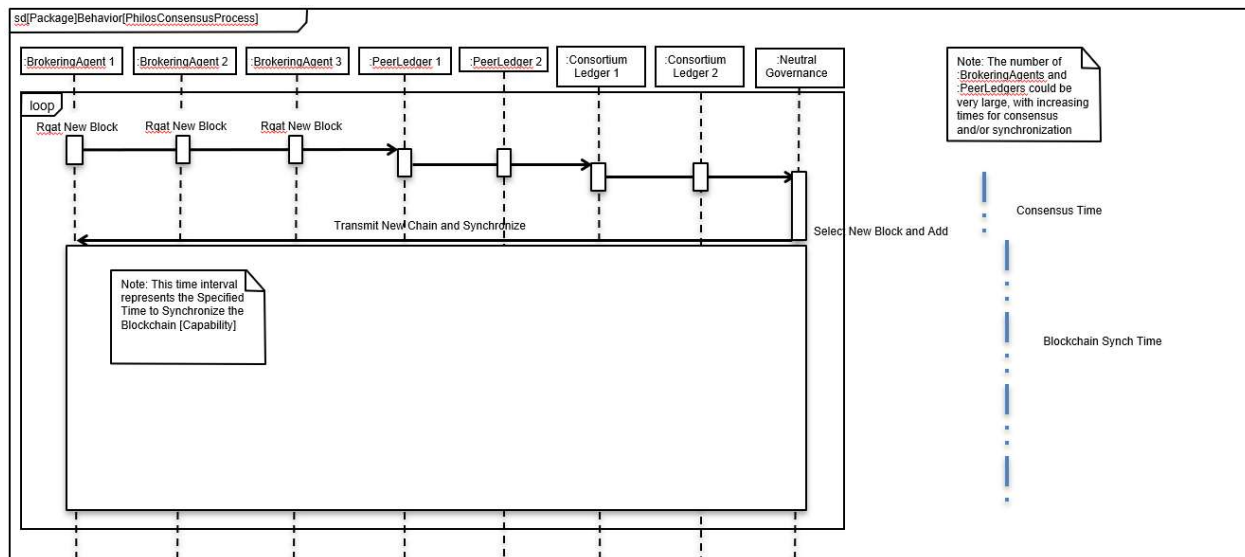


Figure 6.10. Philos™ Blockchain Enterprise Consensus Process

⁹⁴ Specific Philos™ capabilities and/or goals are proprietary and might not be releasable at this time.

The State Machine Diagram in Figure 6.11 represents an example of the high-level relationships of stateful behaviors between the ESP™ Enterprise Domain, and the Industry Utility Registrar™ (IUR™) Domain.⁹⁵ The State Machine Diagram shows the process for the connecting, testing, and provisioning of Logistics Integrated Circuits (LICs), possibly from a hand-held device or from cloud-based web services, to individual LICs within hosted devices. The LICs ensure the host devices are in a cryptologically-secure state. If they are not, they are taken off-line, wiped, and re-provisioned with new cyphers. Other design capabilities (not shown here) allow Proof-of-Origin of all data, so it would be implied that they can be returned to the last known secure state, re-provisioned with new cyphers, and then updates with the most recently available content stored on an associated secure blockchain. If any LIC or host device cannot be restored, they are shut down and isolated until IT intervention or replacements can be obtained (Exit Node). At key positions throughout the process, immutable transaction reports are sent to an associated blockchain for storage and/or recall when necessary. This behavior would be designed to loop endlessly under planned conditions; however, a fault mode could be used to restart at “Query Network.”

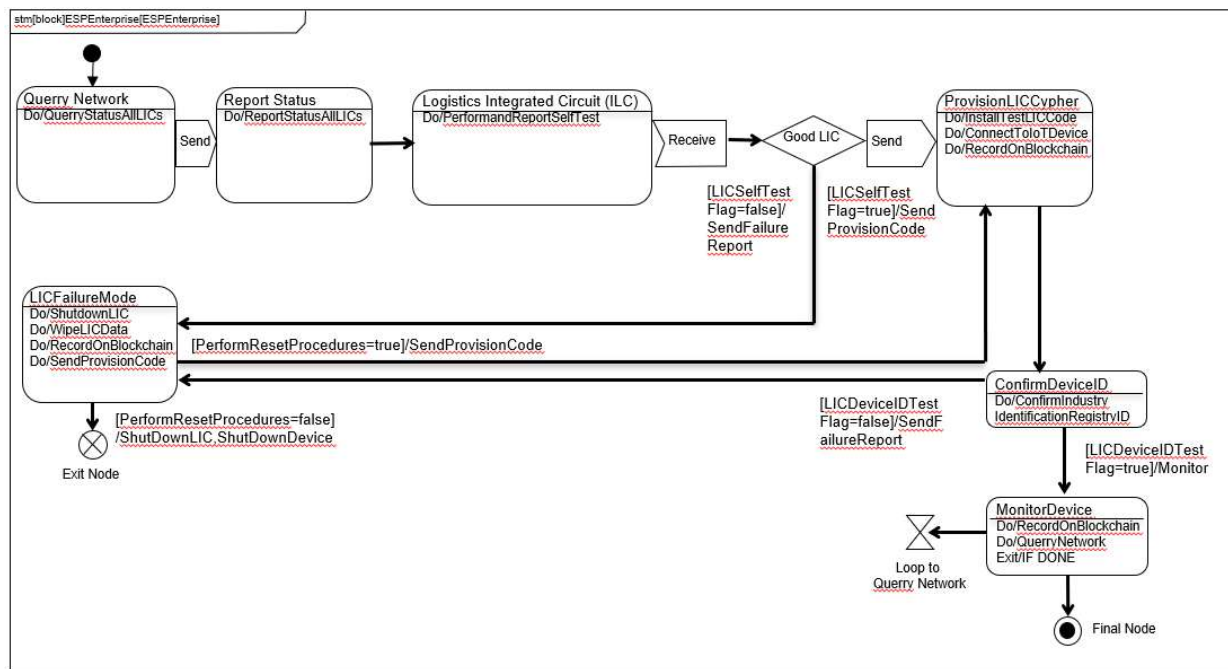


Figure 6.11. Example of a State Machine in the Eco-Secure Provisioning Enterprise

⁹⁵ Gorog, C. *DARPA Cybersecurity Logistics Management for Distributed IOT V2*. Unpublished. BlockFrame, Inc. Source Document Proprietary. 2019.

Data Perspective

Discussion: A Logical Data Model (LDM) Block Definition Diagram for three representative System Classes is shown in Figure 6.12. The diagram shown is not all-inclusive of the entire Enterprise but are key to most Philos™ blockchain data and operations. Note that each System Class could represent a very high level of multiplicity when operational.

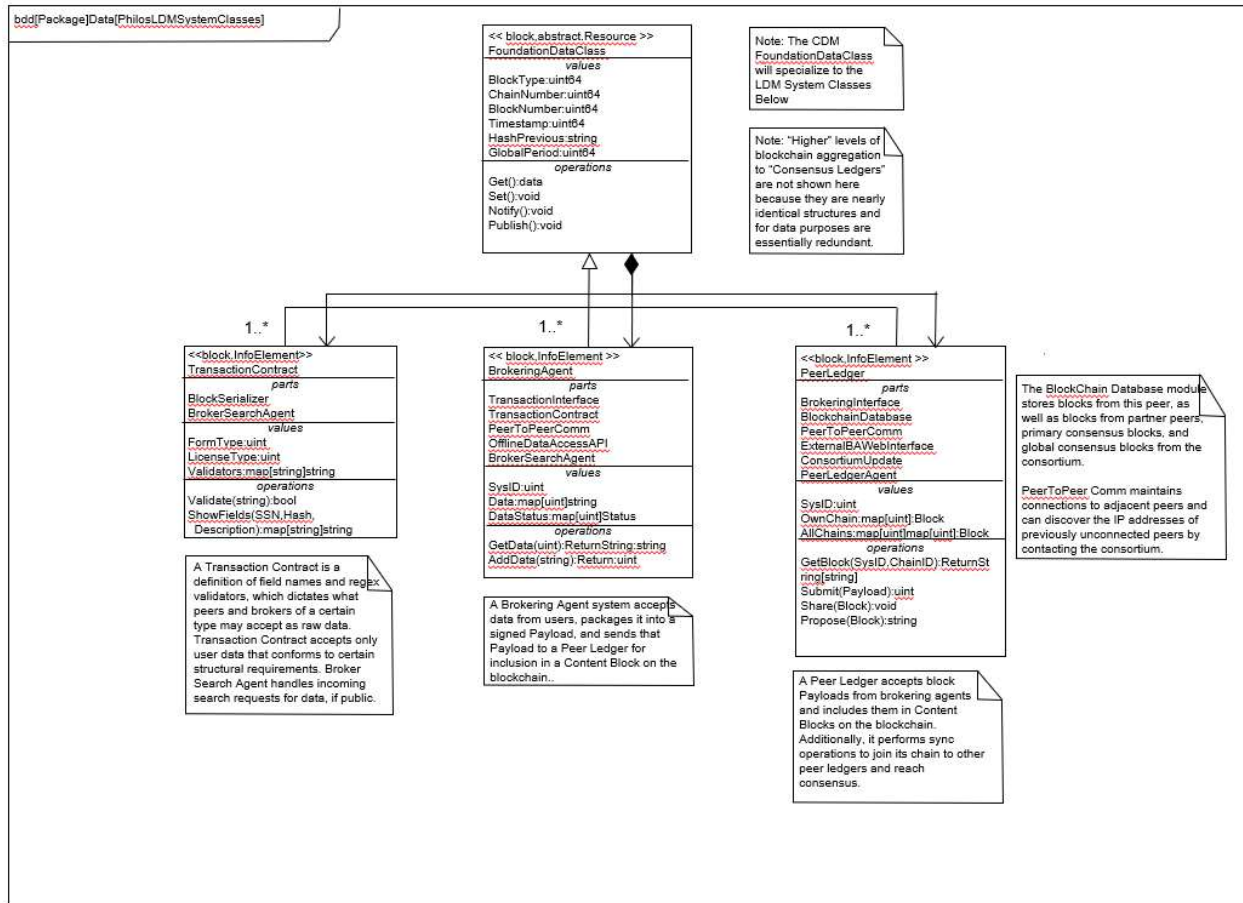


Figure 6.12. Logical Data Model (LDM) Block Definition Diagram with System Classes

Services Perspective

Figure 6.13 decomposes the Philos™ Enterprise Domain and the Neutral Governance Domain to identify Services. As explained earlier, as currently defined, every Domain within the ESP™ and Philos™ Enterprise Domains can be considered as internally accessing every layer of Service. This example of “allocation” of Block Operations shows the Operations and Services information in the Philos™ and Neutral Governance Blocks that mutually connect. Other Domains are not fully decomposed in this example. Note: Figure 6.13 and Figure 6.14 show expanded services taxonomies with additional data in the Block specification columns. Services would be exchanged between the Philos™ Enterprise Consortium Ledger Blocks (which aggregates the Peer Ledger Blocks) and the Neutral Governance Consortium Common Components Blocks. The key Service Operation of the Neutral Governance Domain is to “GetBlock(SysID,ChainID):string”, from recent requests for new blockchain blocks.

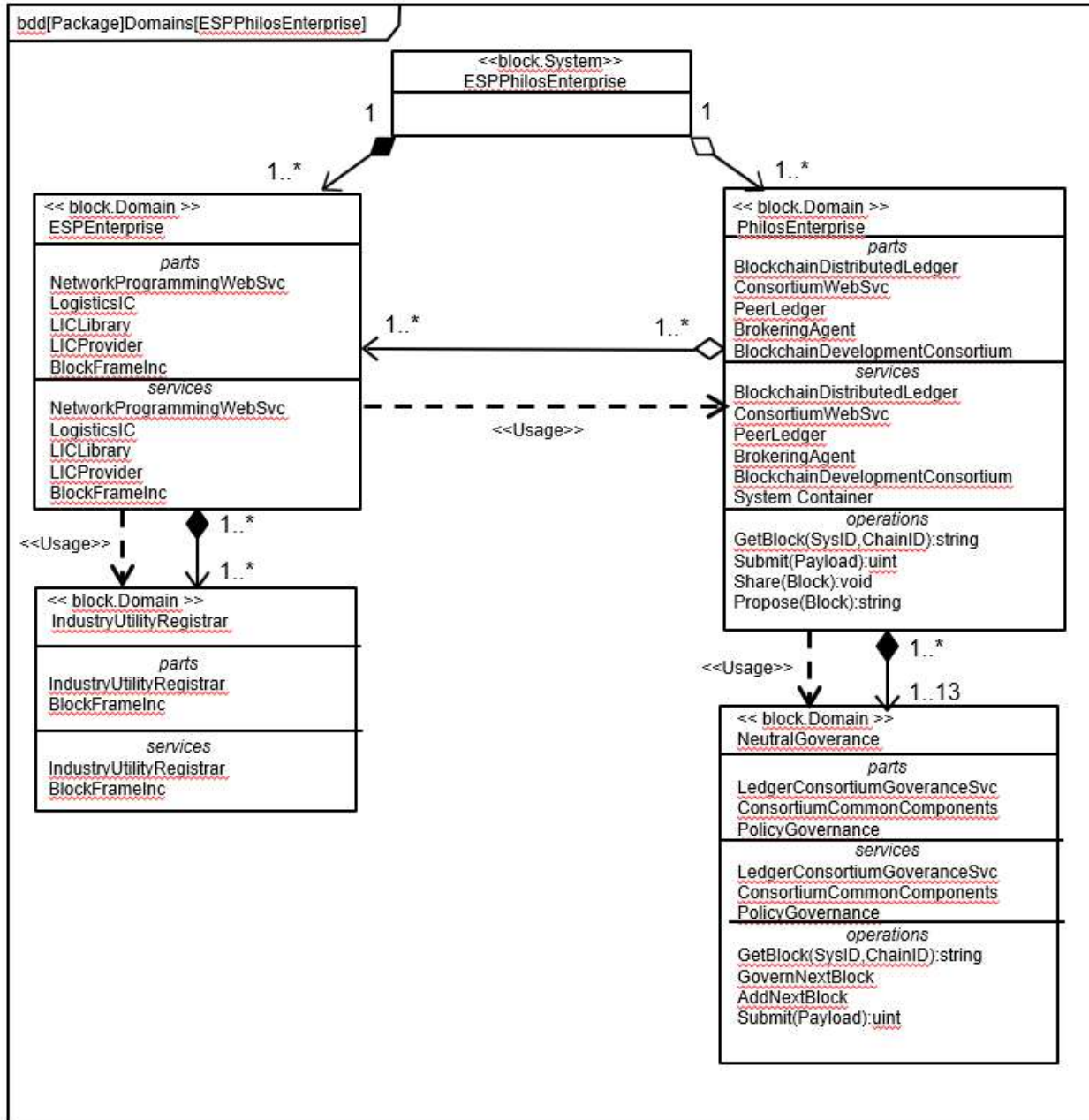


Figure 6.13. Combined ESP™ and Philos™ Enterprises with Allocated Services

The Neutral Governance Services Domain, shown in Figure 6.14, uses the Ledger Consortium Governance Svc, to compare, calculate, and score each candidate block from those requests; add the “winning” block (with block the highest defined value) to the blockchain; and submits (“Submit(Payload):uint”) the new blockchain version back to all the participants on the blockchain with the newly added block. (The “payload” is a cryptographically hashed version of the blockchain data.) In the current configuration, the “Consortium Ledger” Blocks of the Philos™ Enterprise, and the “Consortium Common Components” Block of the Neutral Governance Domain would be

“exposed” to these Service transactions between the Domains, even though both Domains would internally access each of their own “levels of service.” Strictly speaking, the “service” provided by the Neutral Governance Domain is to create the new blockchain version. The “Governance” service may be better described as an internal service within the Neutral Governance Domain to select the contents of the new blockchain version. All data must transit the System Container Service, which is the security level of service for inter- and intra-Domain data flows, but that should not alter the contents of the “Payload.”⁹⁶

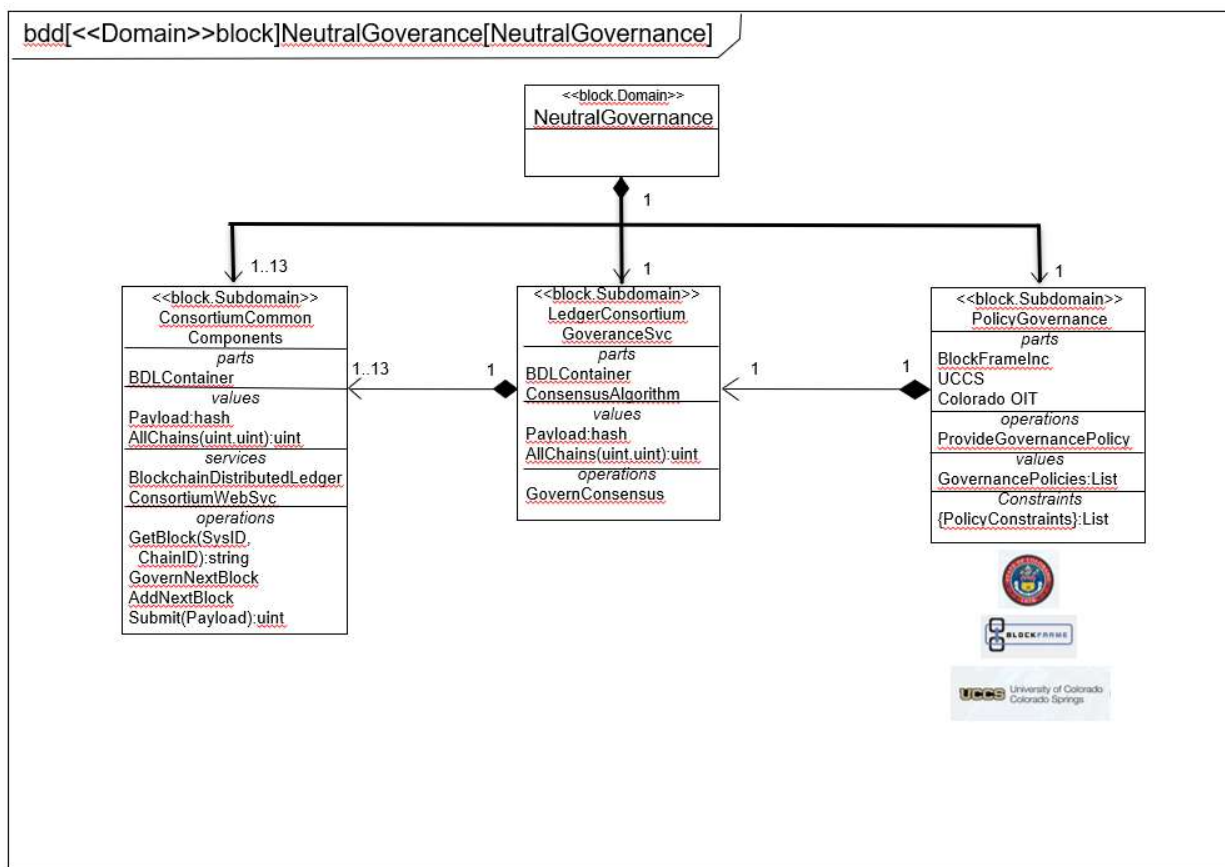


Figure 6.14. Neutral Governance Domain

The architectural layering of services is demonstrated in Figure 6.15 to represent the Domains of the ESP™ and Philos™ Enterprises, plus their “external” Services from the “Industry Utility Registrar™” and the “Neutral Governance” Domains. Each Domain should contain all service layers for internal operations, but their major

⁹⁶ For some applications, the System Container Service could operate independently as a Service for some applications like transmission/reception of secret cyphers and keys, essential software and operating system updates and patches, and other internal administrative information – conceivably even for applications totally outside the blockchain system. However, those are essentially not for immediate operational information used for blockchain genesis, so they are mentioned here, but not emphasized for this example.

services tend to align themselves into particular layers, as shown in Figure 6.15. Since each Domain structure must be independent of all others, and any interactions between Domains must be through their respective security layers, their only mutual interfaces must be route only through those security layers, and not directly between other system components. The “System Container” service shown within the Philos™ Enterprise is a unique service that provides services to every individual secure distributed physical element within all four Domains but may have full ownership only by the Philos™ Enterprise Domain. Therefore, other Security Layers within other Domains might have other additional security features, especially for external applications outside the ESP™ and Philos™ Enterprises. The unique System Container element is shown only within the Philos™ Domain. The use of each Service of any Domain, or any combination of Services, could totally depend on the Use Case being executed, and would likely not be used in all Use Cases. Therefore, for this example, all services to any external Domains must only take place through their respective security layers, as shown.

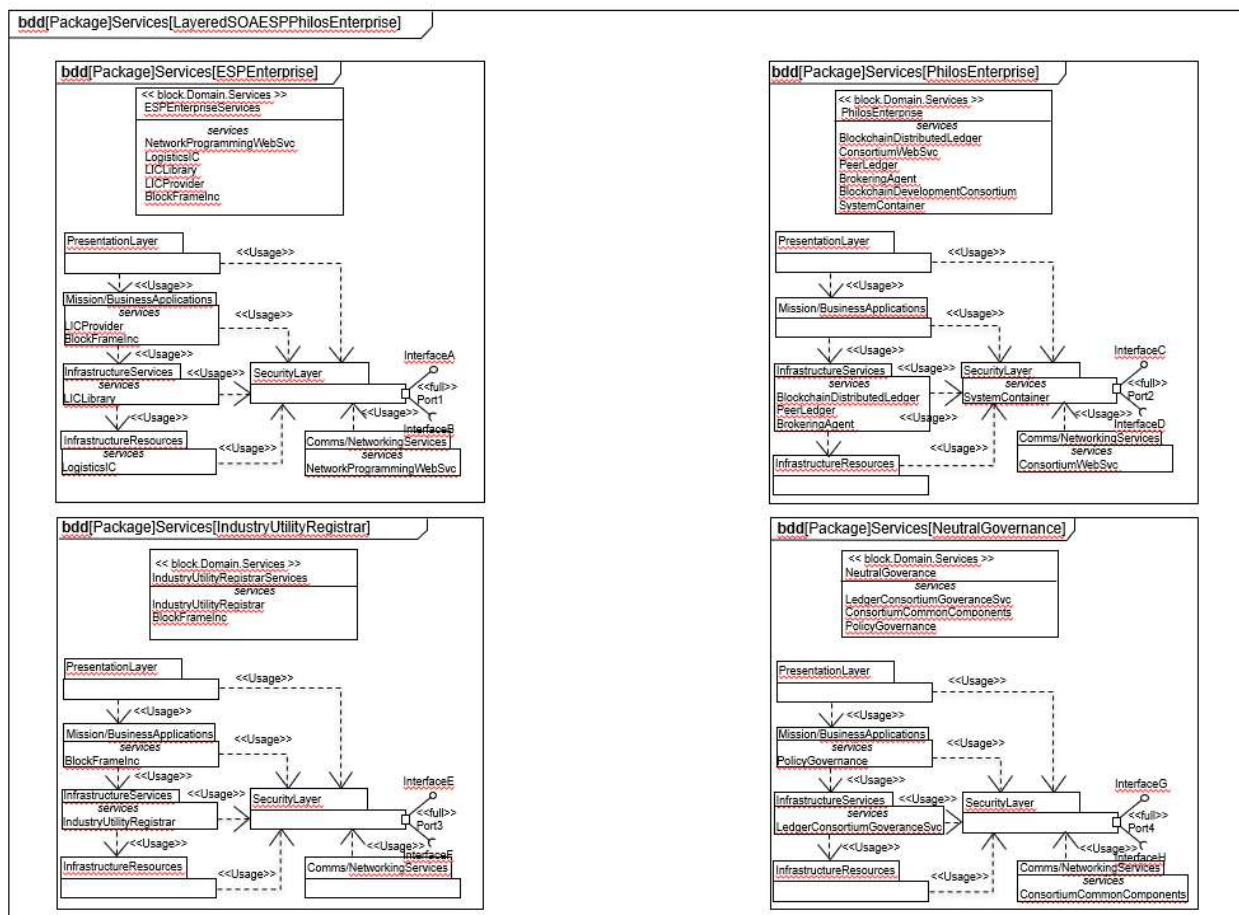


Figure 6.15. Layered SOA Services for ESP™ and Philos™ Enterprises, through Security Layers

Contextual Perspective

Useful contextual content for the Logical/Functional Viewpoint pertaining to the Philos™ Blockchain/Distributed Ledger Enterprise, specifically the motives behind the goals and requirements for blockchain consensus algorithms, is documented in Gorog, C., *Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management*⁹⁷. An excerpt from that document states:

“... After several scale tests in partnership with Colorado State University (Pueblo) the Blockchain Development Community BCDC created Philos™ to address known problems that affect nearly all existing publicly available blockchains as well as inconsistencies with many existing consensus models. The platform was designed to either optimize or eliminate the following common issues experienced with most existing distributed ledger platforms:

1. Wasted power spent on mining
2. Transaction speeds that cannot support large scale operations
3. Lack of incentives to store the ledger
4. The size of the full ledger expands out-of-control
5. Lack of offline support, as each ledger must be online continuously
6. Data End of Life (migration for phased-out data)
7. Non-deterministic size of each block
8. The need to balance visibility and privacy”

Those observations might be considered as current blockchain shortfalls, or as aspirational goals for system requirements/capabilities that are not currently specified as quantified requirements. However, they provide the context for as yet unfinished design goals that are the basis of Philos™ and represent some radical departures from the common stereotypes associated with the common use of blockchains, especially as applied for cryptocurrencies, or cryptocurrency-based programming languages. To expand on some of the “known problems” described above, I would add the following contextual inputs:

⁹⁷ Gorog, C. *Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management*. Unpublished Draft. 2019. All Rights Reserved.

1. Electrical energy worldwide just to mine Bitcoin has exceeded the total power output of Switzerland, and more than half of the power output of Colorado Springs Utilities. This is a terrible and perhaps unethical waste of electrical power to prove the “work” and “stake” often assumed for the mining of common cryptocurrencies. Philos™ would use a different incentive/disincentive structure to “make ethics profitable” by improving collective trust and security.
2. Large blockchains currently require synchronization of large masses of data on a regular basis. Little consideration has been given to how those adversely affect runtimes, network throughput, etc. Philos™ design would attempt to partition operations more efficiently to significantly improve runtimes.
3. If “mining” with electrical energy is the payment for participation and mutually storing distributed blockchains, for many it has also resulted in non-profitability. Many (mostly smaller) “get rich quick” users have “bailed out” as mining and storage have become less attractive because of its high cost. Philos™ would not “mine” energy, so would not participate that that economic distortion.
4. The current size of the full Ethereum blockchain well exceeds a Terabyte, and has become less practical to use, update, and share because of its goal to retain content. Tests with CSU-Pueblo showed that in even modest tests with a Hyperledger Fabric engine, the size of the ledger quickly became unmanageable, apparently because of extraneous “overhead” information contained in the chain. The custom-designed Philos™ system would not attempt to store vast quantities of information on the blockchain itself, but securely “point” to offboard encrypted storage that would not overburden the blockchain itself.
5. If users cannot participate unless they are constantly online, that creates an unacceptable burden for many. Philos™ would be designed to allow users at least temporary periods offline, or temporary interruptions.
6. End-of-Life at the end of the blockchain life cycle has been generally ignored by everybody. Ethereum and others have conducted “hard branches” to new versions of their code to separate from previous ineffective or inefficient versions, but when their end-of-life eventually occurs, whether they will die catastrophically or gracefully is typically ignored. Philos™ specifically considers and manages blockchain growth and end-of-life issues.

7. Nondeterministic block sizes can refer to the sizes of typical hashed blocks. If they are too big, they might be inefficient, if too small, they might be less secure. This issue highlights those tradeoffs and are considered and managed in the Philos™ blockchain.
8. Visibility (transparency) and Privacy are often at odds. If the operations and governance of blockchains are totally non-transparent, they could reward unethical or illegal transactions, but a total lack of privacy could impinge on expected civil rights. The right balance should be designed into the Philos™ system, and into its governance.

Moving forward with the study of the structure and governance of Philos™ development, all of these factors need to be considered, tested, verified, and validated during development. As a result, favorably disruptive tools using blockchain principles could be designed into many practical architectures that have not yet been realized due to current over-identification and over-reliance of many existing blockchains with standard cryptocurrency methods that have proved to be insufficient. The goals of Philos™ could have significant and positive legal, public policy, and social implications, as well as providing personal privacy, supply chain security, and national security benefits. If successful, it could also help fundamentally redefine common understandings and perceptions of blockchain/distributed ledger methods.

6.2.4. Physical Viewpoint

Design Perspective

A previous Domain Diagram showed the merger of two enterprises: the Eco-Secure Provisioning Enterprise (ESP™), and the “Philos™ Blockchain/Distributed Ledger” Enterprise, and is reintroduced here as shown in Fig 16.

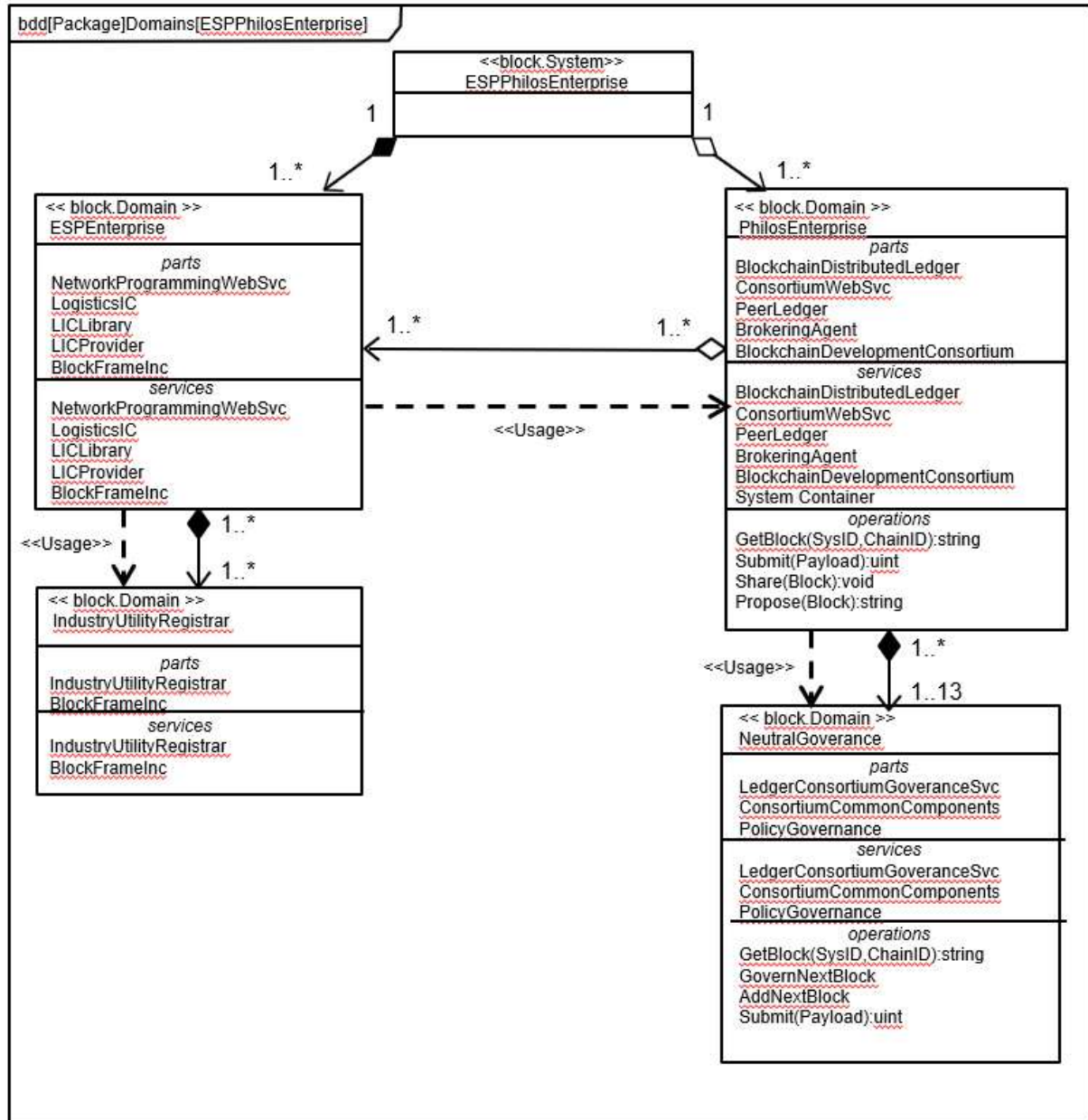


Figure 6.16. Enterprise Domain Diagram.

A Block Definition Diagram for a Logical Data Model (LDM) of the Philos™ Enterprise was also previously shown but was limited to only three key elements of the Philos™ Transaction Contract, Brokering Agent, and Peer Ledger, as shown in Figure 6.17.

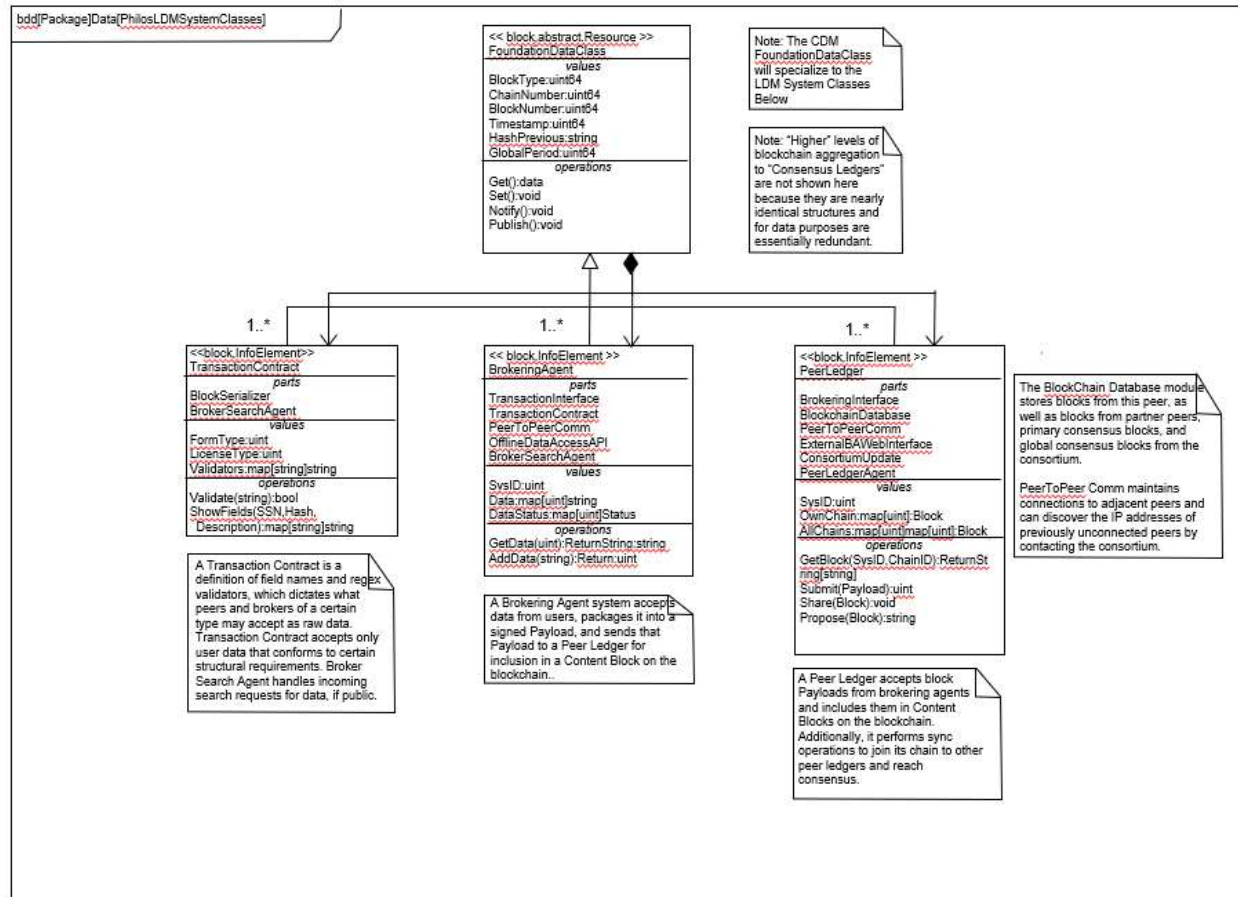


Figure 6.17. Logical Data Model of Selected Philos™ Blockchain Elements.

Those three elements are next modeled using three Block Definition Diagrams to describe the structures of the Philos™ :TransactionContract, the :BrokeringAgent, and the :PeerLedger from the Physical Viewpoint (PV). Note: these examples are almost totally notional because no final physical structural decisions have yet been made at this time. BlockFrame, Inc. and BCDC data were derived from Gorog, C. *Blockchain Consortium Governance Community Sourced Project*,⁹⁸ and (Gorog, C. *Block Chain Cybersecurity Logistics Organization for Globally Scalable Security Management*.⁹⁹ Unpublished. 2019. All Rights Reserved.)

⁹⁸ Gorog, C. *Blockchain Consortium Governance Community Sourced Project. Software Design Document for Blockchain Synchronization Consensus Method*. Unpublished. 2019. All rights reserved.

⁹⁹ Gorog, C. *Blockchain Cybersecurity Logistics Organization for Globally Scalable Security Management*. Unpublished Draft. 2019. All Rights Reserved.

For considerations of data flow, one should think of data flowing in one direction only from one :TransactionContract to its one associated :Brokering Agent to an associated :PeerLedger, which could also be associated with more than one :Brokering Agent. However, from the Physical View, one should think about the :TransactionContract being embedded or nested inside its one associated :Brokering Agent, and perhaps several :Brokering Agents being connected to its associated :PeerLedgers. To depict that architecture in Block Definition Diagrams, a :TransactionContract Block can be considered as a Part of a :Brokering Agent Block, and model it separately in its own Block. Those relationships will be shown in Figure 6.18 as a :TransactionContract Block, Figure 6.19 as a :Brokering Agent Block, and Figure 6.20 as a :PeerLedger Block.

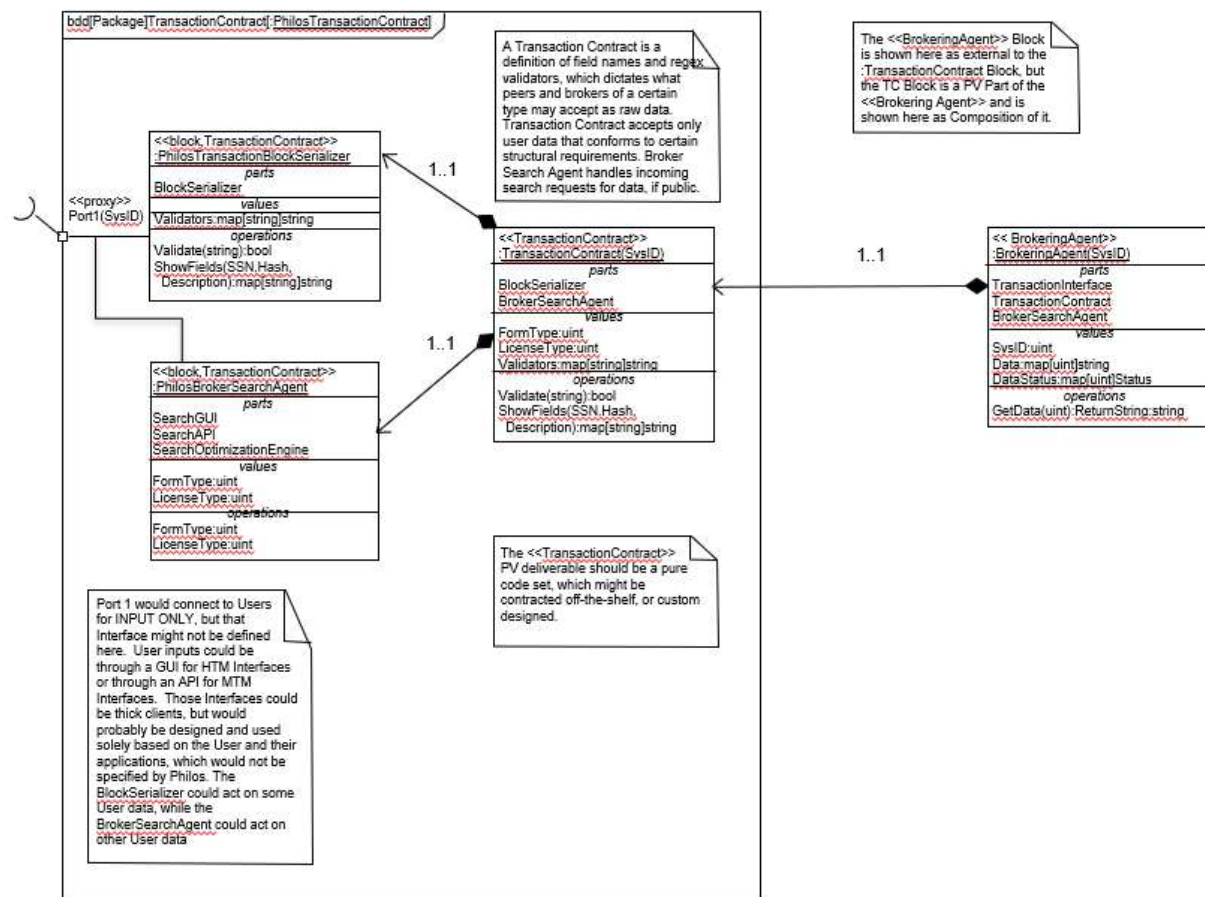


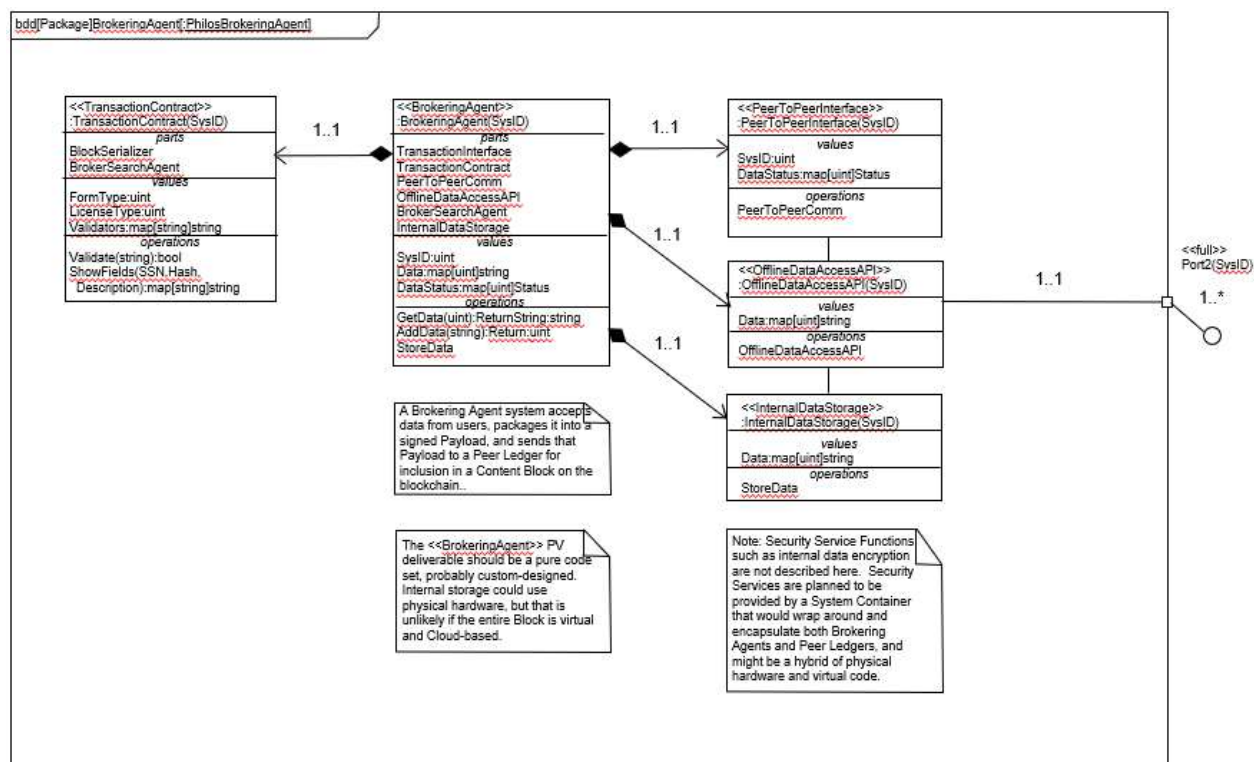
Figure 6.18. Block Definition Diagram of a Transaction Contract

Since in the PV, the :TransactionContract should be embedded internally within the :BrokeringAgent, but is fully Composed as a Part of the :BrokeringAgent, that is depicted by the :BrokeringAgent as external to the

:TransactionContract Block. The Port1 accepts inputs only from User GUIs or APIs but would probably not specify what the User’s would use, other than a generic Port for the connection. Depiction of data flow to the associated :BrokeringAgent should probably not be necessary since the :TransactionContract should be physically embedded within it – at least within its software.

It is possible that the PV requirements of some or all of the Philos™ :TransactionContract could be satisfied by an off-the-shelf product, as the “front end” of the :BrokeringAgent. The feasibility of that has not been determined or tested, but similar generic products could be available, such as from Clause, Inc.¹⁰⁰, which follows a similar logic process. If that were the case, the PV specification could be the Clause, Inc. product. An initial examination, however, has not indicated that product would probably not meet Philos™ requirements.

Figure 6.19 depicts a representative Block Definition Diagram of a Philos™ Brokering Agent. In this Diagram, the :BrokeringAgent collects user data from the :TransactionContract and stores User requests until transmission through an API to the Brokering Agent’s associated Peer Ledger. The associated Peer Ledger could provide services for many Brokering Agents.



¹⁰⁰ <https://clause.io/>

Figure 6.19. Block Definition Diagram of a Brokering Agent

Figure 6.20 presents a Block Definition Diagram of a representative Philos™ Peer Ledger. One Peer Ledger could collect data from multiple Brokering Agents and must perform a process to synchronize their respective blockchains/distributed ledgers. Those synchronized ledgers – now called “Sidechains,” are then combined with ledgers from other multiple Peer Ledgers and sent to :ConsortiumServers to permanently update the main BlockChain. The timing of that process is critical but will be explained later.

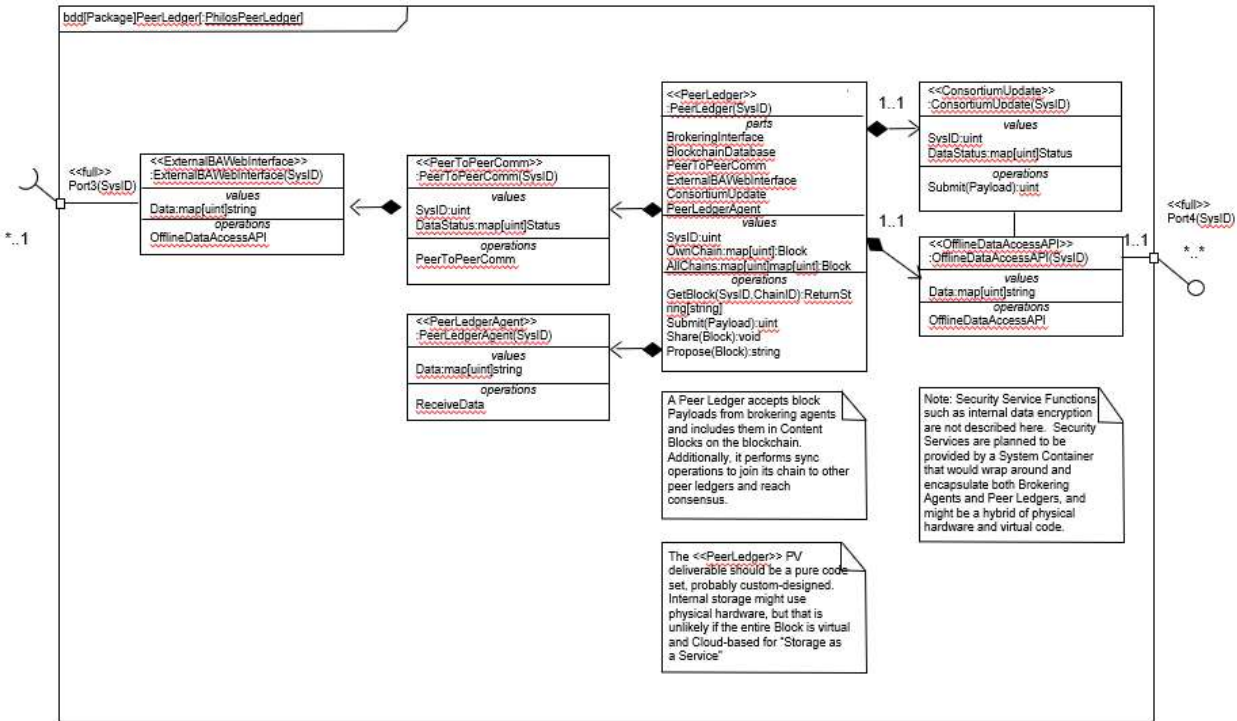


Figure 6.20. Block Definition Diagram of a PeerLedger

To support learning objectives, three specifications will be detailed for important interfaces:

1. The <<OfflineDataAccessAPI>> is a key internal interface type within the Philos™ Enterprise. It is shown as its own Block Part within both the Brokering Agent and the Peer Ledger Block Diagrams and should be a standardized and reusable interface and a common element within all Brokering Agents and Peer Ledgers throughout the Enterprise. Technically, it could be considered an interaction rather than an interface since final interface for all Brokering Agents and Peer Ledgers must be filtered through the security service layers of their respective System Containers prior to being exposed to Web Service communications, but from the Data Perspective, the <<OfflineDataAccessAPI>> should be considered

the data content interface. As shown in their Block Definition Diagrams, the Specifications for the data content should be repeatable and straight-forward Packages of blockchains or other control data.

2. The User GUIs/APIs are essential interfaces to gain access into the PhilosTMBlockchain. Their physical characteristics cannot be determined because they could vary with every user or user application. Because of that, it is likely that Users might want to develop their own thin client as an interface between their applications and the PhilosTM Enterprise. From the PhilosTM side, there must be commonality and standardization at all :TransactionContract points of service, which, as described earlier, must be an internal Part of the Philos Brokering Agent modules, and must be filtered through the security System Containers wrapped around each Brokering Agent. The data specifications for the User GUI/API inputs to a :TransactionContract :Port1(SysID) is shown in Figure 6.21. From the PhilosTM side of the Port (with the SysID to identify which port number is associated with each discrete Brokering Agent in the Enterprise registry), all of the essential data taxonomies for that point of service should be listed.

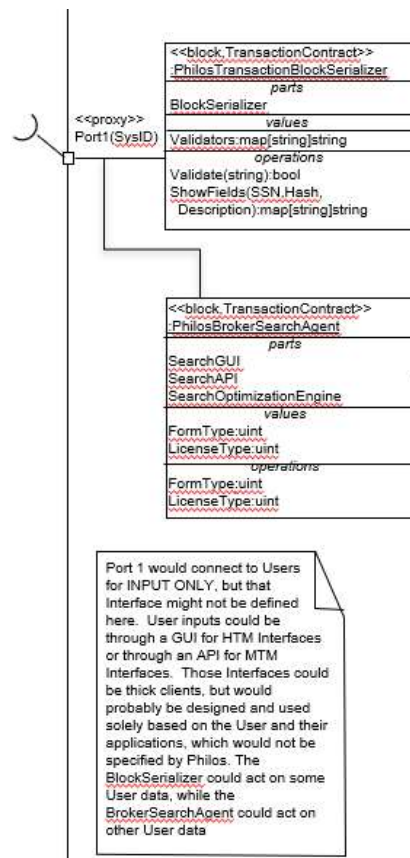


Figure 6.21. Specifications of a Transaction Contract Port

3. A third essential interface in the “Eco-Secure Provisioning™ Enterprise (ESP™),” are connections to a Logistics Integrated Circuit (LIC) embedded within secure devices that houses or connects with client devices and applications and could be devices that are part of the Internet of Things (IoT). Within the Domain Diagram in Figure 6.16, that LIC is identified as “LogisticsIC.” Although a specific LIC product has been identified (source and specifications are sensitive and will not be included here), the common interface that houses the LIC and connects to each physical device has not been selected and could be one of several interface devices. In some cases, custom designs might be necessary for different kinds of user devices, but common units should generally be able to be standardized. One example of the specifications of a product/service that might satisfy some or all those requirements is PIACET <https://piacet.com/home#features>.¹⁰¹ Again, the specific specifications of the LIC that are needed to have compatibility with the secure device are simple: they are just the specifications of the LIC itself. In most planned cases, connections to downstream services would then be through Philos™ Blockchain services, which are described for “Port1” in Interface on Figure 6.21. The PIACET-like API device could be used in conjunction with User machine-to-machine applications to connect their unique devices into the ESP™ Enterprise and/or the Philos™ Enterprise. The exact configurations could be system-dependent, but loose coupling service connections should be the standard requirement.

¹⁰¹ PIACET <https://piacet.com/home#features>.

Standards Perspective

Standards shown in Table 6-1 of Borky and Bradley, 2019. P. 234¹⁰² generally apply, especially for Transport Services using standard Web Services. Development of Security Protocols and Standards are currently in development, but typically exceed standards listed in the Borky Table 6.1, including high encryption standards and methods. For example, Transport Layer Security (TLS) would be used in lieu of the Secure Sockets Layer (SSL) listed in that table. Since the PhilosTM Blockchain is based on unique processes for building and proving trust among the participants, virtually all data is encrypted, and monitored and enforced by the Neutral Governance processes over time. Data descriptions have not been totally resolved, but JSON, GoLang with structured access libraries, some Python, and other languages are routinely being used. SQL-Light was used for initial R&D, but a more fully functional and secure data language will probably be selected later. Most Web Services should naturally use WS-* standards.

Using either a Sequence Diagram from the Logical View or a new one developed for the Physical View, annotate with timing information to show allocation of an overall deadline to the individual steps of the process.

One of the most important driving processes in any blockchain is collecting data from requests for transactions from multiple users and eventually recording them in an immutable and non-repudiable, cryptographically secure blockchain under time constraints. The PhilosTM Enterprise attempts to do that efficiently by collecting requests for transactions from a large number of distributed Users (through dedicated Brokering Agents), and processing (synchronizing) those requests among a distributed set of Peer Ledgers. Then, following governance rules, those collected inputs are synchronized into a new blockchain called a “sidechain” at the level of each Peer Ledger within a determined time limit. Sidechains are then sent in turn to Consortium Ledgers to repeat a similar process under governance rules to create a final enterprise-wide blockchain on a daily basis¹⁰³. In some blockchains, Governance and the creation of new blockchain blocks could be based on the amount of energy spent “mining” for value. On the PhilosTM Blockchain, that does not take place, but the “value function” can be derived from several incentives or disincentives, for example, the behavior of the Users to comply and enforce their own security, or to cooperate with other Users to actively participate. Figure 6.22 and Figure 6.23 show that expanded

¹⁰² Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

¹⁰³ In summary, every transaction among peers, sidechains, and entry into the global enterprise blockchain of the PhilosTM B/DL involves group consensus in throughout. This process mandates, enforces, documents, and stores digitally-signed records on each block to prove that Zero-Trust processes must occur within every activity on the PhilosTM B/DL.

process through two sequence diagrams. Figure 6.22 shows that all Peer Ledgers must Synchronize all of their Brokering Agent transaction requests within XXX minutes. Figure 6.23 shows that the entire Philos™ Blockchain must finish YYY percent of consensus by governance standards on all Peer Ledger Sidechains in the Enterprise within 24 hours¹⁰⁴. The new 24-hour Blockchain is then transmitted for distribution back to all the Peer Ledgers in the Enterprise for storage.

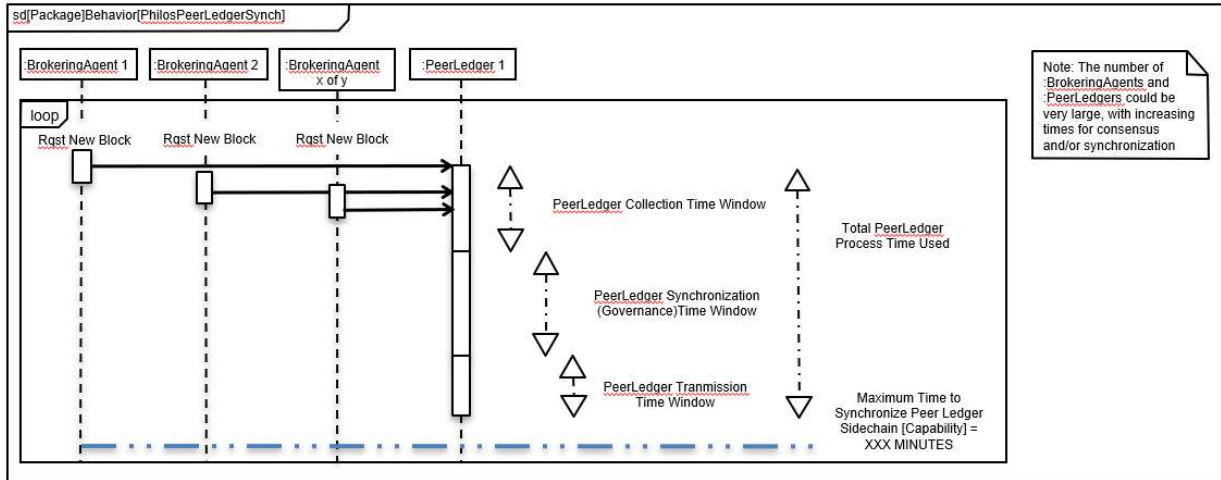


Figure 6.22. Sequence Diagram of Philos™ BDL Peer Ledger Synchronization

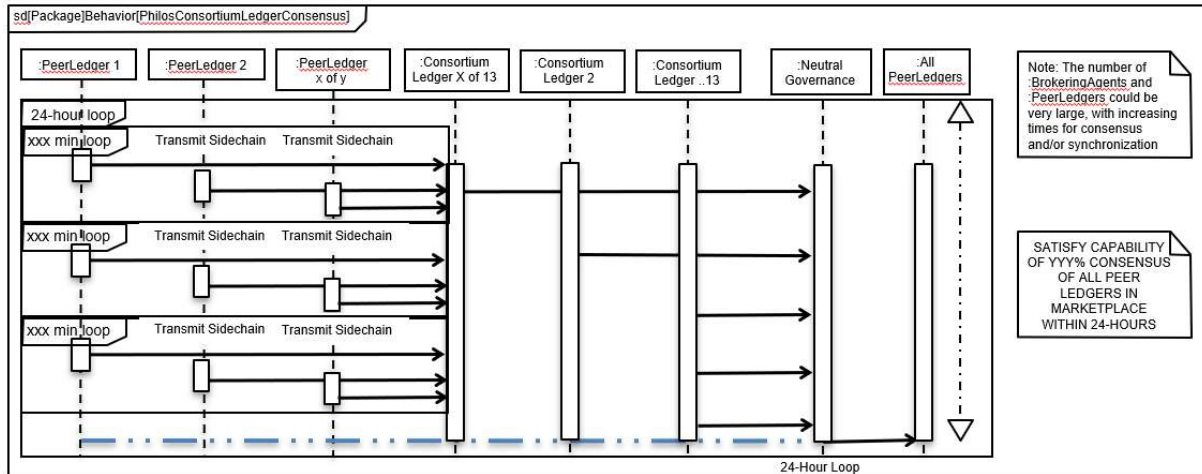


Figure 6.23. Sequence Diagram of Philos™ BDL Consortium Ledger Consensus Interactions

¹⁰⁴ The specific capability values of XXX and YYY are BlockFrame® commercial proprietary.

Services Perspective

The Services Taxonomy for Layered Service-Oriented Architecture Services was shown in Figure 6.15, which already allocated major services to the major Service Blocks of the Enterprises. That diagram is repeated in Figure 6.24 for convenience. Changes to the Block Definition Diagrams in Figures 6.3, 6.4, and 6.5, and the descriptions of specific interfaces shown elsewhere in this project added quite a bit more specificity, especially in service invocation details through GUIs and APIs, and Block descriptions of the Philos™ InfrastructureServices, and the Comms/Networking Services, and some of the Services within the Neutral Governance Domain.

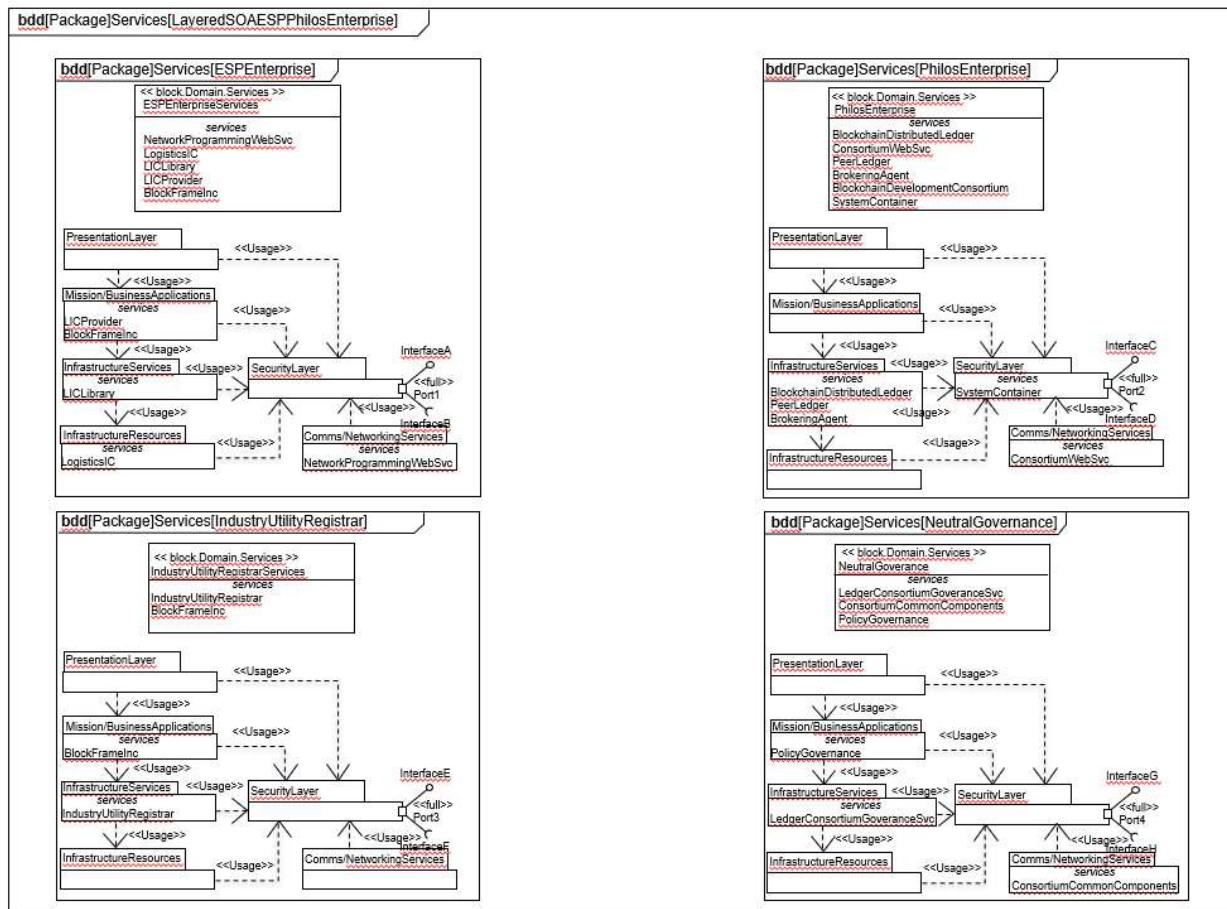


Figure 6.24. Layered SOA Services for ESP™ and Philos™ Enterprises, through Security Layers

To expand to a more complete Services taxonomy, Figure 6.25, shows several `<<usage>>` annotations, and block services among the highest levels of the “Eco-Secure Provisioning Enterprise (ESP™),” `<<block.Domain>>` ESP™Enterprise, and the “Philos™ Blockchain/Distributed Ledger (BDL)” Enterprise `<<block.Domain>>`

PhilosEnterprise, plus two non-enterprise Blocks, the <<block.Domain>> Governance, which acts as an independent service for governance to the Philos™ BDL Enterprise, and the <<block.Domain>> Industry Utility Registrar™, which acts as an independent service for registries for the ESP™ Enterprise.

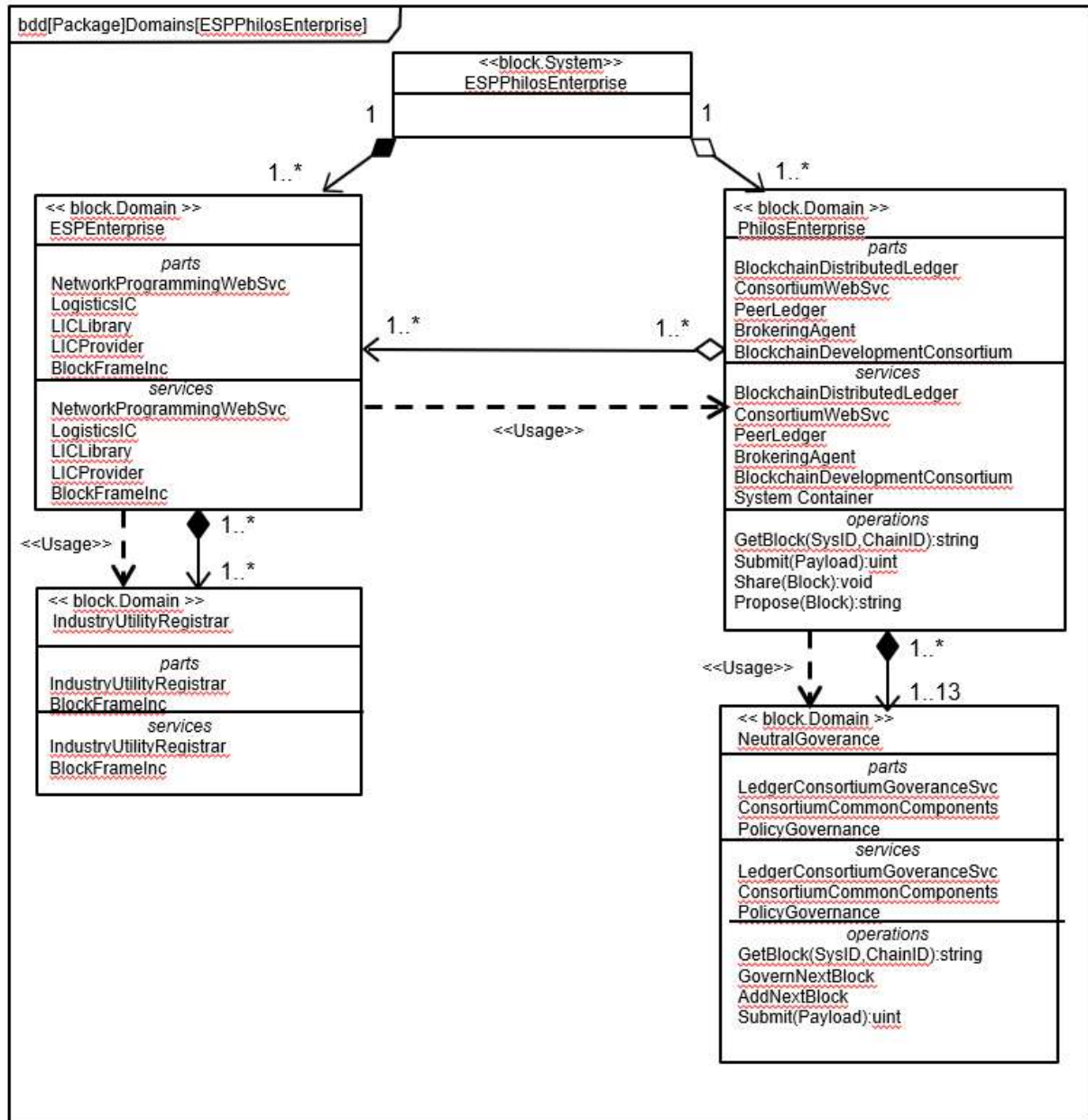


Figure 6.25. Philos™ and ESP™ Domain BDDs showing services.

Table 6.2 summarizes the taxonomy of services provided by each Block in Figure 6.25, the respective user of each Block, whether the service users are internal or external, or both, whether the component is an interface, and to which component or to which user the service is provided. For this visualization, it is assumed that this particular

system is the combined ESP™/Philos™ system of enterprises, is comprised of all four Blocks, and the service interactions occur within that system. However, since the Philos™ Enterprise is also severable and may work independently for other purposes, it might also provide external services to other users. It is assumed that “internal” services occur within each Block, and “external” services occur among external Blocks.

Table 6.2 Taxonomy of Services Among Blocks

Block:	Services Provided:	Service Type:	Interface:	Provides Service To:
<<block.Domain>> ESPEnterprise	NetworkProgrammingWebSvc	External	Yes	<<block.Domain>> ESPEnterprise
	LogisticsIC	External	Yes	<<block.Domain>> ESPEnterprise
	UCLibrary	External	Yes	LogisticsIC
	LICProvider	Internal	No	LogisticsIC
	BlockFrameInc	Internal	No	UCLibrary
<<block.Domain>> PhilosEnterprise	BlockchainDistributedLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	ConsortiumWebService	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises; Neutral Governance
	PeerLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	BrokeringAgent	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	BlockchainDevelopmentConsortium	Internal	No	<<block.Domain>> PhilosEnterprise
<<block.Domain>> NeutralGovernance	LedgerConsortiumGovernanceSvc	External	Yes	<<block.Domain>> PhilosEnterprise
	ConsortiumCommonComponents	Internal	No	Internal
	PolicyGovernance	Internal	No	Internal
<<block.Domain>> IndustryIdentificationRegistry	IndustryIDRegistry	External	Yes	<<block.Domain>> ESPEnterprise
	BlockFrameInc	External	Yes	<<block.Domain>> IndustryIdentificationRegistry

Figure 6.26 shows exemplary block definition diagrams of services within the ESP™ and Philos™ Enterprises, including the Industry Utility Registrar™ (IUR™), and the Neutral Governance Blocks, with annotations for service documentation, modifying diagrams from shown previously. This representation shows only the high organizational levels of the Enterprises and does not represent each component or element of every system. Simple notes are anchored to each respective Block Port to describe the Services that are provided, requested, and exposed. In all cases, the Ports are directly connected to the Block’s security layer. In the case of the Philos™ Enterprise, those security layers take the form of a security System Container, which should be the only exposed component, and would normally encapsulate all other components in that Block. In the Physical Viewpoint, those blocks could be totally virtual, totally physical, or (more likely) a hybrid of both. That design instantiation will depend on cryptographic requirements and will probably not be fully known until prototype testing can be performed. The other components in the non-Philos™ Blocks would generally follow the same security level design, and for this discussion are assumed to be identical common components as the Philos™ System Containers. In all cases, services should transit the security layers, but the security layers themselves would generally not provide any unique services other than security. In other Use Cases, it might be possible that Philos™ System Containers could provide only “Security as a Service” outside these Enterprises, but that would be a separate Use Case not currently under discussion here.

We will assume here that the distributed System Container security layers are simply internal to each Block and are not a separate Service. Consistent with Table 6.2, the external Services of each Block and their respective destinations are described in each simple note.

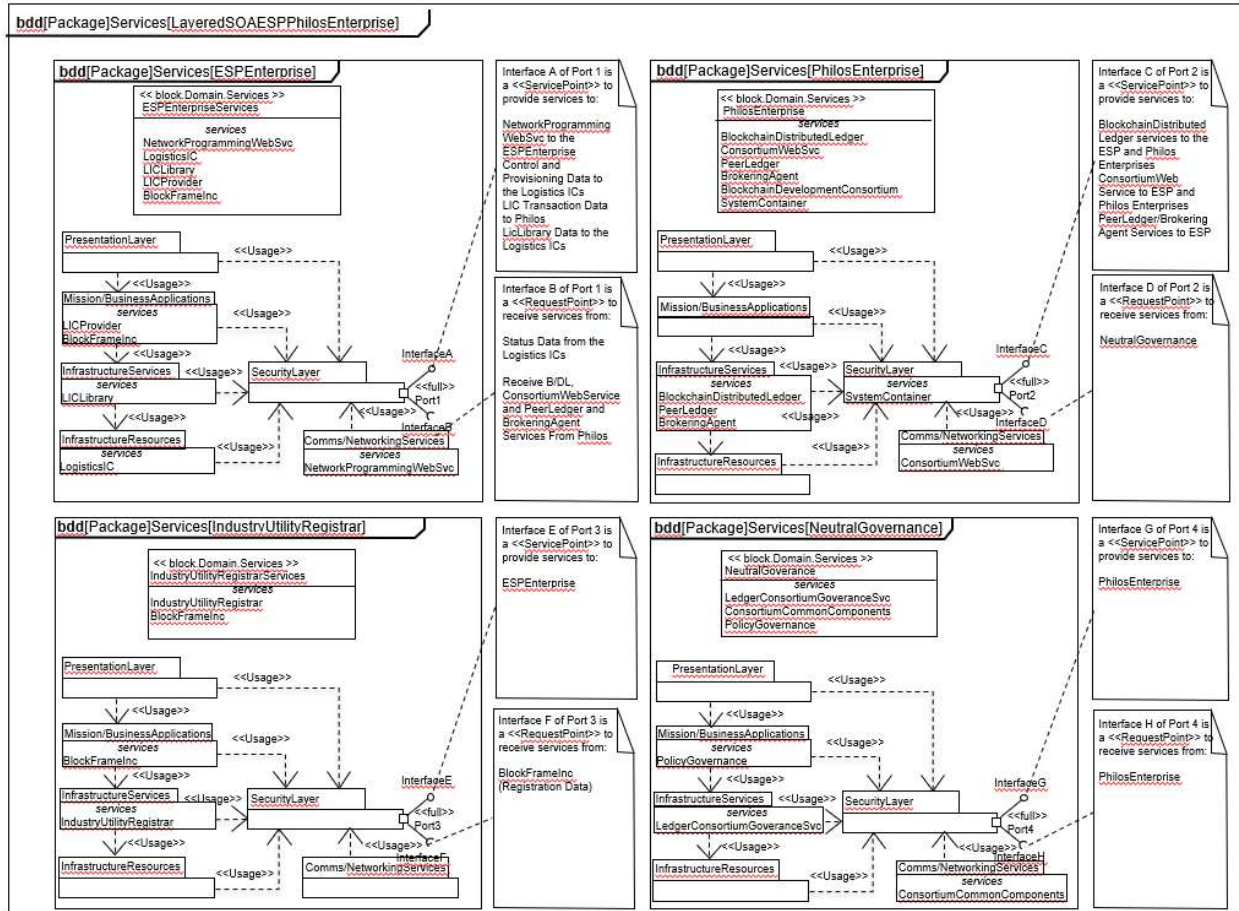


Figure 6.26. Philos™ and ESP™ Domain BDDs showing services

At the ESP™ Enterprise and Philos™ Enterprise levels of organization, all protocols and standards associated with services should be associated with the respective Ports of each BDD Block shown in Figure 6.26. Since each Port is associated and exposed only with its own respective security level component, the necessary common protocols and standards should be the same, or nearly the same, for each Block. Considering the goals of loose coupling, and for simplicity, each Port should be of a common design for each Block at the Enterprise-level of organization. That is not to say that every Port at every organizational level in each Enterprise would need to follow the same standards, as they could be several layers below the Enterprise level. For example, HMIs or APIs used to input user requests would be managed at the individual system or component levels, not on the interfaces at the

Enterprise level. At the Enterprise level, in the Physical Viewpoint, the interactions among the Blocks in Figure 6.26 that process service requests and delivery should all be capable of high-end protocols in the transport layer. Currently, TCP/IP is the most likely candidate, and will probably use a DNS standard on Port 53, and a custom-designed Media Access Control (MAC)-filtering-like process to ensure only known element addresses are allowed to participate in the Enterprises under uniquely assigned System IDs (SIDs). With those considerations, a single representation of the common Enterprise Block Ports/Interfaces is shown in Figure 6.27. Since all service interfaces must be routed through the security layer/System Container, the interface is shown as connected internally to a typical security layer. Since all the Ports in the Enterprises are anticipated to connect to the Internet for communications services, service contracts should also be expected at each Port, but since all the Ports could re-use the same common device architecture, the service contract could be very similar or identical. At least for the Enterprise level of organization shown in Figure 6.26, it is likely that service contracts could be composed into a single service contract for that level. Separate service contracts might be necessary at lower levels of organization for user access, such as Brokering Agents and Peer Ledgers, if they are exposed to uniquely different user requirements.

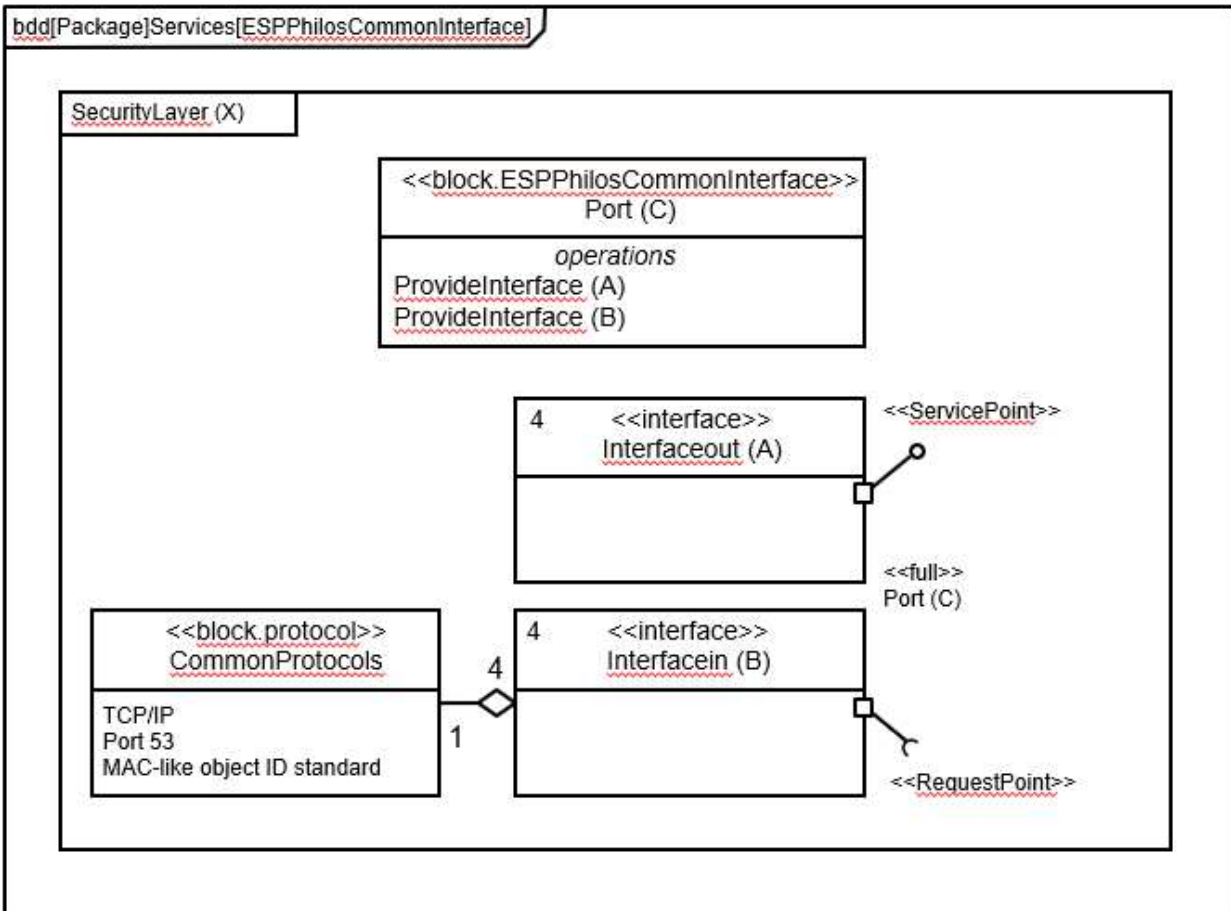


Figure 6.27. Block Diagram of a Common Enterprise Port.

The block definition diagrams shown in Figure 6.24 and Figure 6.25 both show the composition of multiple complex services from (relatively) simpler ones. In these cases, the composition of two separate Enterprises that could act independently demonstrates a vastly larger number of emergent capabilities than either one acting without the other. In fact, the whole design concept of the synergy of the two Enterprises is based on the composition of the two. The de-composition of the Industry Utility Registrar™ and the Neutral Governance Blocks is designed to separate those two elements from the Enterprises to avoid conflicts of interest that might occur if they were internal to the two Enterprises and directly managed by the same organizations.

One example of the flow of a Service execution; is shown in Figure 6.28, a Behavioral Sequence Diagram illustrating the exchange of messages (in the form of encrypted “packets” containing “sidechains”) within the Philos™ Enterprise. These processes operate under specified time constraints (xxx minutes for synchronization of each sidechain), and specified consensus goals (yyy percent of synchronization of all contributing sidechains into a

blockchain consensus block within 24 hours. At the Philos™ Enterprise level, the key Service execution is the transmission of Consortium Ledgers to the :NeutralGovernance Domain, which determines the validity of each sidechain, synchronizes their inputs, builds the next approved block on the Philos™ blockchain, and then retransmits that message (again in the form of encrypted packets containing the new version of the blockchain), back to the Peer Ledgers in the Philos™ Enterprise for distributed storage throughout the distributed ledger system. In this case, the arrival of the multiple sidechain packets at the: NeutralGovernance Domain should initiate the validation and synchronization processes without separate control messages.

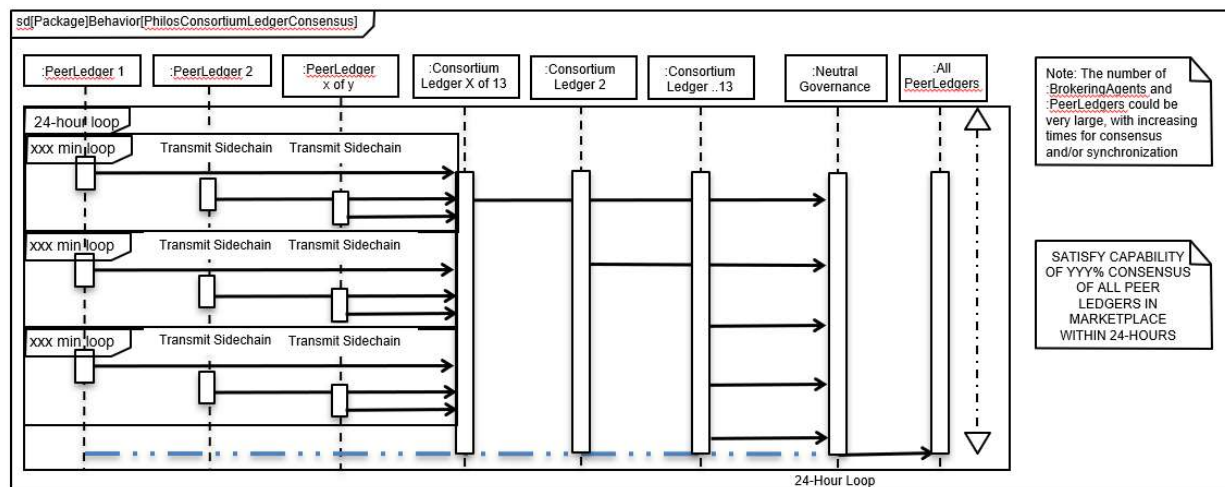


Figure 6.28. Sequence Diagram of Philos™ BDL Consortium Consensus Interactions

Other Perspectives

Real Time

The same Sequence Diagram in Figure 6.28 used to explain Philos™ BDL Consortium Consensus Interactions can also be used to demonstrate time constraints that must be met. The Figure 6.28 shows a Sequence Diagram illustrates the exchange of messages (in the form of encrypted “packets” containing “sidechains”) within the Philos™ BDL Consortium Consensus to work correctly, but the timing requirements are important too. The sequence steps are accomplished under specified time requirements – xxx minutes for synchronization of each sidechain), and specified consensus goals (yyy percent of synchronization of all contributing sidechains into a blockchain consensus block within 24 hours.¹⁰⁵ Although not exactly “real time” criteria, they are so important that “real time” timing analysis and techniques could be used to make the system more efficient, identify needed upgrades, identify inefficient coding methods, identify time-wasting User processes, and hopefully correcting all of those problems collaboratively. As discussed earlier under the Contextual Perspective of the Logical/Functional Viewpoint, almost all current blockchains have timing, cost, longevity, size growth, and auditing-time problems that may eventually cause their demise. In most cases, fixing many of those problems seems to be ignored or deferred. Better timing analysis of any or all of them – and/or tracking the problems with performance metrics could increase their overall health; OR they can start to develop new concepts from scratch, which is exactly the goal of Philos™. Once in full-scale test and evaluation, those metrics should be collected, and a series of timing analyses initiated to stress Philos™ timing capabilities, compare them to more pedestrian systems, and apply more research to make them as efficient as possible.

The MARTE Profile of UML has been specialized to deal with real-time systems and analysis, including testing and analysis of parameters to prove system schedulability. Figure 6.29 is a Sequence Diagram to describe one of the most vital processes for the Eco-Secure Provisioning™ (ESP™) Enterprise. The Figure shows a notional sequence to request a secret provisioning package from a secure source to establish and open a cryptologically-secure gateway/tunnel for secret information. Due to corporate security data, this is an extremely simplified and fictitious representation but will be shown in a Sequence Diagram format to characterize how MARTE annotations might be used. The example shows how a crypto-provisioning source could connect to a target destination device (at the

¹⁰⁵ Those specific performance values will not be specified here. Those requirements could be considered as “soft” time requirements because if they are not met, the information from participants may be rolled over into the next time blocks, but they are important for efficiency because if they start getting “backed up,” it could affect the health of the Enterprise, and cause adverse conflicts with Users and Stakeholders.

request of an engineer physically initiating the process from the target device), pass cryptological keys, initiate a secure synchronous session, pass an executable security package, then execute that package to establish a secure tunnel for further encrypted synchronous communications. [In the real Eco-Secure Provisioning™ (ESP™) Enterprise, the target device is called a “Crypto Trust Center™” (CTC™).] The scenario shown is based on a controlled test experiment to determine a maximum timeout requirement until the process will fail-safe and stop the process. The test includes a notional hostile malware insert, which according to previous “intelligence” and experience, is assumed to be inserted through a robotic man-in-the-middle attack. It is assumed from previous testing that it takes between 30 to 60 seconds for the malware to insert itself into the communications stream. It is also assumed that test engineers have an actual malware package as part of their penetration testing kit and would be conducting this test in a controlled “sandbox” environment. Using the sequence diagram with the MARTE annotation, engineers should expect a normal provisioning process to take between 45 seconds to 60.006 seconds. If the man-in-the-middle attack were successful, an additional 30 to 60 seconds of relative time would be added before the process completes, or a total time of 75 to 120.006 seconds, but during that time, the malware would be successfully inserted. Therefore, for this test, engineers should expect provisioning times of less than 60.006 seconds to be normal; but if provisioning times exceed 75 seconds, engineers should assume that the provisioning was compromised by a malware attack. A good hard criterion for process scheduling would therefore be about 61.0 seconds. Any runtimes that exceed that should terminate, exit, and fail safe. Depending on the circumstances, test engineers could manually restart the process again, but automatic “looping” should not take place until the cause and origin of the malware insert is determined and eliminated. With sufficient verification, the 61 second maximum criterion could then be used within the physical operational system, with a reporting mechanism if the 61 second criterion is exceeded. Part of the rational for pairing the ESP™ Enterprise with the Philos™ Enterprise is to immutably log the facts surrounding that sort of transaction on the Philos™ BDL and go back and find the non-repudiable information about the hostile actor that caused the hack.

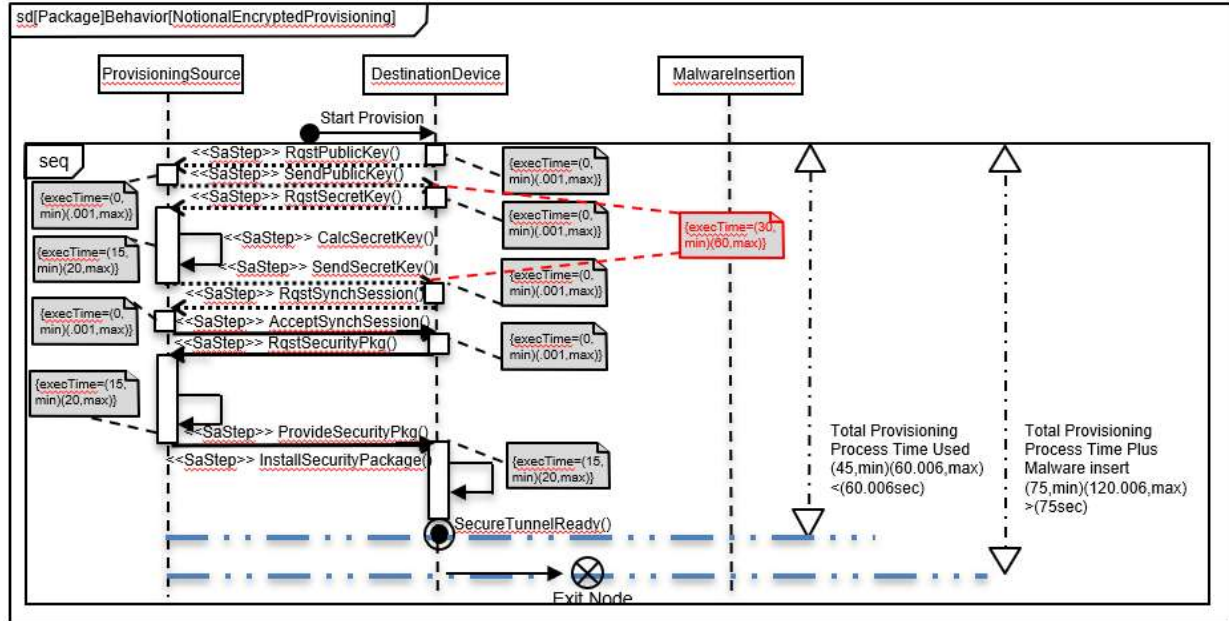


Figure 6.29. Notional Encryption Provisioning

Another time-related behavior of interest that could affect the ESP™ Crypto Trust Center™ (CTC™) is the state of the logic used as LIC chips are being provisioned. Figure 6.30 shows the behaviors of two segments of a CTC™ provisioning process using State Machines to display the states of the process.¹⁰⁶ The first segment of the process begins with a manufacturer's chip in the manufacturer's trusted configuration, as specified by contract, but devoid of any other vendor or operational data. A unique identifier is obtained to identify the chip from the Industry Utility Registrar™ (IUR™) and is recorded on the BlockChain Development Community Philos™ Block Chain. Philos™ permanently, immutably, and non-repudiably records those transactions. The State Machine then initiates the provision of Base configuration software. That action then generates the next segment, the provision the Vendor's IUR™ registration. Philos™ also records that transaction, and the initiates of provisioning the Vendor's unique configuration software. Once that is complete, that transaction is again recorded that on the Philos™ block chain. Following those actions, similar Block behaviors provision other Network-specific and Operational-specific software, and record each of those completed transactions following similar patterns.¹⁰⁷ Note: those sequential steps are not shown here but would be in the actual system architecture.

¹⁰⁶ The numbers used in the state names are also used to represent more detailed steps defined elsewhere in the source design documents. They are retained here for compatibility with those source definitions, and to add clarity for future research.

¹⁰⁷ The data in Table 8.1 of (Borky. 2019. P. 308) is a specific example of schedulability analysis on a real-time system. While the ESP™ and Philos™ Enterprises would probably not follow the behavioral patterns of trying to deconflict competing priorities of parallel processing, it is possible that the Consortium Consensus calculations of Neutral Governance processing of blockchain blocks might have to when a very large number of Side Chains must be synchronized. Therefore, this type of analysis could be useful in the future when such processing units are

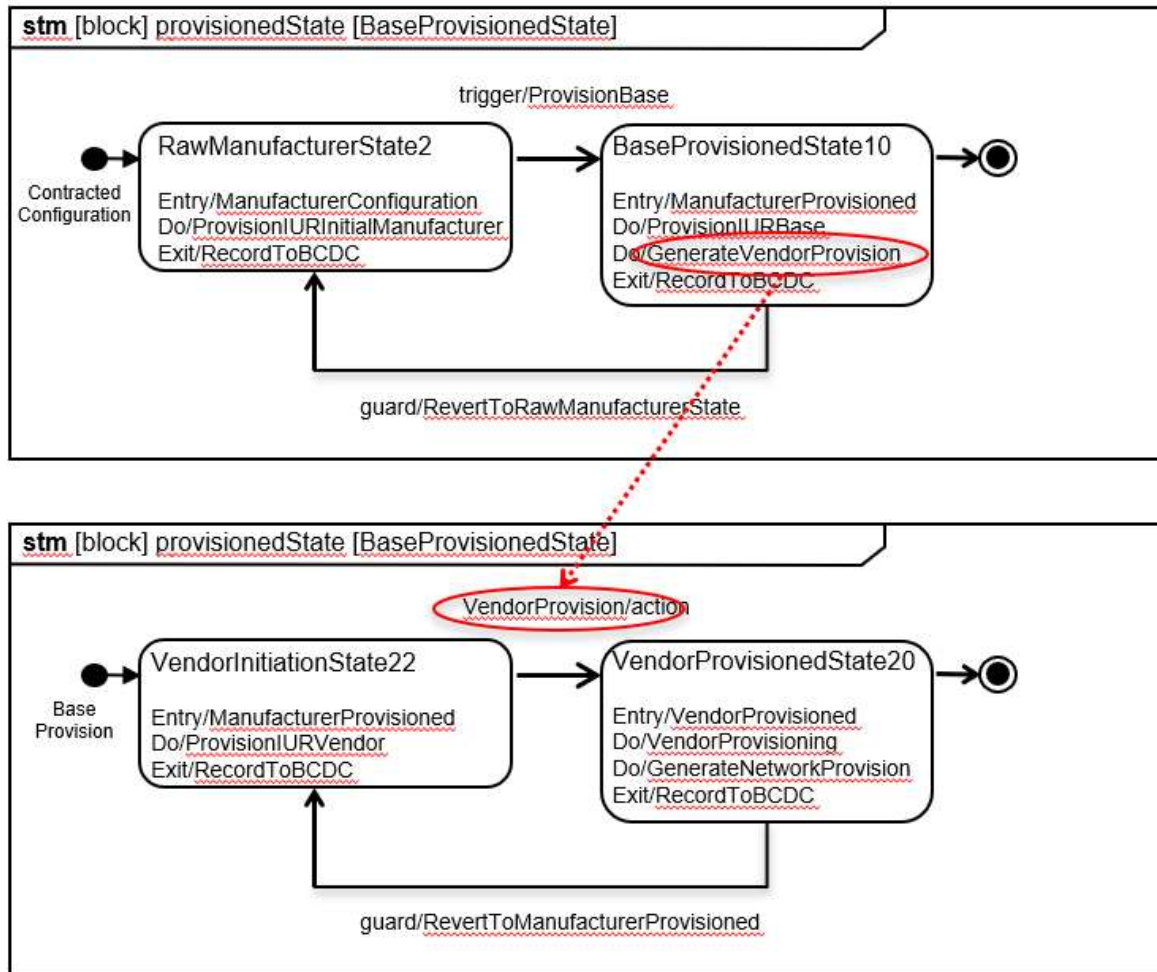


Figure 6.30. State Machine of Crypto Trust Center (CTC) Provisioning (For Two Segments)

Table 8-1 from Borke, 2019, P. 308¹⁰⁸, is reproduced below, with modifications for twice the second data set in the Table, which reduces execution times by half. All other values remain constant. Using that data, Figure 6.32 shows what happens if the throughput of the computing suite is only doubled instead of quadrupled.

being designed or selected in the future. For now, we will only look at the academic requirement, but a similar analysis could also be useful for future analysis for Neutral Governance architecture. Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

¹⁰⁸ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>. P.308.

Task	Execution Time, E μ S	Period, T μ S	Deadline, μ S	Utilization	Importance, C	Priority, P
1	0.1	0.2	0.18	0.5	10	50
2	0.15	0.5	0.45	0.3	8	16
3	0.2	1	1	0.2	5	5
4	0.3	2	1.8	0.15	2	1
5	0.5	3	2.5	0.17	1	0.3
Table 8.1 Rate Monotonic Scheduling with 2x Processing Power						
Task	Execution Time, E μ S	Period, T μ S	Deadline, μ S	Utilization	Importance, C	Priority, P
1	0.05	0.2	0.18	0.5	10	50
2	0.075	0.5	0.45	0.3	8	16
3	0.1	1	1	0.2	5	5
4	0.15	2	1.8	0.15	2	1
5	0.25	3	2.5	0.17	1	0.3

Figure 6.31. Table 8-1 from (Borke. 2019. P. 308), and Data with 2x Processing Power

Figure 6.32 shows the suite priority sharing with twice the throughput, and half the execution times. Since Task 1 always has the highest priority, it will always preempt Task 2 and lower priority tasks; Task 2 will always preempt Task 3 and lower priority tasks; Task 3 will always preempt Task 4 and lower priority tasks; and Task 4 will always preempt Task 5 and lower priority tasks. Due to the higher processing speeds (x2), Task 1 now executes in a duration of 0.05 μ S, but will still restart in its regular period of every 0.2 μ S. Other Tasks' durations and periods are as listed in the modified data set in Figure 6.31. Since the highest priorities start on the bottom, I mapped all events for Task 1 first, followed by the others. Whenever a higher priority preempts a task in progress, I allowed that task to restart after the end of the preemption period, and to run to completion. Working from the highest to the lowest priority tasks, I followed the same pattern until all the executions were complete, or until their periods were so long that they couldn't restart by the expiration of the 2 second examples as shown in Figures 8.5 and 8.6 in (Borke. 2019. pp. 309-310, *as revised*). The operational consequences are that even with only twice the computational speed, all five tasks are able to run, compared to only three tasks able to run in the baseline case. Also, all tasks are able to run to completion within the 2-second example window, but some preemptions still take place that can cause some delays. It also shows that the 4x processing speeds shown in Figure 8.6 of (Borke. 2019. P. 310, *as revised*) are not necessary to service all the tasks, although the 4x case has fewer preemptions and higher performance margins.

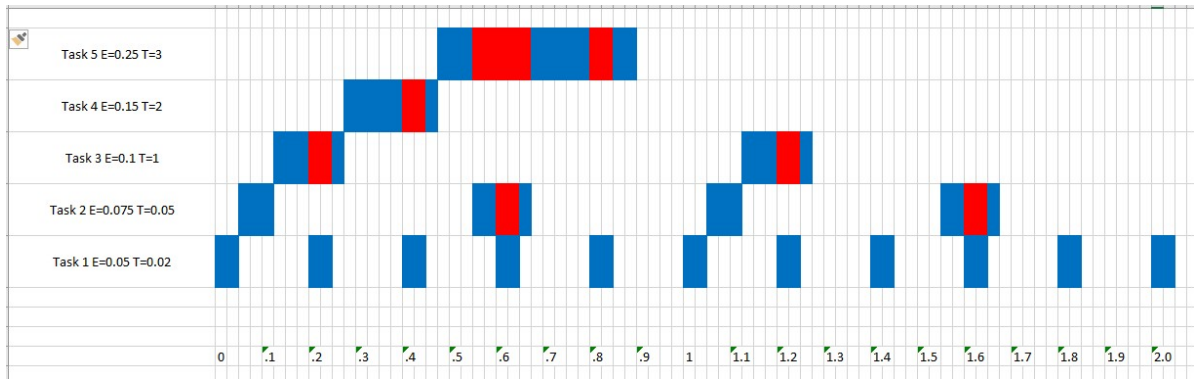


Figure 6.32. Double Throughput of the Computing Suite with Half the Execution Times

Enterprise Diagrams¹⁰⁹

Figure 6.33 shows a BDD to depict the Eco-Secure Provisioning™ (ESP™) Enterprise and the Philos™ Block Chain Enterprise, with their respective domains, and supporting subsystems. Previous BDDs and IBDs were also used to depict various viewpoints, relationships, services, and connections among the two Project Enterprises.

¹⁰⁹ Because the “system” architecture used throughout this project has intrinsically already covered the topic of a merger of two Enterprises – supported by two other systems – Block Definition Diagrams have already been presented. Some of those will be “re-used” here, with some upgrades, modifications, or further explanations to address these new academic questions.

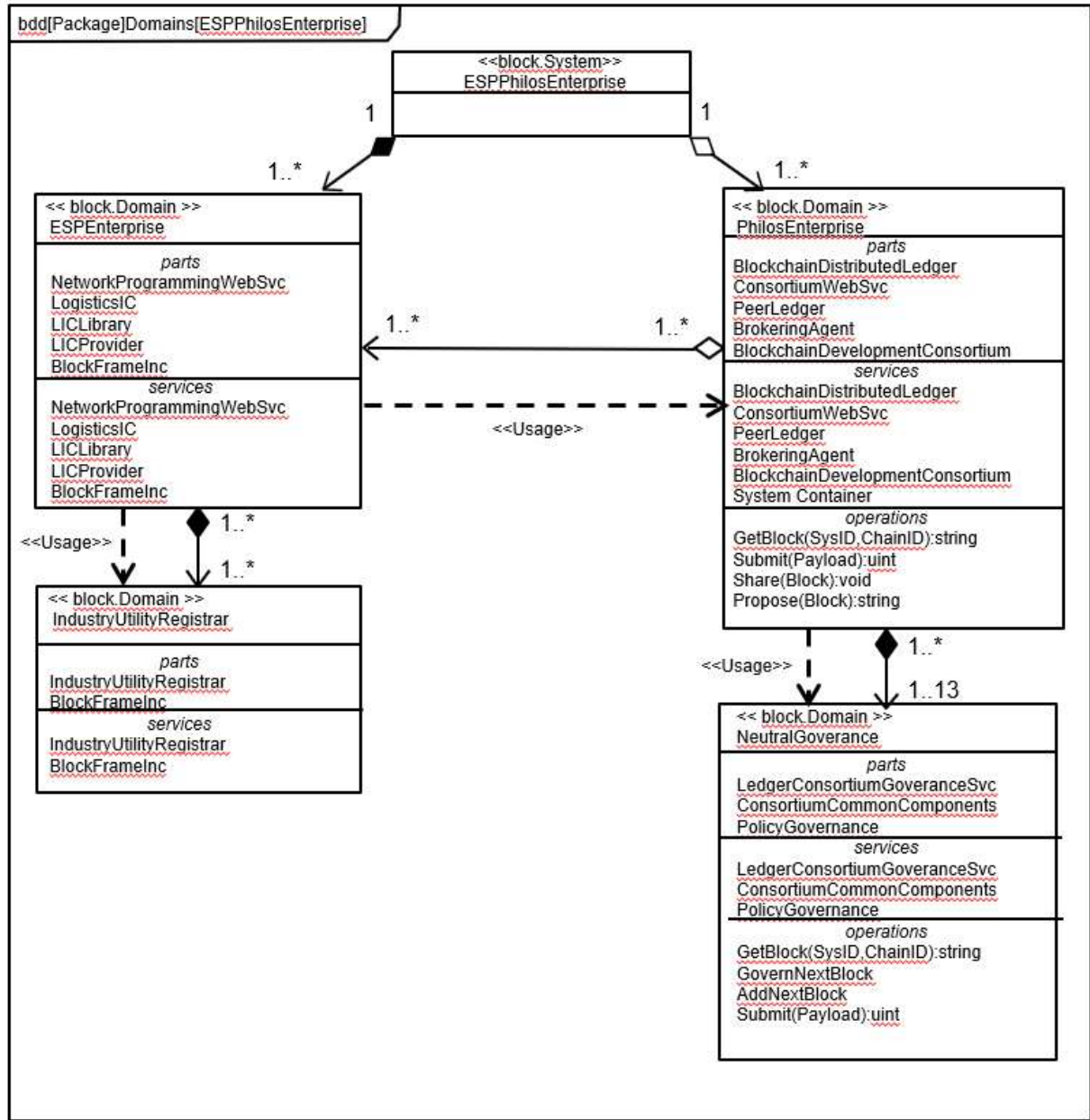


Figure 6.33. Block Definition Diagram of ESP™ and Philos™ Enterprises

Figure 6.34 is a modification of one of the previous BDDs and is used here as an IBD to better emphasize the roles of participants, the logical/functional connections, and the respective services they provide at the highest levels. Ideally, all the connections among the enterprises and supporting systems are considered to be loosely coupled modular interfaces to provide the specific services that are shown. The service orientation of Figure 6.34 should therefore be appropriate to best show the basic exposed service-related structures and the basic exposed interfaces. However, for this diagram, one should notice that all exposed services are routed only through security layers, while

all other services are behind (or actually physically inside) the security layers. Therefore, each IBD Block intends to show the basic structures, the exposed security layer interfaces, the major participant organizations, and the connections to services “behind the curtains” of each Block, all within the same Viewpoint.

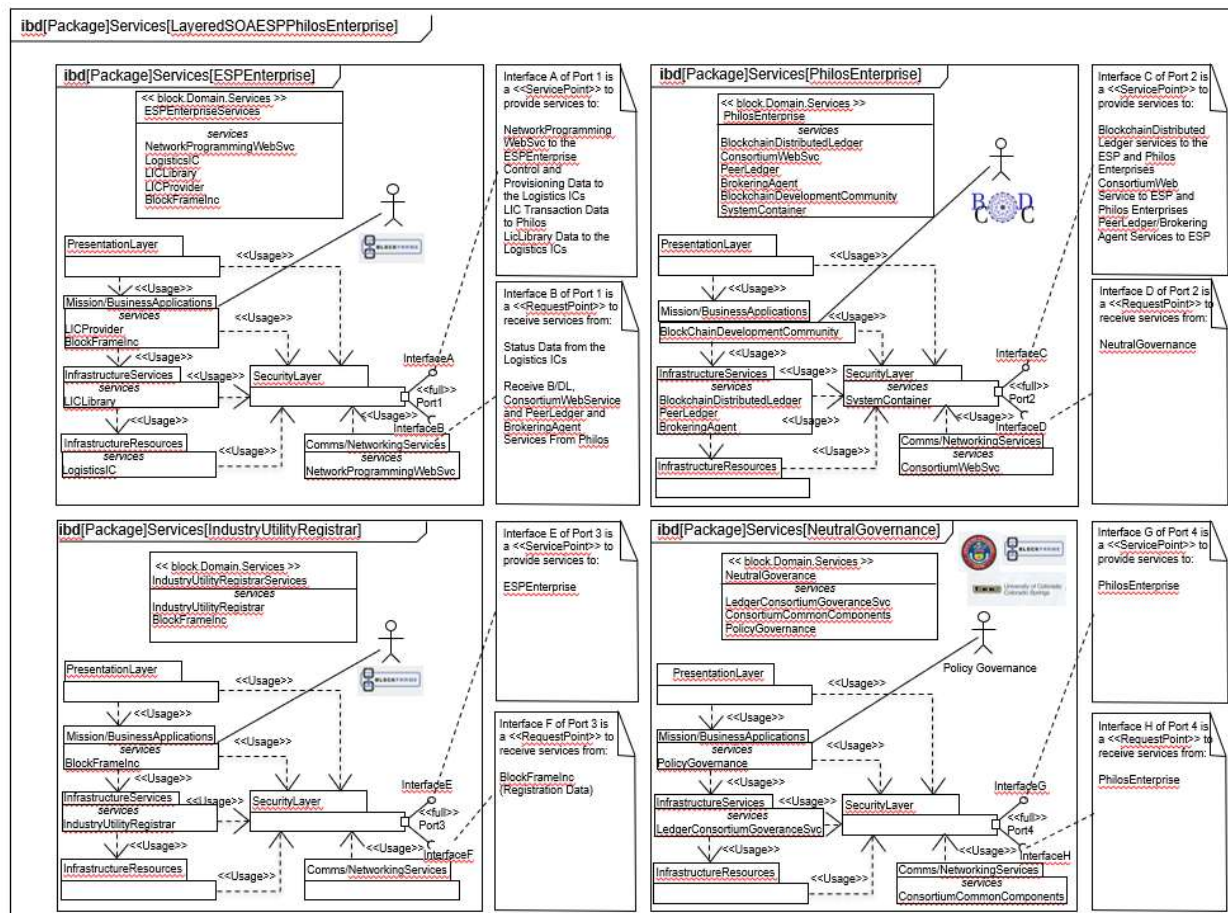


Figure 6.34. Internal Block Diagram of the ESP™ and Philos™ Enterprises

Figure 6.35 is an Activity Diagram to show activities taken to provision a Blockframe Integrated Logistics Circuit (ILC) within the ESP™ and Philos™ Enterprises.¹¹⁰ Initially, the ILC arrives in the manufacturer’s trusted configuration without user/vender-specific software or codes. The ILC engineer requests a trusted Industry Utility Registrar™ (IUR™) identification, which is then generated by the IUR™. The Engineer then does the initial provision of the base software and the IUR™ identification into the ILC. That transaction is then cryptographically stored on the BCDC Blockchain as an auditable, immutable, and non-repudiable record.

¹¹⁰ Some specific details have been omitted here due to security and proprietary concerns. Specifically, Figure 6.35 views the initial steps of the process, but not all of the remaining steps.

The next phase of activity starts when the base-provisioned ILC is passed to the user/vendor. The user/vendor will repeat basically the same activity sequence to add their own provisioning software and IUR™ identification to the ILC, thereby impressing their own additional security standards and ownership of the ILC. Those transactions are again documented and cryptographically stored on the block chain.

Depending on the purposes or use cases of the user/vendor-provisioned ILC, a similar process could be repeated for other security purposes – with each transaction being securely and immutably stored on the blockchain. The advantages of documenting all provisioning transactions include proof-of-ownership, traceable Proof-of-Origin of the data and the devices, time-dated proof of each transaction (for non-repudiation), and other security capabilities.

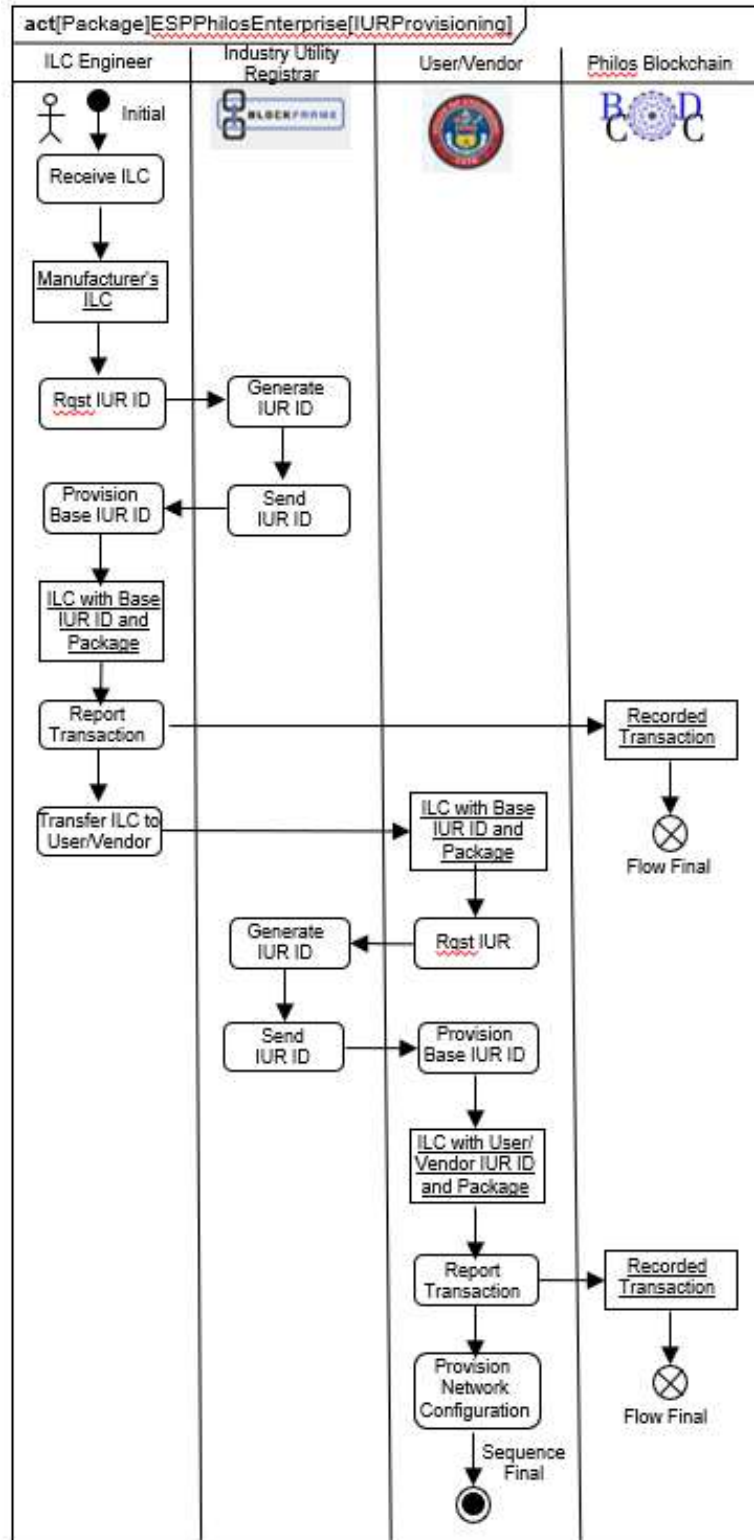


Figure 6.35. Activity Diagram of a Partial ILC Provisioning Process

Figure 6.36 re-uses Figure 6.29 to demonstrate a Sequence Diagram to show message exchanges among systems. In Figure 6.36, MARTE timing annotations show a time-stepped pattern of the provisioning of cryptological

data from a Provisioning Source to a Destination Device. The first several sequences use <<SaStep>> stereotypes to describe timing characteristics of messages, specifically in this case the use of messages to establish asymmetric and symmetric cryptologically-secure communications. Messaging is essential to this process since both the Provisioning Source and the Destination Device must communicate positively and mutually agree to initiate and conduct the communication.

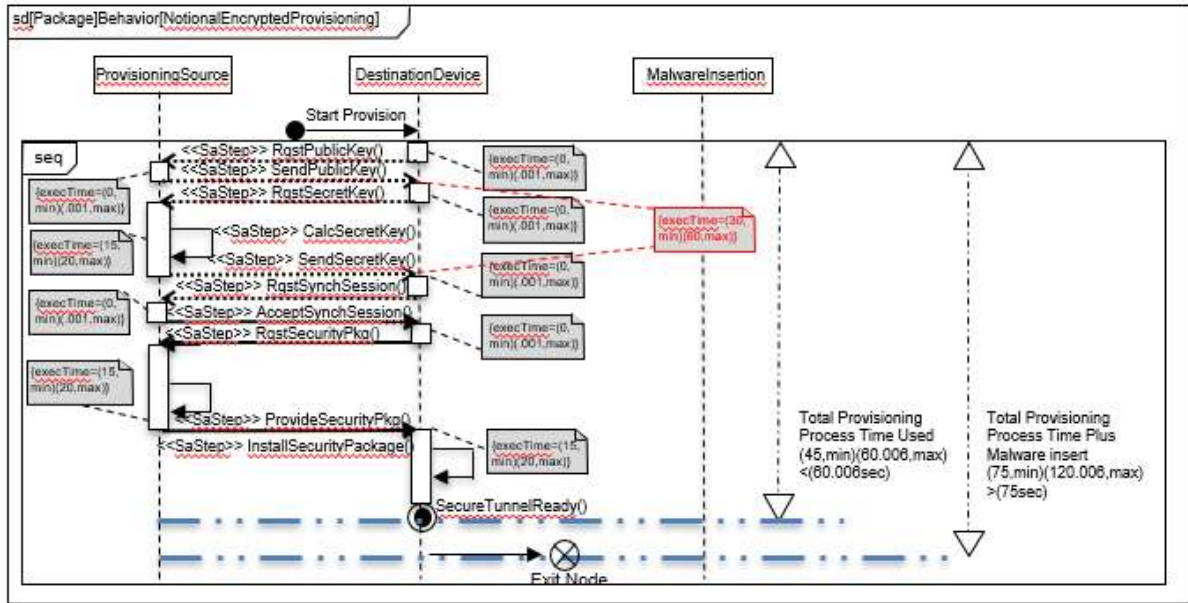


Figure 6.36. Sequence Diagram of Notional Encryption Provisioning

Figure 6.37 is a Sequence Diagram that demonstrates timing data and “enterprise behavior” of multiple systems, in this case, a Philos™ Consortium Consensus Interaction. Specific timing data capabilities are obscured here due to sensitivity, but “xxx” represents specific timing criteria, while “yyy” represents specific performance criteria of a percentage goal to be met within a 24-hour time period. The diagram is reused from my previous Figure 6.23 and Figure 6.28. As shown by previous examples, many system-level blockchain transactions could be recorded and stored on the Philos™ blockchain to document proof of those transactions. In the same way, operational, financial, or manufacturing transaction data could be collected in very large numbers from a very large number of geographically separated systems. Figure 6.37 represents the process of collecting large amounts of data through: PeerLedger’s, and managing those data through the Philos™ blockchain. The data are collected from: BrokeringAgents (not shown in this diagram), through a large multiplicity of :PeerLedgers where data are combined, checked for errors and security trust, and subsequently merged into Sidechains, and sent to :ConsortiumLedger’s. A

similar process combines, checks and merges: ConsortiumLedger-level chains into a Global blockchain, once every 24 hours. Each `:BrokeringAgent`, `:PeerLedger`, and `:ConsortiumLedger` may be considered “systems” in their own right, and each is encapsulated within their own security layers, called “System Containers,” which may be considered the “common devices” or “common nodes” of the overall Philos™ Enterprise or System-of-Systems, as well as connecting the Enterprise to all its users and their applications.

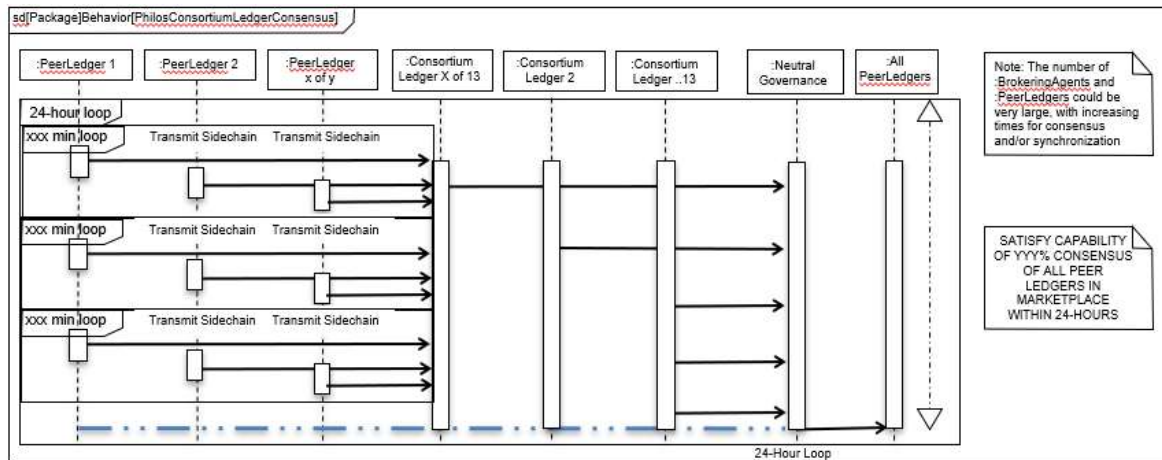


Figure 6.37. Sequence Diagram of the Philos™ Consortium Consensus Interaction

Since the Philos™ Blockchain/Distributed Ledger is inherently designed as an Enterprise, the Conceptual Data Model of each individual Brokering Agent and Peer Ledger comprise a microcosm of the basic data model for the entire Enterprise. Since each Brokering Agent and Peer Ledger is encapsulated (either physically or virtually), within its own security-level Systems Container, every common node of the Enterprise –with possibly a few rare exceptions – may actually comprise the data structure of every other common node and therefore share a standardized data commonality across the entire Enterprise. Figure 6.38 shows a BDD of the Logical Data Model (LDM) of the standard “common node” Brokering Agents and Peer Ledgers that comprise the Enterprise. Since the LDM was already assumed to be used as the design for the common components for the entire Enterprise Eco-System, we assume that it does not need to be “expanded” again here.

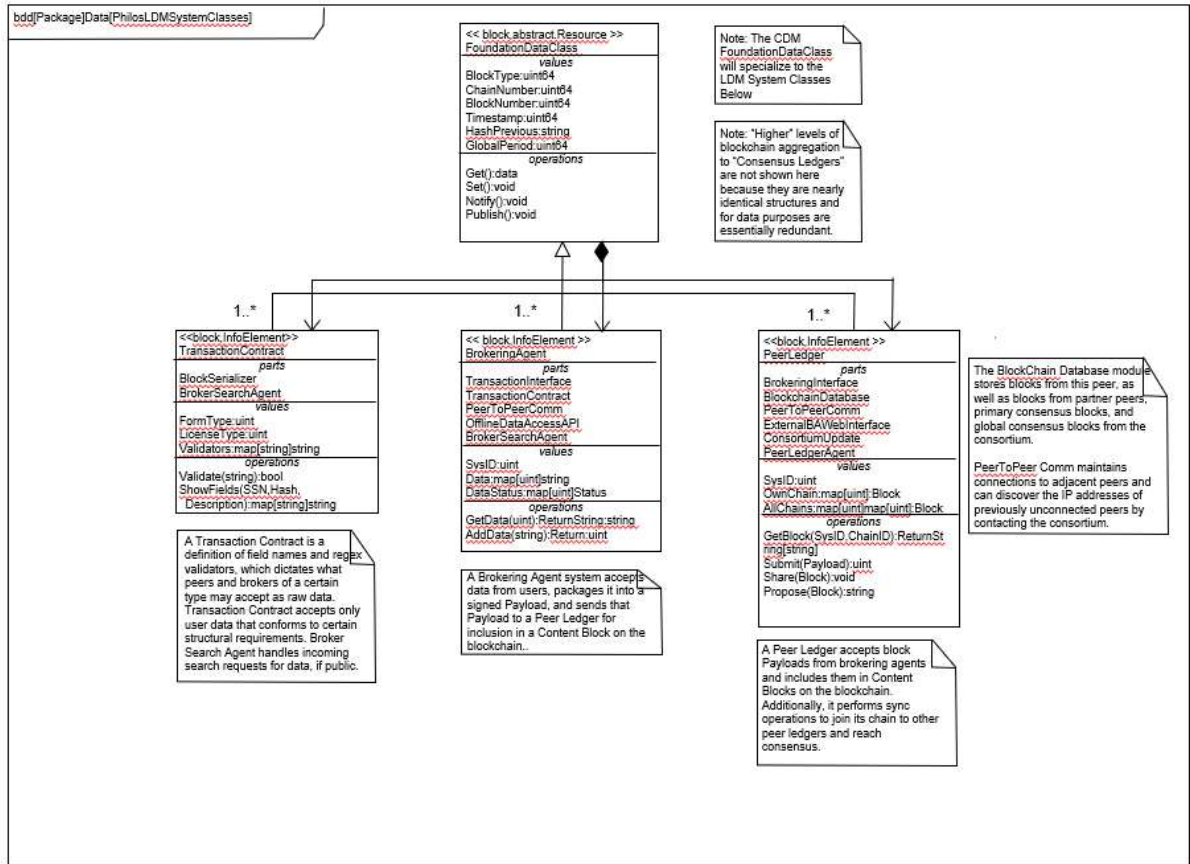


Figure 6.38. Philos™ Logical Data Model

Service-Oriented Architecture Diagrams

Service-Oriented Architecture Diagrams were included in Figure 6.6, Figure 6.7, Figure 6.15, and Figure 6.34, and discussed elsewhere.

Reference Architecture Diagrams

At their current states of development, the current ESP™ and Philos™ have a small percentage of Physical Viewpoint (PV) artifacts, and very few actual products have actually fully specified. Therefore, they are almost at a generic Reference Architecture-level of abstraction already. A few specific structures or behaviors can be made more generic to closely resemble Reference Architectures. However, both the ESP™ and Philos™ contain much more specific data at lower levels of engineering and code-level detail that have not been displayed in this project. Being relatively open architectures already, ESP™ and Philos™ might need to use protected terms, logos, or other specific designations to protect development and production trade secrets, which could be more difficult to protect than more closed proprietary products.

For current purposes, previous ESP™ and Philos™ diagrams will be modified to better resemble Reference Architectures and compared “side-by-side” with more generic “Reference Architectures” at the same levels of detail.

Figure 6.39a. is a simplified, representative Use Case Diagram for the combined ESP™/Philos™ Enterprises. Figure 6.39b. is a Reference Architecture Representation depicting a more generic “Logistics Enterprise” and a “Blockchain/Distributed Ledger Enterprise (BDL),” and more generic actors.

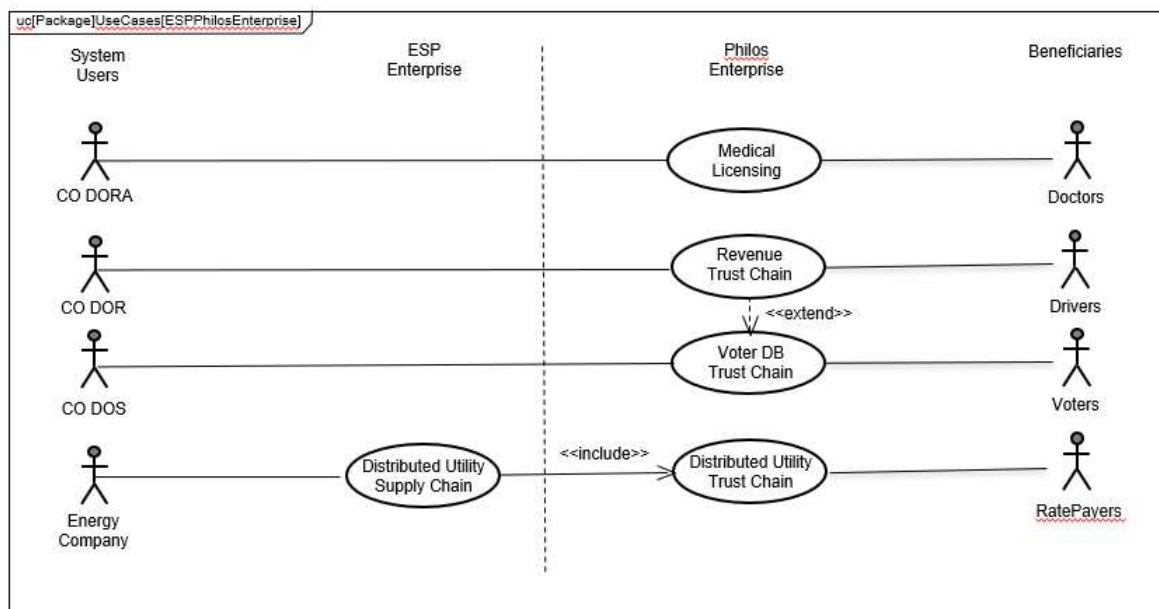


Figure 39a. ESP™/Philos™ Use Case Diagram.

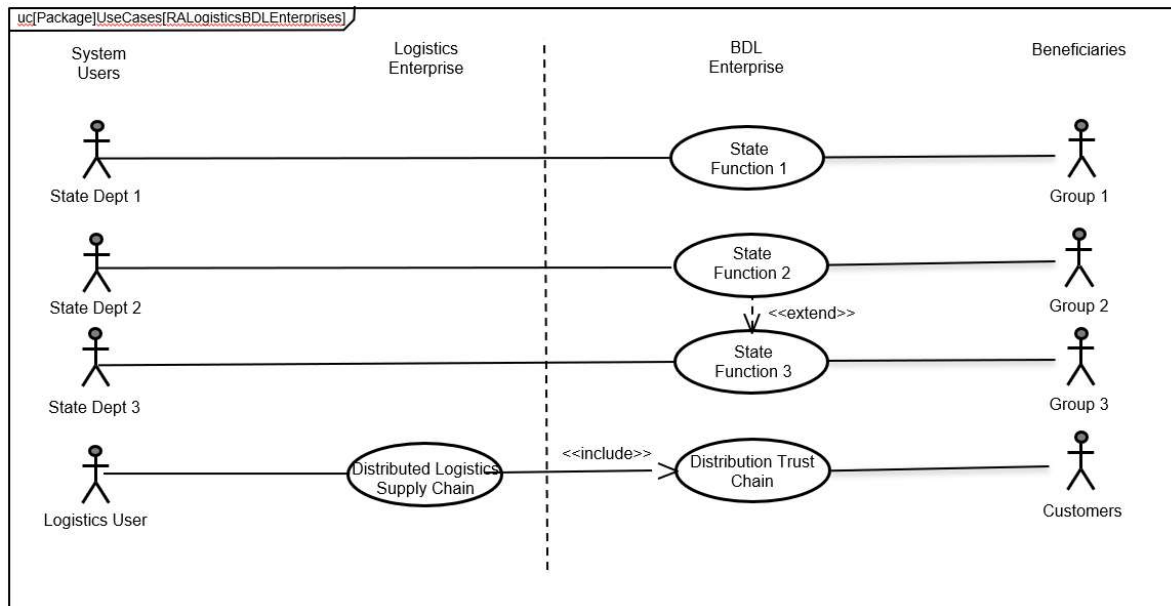


Figure 6.39b. Reference Architecture Logistics/Blockchain Use Case Diagram.

Figure 6.40a. shows an Activity Diagram of the ESP™ and Philos™ Enterprises providing business process trust services for a generic energy company. Figure 6.40b represents a more generic Reference Architecture.

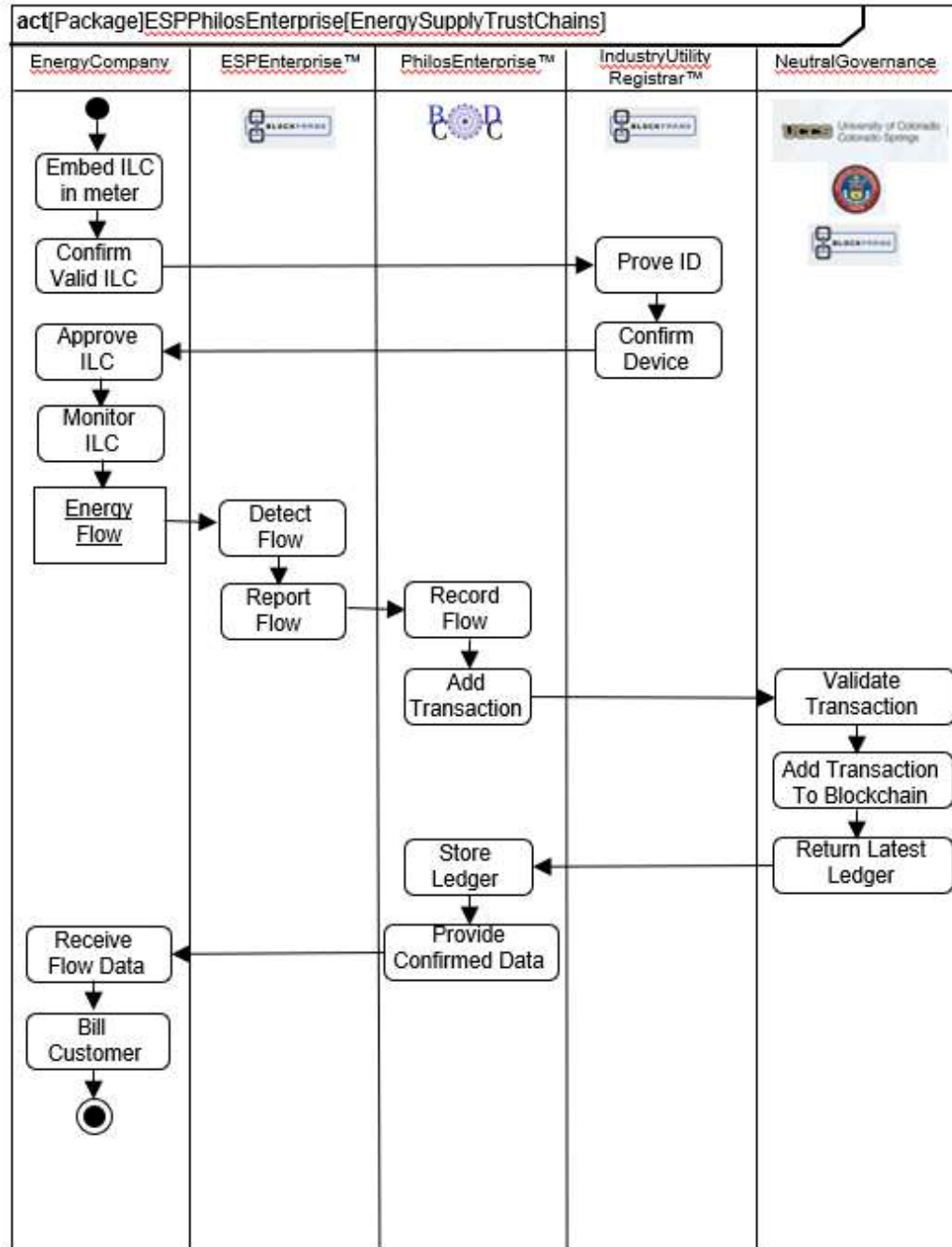


Figure 6.40a. Example ESP™/Philos™ Enterprise Activity for an Energy Company

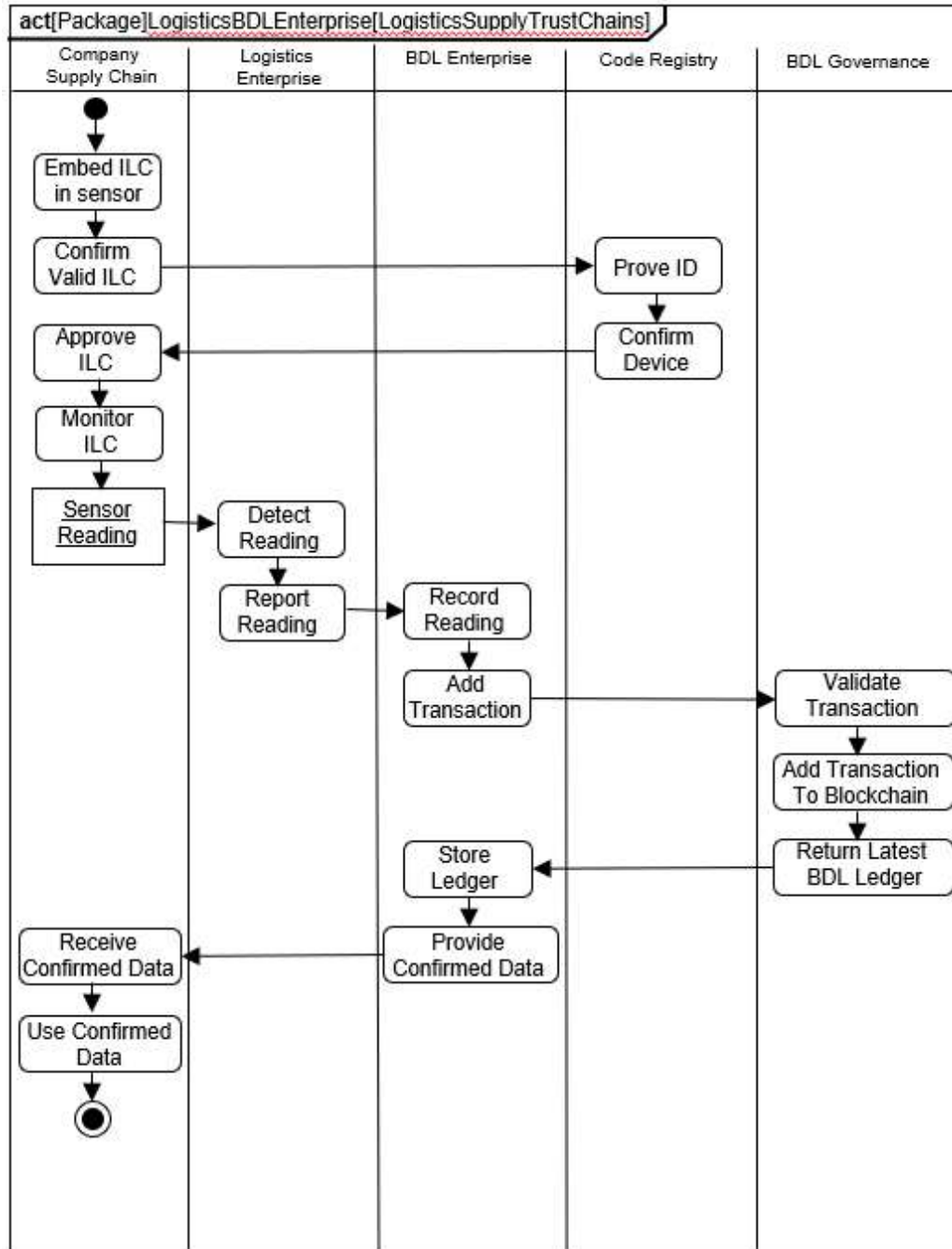


Figure 6.40b. Example Reference Architecture Activity for a Company using a Supply Chain

Figure 6.41a is a Domain Diagram characterizing the Eco-Secure Provisioning™ Enterprise (ESP™), the Industry Utility Registrar™ (IUR™), and the Philos™ Enterprise™. Figure 6.41b represents a Domain Diagram of a more generic Reference Architecture.

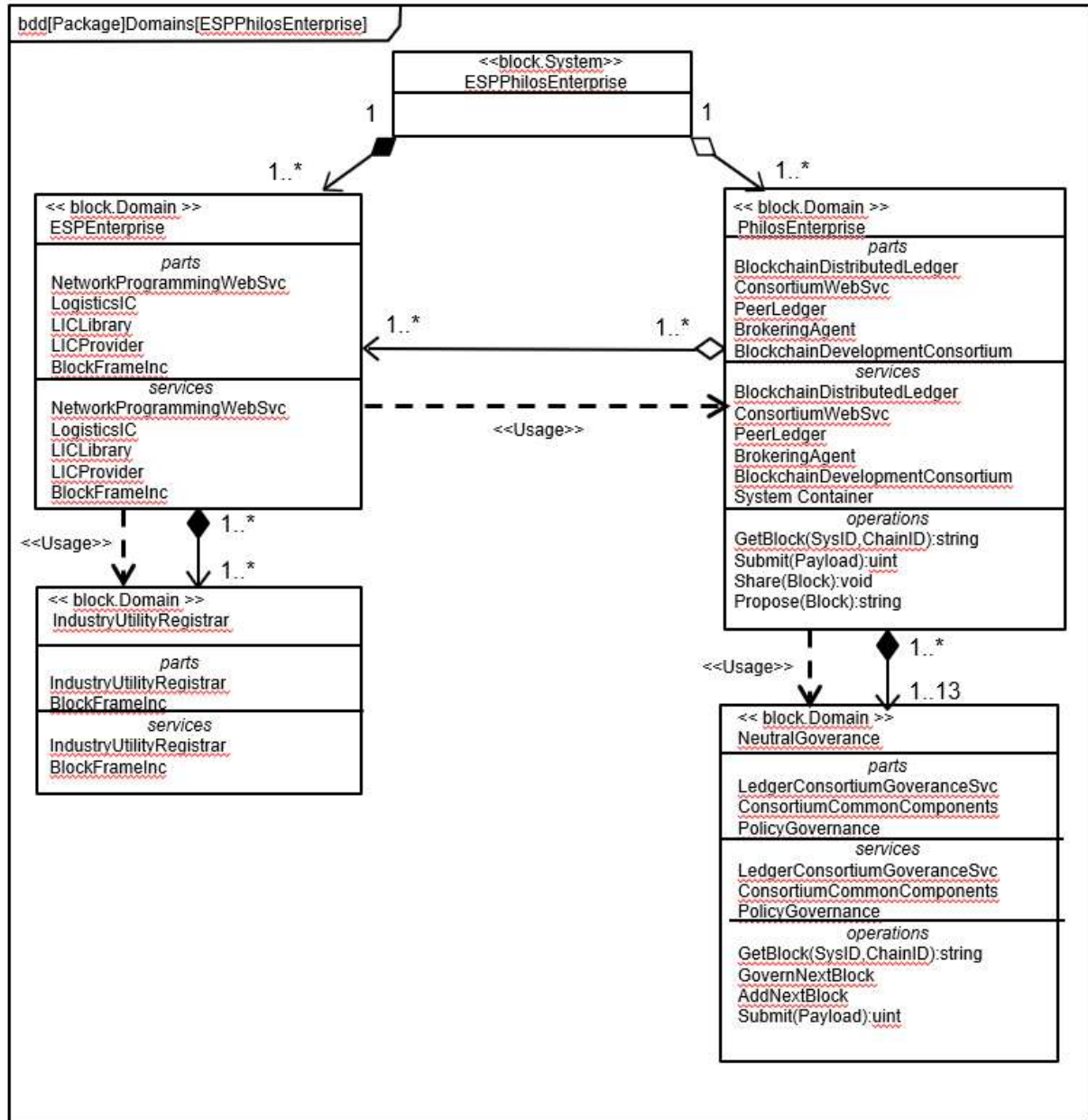


Figure 6.41 a. Example ESP™/Philos™ Enterprise Domains

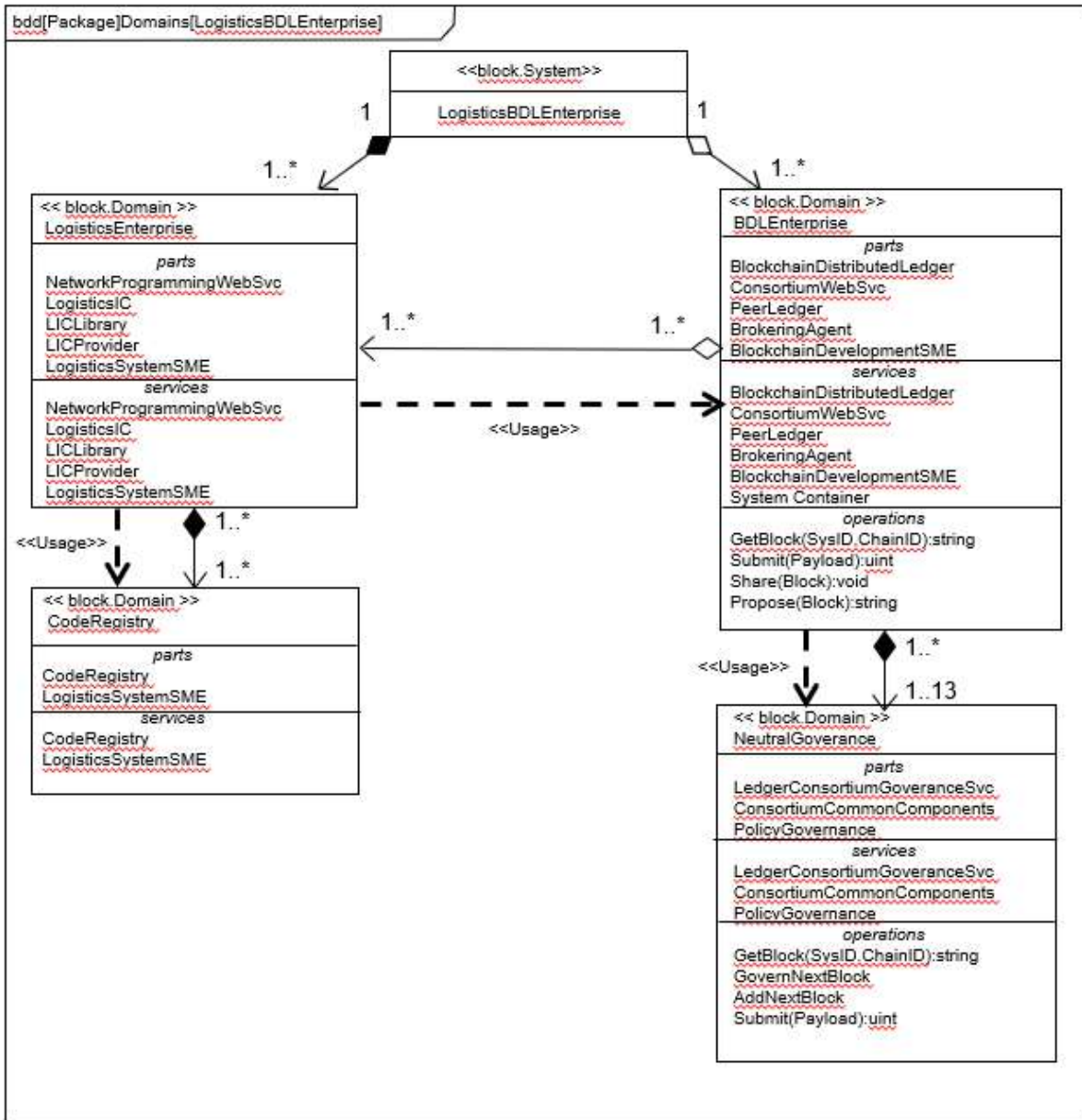


Figure 6.41b. Example Reference Architecture Logistics/BDL Enterprise Domains

Figure 6.42a is a decomposition of the Eco-Secure Provisioning™ Enterprise (ESP™) Neutral Governance Domain. Figure 6.42b depicts a more generic Reference Architecture for neutral governance. In these cases, the diagrams are very similar. Figure 6.42b removes specific corporate and institutional identifications, but without specific information about the Body of Knowledge needed to generate the parts, processes, and encryption standards, the generic Reference Architecture would be unlikely to work without the ESP™-specific Body of Knowledge.

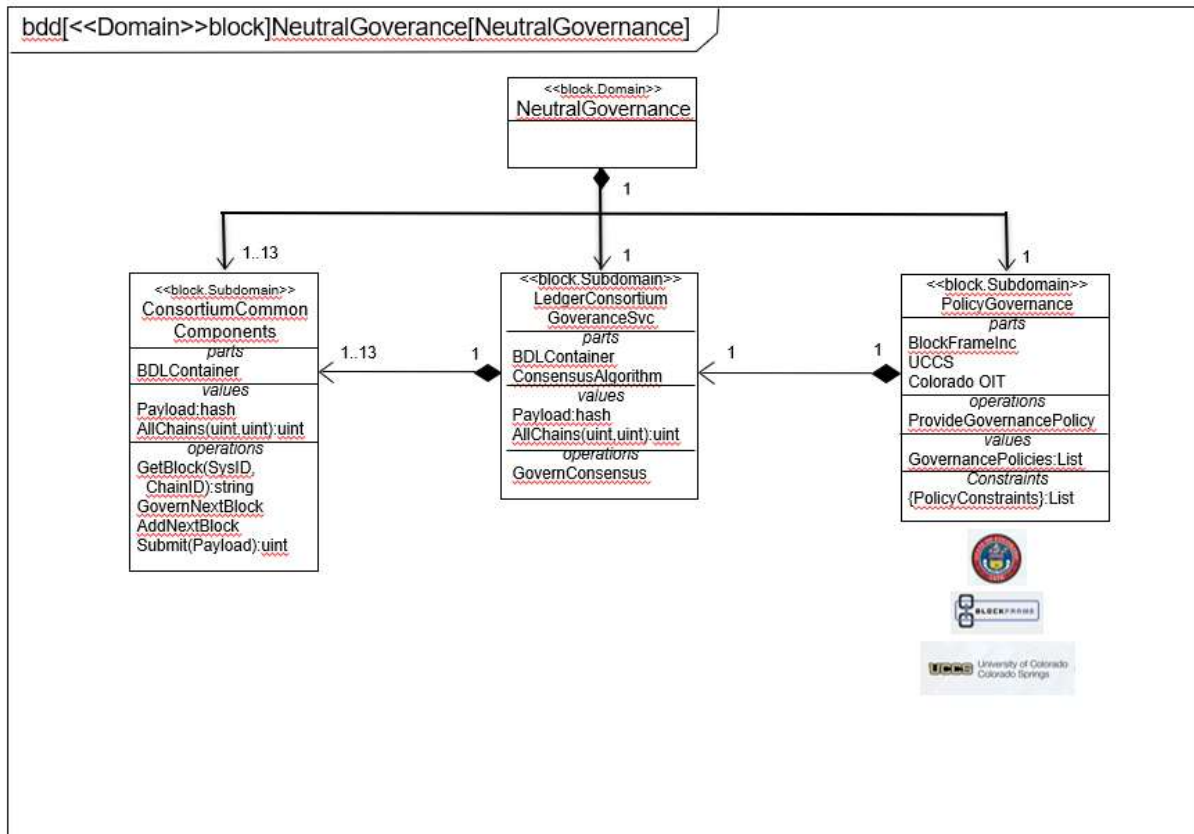


Figure 6.42a. Example Domain Decomposition for ESP™ Neutral Governance

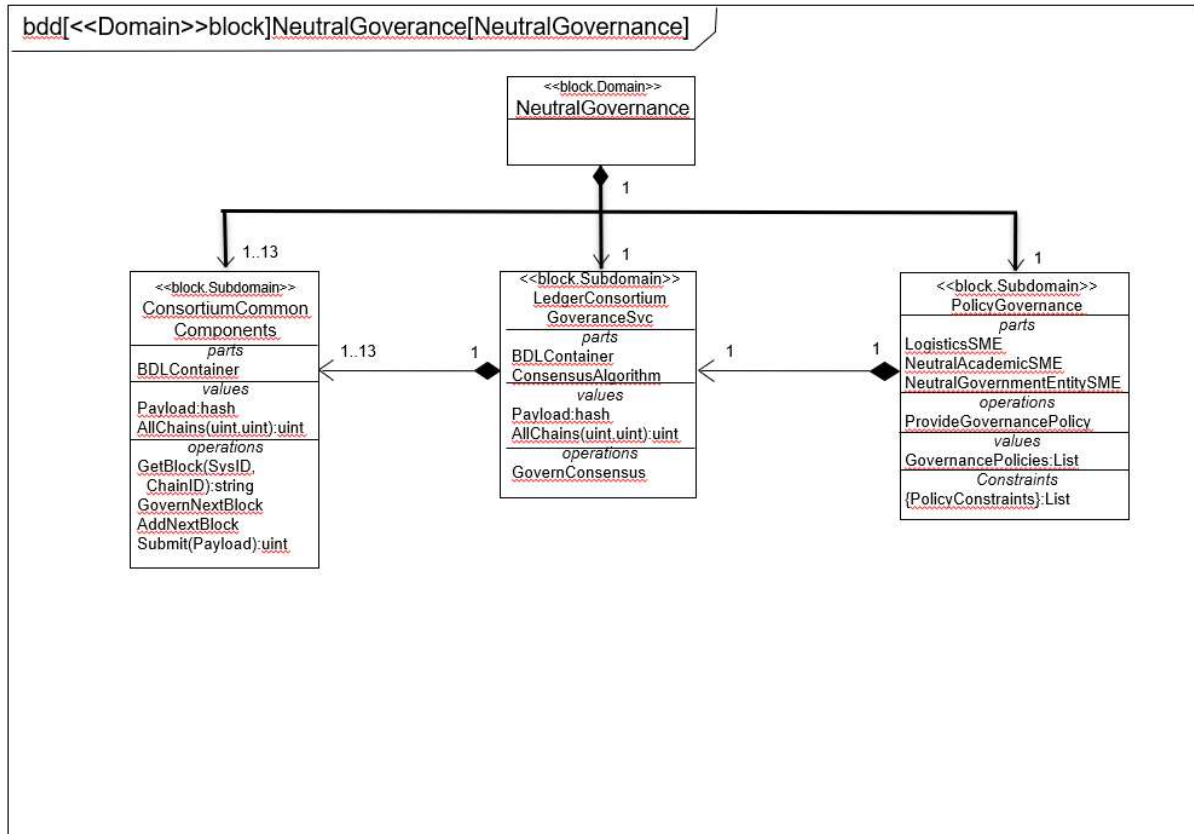


Figure 6.42b. Example Reference Architecture Generic Governance Domain

Figure 6.43a is an ESP™ State Machine Diagram describing the behavior of provisioning Logistics Integrated Circuits (ILCs). Figure 6.43b is essentially identical, except for not specifying the ESP™ as a product. As with the Neutral Governance example in Figures 6.42a and 6.42b, without the body of knowledge needed to generate the parts, processes, and encryption standards, the generic Reference Architecture would be unlikely to work without the ESP™ Body of Knowledge.

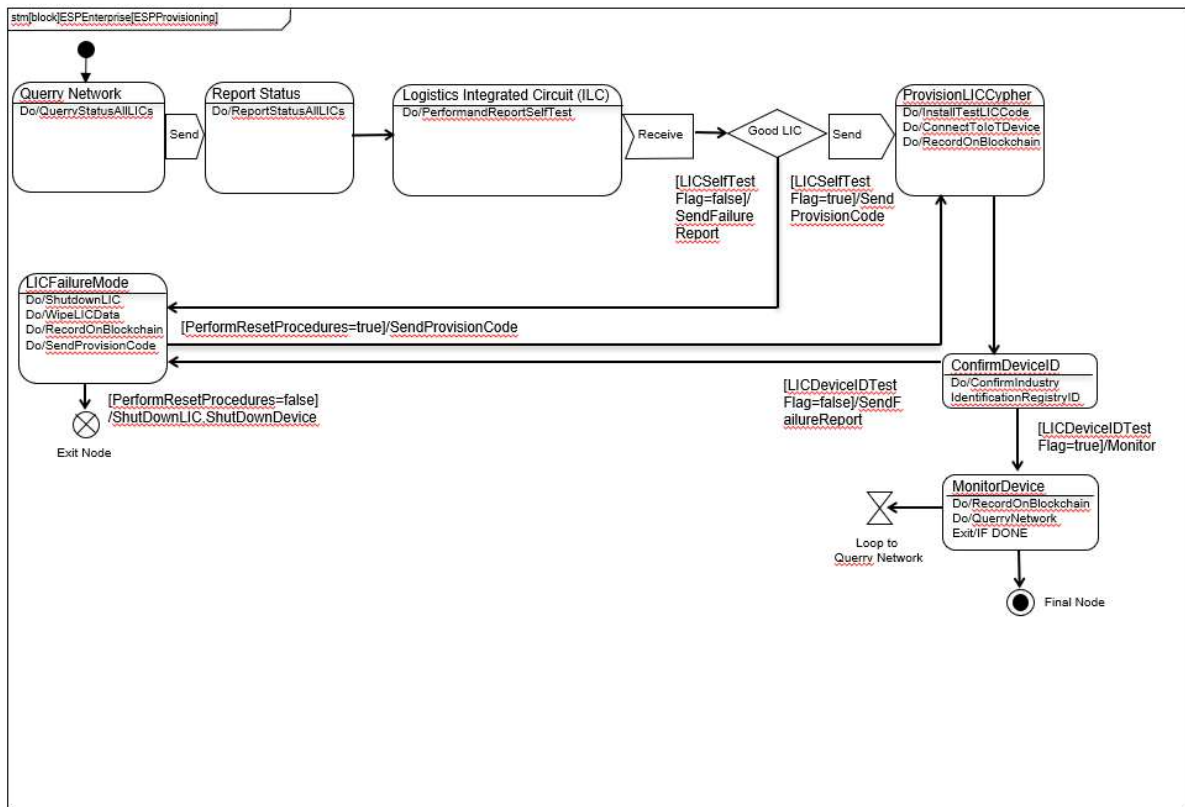


Figure 6.43a. Example ESP™ State Machine for Provisioning Logistics Integrated Circuits (ILCs)

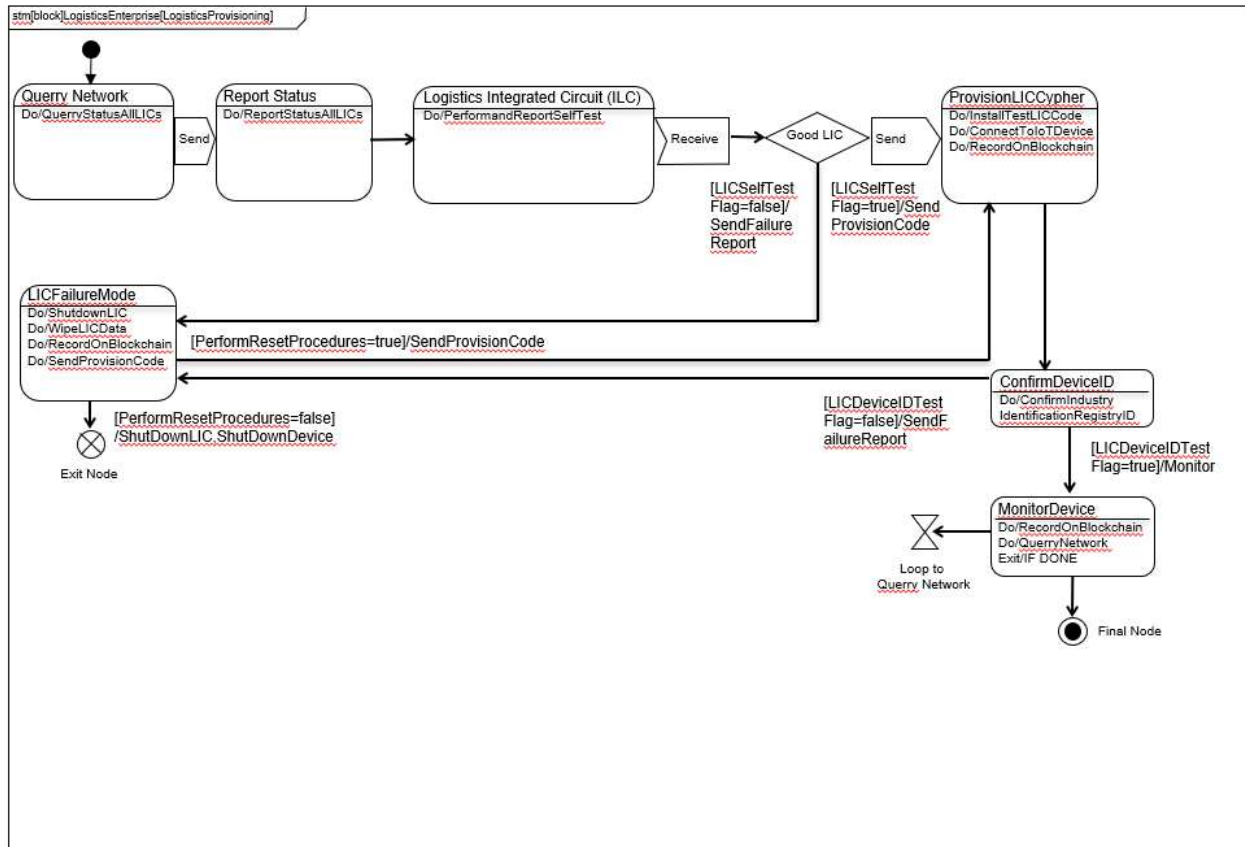


Figure 6.43b. Example Reference Architecture Logistics Enterprise State Machine for Provisioning Logistics Integrated Circuits (ILCs)

Security Features

Protecting the “sensitive assets” of the Philos™ Enterprise might be a more complicated than just a list of physical objects rank-ordered by value or cost. A preliminary analysis and report on the Philos™ System (Lambert and Bishop. *Engineering the Philos™ Blockchain System.*)¹¹¹, indicated that four areas predominated the estimated system costs through 2026, and therefore possibly the whole Enterprise life cycle: financial/legal/insurance; security administration; test & evaluation; and system administration. (See Figure 6.44.) The largest cost driver, financial/legal/insurance, may to a large extent be a surrogate measure for an inverse function of risk. That is, if the system is vulnerable, legal and insurance costs in particular should tend to go up, and vice versa. Included within that concern, if the ownership to the intellectual/digital property rights were lost, the business plan could be vulnerable to fail as well – another legal/insurance issue. Within a new system startup in particular, those systemic business and financial risks could be presumed to be the costliest components that “require protection” from liabilities that could potentially bring down the company, and therefore the system. The takeaway of that is that the biggest systemic risks to value or cost are probably connected to legal liabilities, and not to the vulnerability of physical components.

Looking at the cost/value vulnerability of physical system infrastructure that could be subject to financial loss, one might observe that those could be relatively minimal *relative* to the legal and financial risks mentioned above. One potential business plan might be to reduce the costs of the distributed components – the products are mostly a very large number of distributed logistics integrated circuits (LIC) and/or small components – to a cost breakpoint where they could actually be given away to the users/vendors at no charge. Revenue could be produced by selling “trust/security-as-a-service” through user licenses and/or fees over time. Once provided and provisioned with security codes, ownership of the “down-stream” components and services would be assumed by the user/vendor for their own dedicated applications, and those legal liabilities could therefore be reduced or eliminated by contracts, while still retaining the revenue for usage. Since the physical LICs could have minimal value, are heavily encrypted, and have no stored information value if separated from the active network, they would be assessed to have minimal intrinsic worth. However, if for some reason they lose their data through a breach or data compromise, the network should have the capability to prove the origin of all data on its distributed systems, and ideally each LIC could be re-provisioned back to its previous trusted state with the latest trusted version of data available. The Enterprise would

¹¹¹ Lambert, K., and Bishop, V. *Engineering the Philos™ Blockchain System.* Prepared for Engr 530, Spring, 2019, Colorado State University. May 18, 2019.

certainly have some servers and higher-end computing and storage devices, which could have the highest relative unit cost values, but again, most of the LIC's that do local calculations, and could even be totally virtual machines, might have little intrinsic value.

Considering the loss of value from cyber-attacks, the whole system is designed with defense-in-depth and zero trust in mind. One possible concern is that most distributed system components could be identical, or nearly identical – i.e., if one component is technically compromised by some currently unknown threat, that could lead to the compromise of other “mirror-imaged” network components if they have the same vulnerability.

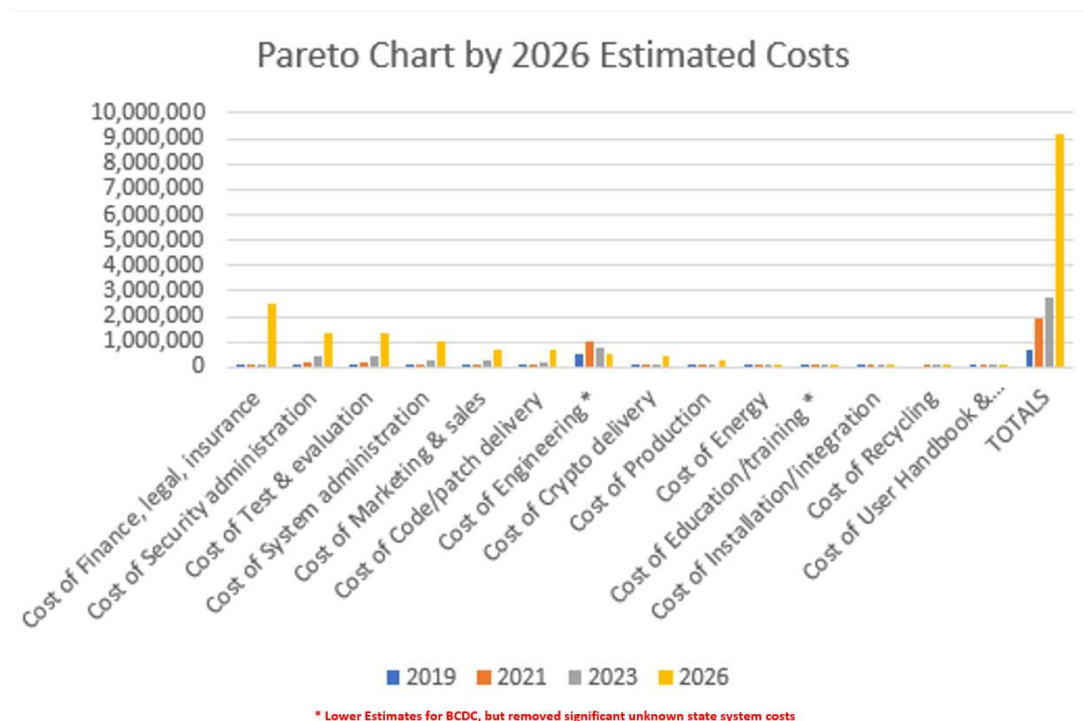


Figure 6.44. Pareto Chart of Relative Costs in 2026 by Cost Driver

Considering the distributed defense-in-depth (DiD) and the zero-trust (ZT) design of the systems, the highest priority for administrative protection is probably with the human operators, insider attacks, and centralized facilities that might be easier to penetrate than the distributed network itself. BlockFrame, Inc., and BCDC should manage physical and cyber security for their own Enterprise components. End users/vendors should manage physical and cyber security for their own system components.

Just as a reference of definitions, Figure 6.45 shows the Services of the ESP™ and Philos™ System-of-Enterprises. Table 6.3 lists various security features and functions with their associated locations. The list does not

include details of all security features but should give a reasonable summary of most of the major considerations and components in each Enterprise and System. Some material was derived from Gorog, C. and Boulton, T. “*Solving Global Cybersecurity Problems by Connecting Trust using Blockchain.*”¹¹² and Lambert, K. “*Cybersecurity for the Block Chain Development Consortium,*”¹¹³ and their derivative sources.

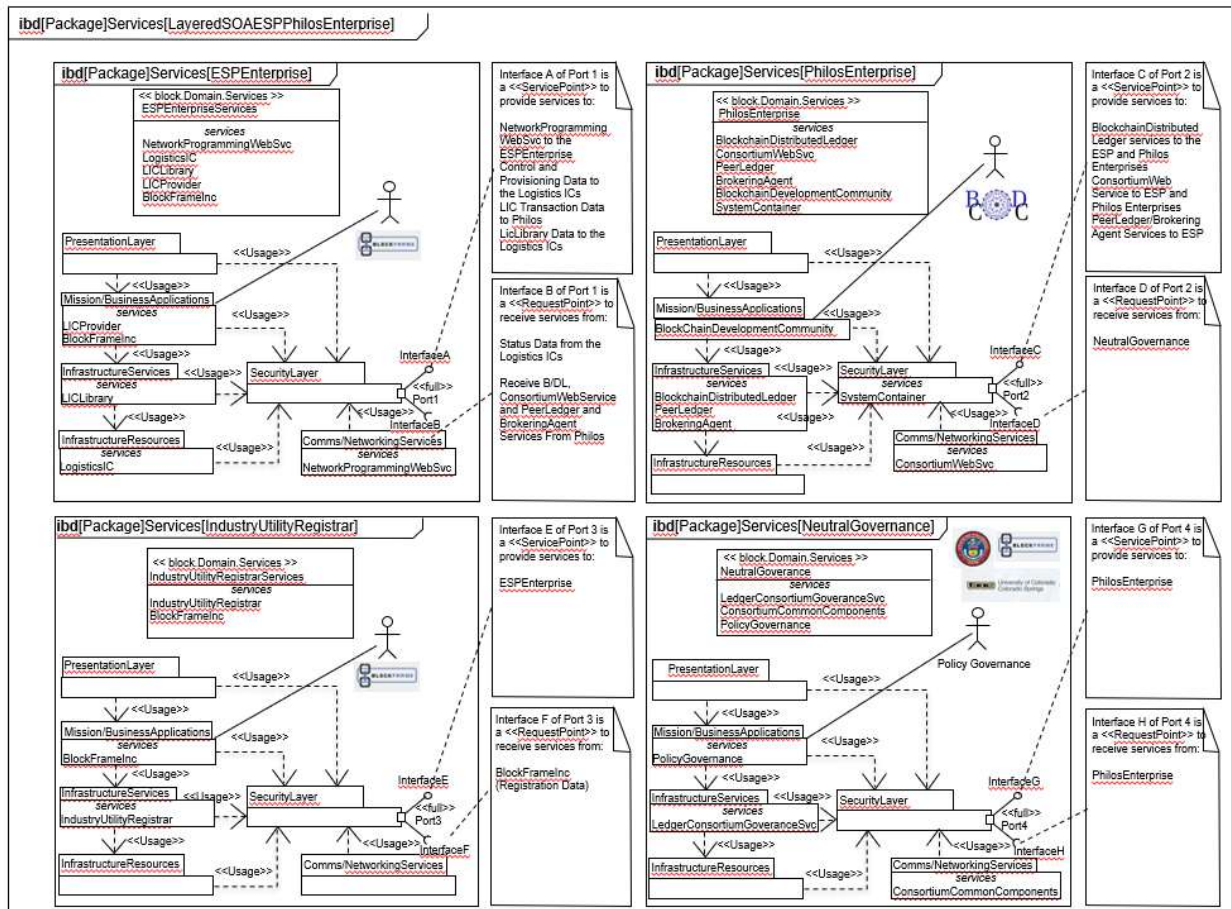


Figure 6.45. Internal Block Diagram of the ESP™ and Philos™ Enterprises

Structural

Probably the most important security component of the Philos™ Blockchain Enterprise is the System Container, due to the fact that it will be used as the security layer for all Brokering Agents, Peer Ledgers, and Consortium Ledgers throughout the Philos™ Enterprise. Each of those are also common components and common

¹¹² Gorog, C. and Boulton, T. “*Solving Global Cybersecurity Problems by Connecting Trust using Blockchain.*” University of Colorado, Colorado Springs. 2018.

¹¹³ Lambert, K. “*Cybersecurity for the Blockchain Development Consortium.*” Colorado State University. May 8, 2019. Engineering 567. Colorado State University. 2019.

interfaces at almost every node of the Enterprise, making them a critically important design component. Figure 6.46 is a Block Definition Diagram representation of the System Container. For now, it should be considered from the Operational Viewpoint, because Block Specifications have not yet been specified sufficiently by designers to verify an accurate Logical/Functional Viewpoint. This level of abstraction might also be considered akin to a Reference Design Pattern or a Reference Architecture because it could be reused with high multiplicity both within the Philos™ Enterprise, and in other security layer systems in other Enterprises or systems. It is also difficult to define and categorize, as the System Container could be considered a Node of an Enterprise, a System within a Node, or an external wrapper containing other complete systems, including Brokering Agents, Peer Ledgers, Consortium Ledgers, or potentially other systems. Table 6.3 contains more descriptive information about key functions of the System Container design, and (Lambert, K., “Cybersecurity *for the Blockchain Development Consortium*.” Colorado State University. May 8, 2019), contains far more details if desired. Even with more detail, the current design is not quite ready yet to validate detailed Logical/Functional Viewpoint specifications.

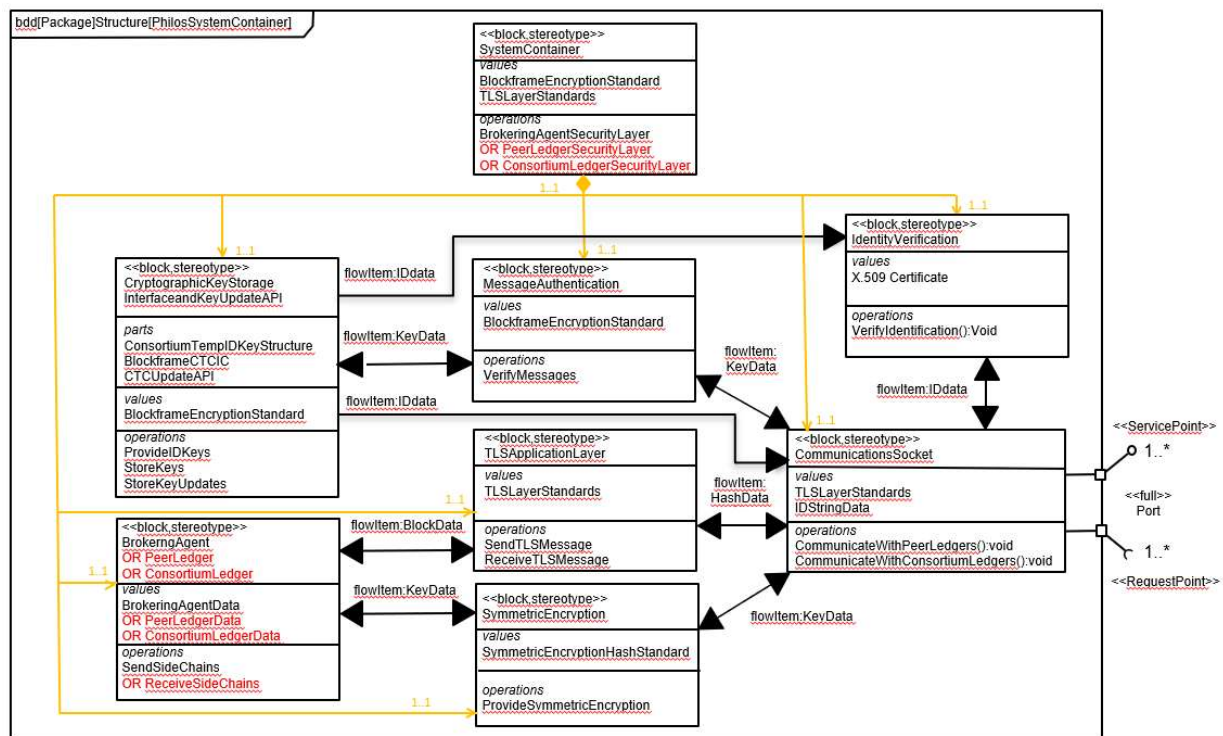


Figure 6.46. Block Definition Diagram of the Philos™ System Container

Table 6.3. ESP/Philos Enterprise Security Feature Summary

Table 3. ESP/Philos Enterprise Security Feature Summary									
Enterprise/System	Resident Component	Security Function							
Philos Blockchain	Excelsior Agent (Transaction Contract)	Transaction Contract Component is "Leading Edge" Boundary Service Point for User/Vendor Input Interface. Secure APIs for access							
	Excelsior Agent (Transaction Contract)	Input Interface services there and throughout system based on loose coupling							
	Excelsior Agent (Transaction Contract)	Transaction Contract Module is serial processor with Broker/Agent, and within the same security System Container							
	Excelsior Agent (Transaction Contract)	Block serializer within the Transaction Contract Module validates user inputs are in standard secure formats, and hashes them into encrypted messages							
	Excelsior Agent (Transaction Contract)	Unidirectional data flow: all data throughout Philos Enterprise is designed to one-way flow to prevent merges with unauthorized data							
	Excelsior Agent (Transaction Contract)	Enforce multi-factor identification and a complex password policy for all users							
	Excelsior Agent	Communicates with other Ledgers, agents, peer ledgers, all data stores only through encrypted data							
	Excelsior Agent	Data stores contain only encrypted data, backed up on regular schedule, regularly audited							
	Excelsior Agent	All Broker/Agents and Peer Ledgers are physically and electronically separated, reducing the attack surface							
	Excelsior Agent	Wide distribution of Enterprise elements reduces the possibility of brute force physical attacks (EMP, or solar wind damage)							
	Excelsior Agent	All Broker/Agents and Peer Ledgers receive patches, OS, and security updates through Centralized Integration/Continuous Delivery (CI/CD) methods							
	Excelsior Agent								
	Peer Ledger	Collects and revalidates data from Broker/Agents for security							
	Peer Ledger	Combines and re-validates all Broker/Agent hashed transactions into new encrypted Side-chains							
	Peer Ledger	Creates Edge Blocks between Broker/Agents and Global Consortium Blocks, thereby separating and decentralizing data flows, and improving distributed security							
	System Container	The primary security capabilities wrapped around all Docking Agents and all Peer Ledgers, with similar security layers for all exposed components of Local Enterprises							
	System Container	Provides robust perimeter defenses to all common SoS components, and defense in depth							
	System Container	Uses smart honey pots to identify and report potential penetrations							
	System Container	Performs and generates cryptographic services for encryption, authentication, and (if needed) for host application services							
	System Container	Local Mutual Authentication Codes (MAC) with coded time completions to authenticate all internal devices (MAC is not the normal definition; more robust)							
	System Container	MAC is based on zero trust enforcement of all connected components							
	System Container	Uses robust internal cyber-security software to detect and eliminate hostile attacks							
	System Container	Provides firewalls and firewall logs for audit, port scan, attack detectors, intrusion detection and protection systems							
	System Container	Uses Advanced Encryption Standard (AES-256) which uses an 8-byte salt, 32-hybrid mode, and a 32-byte password that can only be used for one encryption							
	System Container	Conduct penetration testing of protocols and periodic testing of system components and interfaces							
	System Container	Maintain accurate logging and visibility on all ingress and egress interfaces							
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	System Container								
	Consortium Web Service	Deliver's operational configuration and security updates to Broker/Agents, Peer Ledgers, and other Consortium Devices							
	Consortium Web Service	Provides secure, robust, encrypted communications in Blockchain blocks among the Broker/Agents, Peer Ledgers, and Consortium Servers							
	Blockchain/Distributed Ledger	Provides highly-distributed, redundant, immutable, non-repudiable, consensus-validated data that is auditable, scalable, and highly resistant to tampering							
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger	Primary source of expertise for Philos Development, Security, Operations (DevSecOps), maintenance, sys admin							
	Blockchain/Distributed Ledger	Time Machine for Philos security design, admin, and security master plan (SMP), but security might be done through a neutral third party							
	Blockchain/Distributed Ledger	Security operations: regular audits and enforcement of security standards; conduct surveillance of all logs, CS versions, code versions, patches, upgrades, and port configurations							
	Blockchain/Distributed Ledger	Conduct penetration testing of protocols and periodic testing of system components and interfaces							
	Blockchain/Distributed Ledger	Monitor activity and VM logs for abnormal activity							
	Blockchain/Distributed Ledger	Establish a course of training for operations and security for Philos Users							
	Blockchain/Distributed Ledger	Establish a 24/7/365 Incident Response Team (CSIRT)							
	Blockchain/Distributed Ledger	Provide physical and legal safeguards for the Philos Enterprise							
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								
	Blockchain/Distributed Ledger								

Table 6.4. ESP/Philos Enterprise Security Feature Summary (Continued)

Table 3. CONTINUED ESP/Philos Enterprise Security Feature Summary									
Enterprise/System	Resident Component	Security Function							
Neutral Governance	Ledger Consortium Governance Svc	Governance algorithms to govern, prioritize, and incentivize efficient and secure behaviors on the blockchain							
	Ledger Consortium Governance Svc	Executes algorithms for governance using statistical distributions, state machines, and/or optimization tools to represent policy governance							
	Consortium Common Components	DNS-like structures collect all Enterprise Bridge Blocks; revalidates them, and combines them into new blocks on the Global Philos blockchain							
	Consortium Common Components	Provides rapid, robust, encrypted communications of blockchain blocks among Peer Ledgers, Consortium Servers, and Neutral Governance Servers							
	Policy Governance	Three-organization "non-governing" board to codify enterprise governance policies (Currently Colorado OIT, UCCS, BlockFrame)							
	Policy Governance	Reduces the possibility of conflicts of interest among developers and/or stakeholders							
	Policy Governance	Prioritize blockchain security elements for Confidentiality, Integrity, and Availability (CIA)							
	Policy Governance	Quantitatively enforce CIA through the following five Trust Categories: Suppression; Validation; Reliability; Refutation; Deprivation							
ESP Enterprise	Network Programming Web Svc	Delivers configuration and security updates to User/Vendor apps and devices							
	Network Programming Web Svc	Provides rapid, robust, encrypted communications among User/Vendor apps and devices to Philos Broker/ing Agents							
	Logistics IC (LIC)	Receives MAC codes to associate time-stamped tokenized proof of user/vendor identity							
	Logistics IC (LIC)	Uses time-stamped tokenized tokens to prove each operational and logistics transaction							
	LIC Library	Stores tokenized proof of identity for each user							
	LIC Provider	Provides trusted manufacturer ILC							
	BlockFrame Inc	Provisions manufacturer ILC with base operational and security coding							
	BlockFrame Inc	Provides base-coded ILCs to Users/Vendors for their specific application and security coding							
	BlockFrame Inc	Prime facilitator for ESP's security design, admin, and security master plan (SMP), (but security might be done through a neutral third party)							
	BlockFrame Inc	Establish a course of training for operations and security for ESP Enterprise Users							
	BlockFrame Inc	Establish a ESP Computer Security Incident Response Team (CSIRT)							
	BlockFrame Inc	Provide physical and legal safeguards for the ESP Enterprise							
Industry Utility Registry	Industry Utility Registry	Generates Mutual Authentication Codes (MAC) with seeded time completions to authenticate all internal devices							
	BlockFrame Inc	Manages the IUR							

Behavioral

Figure 47 is a Sequence Diagram showing a notional encryption provisioning process with time-constraints.¹¹⁴ The Figure is a very simplified hypothetical example of a series of steps to establish asynchronous communications keys and links, followed by synchronous communications keys and links for Enterprise operations, or for use by an end-user/vendor. Once the correct security packages are sent, received, and installed, users can (in this example) open a secure communications tunnel to another user or users. Theoretically, the Secure Tunnel can be used to maintain two-way encrypted communications for just a few microseconds, for minutes, or for years.

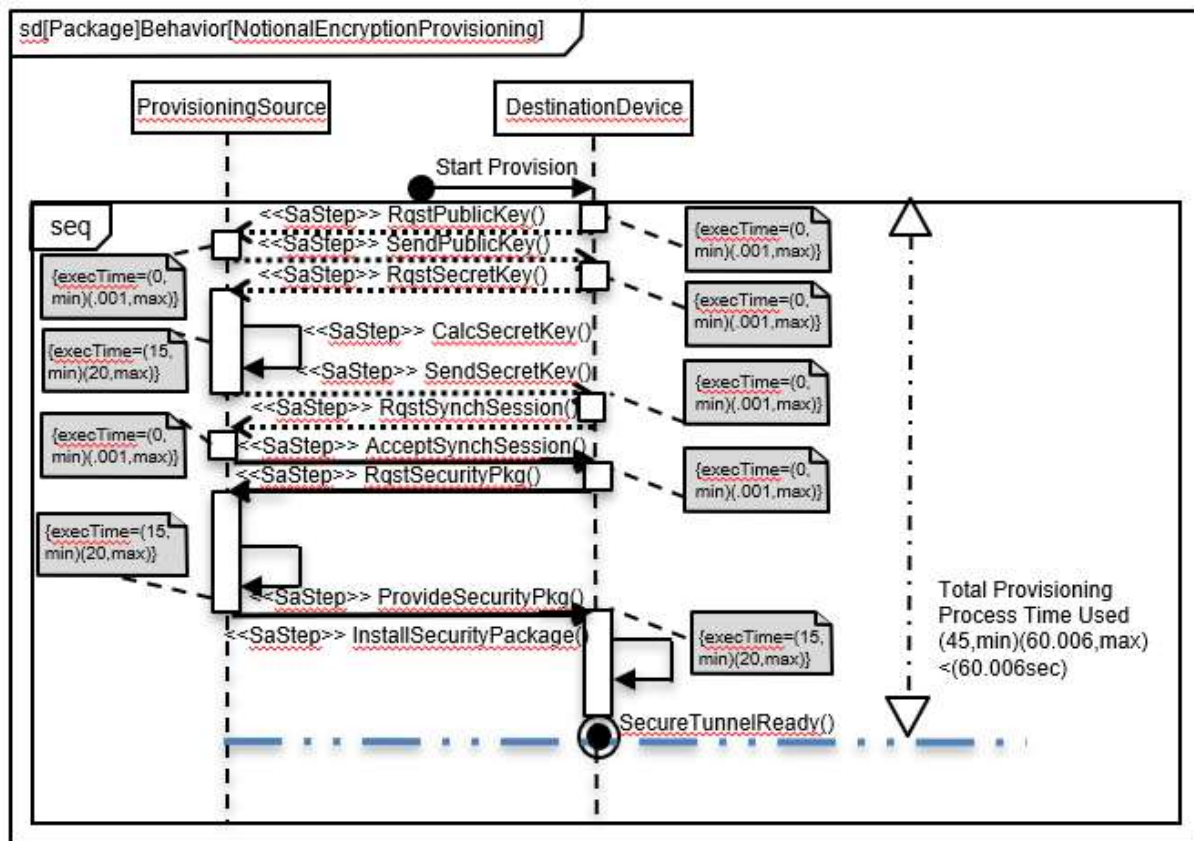


Figure 6.47. Sequence Diagram of a Notional Encryption Provisioning Sequence

An effective Security Administration (SecAdmin) approach includes defined organizational architecture, metrics, and indicators that would be examined as part of continuous assessments and ways to detect malicious actions. As shown in Figure 6.45, the BlockFrame, Inc., Eco-Secure Provisioning (ESP™) Enterprise and the Blockchain Development Community's (Philos™) Enterprises and their supporting systems are composed of four

¹¹⁴ Modified from Lambert, K. *Engr 567 Project Progress Module 11*. Colorado State University. Dec. 2019.

main Block Domains. Each Block is at least “semi-autonomous,” partially for security reasons. Security Administration (SecAdmin) is therefore also coordinated among the four Block Domains. Therefore, to address their SecAdmin, I would address the overall “Global” SecAdmin, and then the unique SecAdmin considerations of each Domain.

“Global” Security Considerations

Since the four Domains include two semi-autonomous Enterprises, some security challenges could come from emergent behaviors. If security problems occur within one enterprise, it is conceivable that it could affect the security of the other enterprise. “Global” security architectures and rules therefore need to be considered among all the Domains. Philosophically, these rules should generally be considered for “global” application: Following initial deployment, routine SecAdmin decision making should be relatively autonomous, and should probably be accomplished to enforcing rules formally agreed-to by all Domain CEOs and executed by a third-party contractor to avoid conflicts of interest and reprioritization of corporate interests for the sake of expediency. Ideally, this SecAdmin should be administered separately for each Domain, and separately from Systems Administration (SysAdmin). While these rules might not be practical during system design and near-term operations due to relatively small staff, it should be at least an aspirational goal as the enterprises mature. As described in Tables 6.3 and 6.4, most common SecAdmin practices would apply to each Domain, including training, security logging and audits, personnel screening, penetration testing, legal mandates, robust authentication and authorization methods and standards, etc.

ESP™ SecAdmin

Since ESP™’s purpose is primarily for logistics and supply chain security, security concerns could focus on security with those unique clients, their locations, their applications, and their devices. Those devices could be highly distributed and could include Internet of Things (IoT) devices. BlockFrame, Inc., and its logistics clients would primarily be concerned with the security details of those use cases. In addition, if processes to log and secure record information is done through the Philos™ blockchain, there should be appropriate security considerations to coordinate those common standards, interfaces, data and encryption standards, etc., with the BlockChain Development Community (BCDC). Due to its relatively closed architecture, its proprietary features, and the nature of its clientele, ESP™ security could be more selective and “controllable” compared to Philos™. ESP™ security

requirements could be based more on contractual agreements than more “blanket”-oriented open agreements with Philos™.

Industry Utility Registrar™ (IUR™) SecAdmin

The security of the IUR™ will primarily be the responsibility of BlockFrame, Inc. It is primarily connected to serve the ESP™ enterprise to assure correct registrations of logistics provisioning (including secure tokens and time-stamps) but is not directly connected to the ESP™ Enterprise to isolate its security and to reduce the possibility of conflicts of interest. In the future, it is assumed to be severable from the ESP™ and governed by a separate body. Since transactions using IUR™ information are planned to be stored on the Philos™ blockchain, there are indirect usages of IUR™ information by the Philos™ blockchain Enterprise and the Neutral Governance system.

Philos™ SecAdmin

The SecAdmin of the Philos™ Blockchain can operate as a service for the ESP™, or for a very broad scope of other users. It is generally designed as an open architecture for (more or less) “public” use (“community” use might more accurately describe some degree of a vetting process) which could provide services to a variety of governmental, non-governmental, and commercial users. The BCDC would primarily be responsible for their own internal and external security designs and operations. The SysAdmin functions would probably be a combination of both internal corporate security functions (corporate training, personnel screening, enforcement, etc.) and external security functions (surveillance, logging forensics, inspection scheduling, penetration testing, etc.), probably performed in the future by a third-party contractor.

Neutral Governance SecAdmin

The Neutral Governance System is a third-party component or Agent, disconnected from the Philos™ Enterprise for security and conflict-of-interest reasons. It is used to provide automated governance services to the Philos™ Enterprise by screening incoming hashed blockchain data, optimizing a process to include blockchain blocks by the consensus of users, adding new blocks to the blockchain, and transmitting the new blockchain version back to the users. This might not be the most technically complex process in the Enterprises, but the optimization policy could be the most controversial among users because it provides mechanisms that could give incentives and/or disincentives to different users. Therefore, the neutral governance policies must enforce agreed-to principles, with as much security, impartiality, and honesty as possible. As shown in Figure 6.45, the governance policies should be controlled regulated by a board of at least three entities: BlockFrame, Inc., the Office of Information Technology

(OIT) of the State of Colorado, and the Subject Matter Experts (SME) from University of Colorado at Colorado Springs (UCCS). A memorandum of understanding (MOU) has already been created to study and cooperate on the development of the Neutral Governance Domain, which should evolve in parallel with the development of the software, network structure, and other governance components, prior to full operational capability.

Metrics and Indicators

Table 6.3 describes several security functions that involve continuous assessments to discover malicious actions. Table 6.4 gives some examples of Quality Attributes (QATs) (metrics and indicators) of security quality, generally over time.

Table 6.5. Example of Quality Attributes

Table 4. Example Quality Attributes (QAs) for the Philos Enterprise									
Quality Attribute	Definition	Metrics			Complicating Issues				
1.5.1 Boundary Protection	The system can resist hostile penetration of firewalls	Percentage of components tested vulnerable to professional penetration testing			Threat actors constantly develop new equipment and tactics that might not be testable				
1.5.2 Password Security	The system can assess passwords policies for complexity	Percentage of passwords tested vulnerable to testing tools			Unauthorized recording of passwords and theft through social engineering is very common				
1.5.3 Training Currency	The system can train and monitor compliance on policy mandates	Percentage of personnel current and compliant with security training and legal agreements			Employees could be reluctant to take training or comply, which for security violations might require termination				
1.5.4 Personnel Reliability	The system can screen personnel for criminal backgrounds	Percentage of employees passing background checks			Many disruptive behaviors (drug/alcohol abuse, disruptive conduct, incompetence) might not be illegal				
1.5.5 Multifactor Identification	The system can enforce authentication through biometric ID devices	Percentage of users with biometric signatures			Digital copies of biometrics could be stolen with permanent compromise, could involve privacy violations				

Conclusion

This chapter discussed the systems architecture development processes of two enterprises: the “Eco-Secure Provisioning™” (ESP™) Enterprise, and the “Philos™” blockchain/distributed ledger (BDL) Enterprise, originally developed through the Block Chain Development Community (BCDC). The research goals are to study and apply systems engineering principles using Model-Based Systems Engineering (MBSE), and the Model-Based Systems Architecture Process (MBSAP), as presented in Borky and Bradley, 2019.¹¹⁵ This chapter generally did not include the use of computer-assisted modeling tools. Other examples using computer-assisted tools are included in Appendix C.

¹¹⁵ Borky, John Michael, and Bradley, Thomas. *Effective Model-Based Systems Engineering*. 2019. Accessed June 22, 2022. <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

6.3. Additional Research Applying SysML Modeling Tools

Research shown in Chapter 6 was conducted using the SysML Framework without using the computer-assisted SysML tools. One such computer tool, the Cameo model, was subsequently used to expand upon Research Objective 2 to demonstrate more advanced methods and capabilities of computer-assisted modeling, but since the systems engineering concepts are virtually identical, all Cameo model diagrams that expand on Research Objective 2 will be presented in Appendix C for easier readability. Cameo was used because of its availability, but other computer-assisted tools are available that also use the SysML framework.

6.4. Philos™ Neutral Governance Architecture

The BlockFrame® Neutral Governance System Architecture is considered and designed as part of the BlockFrame® system-of-systems. Because of the requirement for neutrality, it also requires institutional and legal separation, and is filed and conducts business as a non-profit company, the International Alliance of Trust Chains (IATC) of outside of BlockFrame, Inc. A separate white paper about the architecture for the neutral governance system is largely duplicative of the discussions in this Chapter 6, and is presented as Appendix B.

6.5. Conclusions for Research Question 2

Upon completion of this research, I am now prepared to answer Research Question 2. RQ2 addresses practical MBSE and MBSAP techniques to cybersecurity and digital supply chain requirements and processes. Architectural modeling has been accomplished at the Operational and Logical/Functional Levels to answer the research question using examples from BlockFrame® technologies. This research provides evidence to support the assertion that MBSE and MBSAP are effective approaches with value at organizational and systemic levels. Since these phases of research were finished to satisfy RQ2, more research was conducted and is ongoing, with the intent to expand initial BlockFrame® technology designs, into specific use cases and reference architectures.

Due to resource constraints, the opportunity for cost savings, and the need to move to a more digital environment, it was practical to modify RQ2 Task 3 (RQ2-T3). Instead of a single physical cybersecurity textbook, "Applied Blockchain Technology for Security and Privacy," BlockFrame completed a video training course for initial training for employees, and extensively address blockchain, cryptology, security, and privacy fundamentals, which still satisfies the intent of that task. Other cooperative processes with IEEE are included under Research Question 4 and addressed in Chapter 7.

7.0. Applications of Cryptologic Token Provisioning and Distributed Ledger Governance for Secure and Trusted Commercialization.

This Chapter addresses and satisfies Research Question 3 (RQ3) by demonstrating how the provisioning of cryptologic tokens and the governance of distributed ledgers can create a secure and trusted environment for commercialization. Restated, Research Question 3 (RQ-3), is, “How can Eco-Secure Provisioning™ (ESP) of cryptologic non-fungible utility tokens (NFTs), in conjunction with blockchain/distributed ledger-assisted management and governance, provide secure commercial services, improve cybersecurity, and to enforce trust in systems-of-systems (SoS)?” The research in this chapter examines how functional BlockFrame® system-of-systems capabilities can operate as a sustainable commercial business model and provide effective blockchain, cybersecurity, and supply chain services. This Research Question is also well-aligned to support the philosophy of the goals of the Doctor of Engineering in Systems Engineering program and its emphasis on research with industry.

For this research, system-of-systems terminology will refer to both integrated and independent BlockFrame® systems, as well as external systems used with business-to-business (B2B) clients through common components, interfaces, and Service Level Agreements (SLAs).¹¹⁶ BlockFrame® systems are designed to provide different services for B2B clients tailored to their own requirements and business plans. The guiding BlockFrame® Mission, Vision, and Values can be summarized as:

- “BlockFrame® uses blockchain to organize cybersecurity worldwide – achieving trust and uniformity for secure supply chains,”
- “Provides logistics support to create a secure ecosystem for managing IoT devices using blockchain and distributed ledger technologies,” and,
- “Individualized Privacy, Proof of Origin of data, Zero Trust Networking, Defense-in-Depth Security, Multi-Factor Identification, and Accountable Governance”¹¹⁷.

This RQ3 investigates these key aspects of BlockFrame® systems to better understand how they are applied to practical and applied cybersecure services and transactions. To address these objectives, the content of this chapter is organized as follows:

¹¹⁶ Although BlockFrame® designs are primarily intended to integrate with large-scale B2B clients, each Enterprise may operate independently on smaller scales if that meets the needs of specific customers or use cases.

¹¹⁷ “BlockFrame – BlockFrame Inc.” <https://blockframetech.com/> (accessed Jul. 20, 2022).

- BlockFrame® Functional Architecture. Examine the internal functional behaviors of the BlockFrame® ESP™ and the Philos™ blockchain/distributed ledger to process cryptographic key management and transactions using utility NFTs, processes to allow global scalability, and externally facing monetary interfaces to support commercial purchases for secure transactions to support the overall business model.
- Philos Marketplace™ Operations. Examine the behaviors the BlockFrame® system-of-systems to progressively increase trust factors in a complex, cross-industry global Philos Marketplace™.
- Neutral Governance Actions. Examine and control the Philos™ B/DL process through a neutral governance organization that can set operational parameters for policies to control the consensus trust algorithm.

7.1. BlockFrame® Functional Architecture

The internal functional behaviors of the BlockFrame® ESPTM and the PhilosTM blockchain/distributed ledger to process transactions and cryptographic key management uses utility NFTs and processes to allow global scalability and externally facing monetary interfaces to support commercial purchases of secure transaction services. The PhilosTM distributed ledger uses a “green” design with no energy mining for “proof of stake.” This process intentionally does not attempt to create any monetary value to be used as a cryptocurrency but is designed to permanently record transactions immutably and non-refutably as the primary service. This research objective expands upon the emergence concepts developed in RQ1 to describe system behaviors and economic processes that explain and support that overall business model. Instead of requiring PhilosTM to be used as a financial tool to store and communicate value as a cryptocurrency – i.e., to turn electrical energy into a digital currency – the PhilosTM incentive consensus model is designed to reward participation and mutual compliance with defined uniform security requirements. Therefore, consuming energy as a “proof of stake” to participate in the blockchain is not necessary or desired. In effect, PhilosTM essentially functions as a direct sale for increased security and trust. This allows far more economical uses of PhilosTM as a competitive distributed ledger, blockchain, or “trust chain” when compared to those that use expensive energy mining. Over several years of development, this key concept has proved to be very difficult for many blockchain users or even “experts” to understand. In effect, it is almost inevitable that most users have to “unlearn” previous experiences they might have had with cryptocurrency-based blockchains, previous academic training they might have had about the need for energy mining, or other pre-conceived notions that equate blockchains with very wasteful uses of energy. It is very common that after explaining this concept for an hour or more, students or users will still ask the question, “but how do you do your energy mining?” BlockFrame® has observed that “unlearning” common cryptocurrency-based concepts needs to be planned as part of the “overhead” of learning how the PhilosTM distributed ledger operates, and therefore how the whole BlockFrame® systems-of-systems operates.

Other effects of the difference between the BlockFrame® design and typical cryptocurrencies could be financially disruptive in other ways. Typically, government-regulated distributed energy systems reject systems that might be considered wasteful of energy – which could make BlockFrame® more attractive. While several industries currently assume that they need very large energy sources, including nuclear sources, to operate large-scale server

farms or data centers to “mine” energy to fuel transactions, BlockFrame® technologies have the potential to greatly reduce those costs.

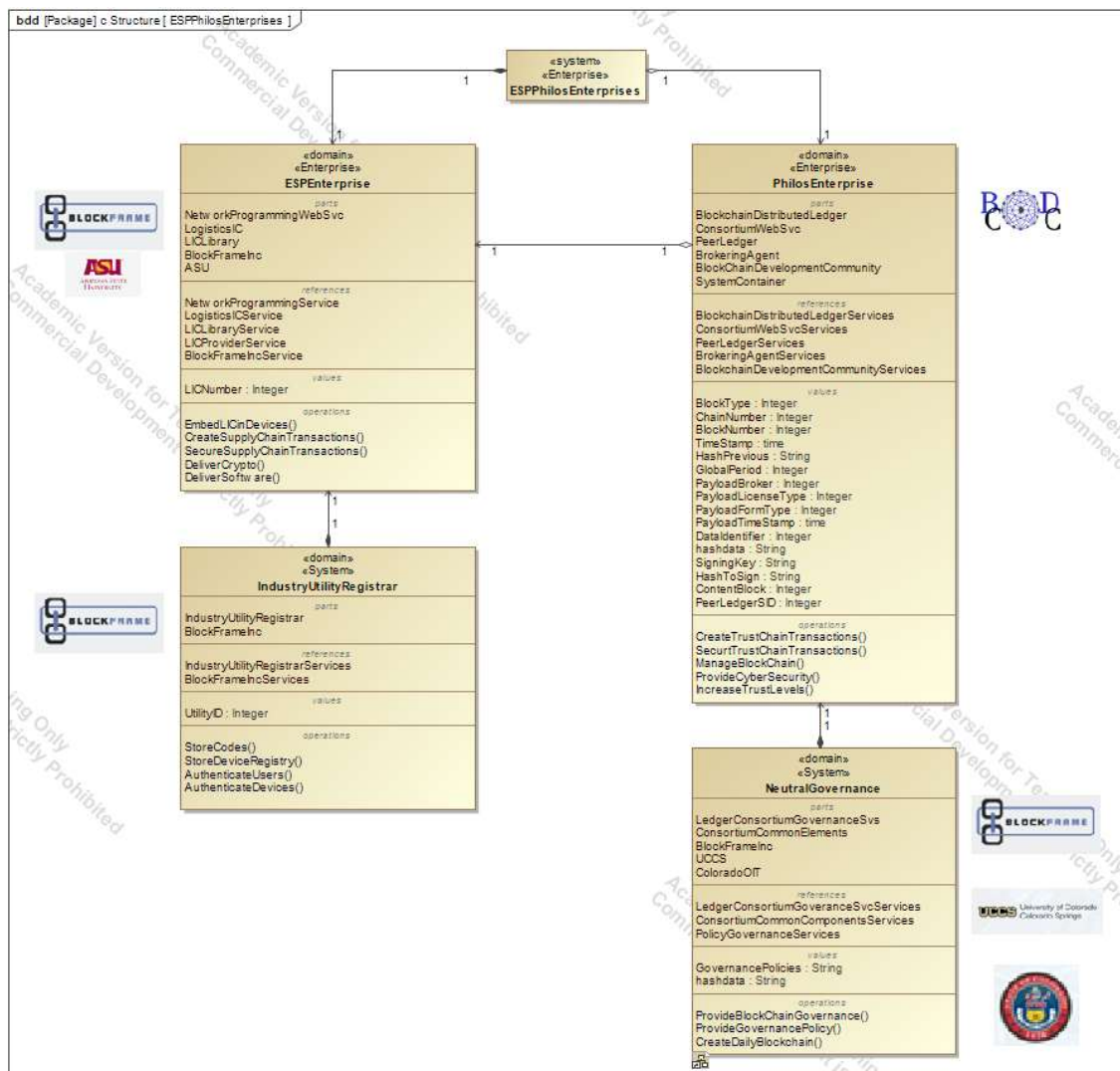


Figure 7.1. Domain Diagrams for the Four ESP™/Philos™ Enterprise Domains

For example, Figure 7.1 shows the four system domains of the BlockFrame® enterprise modeled in SysML. Other diagrams, which are too extensive to be shown here, were used to model each of these domains in greater detail, but can be reviewed in Appendix C. The internal functional behaviors of the BlockFrame® ESP™, and the Philos™ blockchain/distributed ledger to process transaction contracts using utility NFTs, design processes to allow global scalability, and the design of externally facing monetary interfaces to support commercial purchases for secure transactions using the Philos™ blockchain/distributed ledger will be shown in greater detail in this chapter. Exhaustive mapping of requirements and capabilities, such as mapping Cybersecurity Maturity Model Compliance

(CMMC) requirements to each component or behavior in Philos™, ESP™, or other systems or components that satisfied each of those requirements was accomplished. Since the initial design efforts, detailed research was made by Dr Christopher Gorog, et.al., in the white paper “Carbon-Neutral Distributed Ledger,” which was presented at the IEEE Innovative Smart Grid Technologies (ISGT) in April, 2022¹¹⁸. That research added extensive new details about the internal workings of the Philos™ distributed ledger, the use of utility tokens, consensus modeling, trust algorithms, and particular advances for better acceptability for use cases involving transactive distributed energy resources (DERs). Other BlockFrame® efforts have created a statistical surveillance dashboard to monitor the relative behavior trends of each user in near-real time, and overall increases in trust metrics over time.

Internal and External Networks

Except for system development, and possibly for testing within a System Integration Lab (SIL) or test network, the operational Philos™ distributed ledger system itself is designed to have no exposed LAN-connected devices other than cryptologically hardened SystemContainers. The nature and security of the entire Enterprise architecture is largely based on SystemContainer (SC) units that act as autonomous, distributed, common elements of a System-of-Systems Enterprise which communicates through an Internet WAN network using encrypted and hashed messages. Local users of Philos™ might typically use internal LANs to do their own business or mission, and then connect to Philos™ through encapsulated, loosely coupled, and cyber-secure common SystemContainer Application Programming Interfaces (APIs) for Philos™ Services.

Messaging, cryptography services, and other communications are typically in the form of encrypted hashes using the Internet, but not through the World Wide Web. SystemContainer-to-SystemContainer messages through the Internet should comprise virtually all of the blockchain-related business, while each SystemContainer contains its own internal business logic to process most of the information within the System, with a few exceptions. So, it can generally be assumed that each SystemContainer interfaces directly to the Internet through cloud services. Locally, different Users can also access each SystemContainer interface directly but may or may not have use the Internet to do so. In some cases, SystemContainers could be directly connected to User Applications, or directly embedded within User Applications, which is usually more efficient. Typically, the interface architecture on the “User side” of each interface will be the full design and responsibility of each User and will not usually be considered

¹¹⁸ C. Gorog, P. Russell, and T. E. Boulton, “Carbon Neutral Distributed Ledger,” New Orleans, LA, Apr. 2022.

part of the Philos™ network itself. The interfaces for each Philos™ SystemContainer will often either connect to both unique User Applications and to other common SystemContainers Applications, but with mutually exclusive forms of messages and data.

As some components of Philos™ are currently still in development, not all WAN architecture configuration decisions have been made yet.¹¹⁹ Due to the novel requirements and configuration of Philos™, it is designed as a unique ledger design for unique purposes. Therefore, existing NIST, ISO, or other standards might not be sufficient to apply to the unique capabilities of the whole BlockFrame® Enterprise. Some standards for defining items like common hash lengths will continue to be useful, but those might not satisfy the needs of the overall System architecture.

All that said, specific network topologies of SystemContainer patterns have not yet been formally defined or formalized but are currently being researched. I presented that issue at a March 26, 2020, meeting of the Neutral Governance Organization, the International Alliance of Trust Chains (IATC), and some options were discussed for network topology options, such as those described in Selva, et. al.¹²⁰ The first patterns that have utility to help define the Philos™ networking architecture are the “partitioning pattern,” and the “connecting pattern.”¹²¹

Since each Philos™ SystemContainer (as described above) will work autonomously within a distributed framework, it can almost without exception be characterized as a distributed pattern. However, since each SystemContainer can internally perform different functions, those functional characteristics can also drive the topological patterns of how they become interconnected. Figure 7.2 is an abstraction to help explain and simplify the complexity of the common components of the distributed Philos™ Enterprise. The systems containers are truly common components to provide hardware, software, and behaviors to protect their internal components throughout the system-of-systems. They are currently planned as cyber-physical components for reasons of security and backward compatibility with legacy systems, but in fact could be realized as totally virtual devices. When networked together, the systems within the system containers can communicate and perform their own unique functions but still be protected by their own system container. Topologically, they are all true common components, but they also all

¹¹⁹ Lambert, K., “Philos™ Neutral Governance Architecture Overview and Summary Information,” 2020, is available in Appendix B, and represents an initial framework for a cooperative Neutral Governance Organization (NGO) that will ultimately set neutral standards and policies for the total worldwide Enterprise architecture.

¹²⁰ Selva, D., B. Cameron, E. Crawley. *Patterns in System Architecture Decisions. Systems Engineering*, Wiley, 19, no. 6 (2016).

¹²¹ Described in Selva, Figures 6 and 7, P. 492.

contain and perform unique behaviors as a complex system-of-systems with the blockchain/distributed ledger as the central logic mechanism to referee and manage the whole organism. SysML can be useful to help define those architectures, but with the potential huge numbers of components and transactions within the Philos™ economic ecosystem, it is also very difficult to visualize at scale.

At the lowest hierarchy of connections, two functional types of devices within SystemContainers are used: BrokeringAgents (BA), which also internally contain BrokerSearchAgent and BlockSerializer components and functions, and PeerLedgers (PL), which receive and screen messages from a set of BrokeringAgents and process those messages on to higher hierarchies. With few exceptions, such as a dedicated 1-to-1 instance of a PeerLedger implementation, a multiplicity of about 3 to 10+ User Applications Interfaces would typically connect to each BrokeringAgent, and perhaps 3 to 10+ BrokeringAgents would connect to each PeerLedger. Logically, it is anticipated that a star-of-stars topology will typically form under normal conditions, as shown in Figure 7.2. For better visualization, only a relatively low number of connections and instances are displayed, but stars of 10 or more might not be uncommon. In actual instances, that geometry may completely depend on the needed capacity of the Users, and their relative rate of use. PeerLedgers themselves could be connected to other PeerLedgers or to higher levels in the hierarchy to ConsortiumLedgers, which might form a topology more like a mesh pattern. In each User/SC-BA pair, each User may seek unique blockchain services from its paired BA. At that level then, high complexity may come from the high diversity of Use Cases/Scenarios. On the other hand, after a blockchain service request is made, the BrokeringAgent/PeerLedger star topography should begin to follow very structured, non-unique methods to pass blockchain service requests to higher hierarchies where they will eventually be merged into new blockchain blocks daily. At those levels, that complexity is not only from unique applications, but could be complex in its ability to scale-up in size into a massive worldwide-scale of services.

Blockchain Consensus Topography and Behaviors

Jumping ahead to the uppermost hierarchy, new blockchain requests must be processed, merged without losing fidelity or data content, and “accepted” into new daily blockchain blocks through a consensus process by following policies and rules and governance from the Neutral Governance Organization (NGO). While that higher level should be much less complex than the multiplicity of SystemContainers, its network topology involves a more sophisticated decision process. While the “star” process at the level below PeerLedgers seems straightforward and

accepted, the decision process for selecting the best topology of the ConsortiumServers and their governance could be what Selva¹²² might describe as having more “conflict.”

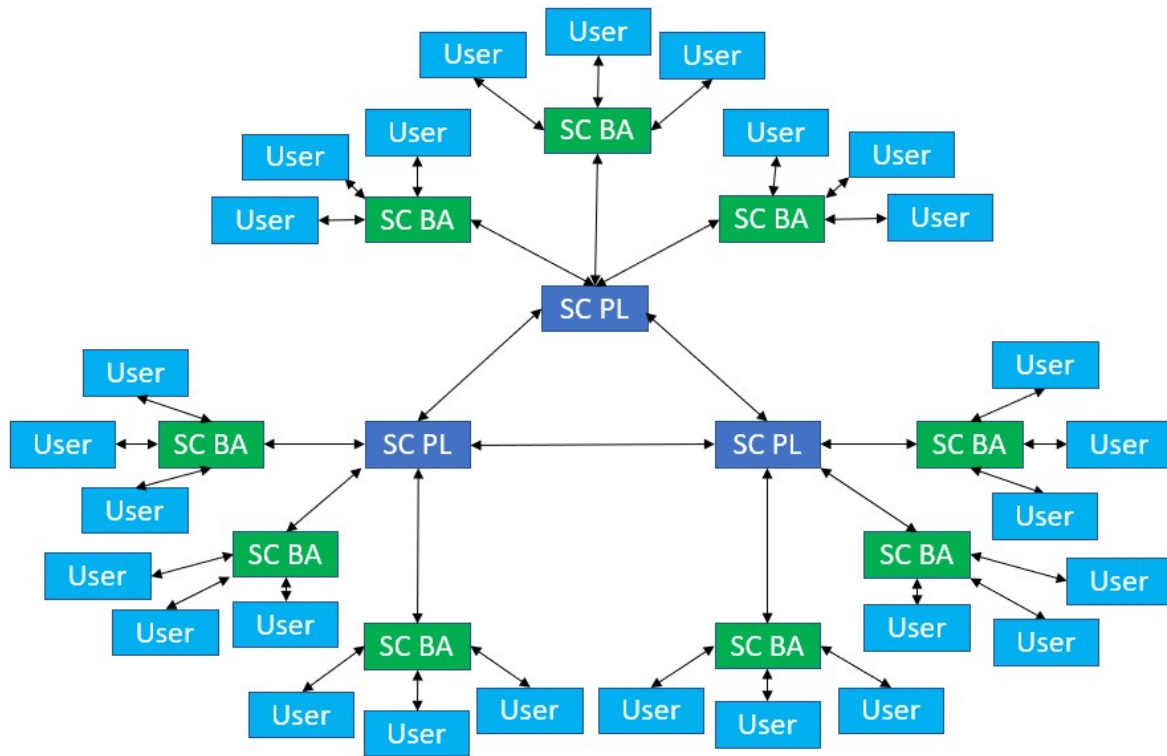


Figure 7.2. “Lower Hierarchy” Star-of-Stars Topology Pattern with Service Interfaces from User Applications to Brokering Agents (SC PL), and then to Peer Ledgers (SC PL)

For design purposes, the number of ConsensusServers is currently assumed to be 13, following the pattern of the Domain Name System (DNS) that currently governs the Internet. The current thinking is that the DNS works well as an international governance pattern, and there may be little reason not to follow that pattern for now. The issue is that the Philos consensus process is uniquely different from Internet governance in that it must centralize and merge data from all sources and follow a governance policy that creates new blockchain blocks and then return those new public blocks back to the Users for their continued and iterative use. Considering the topology patterns presented by Selva, the Consensus topology could be any of those patterns: ring, star, mesh, or tree, perhaps with advantages or disadvantages of each.

Each pattern option should be considered objectively for their respective benefit:

¹²² Selva, D., B. Cameron, E. Crawley. *Patterns in System Architecture Decisions. Systems Engineering*, Wiley, 19, no. 6 (2016).

- A ring pattern could emphasize good “interior lines of communication,” perhaps on dedicated synchronous tunnels to share and collectively process consensus information. In this case, one central decision point might not be necessary, but it raises the question of how consensus can be reached without a single central decision point. Redundancy could be an advantage if all 13 servers can act effectively in concert.
- A star pattern could centralize information and decisions into a single hub structure focused on making all decisions exclusively. An obvious disadvantage would be the potential vulnerability and lack of redundancy of a single point of failure or degradation.
- A tree structure could be appropriate for a more distributed decision process, such as having each ConsortiumServer under the sovereign authority of separate governments or political jurisdictions. A regional governance structure could each have its own tree structure, respect for governance over its own regional authority for rules and consensus, and higher levels of autonomy. The disadvantage is that such a distributed decision structure could increase political and security concerns and does not necessarily answer the question about how separate tree structures can merge to form single universal consensus blocks. Eventually, information has to pass sovereignty boundaries to come into consensus, and it is not clear how that will work from a tree structure that could make sovereign boundaries harder to penetrate. Decentralized consensus decisions could also compromise the whole intent of a common neutral governance structure.
- A mesh structure does not come to mind as a predominant solution but could be considered as a hybrid or compromise connecting solution if other preferable solutions do not emerge.

A couple of other sub-decisions must also be resolved:

- How will ConsensusServers work together to accept, decide on consensus priorities, and create new blockchain blocks. The topological considerations are whether a single physical node must exist where that physically happens and will one node predominate all the others. It is possible that action might occur at any one of the 13 ConsensusServers, but then which one of the 13 will make the final decision, or can that responsibility be rotated among all 13 ConsensusServers? That question has yet to be resolved.

- The international, cross-jurisdictional nature of the Consensus requirements raises important questions about whether governance, or at least selected elements of governance, can cooperate successfully across jurisdictional boundaries, or if they must be centralized into a single policy and automated consensus service. One possible solution might be to federate each ConsensusServer into regional sub-organizations with certain autonomous powers and then collect regionally collected blockchain data elements for final consensus. That question has also not been fully resolved, and in implementation, might depend on dynamic situations. Building trust among the sensitivity of competitive users could also be a function of the level of trust in the robustness of cryptology to keep the content of individual user's secret and secure.

It might make sense to address and resolve the ConsensusServer topologies and Consensus governance behaviors issues prior to describing and specifying the intermediate hierarchy levels between PeerLedgers and ConsensusServers. For convenience, that has been described as the BridgeTransport layers of the hierarchy. It could be assumed that messages could be routed through the BridgeTransport hierarchy through multipath packet switching through the Internet cloud, and perhaps best described as a mesh topology, but other topologies might be possible or preferred depending on the consensus topology. For example, a regional tree structure formed to maximize regional sovereignty might also prefer more control over the lower levels of the hierarchy, including the BridgeTransport layer. Again, it probably makes sense to consider consensus policies and topography first. While those decisions have yet to be fully resolved, the topological terminologies may help describe the issues and at least help define the bounds of the problems, “conflicts,” or dichotomies. The definitions of the BridgeTransport layers are then more likely to fall into place.

The “preferred topology” for the ConsensusServer grid may lean toward a relatively complex mesh structure of 13 servers, with about 20 major auxiliary functions connecting to it. This structure will probably be designed and operated by BlockFrame® but should be governed by NGO policy.

Tasks related to Research Question 3 (RQ3) have used SysML and MP/G models to provide detailed and comprehensive definitions for proposed distributed ledger and EPS™ secure provisioning system architectures and behaviors, and to describe how functional capacities of the BlockFrame® system-of-systems enterprise can be applied to commercial business processes, to verify and validate system behaviors, and to organize externally-imposed system requirements such as CMMC and FERC guidelines, and to identify components and processes that satisfy

those requirements. However, more research and actual applications in this area will continue to help demonstrate the utility of NFTs, the carbon-neutral blockchain, digital supply chains, transactive energy services, and other use cases and capabilities. All of these must be considered to create a viable, robust, and sustainable commercial marketplace.

7.2. Philos™ Marketplace Operations

To realize an effective Marketplace for BlockFrame® capabilities, we design and deploy blockchain supply chains, governance, and security common units/common functions to create a system-of-systems infrastructure to increase the security and trust levels of government and private IT systems. To do that, we need to understand and satisfy two key requirements. First, there must be a robust and trusted way to mitigate systemic risks in environments with negative emergence. Second, there must be robust and sustainable financial and economically sustainable processes to maintain and grow the system-of-systems.

The function of BlockFrame® technologies to enforce trust and security uses an algorithmic approach to monitor, govern, and differentiate trust scores from all participating users, and to incentivize or dis-incentivize users through the decisions of their peers and while in competition to add blocks to the Philos™ blockchain. A dashboard tool uses statistical surveillance to constantly monitor the relative behaviors of each real-world user of the Philos™ blockchain, the statistical trends of their behaviors, and overall increases or decreases in their trust scores over time. This work results in an experimental assessment of the effectiveness of the ESP/Philos™ system design and can be used or modified to manage and govern the overall system-of-systems.

Considering Risk Factors in Philos™ Blockchain Development and Operationalization

When operational, Philos™ is expected to “optimize” trust and security. Trust can be defined in different ways, but an effective design must be able to enforce consistent rules throughout the system-of-systems. The main question for calculating the governance algorithms might be, “How do you measure and “trust” data when many data, personnel, hardware, software, and processes should initially not be trusted due to a “zero-trust” cybersecurity strategy? The key to a zero-trust strategy might be to assume that trust starts at zero, but then it can be increased and over time only as all sub-systems, designers, and users cooperate to add to the trust levels, and are willing to do so. Since the goal of the Philos system should be to optimize trust and security, it is convenient to use optimization terminologies of an “objective function,” and “decision variables” and “constraints” to describe the system-of-systems goals. To maximize trust, it first needs to be defined. Trust is currently under extensive academic discussion,

but identifying and selecting specific metrics has not been definitive and sometimes meets academic resistance. When discussing trust, opinions are inevitably expressed about how closely trust is defined or related to privacy and security.¹²³ While there are technical, legal, and cultural arguments about that, those will not be discussed in this section, but the dynamics of trust, privacy, and security are covered more extensively in Section 8.4., Digital Privacy: Technical, Legal, and Legislative Aspects, Section 8.5., IEEE Digital Privacy Model, and Section 8.6., Bridging the Technologist/ Policymaker Divide in Digital Privacy.

This discussion first presents six considerations of ways to subjectively think about how “trust” might be used and applied during system development. That is a critical design question that should be addressed and “optimized” to be effective. None of these considerations are “wrong,” but none of them alone fully satisfy the need for a commonly applied algorithm for the complex, conflicting objectives of blockchain governance. It is possible that an objective function to “satisfice” conflicting objectives might be the most acceptable method. Even if the Philos™ blockchain can successfully operate with a single relatively simple algorithm, further research in this area should be conducted to help select the best optimization process.^{124 125}

1. ***Trust = P (Trust)***. The baseline algorithm currently being considered for Philos™ engineering tests is a two-factor beta scoring distribution which results in single scalar values to rank blockchain priorities. Representing “trust” by a probability (0.0 to 1.0) is relatively simple but might not capture the conflicting real-world dynamics of conflicting user priorities and/or system governance.
2. ***Trust = 1/Risk***. If other risk analysis algorithms more accurately represent or mitigate risk within the users’ systems, would those be more appropriate? If so, would “trust” be better measured as the inverse of risk, and could a risk analysis approach be more effective due to its more extensive body of research?
3. ***Trust = 1/ Liability Insurance (\$)***. Cybersecurity insurance is currently a major growth industry because of the subjective perception that it might lower financial risks or increase trust. This has not been proved, and there are a lot of unanswered questions about how payouts would be made, or whether there could be some sort of actuarial table to help define the “cost” of trust or even defining whether

¹²³ While there are technical, legal, and cultural arguments about that, those will not be discussed in this section, but the dynamics of trust, privacy, and security are covered more extensively in Section 8.4., Digital Privacy: Technical, Legal, and Legislative Aspects, Section 8.5., IEEE Digital Privacy Model, and Section 8.6., Bridging the Technologist/ Policymaker Divide in Digital Privacy.

¹²⁴ One might argue that the following definitions should probably be considered “proportions” rather than equalities, but we assume that “equalities” are easier for this discussion.

¹²⁵ Gorog, C., and Boulton, T. 2018. *Solving Global Cybersecurity Problems by Connecting Trust using Blockchain*. University of Colorado in Colorado Springs.

things like cyber breaches might be insured for the consequences of just one breach, or for millions of consequences from just one breach. An interesting concept is whether using blockchain technologies can be used to “sell” more reliable “trust” than cyber insurance can “sell” more reliable mitigation of “risk”? From an economic point of view, should they be inversely proportional? If not, the insurance might not be actuarially sound. For the State of Colorado, this might be a useful study since the state already committed to purchasing cyber insurance, probably without adequate analysis.

4. ***Trust = “Credit Score.”*** Can “trust” be measured like a dimensionless “credit score,” that may only follow relative and subjective trends over time, or could that “rating” of each participating user be adequate by using their “relative” trust?
5. ***Trust = f (Trust (t)).*** Is trust truly a function of time that builds up only after practical proofs of experience? That may be true, but for trust within government or business systems, is that conclusion objectively quantifiable, or would it be based only on subjective perception?
6. ***Trust = f (Immutability, Non-Repudiation, Distributed Storage, Consensus/Proof of Work/Proof of Stake).*** If all these factors are inherent beneficial characteristics of blockchains/distributed ledgers, should they all be considered and scored as part of a multiple-criteria decision analysis?

For discussing trust, we would observe that all these considerations are useful, but none of them may be adequate to be used as directly quantifiable measurements that would be needed for accurate tools or algorithms within digital optimization models. More measurable and precise metrics are needed. However, for the design of a practical blockchain consensus algorithm, a critical question is how “perfect” does that algorithm need to be? As the “referee” of trust or risk within a digital system, it needs to be perceived to be as “fair” and mathematically consistent as possible, but as a digital representation it is still only representing subjective values. By breaking down the consensus algorithm into examinable components, a governance structure can better observe and control those values with the understanding that “perfection” might not be possible, and there will probably be a point of diminishing returns in trying to design too much into the algorithm itself. The best design goal might be to provide the governance organization with the best digital tools to monitor and control the subjective factors.

Ten Representative Risk Factors

A list of all possible trust/risk factors for the whole system-of-systems would be extensive and possibly unworkable. To focus on this discussion, ten representative factors are listed below to illustrate the process. The first

five represent general managerial risks. The second five are specific “categories of trust” to be used within the Philos™ blockchain system and designed to be applied together to collectively increase separate aspects of trust – and thereby decrease risk – over time.¹²⁶ If more “trust” capabilities translate into less risk, then mitigating specific sources of risk should also be able to mitigate the most important factors.

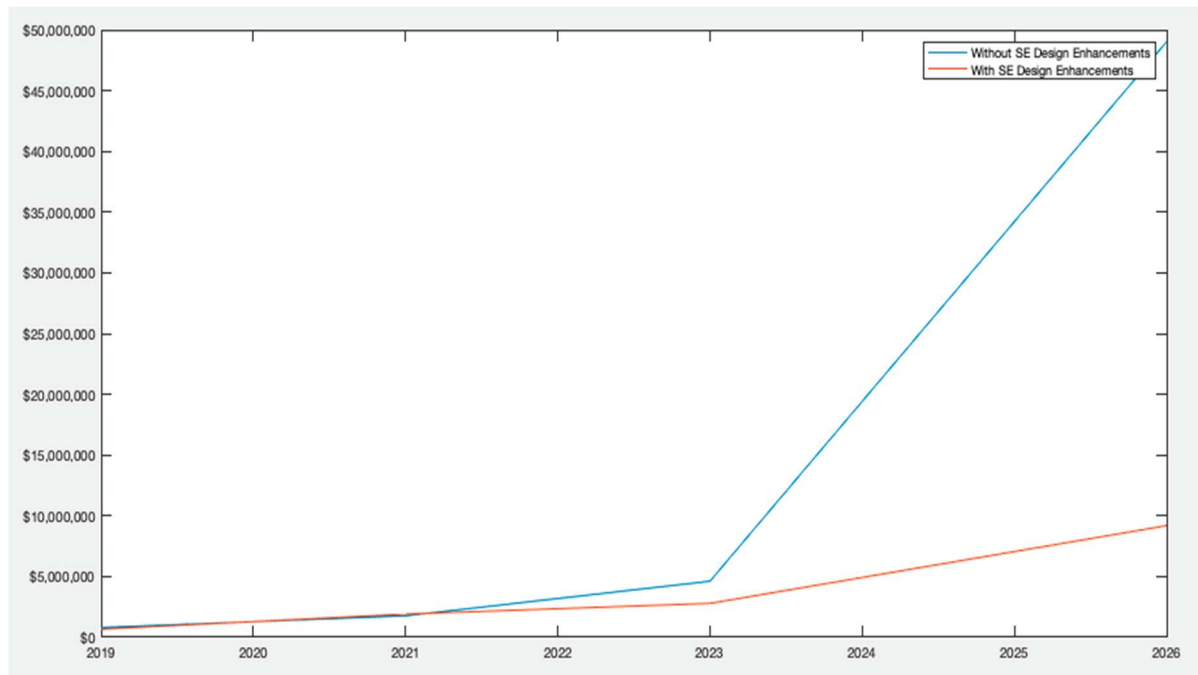
The technical risks are specifically selected to decrease the risks of confidentiality and integrity. While in information security, the factors of confidentiality, integrity, and availability (CIA) are typically considered most important, but for the purposes of blockchain risks, confidentiality and integrity are relatively more important than availability, which is assumed to be of minor importance within a closed blockchain system. Instead, other factors have been identified as more measurable and appropriate for the BlockFrame® system-of-systems. These include privacy, authentication, integrity, and non-repudiation (PAIN). A fifth factor, deprivation trust, or “experienced deprivation” is also included. These technical risks are explained in more detail below.

Five Representative Managerial Risks

1. **Funding/cost risk:** Assuming a startup with zero cash on hand, research and development funding commitments from users were estimated almost immediately. As an example, the probability that revenue can be secured for all for initial system, subsystem, and component R&D, production, and operational cost goals to meet all research, development, and integration to support an estimated 3 users by 2021, 10 users by 2023, and 100 users by 2026, and then sustain a positive cash balance of 5% of the revenue. Without adequate upfront funding, the project would not be able to attain a point of long-term sustainability from other funding sources. Figure 7.3 shows notional cost data to exemplify some of the cost and risk considerations that could be incurred for a typical business startup. Cost estimates in Figure 7.3 were assumed. Figure 7.5 shows also notional growth potential. The main point is that these risks should follow a quantitative and predictable risk pattern and should be scrutinized over time to identify and mitigate funding and cost risks, and to make adjustment decisions as necessary.

¹²⁶ Gorog C. and Boulton T. 2018. *Solving Global Cybersecurity Problems by Connecting Trust using Blockchain*.

Notional Cost Estimates



YEAR	2019	2021	2023	2026	
Cost of Finance, legal, insurance	17,000	96,000	96,000	2,512,000	=
Cost of Security administration	38,000	230,000	478,000	1,388,000	+
Cost of Test & evaluation	40,000	175,000	444,000	1,319,000	=
Cost of System administration	18,000	108,000	244,000	1,032,000	+
Cost of Marketing & sales	20,000	94,000	279,000	721,000	=
Cost of Code/patch delivery	6,000	80,000	194,000	683,000	=
Cost of Engineering *	500,000	1,000,000	750,000	500,000	--
Cost of Crypto delivery	6,000	58,000	135,000	441,000	=
Cost of Production	1,450	1,270	3,530	252,650	---
Cost of Energy	6,000	6,000	29,000	113,000	=
Cost of Education/training *	9,000	50,000	100,000	100,000	--
Cost of Installation/integration	200	1,020	9,410	67,600	---
Cost of Recycling	0	3,000	20,000	50,000	=
Cost of User Handbook & Training materials *	8,000	10,000	10,000	10,000	-
TOTALS	669,650	1,912,290	2,791,940	9,189,250	--

* Lower Estimates for BCDC, but removed significant unknown state system costs

Year	2019	2020	2021	2022	2023	2024	2025	2026	Total
Annual Cost \$M	0.7	1.3	1.9	2.4	2.8	4.9	7.0	9.2	30.2

Figure 7.3. Notional Cost Data¹²⁷

¹²⁷ Data from: Lambert K and Bishop V. *Engineering the Philos™ Blockchain System*. Final report, Engr 530. Colorado State University. 2019. (Actual future data may vary significantly from these initial estimates.)

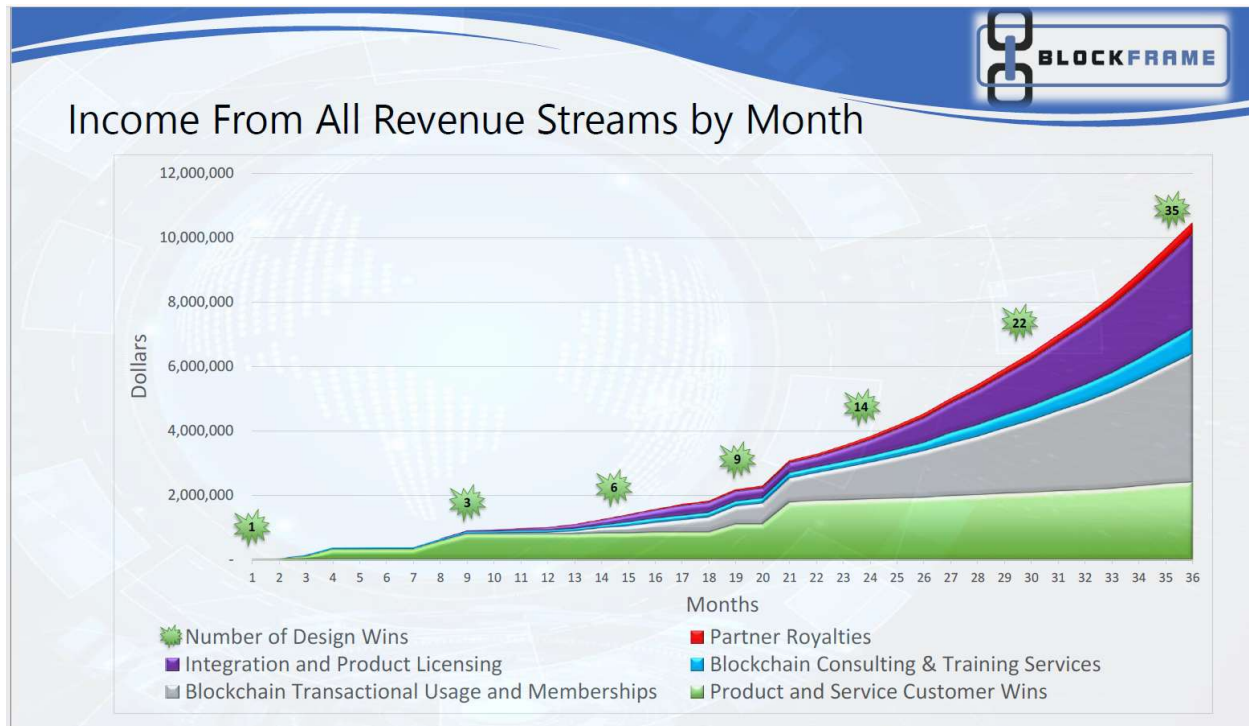


Figure 7.4. Projected Income from All Revenue Streams by Month

2. **Integration risk:** Many users already have a critical need for the supply chain management and security that can be provided by Philos™, but they must be able to integrate it with their own unique systems, some of which may be legacy equipment. What is the likelihood that host (government or business) servers and applications are fully compatible with system hardware, software, and procedures? If they are not, or if Philos™ is incompatible, the system-of-systems will either not function, or performance will be degraded.
3. **Schedule risk:** Funding, cost, and integration timelines all threaten the completion of production milestones. What is the probability that end users will be able to operate the final products within contracted time frames, estimated here to be for 3 users by 2021, 10 users by 2023, and 100 users by 2026? Failure to do so will increase risks to state and federal programs in an environment of rapidly increasing cybersecurity threats. In practice, we found that these initial estimates were not realized as fast as we originally forecasted. We attribute that primarily to the unexpected effects of COVID, which delayed many development efforts and capital investments by at least two years.
4. **Threat/security risk:** Hostile cybersecurity threats are ubiquitous and growing rapidly. What is the likelihood that the security of Philos™ computer hardware, software, and procedures are secure from hostile

intrusion and misuse? What then are the likely outcomes to mitigate the risks of end user systems once they are integrated with Philos™?¹²⁸

The most valuable assets of the BlockFrame® company, the potential for growth and profitability notwithstanding, is the value of its intellectual property. Therefore, it should also be seen as the highest potential risk factor. While that is not a factor that would be considered or included within Philos™ operations, the theft or compromise of intellectual property is considered a major financial and legal risk. BlockFrame® has made extensive efforts to mitigate those risks by establishing several US and international patents. Figures 7.5 and 7.6 show those graphically.¹²⁹

¹²⁸ According to Lloyd's of London, it is estimated that cyberattacks cost businesses \$400 billion globally per year. In 2014, one billion records were compromised." "... the founder and head of XL Catlin, the largest insurer in Lloyd's of London, said cybersecurity is the 'biggest, most system risk' he has seen in his 42 years in insurance." The source also qualitatively describes the likelihood of a "major breach cascade," "... which will eclipse what has been seen to date." (SOURCE: Geer, D and McClure, S. 2016. *How to Measure Anything in Cybersecurity Risk*. Wiley, Hoboken, New Jersey. pp 9, 11.).

¹²⁹ The most current patent information can be accessed at <https://www.blockframetech.com/patents/>.

BlockFrame® Patents Portfolio to Product Relationship

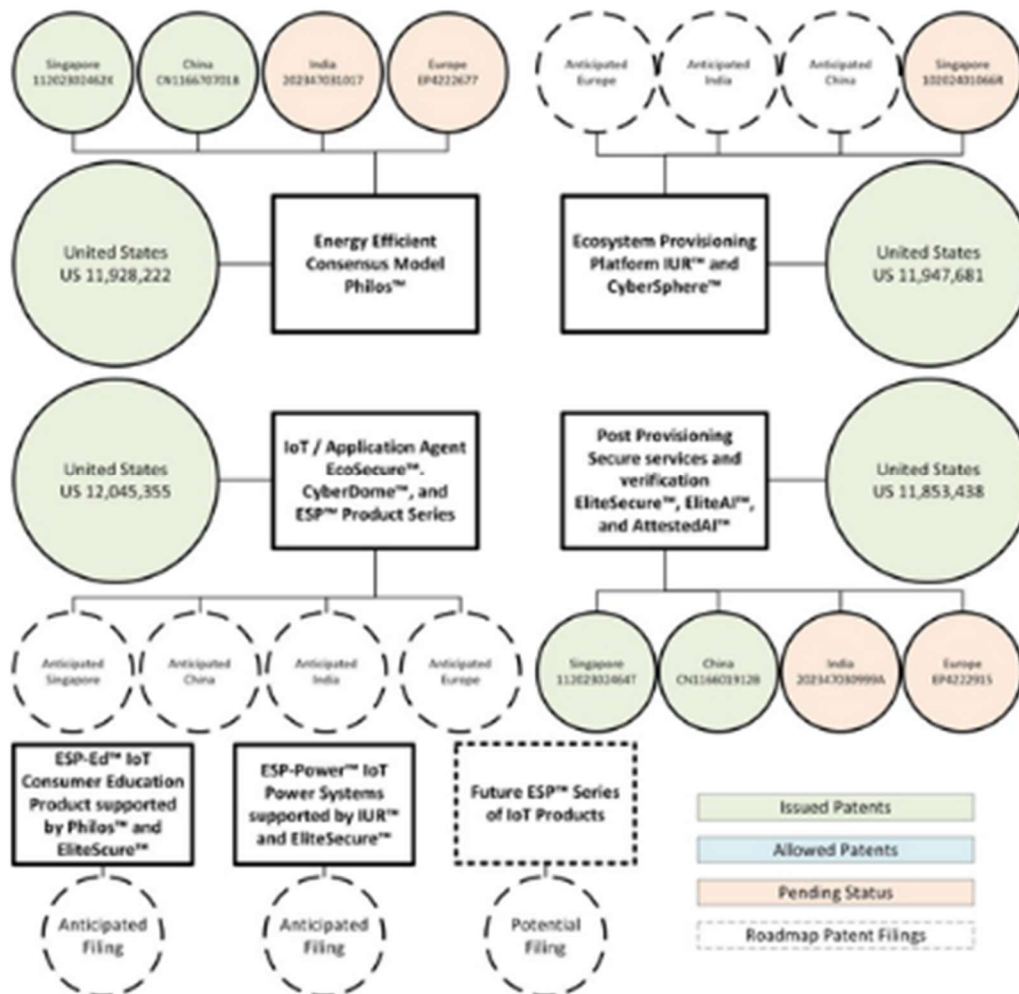


Figure 7.5. BlockFrame, Inc., Patent Portfolio and Product Relationship

Our portfolio of patents enables a position of strength to empower future proof business models protecting our core technologies. Current patents on four technology components are issued in the United States and are extended to five of the major economic centers of the world. These patents are pivotal to our innovation and enable the protection of our customers' ability to control the aspects of their products in their individual markets. Our customers can leverage exclusivity if desired in a particular market segment or industry to position them over their competition.

Energy Efficient Consensus Model

This protected operation implemented by our Philos™ Distributed Ledger employs the only patented blockchain consensus model on the market. The consensus model does not utilize any mining or high computational algorithms in its operation thus enables the ability to provide the more advanced capability over any competition blockchain while eliminating mining.

Ecosystem Provisioning Platform

The unique provisioning of security credentials in a protected manner utilized in our CyberDome™ and CyberSphere™ products is the first of a kind trust management for cyber-physical and cryptographic certificate and key distribution using distributed ledger. Many recent future looking technology publications including Gartner, IEEE Industry Advisory Board, and AI Articles cite that trusted systems using blockchain are a top need for the future of the data revolution. We are decades ahead of this trend and with this patented technology and are the only providers able to implement distribution of trusted components using blockchain.

IoT/Application Agent

Our secure provisioning of trusted identification components enables each end system, Internet of Things (IoT) device, mobile app, or distributed application to effectively become a blockchain wallet. Every data item received from or produced by the end node system is uniquely identifiable containing EtiSecure™ metadata. This patent protects the on-system agent operations and enables the containment of base components in each end node while also mapping operation across the supply chain enabling customers and product owners to operate privately and securely over the entire product lifecycle from cradle to grave.

This patent enables options for legacy system integration using software solutions, but also provides highly secure cyber-physical components. This creates a unique position enabling IP protection on the physical level, through logistical distribution across the logistics of deployed products into the application level, and business tiers. Our customers are enabled with granular control to empower top-down individualized asset business models across large amounts of deployed products.

Post Provisioning Services and Verification

This patent protects processes which enable our EtiSecure™ lineage from data production through distribution, and storage. Data items from any CyberSphere™ entity can be trusted and this underlying trust used for clients to empower Quality of Service for applications, machine learning algorithms, or Artificial General Intelligence (AGI). Countless use cases can leverage these services and the underlying trust of each system, data item, or participant identity.

ESP (EtiSecure Provisioning) Family of Products

Several of the BlockFrame anticipated products that are targeted for end systems which leverage our family of services in a specific industry or consumer applications which are on our internal roadmap for future Patent filings. Some of these are set for internal release in the near term while other concepts are in early design stages. The ability to leverage each BlockFrame product once any of the others are in circulation provides the unique ability for our clients to gradually corner their industry or market with trusted operations that set them apart from their competition.

Figure 7.6. BlockFrame, Inc., Patent Portfolio and Product Relationship Explanations

5. **Logistics risk:** The ultimate risk to many end users is supply chain disruption, diversion, or denial of service from hostile cyberattacks to either their physical or informational logistics systems. What metrics describe normal logistics performance, loss of performance under attack, and mitigation of those risks due to the Philos™ system?¹³⁰ Each individual end user system could have different performance and risk profiles. For future estimates, we assume that 20% of user organizations are “small,” 70% are “medium,” and 10% are “large.” Philos™ is “indefinitely expandable,” and could address many problems, but estimating how major cybersecurity risks might be mitigated could be massive, complex, and slightly elusive. To approximate these risks in the future, it might be useful to consider samples of “typical” business scenarios

¹³⁰ For context: according to the United Kingdom National Cyber Security Centre, 81% of large businesses reported a breach, with the average cost of each breach of £600k - £1.15m. (SOURCE: <https://www.ncsc.gov.uk/static-assets/documents/NCSC%20Cyber%20Attacks.pdf>)

with typical cyberattack scenarios, approximate ranges of expected effects, and represent combinations of possible mitigation techniques. With the current dearth of actual data, simulated Monte Carlo estimates using parametric ranges of variables might be useful to “bound” the problem and to calculate expected variabilities.¹³¹

Five Technical Risks¹³²

6. ***Suppression Trust (TSP) Risk:*** “a blockchain transaction entry incorporating the resultant signature related to the proof of operation of a single privacy encoding cryptographic operation.” This is a quantifiable measure of trust related to cryptographic operations to prove privacy and confidentiality over time to reduce the risk of loss of confidential data.
7. ***Validation Trust (TVL) Risk:*** “a Blockchain transaction entry incorporating the resultant signature related to the proof of a valid single successful comparison of dynamically created vs. stored cryptographic digest creating an authentication sequence relating two entities.” This is a quantifiable measure of trust related to cryptographic operations to prove authentication and confidentiality over time to reduce the risk of unauthenticated threats from accessing the systems.
8. ***Reliability Trust (TRL) Risk:*** “a blockchain transaction entry incorporating the resultant signature related to the proof of operation of a single Data Identification operation, incorporating a digest, in relation to a given uniform resource identification (URI).” This is a quantifiable measure of trust related to cryptographic operations to prove data integrity and authenticity over time to reduce the risk of unreliable data within the systems.
9. ***Refutation Trust (TRF) Risk:*** “a blockchain transaction entry incorporating the resultant non-repudiation cryptographic transaction in relations to a Reliability Trust Transaction.” This is a quantifiable measure of trust related to cryptographic operations to prove data integrity and non-repudiation over time to reduce the risk of breaking previously agreed-to transactions.
10. ***Deprivation Trust (TDP) Risk:*** “a blockchain transaction entry incorporating the resultant Degradation information in relation to a Reliability Trust Transaction.” This is a quantifiable measure of trust related to

¹³¹ Pareek, M. 2012. *Using Scenario Analysis for Managing Technology Risk*. ISACA JOURNAL VOLUME 6, 2012.

<https://www.isaca.org/JOURNAL/ARCHIVES/2012/VOLUME-6/Pages/Using-Scenario-Analysis-for-Managing-Technology-Risk.aspx>.

¹³² Gorog, C., and Boulton, T. 2018. *Solving Global Cybersecurity Problems by Connecting Trust using Blockchain*. University of Colorado in Colorado Springs.

cryptographic operations to prove data availability and examinable degradation over time to reduce the risk of abnormally losing information that could be examined and corrected.

Taken together, these five technical risks can be used together to quantitatively assess the relative risks of each blockchain user, and therefore their overall level of trust. Those are then used in the Philos™ Marketplace to incentivize (or de-incentivize) users to work collaboratively and in their own self-interest to choose trusted partners to conduct smart contracts and to improve their ability to store their content on the blockchain. Because the use of these five trust/risk factors are the basis of the Philos™ blockchain/distributed ledger consensus model, they will be discussed further in that section. As a practical process, the digital data collected through individual transactions and consolidated through the process of constructing the blocks in the blockchain, can be viewed as a process of converting that data into a governance decision structure that can be observed and controlled by human intervention. That interface must consider human factors and can be viewed as a process of decision analysis. Following the “rules of thumb” of decision analysis, those factors should ideally be measurable, mutually exclusive, and collectively exhaustive. While that ideal might not be perfectly attainable, the five risk factors should be adequate to provide governance authorities with adequate capabilities to control the process. For example, the five trust/risk factors are specifically measurable and mutually exclusive but might not be perfectly collectively exhaustive. However, if they adequately represent the major factors of trust/risk that affect the Philos™ Marketplace, and can be effectively managed by good governance practices, they should be very adequate. It should also be noted that this is consistent with the philosophy of emergence theory in that in an environment of high negative emergence environment, the goal should be to mitigate the chaos to the point where it can be better controlled by known methods. We believe that the BlockFrame® system-of-systems can successfully do that through this defined process.¹³³

¹³³ While this section has addressed several aspects of trust and risk, and how they can be effectively managed through an algorithmic approach, a great deal more research was conducted on risk analysis on the BlockFrame® systems that is not essential to support this immediate research objective. That research will therefore be reserved and discussed in Appendix D.

7.3. Neutral Governance Actions

This section explains the capability of the BlockFrame® system-of-systems to collect, act on, and report data over time to evaluate the overall effectiveness and trust levels of the system-of-systems through a Neutral Governance Organization (NGO), International Alliance of Trust Chains (IATC).

Consensus Model/Trust Behaviors

Figure 7.7 shows actual test data from Philos™ B/DL servers, performance parameters, and trust parameters that increase and converge over time. Simulator testing shows that improvements in the collective levels of trust among users should inevitably increase over time. Some or most of the factors represent functional performance parameters but they can also be considered non-functional quality attributes (QATs) that are appropriate to track as quality metrics and behaviors over time. Examples are given in metrics like, “Determine Merchant Performance,” “Ownership Track Record,” and “Identify High Risk Items/Organizations,” that can monitor trends in marketplace QATs, and rapidly identify quality problems that might compromise trust within business transactions, users “gaming the system,” or other trends that could compromise the goals and performance of the trust chains. This also demonstrates the importance of the consensus algorithms, and the incumbent risks that might occur if those algorithms were compromised, inefficient, or possibly failing economic stress tests.

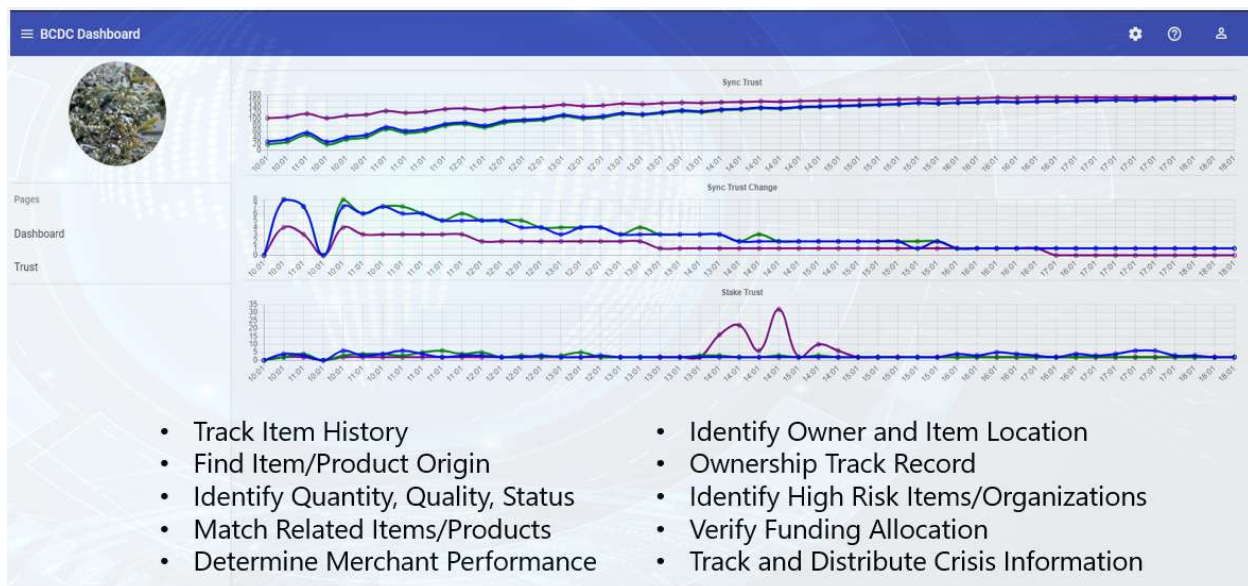


Figure 7.7. Philos™ Dashboard

Network Topography Considerations

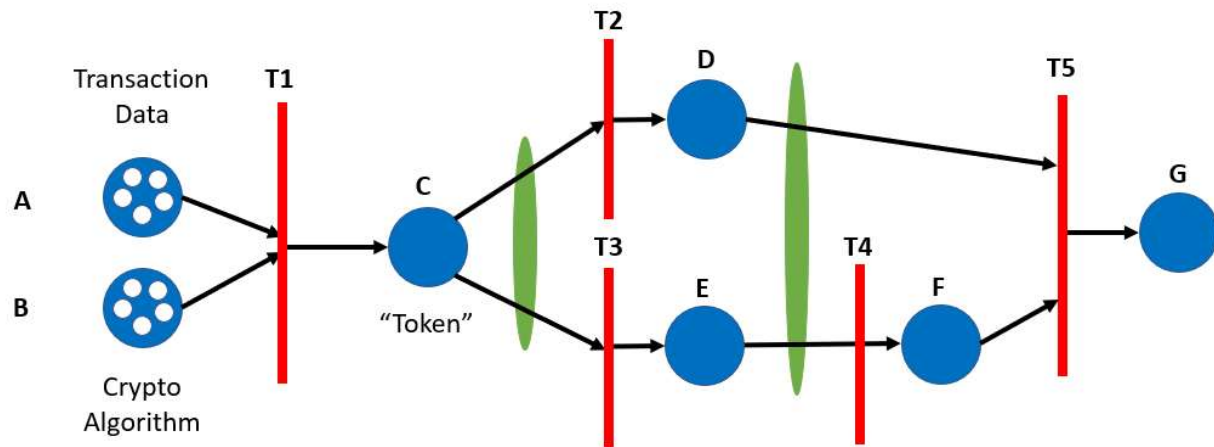
Two high priority behaviors that are performed by Philos™ include tokenization of service transactions, and economic/financial tokenization for blockchain services. The terminology and visualization capabilities of Petri Nets¹³⁴ make them a convenient way to discuss and describe both processes.

An initial discussion using Petri Net “basic graphics and text” is a good place to start, and for now discussion will be limited to that, but Petri Nets could also themselves be used as inputs into SysML modeling.

The first UseCase might be called, “TransactionTokenization.” In current popular business contexts, that would not be the same definition of tokens that are used in SysML but would follow the business standard of that industry. Their use of the term “token” describes a process to receive clear text information, and then to encrypt each data string into encrypted hash strings which we can call the “token”. Philos™ can do that using arriving input strings and encrypting them into “tokens” in a virtual sense. SysML should also be able to represent that process, at least to follow the path of the inputs and the resulting tokens, using a Petri Net-like logic, and modeling it using either Activity Diagrams or State Machines.

While this part of the process might be somewhat trivial, the next step could be novel. One problem with the current industry practice is that their “tokens” are typically held by a third party which might be trusted, or they might not – i.e., possibly introducing a third-party counter-risk from distortion, loss, or theft of data. With no other options available, users might feel compelled to use the third-party system, or not to do business at all. That’s where the option of using Philos™ can be useful. Instead of having to trust third parties, the tokenization transaction – which could represent a very wide variety of consumer or business transactions – would not just be recorded with the third party, but also recorded automatically on the Philos™ blockchain, making it encrypted, immutable, distributed, and a proof that is non-repudiable. Instead of only trusting a third party, the owner of the data, as a participant of the transaction, can also personally own the proof that the transaction took place because they have a proved digital copy of the token which can also be decrypted back to its original clear text. Figure 7.8 is an example of a Petri Net showing the principle of this Use Case.

¹³⁴ Herber and Borky, CSU “Engr 667: Session 8: Mathematical Network Models 2 and Pattern in Architecture Decisions.”

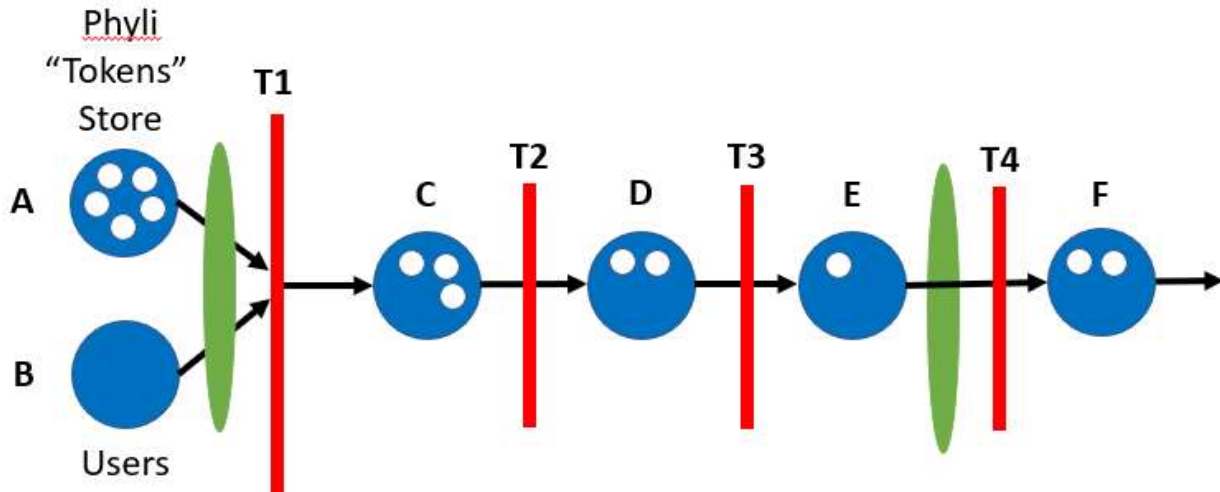


Place A is an unencrypted Transaction
 Place B is a unique cryptology key
 Transaction T1 encrypts the Transaction
 Place C is the resulting Token
 T2 stores the Token with a 3rd Party at Place D
 T3 stores the Token on the Blockchain Place E
 If the owner suspects a false Transaction, a “conflict”, T4 will retrieve and decrypt the Token
 Place F is the decrypted Token from the Blockchain
 T5 is a process to compare the 3rd Party Token to the Blockchain Token to Prove Trust
 Place G is the resulting Token with Proof of Trust

Figure 7.8. Notional Example of “Transaction Tokenization”

The second UseCase, shown in Figure 7.9, shows economic/financial tokenization for blockchain services, could be a key method for Philos™ sustainability through competitive and sustainable neutral accounting for various services. The concept would be to purchase non-currency tokens, called Phylia™ (plural), from central brokers for cash, purchased in dollars or other currencies, for Philos™ services depending on what each User needs. In all cases, BlockFrame® Phylia™ should be considered as “utility tokens” for internal System transactions, rather than as representing currencies. Controls by the Neutral Governance Organization (NGO) would set rates for blockchain services based on competitive market values and sufficient profitability to sustain the system. Users would buy enough Phylia™ for their individual needs. Since the NGO non-profit status might not allow it to conduct commercial transactions directly, a separate scope-limited for-profit corporation is probably necessary to handle the cash transactions to avoid conflicts of interest. When requests for services are made, accepted, and processed, the User’s account (or virtual “purse” or “wallet”) would be debited by the appropriate amount as true fees-for-service. For example, a successful request to secure a document, and to record that mutual transaction on the Philos™ blockchain

might cost the equivalent of five cents but would be denominated in Phylli™ for internal accounting purposes. Using Phylli as strictly internal utility tokens also distinguishes them from monetary currencies or cryptocurrencies that could be more subject to government regulation. As more transactions are requested, more Phylli™ would be spent by the Users. The proceeds from cash sales would help pay for System costs, including development and maintenance of software, hardware, installation, security, insurance, healthcare, salaries, and basically all System overhead to run the operational business. Since Philos™ never uses wasteful data mining for proof of work, a fee-for-service business plan would be proportional to the volume of User transactions. Since a major factor in the sustainability of the blockchain is to expand and maintain the number of voluntarily participating Users, distributed blockchain storage servers, and other factors, Phylli™ can also be used to discount services or repay distributed Users for providing blockchain storage space, or perhaps for enrolling new participating members. Regulating the Phylli™ value of services would be done by the central policies of the NGO. The NGO controls the Phylli™ through relative fee-for-service values to assure the Phylli™ do not have liquidity or inflationary extremes and would be incentivized to maintain stable pricing. High costs would lose customers, and low values would undercut System sustainability. Theoretically, a well-defined policy could be executed algorithmically and maintain stable pricing through automated control logic. Transaction debits and credits would be stored on the blockchain as transactions. Many more complex transactions like fees-for-service, or incentive credits for active participation could result in sustainable growth and well-resourced services.



Place A is the Phylli™ Store

Place B is a queue of Philos™ Users

A User decides to participate in blockchain services

T1 is a transaction to purchase three Phylli™

Place C is a qualified User with three Phylli™

T2 and T3 are transactions to spend one Phylli™ value for blockchain services

The User decides to provide a server for the storage of the Philos™ blockchain

T4 is a transaction to credit the User one Phylli™ for the User's storage services

Figure 7.9. Notional Example of Economic/financial Tokenization for Philos™ Services

These factors were discussed in general terms with the BlockChain Development Community (BCDC) on March 28, 2020, and demonstrated a positive interest in pursuing this methodology to describe, clarify, and visualize how “Phylli™ Economics” might work to meet the stated goals. Since that time, there have been many discussions to refine those concepts and continue development that are discussed in part below.

Comparing and Contrasting SysML Behavior Modeling

Examples from Selva,¹³⁵ the Petri Net¹³⁶ examples above, and others covered in this section, could be considered as heuristic decision-making processes and tools that can help guide thinking to reduce the complexity of problems by applying a series of techniques without the time and effort inherent in more formal computer models. However, they can also explore and set patterns that can flow nicely into SysML behaviors like Activity, Sequence, and State Machine diagrams used in more detailed computer-assisted models.

¹³⁵ Selva, D., B. Cameron, E. Crawley. *Patterns in System Architecture Decisions. Systems Engineering*, Wiley, 19, no. 6 (2016).

¹³⁶ Herber and Borky, “Engr 667: Session 8: Mathematical Network Models 2 and Pattern in Architecture Decisions.”

In the discussions about topography, decisions need to be made that could involve complex political issues and constraints that might not be possible or desirable to model quantitatively alone because of complex jurisdictional and legal boundaries. SysML can probably model most typical topographies, but selecting an acceptable topography to model complex networks will probably be an input to SysML, rather than an output. Once that topography is selected, it can probably be modeled in more detail using computer-assisted models. As with any systems engineering process, more than one alternative might need to be modeled prior to making a final decision, but if heuristics or computer modeling can “down select” options with a high degree of acceptance, it might not be necessary to use a complex model for something that is already obvious. On selecting the best topography for distributed worldwide consensus servers, for example, some of the biggest issues may involve political considerations more than network efficiency, so a mix of heuristics and computer modeling might make sense.

In the discussions of Petri Nets, two business models were examined that could motivate high demand and investment from Users. Knowing that there should be a high demand for the described services, simple and graphic relational models – like Petri Nets – could go further in educating potential Users about how their participation would work, rather than 100,000 lines of JSON code which might have the opposite effect. Blockchain operations are sometimes terribly misunderstood, even by self-proclaimed experts, but when simple graphical representations can tell a story about how it works from the stakeholders’ point-of-view, they might not need to know much more than that. Users and stakeholders certainly do not need to know how all the black boxes work. Design engineers, on the other hand, need to know the details to make systems work, but might care less about how the stakeholders or Users would use them. That might sum up the “contrasts.” The top-down, bottom-up system engineering process needs both, and should use both to find the appropriate decision sets.

Among the topological patterns presented in Selva¹³⁷, including the “partitioning pattern” and the “connecting pattern,” the partitioning pattern is not very interesting at this point because the keystone of the Philos™ design is already based on distributed blockchain resources. The connecting pattern, however, could be of great current interest to provide a set of common terms and patterns for discussion about the consensus model. While the claim that six canonical patterns of architectural decision analysis methods can form a “collectively exhaustive” set seems overly-simplistic, it is probably correct that you can do a lot of work with a hammer, a saw, a wrench, a file, a paintbrush, and a ladder – i.e., a limited set of six tools. Having a well-vetted and well-organized toolkit of heuristic

¹³⁷ Selva, D., B. Cameron, E. Crawley. *Patterns in System Architecture Decisions. Systems Engineering*, Wiley, 19, no. 6 (2016).

decision theory methods is nice to have. Having 35 years of experience is nice too, and Selva's article seems to capture some very useful patterns from the long experience of others. The partitioning and connecting patterns stood out as being the most relevant to the Philos™ architecture, but the connecting pattern stood out as being a more immediate need.

In our example of trying to discover connecting patterns for the Philos™ ConsensusBlock model, the next step might be to compare all the topological options and attempt to use the “downselecting” pattern to find options that dominate others and eliminate the weaker topology options from consideration. Not all the relevant factors have been discussed yet, but heuristic patterns certainly might help in those decisions.

Major decisions using the combining, assigning, or permuting patterns do not immediately seem useful as decision tools for decisions that need to be made for the Philos™ Enterprise in the near term. Some decisions, like the configuration of cybersecurity components within SystemContainers probably need more consideration, but it is not clear which patterns might contribute to those decisions. Personnel selection for a mix of effective skills sets among future corporate employees is likely to be more important in the future, but again it is not clear whether these decision patterns could be useful yet.

Problem and Background

The Philos™ Blockchain/Distributed Ledger (B/DL) is designed to operate on a fee-for-service basis. Reasonable prices for Users must consider fair costs to build, sustain, and maintain the B/DL, and in some cases use embedded Eco-Secure Provisioning™ Integrated Logistics Chips (ILCs), to provide trusted cryptographic data transmission and storage services, while keeping costs low enough to attract new customers. Components of the Philos™ Blockchain are now being tested, but to create a usable Service Level Agreement, an initial estimate of the cost per service must be included. That needs to be determined for the system to work. This research is an initial “cold start” to begin to calculate those costs, in parallel with work on other time-urgent components.

Objective: to identify a reasonable estimate for the initial dollar price of Phylia™ – which are virtual utility tokens used to pay for Philos™ blockchain smart transaction services, and for use with other services within the Philos™ Enterprise.

Assumptions and Definitions:

- A Phylia™ (plural Phylia™) is a virtual pre-paid token-like programming object, that is limited in its function to pay for services within the Philos™ B/DL at the Peer Ledger level.

- The term, “virtual utility token,” is similar to the Technopedia¹³⁸ definition: “A token is a special frame that is passed from node to node around a ring network. When it gets to a node that needs to transmit data, the node changes the token into a data frame and transmits it to the recipient.” For the Philos™ Phylī™ instantiation, the System is not strictly a ring network, but it is a *closed* network and operates only within Peer Ledgers networks where the Phylī™ tokens represent a common measurement of value to pay for other Services accessed through the blockchain. In this case, “virtual token” is *not* intended to represent a cryptocurrency.
- For this research example only, we try to determine an “optimal” value of one Phylī™ to pay for exactly one use of the Philos™ Blockchain to successfully process one transaction into a blockchain block. This simplification is made for mathematical convenience to calculate the approximate corresponding US Dollar value of that fee-for-Service “administrative fee”.
- In practice, Phylī™ will have to “float” slightly in “value,” but short-term dynamic changes should be controlled in the long run to return to a central-tendency dynamic stability. That dynamic will not be examined extensively in this report, but a simple explanation is that minor instabilities should normally be corrected within expected ranges by the consensus algorithm, while unexpected excursions might have to be directly monitored by the NGO, and human interventions and corrections might need to be made.
- Unlike other blockchains, Phylī™ are not “mined” as “proof of work,” or “proof of stake.” They are designed to manage a common Service-fee value structure and are not planned to be used as an exchangeable cryptocurrency outside the blockchain either for speculation or profit. They are primarily used to account for the value of internal transactions to charge Users fees-for-service for completed transactions. Those transactions would include the overhead costs of operating the blockchain infrastructure, the cost of creating new blocks, and paying for the cost of other services desired by the Users.
- Many types of transactions for User Services could be purchased for exchanges within Philos™ Blockchain transactions. It is assumed that the prices for those Services will be pre-negotiated and pre-

¹³⁸ Technopedia. <https://www.techopedia.com/definition/4140/token>.

paid through third parties, but within the Blockchain they are accounted for by exchanges of Phylia™. To create a uniform definition, the values for performing those non-Philos™ Services are generically called “Stamps,” which will be denominated in Phylia™ when traded within the Philos™ B/DL. For example, suppose a Philos™ Blockchain User established a Service Level Agreement (SLA) with the State of Colorado Department of Regulatory Agencies (DORA) to register Physicians’ licenses for \$30 each, which would be the cost of one Physician License Stamp. The User would also enter an SLA with Philos™ administrators to set the terms for their use of the blockchain. If the cost to automatically process each license and permanently record that transaction on Philos™ costs a common fee of \$2 USD to pay for the Philos™ Service (here the equivalent of 1 Phylia™), the total cost of that transaction would be \$32 dollars, or 16 Phylia™. Philos™ would retain one Phylia™ for their Service, and 15 Phylia™ would account for the “pass-through” Service with the Colorado DORA. The User’s “account” or “credit” on the Blockchain would then be decremented by 16 Phylia™.

- In practice, Phylia™ can also be used to pay back System Users for the use of computational power or storage on their own servers, but for simplicity those factors will not be considered here in detail. If they are included later, they could be considered within the overhead cost of running the blockchain and accounted for in that way.
- The cost of each Phylia™ would normally be paid for in US Dollars through an exchange broker or mechanism, and probably a separate private company to avoid conflicts of interest. Other currencies might also be used to purchase Phylia™, subject to law, regulations, and in compliance with sovereign states, countries, or international norms. Whatever currency a purchaser uses, the conversion into Phylia™ is important to allow the use of a common pre-determined value set for all transactions within the blockchain.
- Discounts or premiums on the price of Phylia™ or other rules might be considered to account for various levels of the quality of user trust, or to incentivize the long-term use of the blockchain. It is assumed that it will not be considered or modeled as part of this research.
- For this research, it is assumed that a “fair price” of a Phylia™ will equal the “optimized” minimum cost of a fee-for-Service to use the Philos™ Blockchain. The estimated cost of Philos™ Services that should be considered could include:

- An estimate of the overhead for business operations, the average annual cost per employee parameters includes average costs for salary, infrastructure/rental space, server/local network/ADP costs, local data storage, taxes, energy, education & training, research and development, test and evaluation, system management, security management, health and liability insurance. Any of these parameters may vary within the model or could be changed parametrically to conduct “what-if” analyses.
- The total per capita cost for “full-time equivalents” is calculated to be \$95,000, as shown in Table 7.1 “Parameters”.

Table 7.1. “Parameters”

Parameters	
Nodes per Employee per Year =	1,000
LICs per Employee per Year =	10,000,000
AVG Salary per Employee per Year =	\$50,000
AVG Building Rent per Employee per Year =	\$10,000
AVG Server & ADP Cost per Employee per Year =	\$10,000
Taxes Workers Comp =	\$15,000
Education/Training =	\$5,000
Insurance =	\$5,000
Total Assumed Annual Cost per Person =	\$95,000

- For this research, it is assumed that a 20% premium will be added to the cost of the Phylia™ fee to account for corporate profits, reinvestment back into the blockchain, and for some margin to mitigate risks. (I.e., margin for “unknowns” that might occur in fielding of the new technologies and Use Cases.)
- After all the above costs are considered, fee-for-Service costs would be capped to incentivize low, competitive costs for all Users. Even if there are no existing competitors to Philos™ for these Services, high costs could encourage new solutions and increase competition, so it will just be assumed here that the Users would not be charged arbitrarily high premium rates.
- All Users would agree to Service terms through Service Level Agreements.

- Purchase or sales of Phylit™ will be assumed to be restricted to Services for member Users only.
- Governance, including the determination and regulation of costs and services would have oversight from the Neutral Governance Organization (NGO), which would have authority to conduct audits, set standards for the economic stability of Phylit™, and to select and approve algorithmic economic controls to include dynamic stability controls on Phylit™ to help prevent Users from gaming the system.

Objective Function: To minimize the US Dollar cost of Phylit™, subject to the costs to support each Use Case Scenario will be based on additive costs estimates as explained in Assumptions and will use the following assumptions and calculations to estimate the operational costs in US Dollars.

- One administrator can manage a maximum of 1,000 “Nodes” per year, such as System Containers operating as points-of-service.
- If Logistics Integrated Circuits (LICs) are included in a Use Case Scenario, it is assumed that one employee can manage a maximum of 10,000,000 networked LICs.
- To calculate the necessary number of employees, those surrogate planning factors are used to represent the approximate workload per employee.
- If a Use Case requires the use of a Secure Classified Information Facility (SCIF), it is assumed that \$1 Million USD would be required as a one-time cost for that Use Case. It is also assumed that SCIFs cannot be shared with Users for other Use Cases to prevent co-mingling of classified data centers.
- Growth of User participation will be represented over three “Epochs” as described in Part B. Over time, the addition of each Epoch will be additive. The purpose of dividing the Use Cases into Epochs is to represent the growth of various Use Cases over time, and to determine whether and how that growth affects the routine costs of transactions, and therefore the fee-for-service, as measured by the cost of the Phylit™.
- Total cost and resource constraints will *not* be used for this research. That information is not currently available, but this also allows the system to expand to its upper bounds based on using all personnel calculated to perform the tasks. When more information and relationships are better known, more realistic cost constraints can be added. Since the goal for this research is only to compare some

hypothetical Scenarios and time slices to compare different fees-for-Service, that lack of realism may be irrelevant to meet this goal.

Table 7.2 represents eleven highly simplified Eco-Secure Provisioning™ and Philos™ Blockchain/Distributed Ledger Use Case Scenarios. These are notional data only to demonstrate differences in scaling considerations and dynamics. Although all these Scenarios might represent realistic Philos™ Use Cases, the specific numbers of Nodes, Transactions, SCIFs, LICs, and Crypto Key Rolls have been selected to represent totally hypothetical stress points to highlight extreme system behaviors for discussion, and to determine whether the costs per blockchain Service instances are very similar or highly dissimilar.

Table 7.2. Use Cases and Characteristics

Table 10. Use Cases and Characteristics	Nodes	AVG Trans/ Node/Yr	AVG Trans/Yr	BF SCIF?	LICs	Provisions/Yr	AVG Rolls/Yr	Epoch
CDE teacher licenses	6	365	2,190	NO	0	0	0	2020-2022
DMV, Voting Records (DOR/DOS)	6	365	2,190	NO	0	0	0	2020-2022
State Medical Licenses (DORA)	6	365	2,190	NO	0	0	0	2020-2022
COANG	18	100	1,800	YES	50,000	365	18,250,000	2023-2025
Missile Defense Agency	36	105,120	3,784,320	YES	50,000	365	18,250,000	2023-2025
Representative IoT Provisioning	36	12	432	NO	10,000,000	12	120,000,000	2023-2025
Utility company	6	105,120	630,720	NO	500,000	12	6,000,000	2023-2025
FEMA	36	365	13,140	NO	100,000	12	1,200,000	2026-2029
AF Space Surveillance System	36	105,120	3,784,320	YES	100,000	365	36,500,000	2026-2029
F-35 Air Force Supply Chain	36	365	13,140	YES	50,000	365	18,250,000	2026-2029
TRANSCOM	36	365	13,140	YES	100,000	365	36,500,000	2026-2029
Total	258		8,247,582		10,950,000	1861		
Epoch (Cumulative)								
2020-2022	18		6,570	0	0	0	0	2020-2022
2023-2025	114		4,423,842	2	10,600,000	754	162,500,000	2023-2025
2026-2029	258		8,247,582	3	10,950,000	1,107	254,950,000	2026-2029
Table 11. Assumed Costs by Use Case Scenario	Number of BCDC FTE	Annual Employee Costs	Annual Cost Plus 20% Profit and Risk	SCIF COSTS	Total Overhead Costs/Year	Avg Trans/Yr + Avg Provs/Yr	Average Cost per Phylli	Epoch
CDE teacher licenses	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
DMV, Voting Records (DOR/DOS)	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
State Medical Licenses (DORA)	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
COANG	0.02	\$2,185	\$2,622	\$1,000,000	\$1,002,622	18,251,800	\$0.0549	2023-2025
Missile Defense Agency	0.04	\$3,895	\$4,674	\$1,000,000	\$1,004,674	22,034,320	\$0.0456	2023-2025
Representative IoT Provisioning	1.04	\$98,420	\$118,104		\$118,104	120,000,432	\$0.0010	2023-2025
Utility company	0.06	\$5,320	\$6,384		\$6,384	6,630,720	\$0.0010	2023-2025
FEMA	0.05	\$4,370	\$5,244		\$5,244	1,213,140	\$0.0043	2026-2029
AF Space Surveillance System	0.05	\$4,370	\$5,244	\$1,000,000	\$1,005,244	40,284,320	\$0.0250	2026-2029
F-35 Air Force Supply Chain	0.04	\$3,895	\$4,674	\$1,000,000	\$1,004,674	18,263,140	\$0.0550	2026-2029
TRANSCOM	0.05	\$4,370	\$5,244	\$1,000,000	\$1,005,244	36,513,140	\$0.0275	2026-2029
							Averages =	
Total	1	\$128,535	128,536	\$5,000,000	5,154,242	263,197,582	\$0.1047	

Assumptions:

- “Nodes” as used here are entities that have a point-of-service to do Philos™ transactions. They are listed here with a notional average number of transactions per node per year, to estimate the number of Philos™ Service fees for transactions per year. For this research it is assumed that each Node represents a Philos™ Systems Container with either a Brokering Agent or a Peer Ledger being used as a customer point-of-service. For simplification, is further assumed here that each Node may serve multiple

customers, but the number of Nodes associated with each Use Case Scenario will represent a total level-of-effort to estimate average system costs.

- If a Use Case requires the use of logistics integrated circuits (ILC), the cost of production and base provisioning of the ILCs themselves is assumed to be paid as a cost by BlockFrame, Inc. and not specifically included here as a cost of blockchain operations. The cost of the specialized provisioning of ILCs by Users will also be absorbed as a cost by the individual Users, outside of the cost of blockchain operations. In both those cases, for security purposes, provisioning actions are recorded as transactions on the Philos™ Blockchain and would be charged a fee that will be counted here as part of operational costs.
- Cryptographic Key “Rollovers” for each Node involve consideration of various levels of security. For this example, we assume that Users processing highly classified material, like military combat systems, would plan on one key rollover per day, while Users that handle less sensitive information might typically roll their keys on an average of once per month.
- It is assumed that within any Use Case/Scenario, the number of average transactions per year (for operational purposes) and the number of crypto key rollovers per year for security purposes are mutually exclusive, but both would require Philos™ Blockchain Transactions, and for analytic purposes they are accounted for separately.
- Assumed Costs by Use Case Scenario Table 7.3 uses the definitions and assumptions above to derive some initial calculations for the costs of operational blockchain Services for the given hypothetical Use Case Scenarios.

Table 7.3. Assumed Costs by Use-Case Scenario

Table 11. Assumed Costs by Use Case Scenario	Number of BCDC FTE	Annual Employee Costs	Annual Cost Plus 20% Profit and Risk	SCIF COSTS	Total Overhead Costs/Year	Avg Trans/Yr + Avg Provs/Yr	Average Cost per Phylli	Epoch
CDE teacher licenses	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
DMV, Voting Records (DOR/DOS)	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
State Medical Licenses (DORA)	0.01	\$570	\$684		\$684	2,190	\$0.3123	2020-2022
COANG	0.02	\$2,185	\$2,622	\$1,000,000	\$1,002,622	18,251,800	\$0.0549	2023-2025
Missile Defense Agency	0.04	\$3,895	\$4,674	\$1,000,000	\$1,004,674	22,034,320	\$0.0456	2023-2025
Representative IoT Provisioning	1.04	\$98,420	\$118,104		\$118,104	120,000,432	\$0.0010	2023-2025
Utility company	0.06	\$5,320	\$6,384		\$6,384	6,630,720	\$0.0010	2023-2025
FEMA	0.05	\$4,370	\$5,244		\$5,244	1,213,140	\$0.0043	2026-2029
AF Space Surveillance System	0.05	\$4,370	\$5,244	\$1,000,000	\$1,005,244	40,284,320	\$0.0250	2026-2029
F-35 Air Force Supply Chain	0.04	\$3,895	\$4,674	\$1,000,000	\$1,004,674	18,263,140	\$0.0550	2026-2029
TRANSCOM	0.05	\$4,370	\$5,244	\$1,000,000	\$1,005,244	36,513,140	\$0.0275	2026-2029
							Averages =	
Total	1	\$128,535	128,536	\$5,000,000	5,154,242	263,197,582	\$0.1047	
Epoch								Epoch
2020-2022	0	\$1,710	2,052	\$0	2,052	6,570	\$0.31	2020-2022
2023-2025	1	\$111,530	133,836	\$2,000,000	2,133,836	166,923,842	\$0.15	2023-2025
2026-2029	0	\$128,535	154,242	\$3,000,000	5,154,242	263,197,582	\$0.10	2026-2029

As used above, the constraints have been simplified with assumptions on the numbers of employees necessary to perform work tasks to manage transactions from Nodes and ILCs; security constraints driving the numbers of Crypto Key “Rolls,” the one-time costs of SCIFs, and the continuous costs to support a growing number of employees. All cost constraints have been relaxed here to initially observe extreme boundary limits, but those could be added later.

In the long run, a linear programming solver or a mixed-integer solver might be a good approach and should be attempted when sufficient data are available. A purely integer programming solver might use more computational resources to determine exact integer answers, but for this research it would not be worth that level of accuracy.

In the short term, a simple spreadsheet approach seems to be more transparent and lends itself to better “what if” Delphi-like brainstorming with developers and stakeholders to use for sensitivity analysis at the current level of fidelity. This simple heuristic approach and sensitivity analysis could identify certain scenarios to down-select, how scenarios can be compared financially to others, or whether the general characteristics of each scenario can be better communicated to Users for rough order-of-magnitude cost estimates for negotiations or to set terms for service-level agreements. Probably the biggest reason for this framework now is because of the current lack of a credible pricing mechanism, and the “real world” time-urgency to develop one quickly.

Observations:

- To avoid inconsistencies and misunderstandings about definitions within working groups and discussions, a common set of understandable formal definitions needs to be set, and used consistently to accurately write code, set financial exchange values, etc. With the uncertainties of how Phylli™ will

be used in certain use cases, and what market demand will be in the future, it is not yet known what the actual US Dollar value should be.

- Under the assumptions used, the average US Dollar fee-per-Service to pay for one instance of adding data to the blockchain, defined here as one Phyli™, is about \$.1047. That may still be higher than what some people expected, possibly because the assumptions are over-simplified, or other factors were not correctly considered.
- As the System expands to include more large Use Cases, the price per Phyli™ drops by about 35 percent from the first Epoch to the third Epoch. That seems to be more reasonable, and probably in line with current expectations.
- The Average Cost per Service Fee, here defined as the value of one Phyli™, varies wildly among the different Use Case Scenarios, from a high price of \$.3123 to a low price of \$0.001. Again, although some of the high prices occur within high-transaction Scenarios, economies of scale are not yet reflected in the outcomes – probably because the number of personnel that are assumed to be required might be too high.
- Selected sensitivity analyses can be effective to verify, validate, and better understand outcomes. For example, the average cost of Phyli™ for the “Representative IoT Provisioning” Use Case is currently calculated at \$.001 per transaction. If the assumed 1.04 necessary employees were changed – perhaps by reevaluating the assumptions or by adding more automation – would result in changing the USD cost per blockchain Service transaction. So, even without a robust optimization model, this heuristic “triage” phase can help with moving toward a more optimal decision.

Next Steps:

- Assumptions, definitions, and methodology all need to be vetted for accuracy. Preliminary discussions on this topic were conducted on April 10, 2020, with developers and stakeholders. The ideas in this section serve as an initial starting point for future actions when commercial marketplace sales of Phyli™ become possible. When that occurs, other considerations will probably also be broader than discussed here, including simulation models of Phyli™ and “Stamp” tokenization flows using Petri Nets, growth and cost constraints, economic dynamics within the Philos™ blockchain, and several other topics.

- This analysis might be considered a good example of how Agile Systems Engineering was applied, and the original data were rapidly considered, modified, and replaced due with new assumptions. Three iterations took place, and the process was discussed extensively on May 7, 2020, with key BlockFrame leaders and developers. Rapid turnarounds helped to quickly verify and validate several assumptions. It is interesting that while this is a very complex problem, the rapid iterations helped to “bracket” the problem to something more manageable relatively quickly. Full resolution, however, might require more detailed heuristic models or simulations, possibly using Petri Net analysis or something similar.

Non-performance system characteristics (Quality Attributes)

BlockFrame® Eco-Secure Provisioning™ (ESP) using non-fungible utility tokens (NFTs) can be used in conjunction with blockchain/distributed ledger management, to provide secure commercial services, improve cybersecurity, and to enforce trust in systems of systems (SoS). Various business-to-business (B2B) clients and use cases can access the NFTs through common component interfaces and exchange them for the value of services according to defined Service Level Agreements (SLAs). BlockFrame® systems can provide many different services for B2B clients, using blockchain to organize cybersecurity, trust, and uniformity behaviors.

Some system characteristics have already been mentioned that could be measured through Quality Attributes (QATs) that currently have high priority requirements for stakeholders. The Neutral Governance Organization (NGO) has a key role in monitoring both performance metrics and QAT metrics to govern the economic health and quality of the System, to comply with legal and regulatory requirements for the Philos™ Marketplace, the state of the levels of trust, the physical state of hardware and software components, and other necessary considerations to run the system.

1. CMMC, SIP-13, and other requirements. Depending on specific UseCase requirements, many federally mandated requirements must be directly connected with functional system performance. They can also be compliance requirements which can and should be tracked as Quality Attribute metrics over time, and should be demonstrated, verified, and validated by system components and processes through testing and evaluation, with oversight of the NGO. Ultimately, the quality of NGO participation and oversight directly affects the health of the Philos™ Marketplace for users.
2. Corporate Cash Flow. Due to the young nature of BlockFrame® and the Neutral Governance Organization (NGO), which are separate entities but mutually dependent as explained above, cash flow

risk has the potential to delay or halt business growth and sustainability. Therefore, operations need to be monitored on a regular basis to ensure projects stay on schedule and revenue flows stay within expectations for growth. Figure 7.10 is a notional depiction of revenues that can be quantified and projected into the future to monitor future risks. “Design Wins” might not be quantitatively defined, but could represent events like R&D or CRADA grants, contracts, revenue from stock shares, etc. As these are quantified, identifiable periods of future risk have been identified as causes for concern. When “Wins” do occur, those concerns could be delayed or mitigated. Ideally, this should be assisted using a set of non-functional QAt metrics that can be followed on a near-real-time basis through the Consensus Dashboard and modified throughout the system life cycle to support management decisions and status.

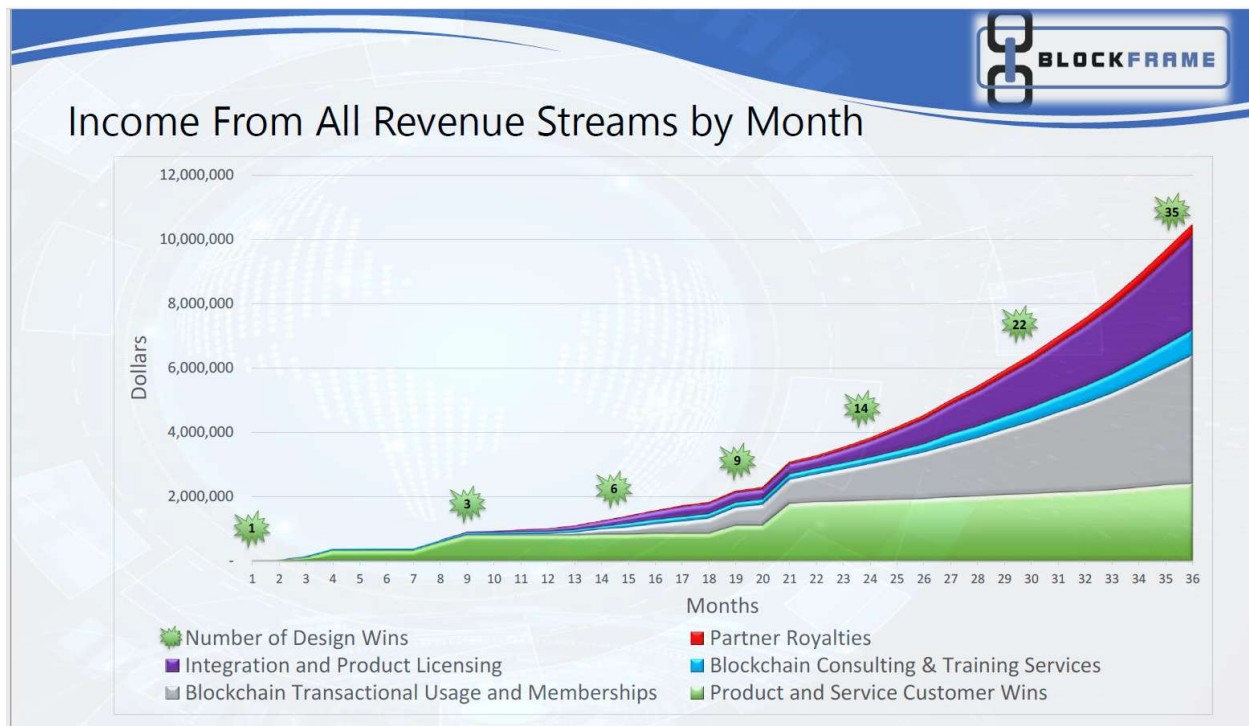


Figure 7.10. Income from All Revenue Streams by Month

3. Defining the Governance Objective Function for Trust.¹³⁹ The Philos™ Consensus Model must not only manage the process to accurately add blocks to the B/DL, but it must also accurately monitor and optimize the trust behaviors of the Users. If the B/DL performs outside the normal ranges of

¹³⁹ Lambert, Kent, and Vincil Bishop, Engineering the Philos™ Blockchain System, May 2019.

performance or quality objectives, the NGO must intervene to correct or mitigate anomalies. Ideally in practice, the B/DL and the NGO constantly maximize trust among the Users and mitigate risks. Figure 7.7 shows actual test data from Philos™ B/DL servers, performance parameters, and trust parameters that increase and converge over time. Again, some of these may represent functional performance parameters, but can also be considered non-functional quality attributes (QAts) that are also appropriate to track as quality metrics and behaviors over time. Examples are given in Figure 7.7 for metrics like, “Determine Merchant Performance,” “Ownership Track Record,” and “Identify High Risk Items/Organizations,” that can monitor trends in QAts, and rapidly identify quality problems that might compromise trust within business transactions, Users “gaming the system,” or other trends that could compromise the goals and performance of the trust chains. This also demonstrates the importance of the Consensus algorithms, and the incumbent risks that might occur if those algorithms were compromised, inefficient, or possibly fail economic stress tests.

4. Economic Stability of Phylia™. Stakeholder concerns about the initial pricing of the in-system utility tokens (Phylia™) are largely due to the unknowns concerning their economic behaviors. The initial pricing mechanism is essential because it must behave smoothly and nearly as expected for initial demonstrations, or there could be risks of perceived failures of the whole B/DL. Although Phylia™ should not be considered cryptocurrencies that could be used in open markets and could be subject to changes in value due to speculative investor behaviors as utility tokens they should be used for exclusively for administrative or accounting purposes to ensure accurate assure payments of fees-for-services. However, safeguards must be created and constantly monitored throughout the system lifecycle to ensure that dynamic instabilities, or even too much dynamic stability, could adversely affect fee payment structures. Since the initial startup of Philos™ and the active use of Phylia™ could begin with no transaction history, that initial phase could be very unpredictable and risky. Like a parent teaching their child to ride a bicycle, the neutral governance system might have to tightly control the initial conditions until initial stability appears to be sustainable – like the point parent safely release their grip on the bicycle and the child can confidently start pedaling on their own. A simulation model will be very useful to test the behaviors of that initial phase, but in the long run, the stability levels

could be considered as QAts that should be self-sustaining but also require robust monitoring for inserted events that could break the predicted behaviors.

7.4. Synthesis

Restated, Research Question 3 (RQ-3) is, “How can Eco-Secure Provisioning™ (ESP) of cryptologic non-fungible utility tokens (NFTs), in conjunction with blockchain/distributed ledger-assisted management and governance, provide secure commercial services, improve cybersecurity, and to enforce trust in systems-of-systems (SoS)?” This chapter satisfies Research Question 3 (RQ3) by demonstrating that provisioning cryptologic tokens and the governance of distributed ledgers can create a secure and trusted marketplace environment for commercialization. This research showed that functional BlockFrame® system-of-systems capabilities can operate as a sustainable commercial business model and provide effective blockchain, cybersecurity, and supply chain services. This Research Question is also well-aligned to support the philosophy of the Doctor of Engineering in Systems Engineering program emphasis on research with industry.¹⁴⁰

The commercialization strategy of the BlockFrame® suite of designs was examined by a third-party analysis conducted with the Simon-Everett company in September 2020.¹⁴¹ That report favorably compared the BlockFrame® blockchain/distributed ledger and cybersecurity designs with all other known similar designs in competitive markets using three alternative applications across twelve competitive commercial factors.

The BlockFrame® system of systems was compared to the capabilities other cyber security and blockchain systems using an expert industry evaluation and comparison approach through a grant from the Colorado Springs Economic Development Company, and a contract with the Simon Everett Corporation. This task will evaluate and assess the MBSE artifacts from Task 3.2.1 and 3.2.2 for metrics of performance including cost, sustainability, reusability, quality, and ease of use. These evaluations will be used in the process of writing patents to both mitigate risks to the product intellectual property and illustrate the uniqueness of the BlockFrame® technologies and capabilities.

From the Summary Observations, the Simon-Everett report concluded:

BlockFrame has the potential to create a blue ocean. Conventional cyber solutions are competing with each other on price and features in the margins (a saturated, cutthroat “red ocean”), whereas conventional blockchain solutions are of limited practical application. However, BlockFrame truly diverges from its competition in two areas (an open, competition-free “blue ocean”): energy efficiency and sustainability.

¹⁴⁰ More detailed work that supported this chapter in the areas of functional architecture, marketplace operations, and risk analysis is contained in Appendix C and Appendix D.

¹⁴¹ Daniel Spector and Brian Shea, Principals, *Simon-Everett. BlockFrame® Strategy Canvas*. September 2020.

This chapter satisfies the research question and shows that the BlockFrame[®] technologies and systems-of-systems architecture has the potential to provide highly successful services in a highly competitive and in-demand cybersecurity marketplace.

8.0. Research in Professional and Public Policy Applications

This Chapter addresses and satisfies Research Question 4 (RQ4), “What are the implications of this research for public policies and industrial practice for cybersecurity, digital supply chains, and digital privacy?” by presenting research done to translate industrial and professional efforts into practical and academic contributions. Translational contributions from this research have involved technologies from BlockFrame, Inc., systems engineering processes with Colorado State University, and several activities with companies and professional industrial organizations, primarily the Institute for Electrical and Electronic Engineers (IEEE), in the communities of practice of digital privacy and blockchain/distributed ledger applications. Ubiquitous world-wide problems with cybersecurity, digital supply chain management, national security, international commerce policy, and the loss of personal privacy have elevated these issues to the highest importance. Education in both the public and private sectors is essential to understand these issues, but even more important is to understand how to mitigate the problems. The IEEE has been proactive in several of these areas. The IEEE Future Directions Committee is responsible for major cross-Society programs. They supported the creation and funding of the IEEE Digital Privacy Initiative to demonstrate a deep commitment to digital privacy concerns. The activities associated with this chapter include the following:

- The IEEE Digital Privacy Initiative (DPI)¹⁴² examines digital privacy across the scope of all IEEE organizations nationally and internationally. On the DPI, I served as the subgroup chair for digital privacy policy and legislation.
- Representing the IEEE Digital Privacy Initiative, I prepared and gave a presentation to the National Council of State Legislatures (NCSL) Cybersecurity and Privacy Task Force, at their annual convention in July 2022¹⁴³. This was intended to provide mutually beneficial support on selected state and interstate cybersecurity and digital privacy issues to bridge gaps between technologies and public policy. This seems to have been the first such combined effort to address technical cybersecurity and privacy policy issues across all US state legislatures.
- Representing the DPI, I moderated an international IEEE Future Directions Digital Privacy Workshop on “Policies and Frameworks around the World,” with panelists Stefano Leucci, European Data

¹⁴² IEEE. IEEE Digital Privacy Initiative. <https://digitalprivacy.ieee.org/>

¹⁴³ National Council of State Legislatures Task Force on Cybersecurity. <https://www.ncsl.org/ncsl-in-dc/task-forces/task-force-on-cybersecurity.aspx>

Protection Supervisor (EDPS), Jason Lau, International Association of Privacy Professionals, and Dylan Gilbert, National Institute of Standards and Technology, who respectively presented on the European Union General Data Protection Regulation (GDPR), the China Personal Information Protection Law (PIPL), and National Institute of Science and Technology (NIST) policies and activities concerning digital privacy¹⁴⁴.

- I participated in and supported IEEE Standards Working Group “Task Force 1,” of the IEEE Special Committee on Security & Privacy to contribute to cybersecurity standards research of the Department of Defense Mitre Att&ck research on Industrial Control Systems (ICS) and Cybersecurity Maturity Model Compliance (CMMC) models¹⁴⁵, and analysis efforts to support the development of IEEE cybersecurity standards. That working group produced a series of four white papers to define useful frameworks for blockchain technologies (BCTs) with an emphasis on distributed energy use cases, to identify cyber tactics that might be used to attack BCT components, and to suggest mitigation steps to improve cybersecurity resilience. I was a co-author of the last two papers, which are presented in Sections 8.2 and 8.3 respectively.
- I was the lead co-author of, “Digital Privacy: Technical, Legal, and Legislative Aspects,” which is presented in Section 8.4.
- I was the co-designer of the IEEE Digital Privacy Model¹⁴⁶ and the author of the article, “Bridging the Technologist/Policy maker Divide in Digital Privacy,”¹⁴⁷ that was published in IEEE Insights magazine, which are presented in Sections 8.5 and 8.6, respectively.
- I prepared and delivered a briefing at the request of the Colorado County Clerks and Records Association titled, “Making Distributed Ledger a Reality in Colorado.” That is summarized in Section 8.7.

¹⁴⁴ IEEE Future Directions. Digital Privacy Workshop. “Policies and Frameworks around the World.” Oct 7, 2021. Accessed July 22, 2022. <https://cmte.ieee.org/futuredirections/events/digital-privacy-workshop/>.

¹⁴⁵ Mitre. “Mitre ATT&CK for Industrial Control Systems.” Accessed July 21, 2022. https://collaborate.mitre.org/attackics/index.php/Main_Page.

¹⁴⁶ IEEE Digital Privacy Model. <https://digitalprivacy.ieee.org/about/digital-privacy-model>.

¹⁴⁷ Lambert, Kent. *Bridging the Technologist Policy maker Divide in Digital Privacy*. Published in IEEE-USA InSight; (accessed Feb 26, 2024).

- I presented a live online lecture at the request of the IEEE Region V training seminar, entitled “A Call-to-Action Uniting Tech Innovators, Policy Leaders, and Businesses to Safeguard Digital Privacy.” That is summarized in Section 8.8.
- At the request of the IEEE Digital Privacy Initiative, and as Chair of the Policy and Legislative Subcommittee, I participated with an IEEE-USA working group to revise the tri-yearly IEEE-USA Digital Privacy Policy Statement. That is summarized in Section 8.9.

Together, these activities and work products document the applied and translational outcomes of this research.

8.1. Prior Articles for Context

Two IEEE papers are presented in Sections 8.2 and 8.3, in which I am a co-author. Two other papers preceded those prior to my participation as a member of the IEEE Special Committee on Security & Privacy. They are mentioned here to provide useful context and may be necessary to fully understand the technical direction of the papers that followed. “Standardization of the Distributed Ledger Technology cybersecurity stack for power and energy applications,” was the original foundational document for the research, and “Application of DLT cybersecurity stack to TES applications for a scalable, cybersecure, and interoperable future,” which was the first of the series of three articles published by IEEE. Only the abstracts of both of those articles are presented here as a useful introduction to Sections 8.2 and 8.3.

Standardization of the Distributed Ledger Technology cybersecurity stack for power and energy applications¹⁴⁸

Authors

Sri Nikhil Gupta Gouriseti, Ümit Cali, Kim-Kwang Raymond Choo, Elizabeth Escobar, Christopher Gorog, Annabelle Lee, Claudio Lima, Michael Mylrea, Marco Pasetti, Farrokh Rahimi, Ramesh Reddi, Abubakar Sadiq Sani

Abstract

The global trend toward integration of is opening doors to advanced, complex, and distributed marketplaces. Such advanced ecosystems, where utility-owned and non-utility-owned assets can contribute toward grid operations, generally require distributed communications and grid architectures. We posit the potential of using (DLTs) in supporting such applications, although their full potential has not been fully used, for example in designing long-term scalable solutions in operational technology applications. This is partly due to the lack of standardization across and between different DLTs, as well as other supporting building blocks (*e.g.*, communication protocols). This paper attempts to address this gap by proposing a DLT cybersecurity stack specifically designed for researchers, DLT technology developers, and end users (such as utilities). The DLT cybersecurity stack has been notionally mapped to related cybersecurity components, namely the Open Systems Interconnection (OSI) model, the Transmission Control Protocol/Internet Protocol (TCP/IP) suite, and existing Smart Grid architecture frameworks. In addition, the

¹⁴⁸ The full published article is available at <https://www.sciencedirect.com/science/article/pii/S2352467721001247>.

paper discusses several cybersecurity implications and demonstrates the potential uses of the DLT cybersecurity stack through multiple power and energy use cases. It is important to note that the stack can be also applied to DLT use cases that are outside the power and energy domain. This work was performed by the Cybersecurity Task Force under the IEEE P2418.5 Blockchain for Energy Standard working group that is part of the IEEE Power and Energy Society's Smart Buildings, Loads, and Customer Systems (SBLC) technical committee.

Application of DLT cybersecurity stack to TES applications for a scalable, cybersecure, and interoperable future¹⁴⁹

Authors:

D. Jonathan Sebastian-Cardenas, Sri Nikhil Gupta Gouriseti, Annabelle Lee, Michael Mylrea, Umit Cali

Abstract:

Transactive Energy Systems (TES) are expected to improve upon existing grid operations and capabilities by enabling the integration of traditional grid resources with distributed energy resources (DER). Distributed Ledger Technology or DLT (e.g., blockchain) presents itself as a viable instrument to support decentralized, autonomous, and tamper-evident applications, which can be leveraged within TES's ecosystem. DLTs can provide pertinent security controls including access controls, data immutability, and traceability in addition to other well-known advantages such as decentralization and scalability. This work demonstrates the DLT cybersecurity stack and its applicability to TES-based use-cases/applications. The seven-layer DLT cybersecurity stack is a DLT-agnostic framework that can quickly be used to classify and group the individual needs of an application into the different processing and cybersecurity layers offered by a DLT using a common taxonomy and an architectural mapping framework. This enables application engineers to demystify and strengthen the overall security aspects of their systems while maintaining an open perspective towards features and drawbacks that may hinder their performance in real-world scenarios. The paper leverages the work performed by the IEEE P2418.5 Blockchain for Energy Standards working group.

¹⁴⁹ The full published article is available at <https://ieeexplore.ieee.org/document/9917260>.

8.2. Assessing Cybersecurity Resilience of Distributed Ledger Technology in Energy Sector Using the MITRE ATT&CK® ICS Framework¹⁵⁰

Abstract

Digitization in the power industry enables wide connectivity among multiple new entrants such as DERs, prosumers, and P2P counterparts within or outside the Distributed Ledger Technology (DLT). The use of DLT to improve resilience in the power grid has growing support, but new technology provides new opportunities for adversaries to cause harm. This work completed by the Cybersecurity- focused task force of IEEE SA P2418.5 evaluates the potential risks by applying the MITRE ATT&CK® ICS matrix to the DLT Engineering and Cybersecurity Stack designed for power systems applications.

SECTION I. Introduction

Distributed Ledger Technology (DLT) has shown the potential to improve the resilience of electricity infrastructure. The objective is to have a more agile infrastructure that incorporates distributed energy resources to better respond to all hazards, from naturally occurring environmental hazards to cyber-attacks. However, the benefits of resilience offered by DLT must be weighed against the potential for additional cybersecurity risks. Digital transformation of the energy and other critical infrastructures is increasing external connections to Industrial Control Systems (ICS). Yet, organizations often lack monitoring, visibility, and defenses for their ICS environments [1]¹⁵¹. The cyber threats seeking to exploit this expanded attack surface may exploit people, processes, and technology aspects of DLT with various Tactics, Techniques, and Procedures (TTPs). This article presents one of the first explorations of these DLT opportunities and challenges through the lens of the MITRE ATT&CK® ICS matrix [2]–[4]. The MITRE ATT&CK® ICS matrix examines how the TTPs from real-world cyber events can occur, with tactics representing the short-term goals of an adversary and techniques representing how an adversary realizes those goals [5]. Adversarial TTPs are considered with the emerging architecture of DLT where adversaries can consistently use various TTPs to exploit energy delivery systems and infrastructure vulnerabilities. This paper provides a novel

¹⁵⁰ This section was previously published with K. Lambert as co-author as S. N. G. Gouriseti *et al.*, *Assessing Cybersecurity Resilience of Distributed Ledger Technology in Energy Sector Using the MITRE ATT&CK® ICS Framework*, 2022 IEEE 1st Global Emerging Technology Blockchain Forum: Blockchain & Beyond (iGETblockchain), Irvine, CA, USA, 2022, pp. 1-6, doi: 10.1109/iGETblockchain56591.2022.10087078. The published paper can be accessed at <https://ieeexplore.ieee.org/document/10087078>.

¹⁵¹ Source citations for Section 8.2 are listed parenthetically at the end of the section, as they were cited in the original document.

examination of how the MITRE ATT&CKICS matrix is applied to each layer of the DLT engineering and cybersecurity stack [6]. It provides insight on how to leverage the MITRE ATT&CK® ICS matrix with its intended purpose [7] to mitigate weaknesses when applying DLT to advance the grid modernization goals in the electricity infrastructure.

The rest of the paper is organized as follows: Section II provides an overview of DLT and current exploratory applications in the energy sector. Section III provides a deep-dive into DLT-related cybersecurity efforts and defines the need for cyber resilience research and standards. Section III also provides a high-level overview of the DLT Engineering and Cybersecurity Stack and the MITRE ATT&CK® ICS Matrix. Section IV provides a deep-dive into the DLT Cyber Resilience Model (CRM) design and defines the process designed to map the DLT Engineering and Cybersecurity Stack [6] and the MITRE ATT&CK® ICS Matrix. Section V discusses future work and conclusions.

SECTION II. Overview of DLT and Its Applications in the Energy Sector

In recent years, much attention has been paid to the potential use of DLT in energy applications [8]. Examples include the use of DLT in peer-to-peer and peer-to-market Energy Trading, tracking of Renewable Energy Credits, Electric Vehicle (EV) charge management, and management of gas contracts [9]. DLT is based on information and communication infrastructures and protocols that allow simultaneous distributed access, validation, and record updating of a distributed ledger in an immutable manner across a network of multiple stakeholders, entities, and locations. The information stored in the ledger(s) via using keys and cryptographic signatures becomes an immutable database, and any further processing is governed by the rules of the underlying DLT network. The structure of the generated distributed ledger relies on a cryptographic hash, making it resilient to cyber-attacks by allowing the detection of any possible content alteration.

Distributed ledgers can be classified as public or private, permissionless or permissioned, or any combination above. However, most energy DLT applications prefer permissioned DLT networks where access is restricted to users with defined privileges. “Consensus” plays a central role in the DLT architecture to verify and validate transactions. Based on the primary objective of the specific DLT network, the widely used consensus protocols include Proof-of-Work (PoW), Proof-of-Stake (PoS), Proof-of-Authority (PoA), and voting based. Energy DLT applications often adopt the PoA (relies on the partial trust of publishing nodes) or voting-based (use principles

from Byzantine Fault Tolerance (BFT) and Crash Fault Tolerance (CFT)) consensus models, with some experimenting with using PoS (based on the stake a user has invested into the system) consensus.

SECTION III. DLT Cybersecurity and Cyber Resilience

Several energy sector use-cases that either propose or experiment with use of DLT seem to involve some level of interaction with Industrial Control Systems (ICS) or Operational Technology (OT) systems [8], [10], [11]. ICS networks are often involved in driving continuous, discrete, and batch processes, and such processes cannot afford loss or degradation of operations. For instance, a DLT-based DER management and energy market use-case may need access to DER data, customer information, and operational information [12], [13]. Similarly, a DLT-based EV charging and coordination use-case may need access to the charging station information, EV information including battery state and other metrics [14], [15]. However, ICS networks in the energy sector have several cybersecurity challenges that have been captured and emphasized in the literature [16]–[18]. Recent research has also focused on cyber resilience in addition to cyber security [3], [19]–[21] pertaining to ICS operations. From the Energy DLT perspective, it is imperative to evaluate each DLT security implication on the energy systems networks' operational processes before deploying the technology. To address this gap, the IEEE Standards Association (SA) P2418.5's cybersecurity task force developed the DLT engineering and cybersecurity stack [6]. The seven-layer stack is intended for DLT architects and application owners to ensure that cybersecurity is inherent at every step of the design and deployment process. Further-more, the stack offers recommendations and mitigations for known vulnerabilities and attacks and is expected to address the cybersecurity challenges and ensure secure operations. At a high-level below are the definitions of each of the seven layers:

Application Layer Applications that trigger rule-bases and program code.

Execution Layer Rule-bases and program code.

Consensus Layer Consensus protocols: lottery-based, voting-based, etc.

Data Model Layer Application data storage, including time synchronization.

Network Layer Peer-to-peer communications, transaction broadcast/discovery.

Infrastructure Layer Physical computing and data storage entities.

Physical Layer Users, systems, and physical devices (e.g., system actors).

Cybersecurity resilience is the ability of a system to continue operating under adverse conditions, stress, or an attack. The system may operate in a degraded mode while maintaining essential operational capabilities [4]. Cybersecurity resilience is critical to energy systems since an *ICS/OT* network generally cannot be disconnected if an adversary or attack is detected. Similarly, cybersecurity resilience is crucial for DLTs since the DLT cannot be taken off the network if adversaries or attacks are detected. For DLTs, cybersecurity resilience may be achieved by implementing fault-tolerant principles in the design of the DLT, for example, CFT or BFT. Although private and permissioned DLT deployment for a use-case could minimize the risk of adversarial presence, a controlled environment does not eliminate cybersecurity risk. Therefore, one objective of assessing DLT-based use-cases and applications in the energy sector and specifically in the *ICS/OT* environment is to ensure the cybersecurity and cybersecurity resilience of the energy *ICS/OT* system/network is not deprecated but enhanced [3]. For example, suppose a DLT is used to assist with energy operations, data aggregation, and analytics. In that case, the cybersecurity resilience analysis is as follows: 1) resilient operations of the electro-mechanical systems contributing to operations (hardware perspective), 2) resilience of the DLT assisting in the network operations, including executing distributed applications (software perspective), and 3) resilience analysis of the overall system (both hardware and software combined perspective).

The MITRE ATT&CK® ICS matrix uses principles from the cyber kill chain [22] and information acquired from past cyber incidents to evaluate what an adversary can do to a system and how the adversary can accomplish their goals. A simplified overview of the MITRE ATT&CK® ICS Matrix tactics shown below illustrates the 12 tactics and what the adversary is trying to achieve:

Initial Access Get a foothold into the ICS environment;

Execution Run code or manipulate system functions, parameters, and data in an unauthorized way;

Persistence Maintain their foothold in the ICS environment;

Privilege Escalation Gain higher-level permissions;

Evasion Avoid security defenses;

Discovery Locate information to assess and identify their targets in your environment;

Lateral Movement Move through the ICS environment;

Collection Gather data of interest and domain knowledge on the ICS environment to inform their goal;

Command and Control Communicate with and control compromised systems, controllers, and platforms with access to the ICS environment;

Inhibit Response Function Prevent the safety, protection, quality assurance, and operator intervention functions from responding to a failure, hazard, or unsafe state;

Impair Process Control Manipulate, disable, or damage physical control processes;

Impact Manipulate, interrupt, or destroy the ICS systems, data, and their surrounding environment.

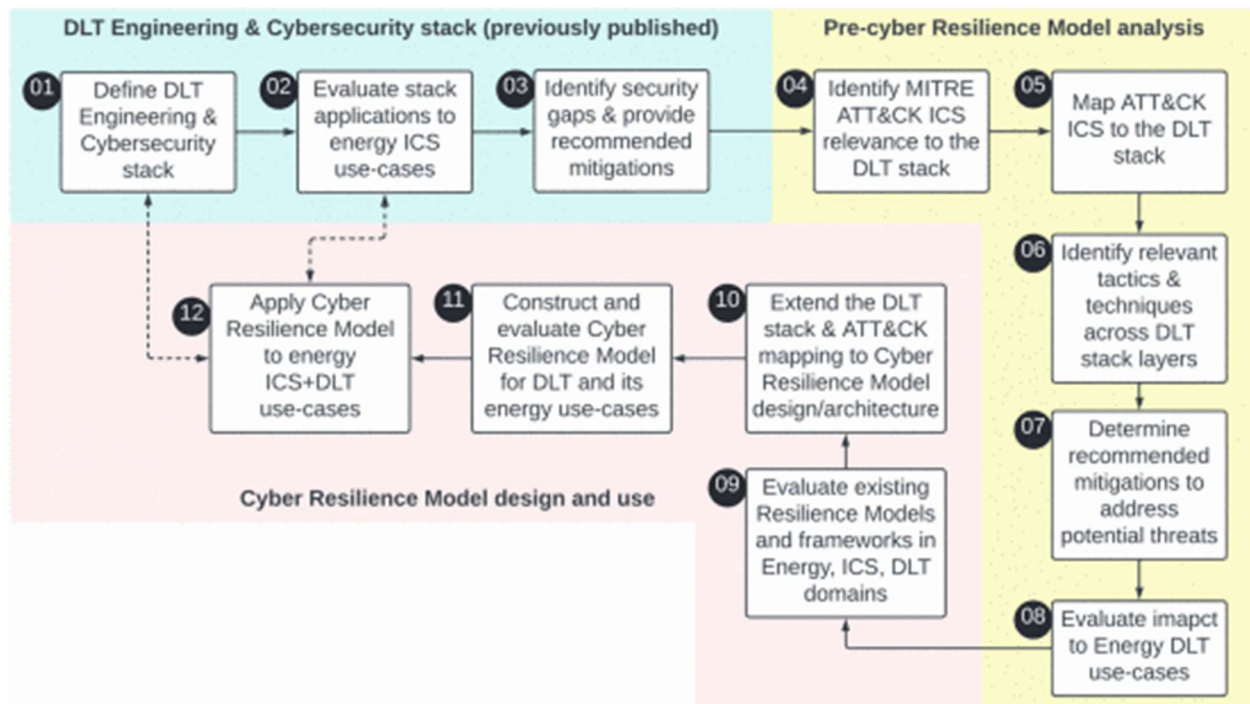


Fig. 1. Cyber resilience model development process flow/roadmap

Even during a cyber-attack, detecting anomalies, including vulnerabilities that may be used to exploit operations, and assessing the overall risk of the operations/use-cases to implement defensive and reactive response actions is essential. To meet this goal, the IEEE SA P2418.5 working group's cybersecurity task force investigates cyber resilience of DLT-based energy applications in the ICS/OT environment against the MITRE ATT&CK® ICS matrix through the following activities:

1. The task force has been assessing the mapping between the DLT engineering and cybersecurity stack against the MITRE ATT&CK® ICS matrix. Mapping the stack and the matrix can help identify the implications of various tactics and techniques that an adversary may use to compromise one or more of

the seven DLT stack layers and can provide valuable insights into possible exploits in DLTs and assist in defining security and resilience measures.

2. Identifying these tactics/techniques can also be used in a risk assessment for an ICS/OT system or network. The objective is to identify mitigation strategies to address cybersecurity resilience, including proactive and reactive cybersecurity controls.
3. The task force aims to use the mapping to provide guidance that can assist DLT designers and DLT users (application owners) in increasing the overall cyber resilience and cybersecurity for energy sector ICS/OT systems and networks - leading to a cyber resilient model.
4. Developing a cyber resilient model will support a comprehensive cyber risk management framework with well-defined multi-layer and preventive cyber security controls and measures to safeguard the entire digital chain. Further-more, an acceptable risk policy will allow all market agents with different cyber risk appetites and tolerance thresholds that differ according to the asset size and technology to self-assess their risk exposure according to the same set of rules.

The task force's current effort has been to develop the mapping. The detailed maps between the stack and the matrix are beyond the scope of this paper, however, the following section provides an architectural process flow that explains how the mapping has been performed.

SECTION IV. DLT Cyber Resilience Model Roadmap and Mapping Process Flow

Fig. 1 shows the overall process flow of the Cyber Resilience Model (CRM) design efforts - which is divided into 12 steps.

- The first three steps involve the development builds of the DLT Engineering and Cybersecurity Stack, which was completed by the task force and presented in [6].
- Steps 4–8 of Fig. 1 encompass the pre-CRM analysis that evaluates the relevance of various tactics and techniques from the MITRE ATT&CK[®]ICS matrix against the stack. The pre-CRM phase analysis is complete but will be published in a forthcoming paper due to its complexity.
- Using the mapping information between the stack and the MITRE ATT&CK[®] ICS Matrix along with the stack description, the task force then will focus on the completion of the CRM phase (steps 9–12 of Fig. 1).

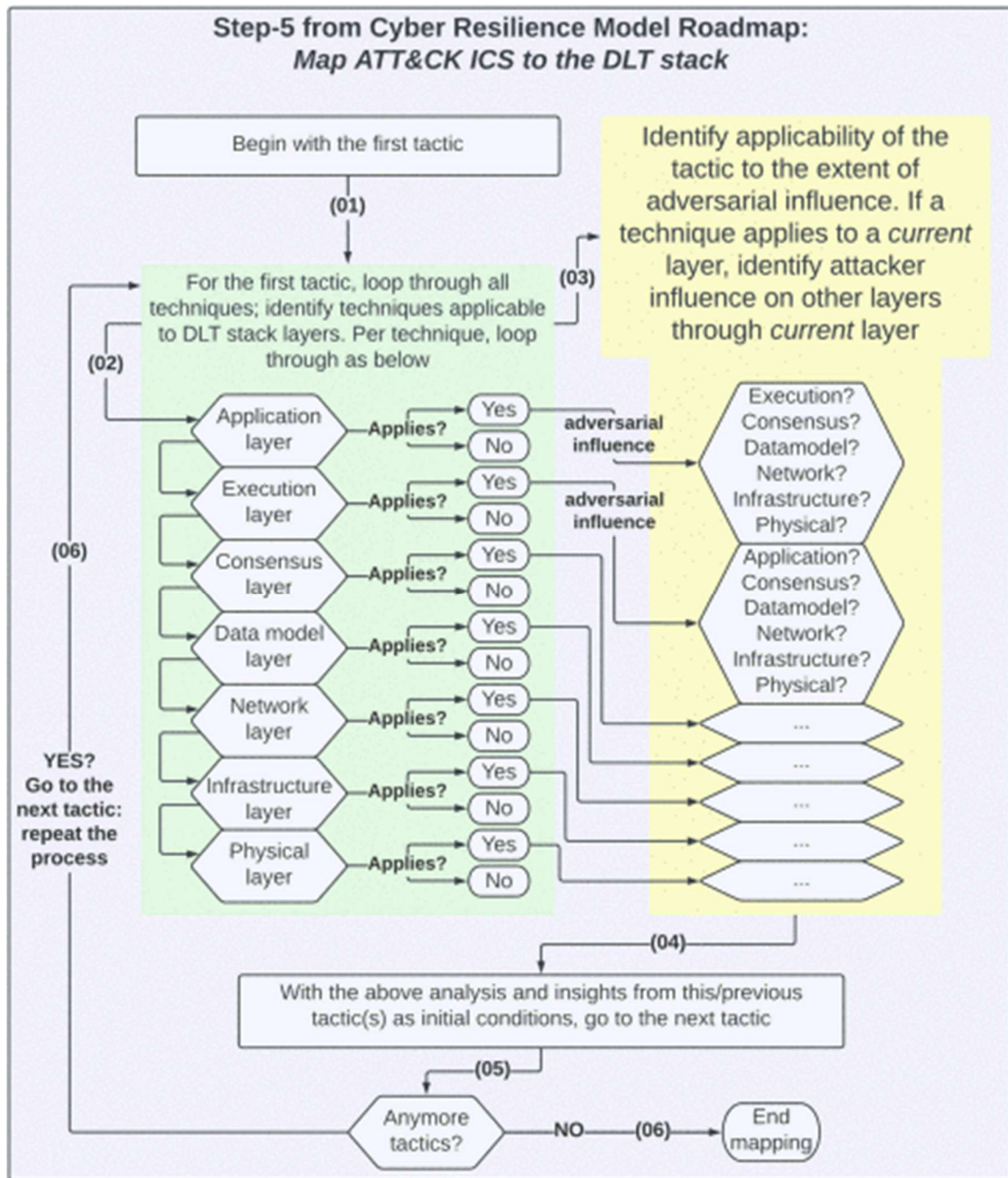


Fig. 2. Top-down deep-dive into step-5 from crm development process flow/roadmap
The most critical and complex step of pre-CRM analysis is step-5 and a top-down view into step-5 is shown in Fig. 2.A walk-through of Fig. 2 follows:

Since the sequence on which the tactics listed by MITRE (left-to-right) is based on adversarial propagation on a kill chain or similar construct, the same sequence has been followed for this mapping. Starting with the first tactic (Initial Access), each relevant technique has been reviewed to evaluate its applicability to each of the stack layers. If there is relevance, the relationship is further analyzed to determine the applicability of the technique (and the overall tactic) to the extent of adversarial influence. In other words, the following question is answered by further analysis: *Can the adversary compromise a DLT component that belongs to a layer and affect the DLT components at other stack layers?* This process is repeated for all of the 12 tactics. The end result is a detailed map between the DLT stack and the MITRE ATT&CK® ICS tactics and techniques. The mitigation recommendations (step 7 of Fig. 1) will be tailored using MITRE's existing mitigations for the tactics and techniques. Once the mapping and general mitigations are in place, energy sector use-cases will be used as illustrations to demonstrate the use of mapping and mitigations with guidance on how to tailor the mitigations to an energy use-case (step 8 of Fig. 1). The above exploration is based on a number of assumptions as cyber resilience and cybersecurity are broad areas for any application. The assumptions of the presented work are:

Assumption-1 Every DLT architecture includes the following three components: 1) Smart contracts or similar artifacts that allow defining rules, programs, etc.; 2) A distributed ledger or a database-style artifact that is immutable and captures transactional history; 3) A consensus mechanism or similar artifact that allows for group agreement, is fault tolerant (crash or byzantine), and is subject to a governance model;

Assumption-2 Immutability of the ledger is achieved through cryptography. If the cryptographic algorithm is compromised, the immutability of the ledger should be assumed to be compromised;

Assumption-3 The DLT includes the seven layers defined in the DLT engineering and cybersecurity stack;

Assumption-4 Only permissioned or private DLT is encouraged for use in the energy sector ICS/OT applications/use-cases. This is important because the participating entities (e.g., utilities, regulators, auditors, etc.) require mechanisms to ensure non-repudiation in the case of anomalous behavior. This assumption may be backed by the extensive literature that indicates the importance of visibility in DLT-based ICS/OT use-cases, unlike many public DLTs [23], [24];

Assumption-5 The standard security practices, for instance, secure software design and development, secure product life-cycle management, implementation of a security framework, and security versioning and patch management, are implemented for the ICS/OT systems and networks.

SECTION V. Conclusion

This work has laid out an approach to evaluate and mitigate the security threat of **DLT** applications in the energy infrastructure through the lens of **MITRE ATT&CK**[®] ICS matrix. The ongoing work will comprehensively map **TTP** enumerated in the **MITRE ATT&CK**[®] ICS framework to each layer of the **DLT** stack to identify new cybersecurity assurance requirements in the technology. Future work will propose a holistic approach to address specific security needs for the entire **DLT** for upstream and downstream applications: 1) the governing principles to cyber security risk management, 2) a cybersecurity framework for the agents in any downstream or upstream connected to a **DLT**, 3) the principles for cyber risk assessment by technology type such as a remote controlled **DER**, 4) the reporting mechanism for risk based incidents and threats, 5) a governance and oversight of internal controls and risk mitigation techniques, and 6) a disclosure policy with respect to cyber security assessment and identified threats.

References

1. *Dragos ICS/OT Cybersecurity Year in Review (YIR) Report*, [online] Available: <https://www.dragos.com/resource/dragos-2021-industrial-cybersecurity-year-in-review-announcement/>.
2. MITRE Corporation, *ICS Matrix*, [online] Available: <https://attack.mitre.org/matrices/ics/>.
3. R. Ross, V. Pillitteri, R. Graubart, D. Bodeau, and R. Mcquaid, "NIST Special Publication 800–160", *Rev. 1: Developing Cyber-Resilient Systems: A Systems Security Engineering Approach*, vol. 2, 2021.
4. *Joint Task Force Transformation Initiative of the National Institute of Standards and Technology (NIST)*, 2011, [online] Available: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-39.pdf>.
5. B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, *MITRE ATT&CK®: Design and philosophy*" MITRE Corporation McLean VA USA Technical Report MP180360R1, March 2018.
6. S. N. G. Gourisetti, U. Cali, K.-K. R. Choo, E. Escobar, C. Gorog, A. Lee, et al., "Standardization of the Distributed Ledger Technology cybersecurity stack for power and energy applications", *Sustainable Energy Grids and Networks*, vol. 28, 2021.
7. O. Alexander, M. Belisle and J. Steele, "MITRE ATT&CK® for industrial control systems: Design and philosophy", *MITRE Corporation McLean VA USA Technical Report MP01055863*, 2020.
8. M. Andoni, V. Robu, D. Flynn, S. Abram, D. Geach, D. Jenkins, et al., "Blockchain technology in the energy sector: A systematic review of challenges and opportunities", *Renewable and Sustainable Energy Reviews*, vol. 100, pp. 143-174, 2019.
9. U. Cali, M. Deveci, S. S. Saha, U. Halden and F. Smarandache, "Prioritizing Energy Blockchain Use Cases Using Type-2 Neutrosophic Number-Based EDAS", *IEEE Access*, vol. 10, pp. 34260-34276, 2022.
10. S. P. Siano, G. De Marco, A. Rolan and V. Loia, "A survey and evaluation of the potentials of distributed ledger technology for peer-to-peer transactive energy exchanges in local energy markets", *IEEE Systems Journal*, vol. 13, no. 3, pp. 3454-3466, 2019.
11. A. A. Monrat, O. Schelén and K. Andersson, "A survey of blockchain from the perspectives of applications challenges and opportunities", *IEEE Access*, vol. 7, pp. 117134-117151, 2019.

12. F. A. Rahimi and S. Mokhtari, "Distribution Management System for the Grid of the Future: A Transactive System Compensating for the Rise in Distributed Energy Resources", *IEEE Electrification Magazine*, vol. 6, no. 2, pp. 84-94, 2018.
13. Y. Wu, Y. Wu, J. M. Guerrero, and J. C. Vasquez, "Digitalization and decentralization driving transactive energy Internet: Key technologies and infrastructures", *International Journal of Electrical Power and Energy Systems*, vol. 126, 2021.
14. S. Rinaldi, M. Pasetti, E. Sisinni, F. Bonafini, P. Ferrari, M. Rizzi, et al., "On the mobile communication requirements for the demand-side management of electric vehicles", *Energies*, vol. 11, no. 5, 2018.
15. S. Rinaldi, M. Pasetti, A. Flammini, P. Ferrari, E. Sisinni, A. Depari, et al., "Analysis of communication requirements for enabling the coordinate charge of sustainable multimodal electric vehicle supply equipment", *2021 International Conference on Electrical Computer Communications and Mechatronics Engineering (ICECCME)*, vol. 102021, pp. 1-6.
16. D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan and N. Meskin, "Cybersecurity for industrial control systems: A survey", *Computers and Security*, vol. 89, 2020.
17. T. Alladi, V. Chamola and S. Zeadally, "Industrial control systems: Cyberattack trends and countermeasures", *Computer Communications*, vol. 155, pp. 1-8, 2020.
18. P. Ferrari, E. Sisinni, P. Bellagente, S. Rinaldi, M. Pasetti, A. O. de Sá, et al., "Model-Based Stealth Attack to Networked Control System Based on Real-Time Ethernet", *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7672-7683, 2021.
19. R. F. Babiceanu and R. Seker, "Cyber resilience protection for industrial internet of things: A software-defined networking approach", *Computers in Industry*, vol. 104, pp. 47-58, 2019.
20. F. Januário, A. Cardoso and P. Gil, "A distributed multi-agent framework for resilience enhancement in cyber-physical systems", *IEEE Access*, vol. 7, pp. 31342-31357, 2019.
21. M. Azimian, V. Amir, S. Javadi, S. Mohseni and A. C. Brent, "Resilience-oriented planning of multi-carrier microgrids under cyber-attacks", *Sustainable Cities and Society*, vol. 79, 2022.
22. E. M. Hutchins, M. J. Cloppert, R. M. Amin et al., "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains", *Leading Issues in Information Warfare & Security Research*, vol. 1, no. 1, pp. 80, 2011.
23. S. Saha, N. Ravi, K. Hreinsson, J. Back, A. Scaglione and N. G. Johnson, "A secure distributed ledger for transactive energy: The Electron Volt Exchange (EVE) blockchain", *Applied Energy*, vol. 282, 2021.
24. S. Wang, A. F. Taha, J. Wang, K. Kvaternik and A. Hahn, "Energy crowdsourcing and peer-to-peer energy trading in blockchain-enabled smart grids", *IEEE Transactions on Systems Man and Cybernetics: Systems*, vol. 49, no. 8, pp. 1612-1623, Aug 2019.

8.3. Assessment of the Distributed Ledger Technology for Energy Sector

Industrial and Operational Applications Using the MITRE ATT&CK® ICS Matrix¹⁵²

Abstract

In recent times, Distributed Ledger Technology (DLT) has gained significant attention for its potential application in the energy sector. Utilizing blockchain and DLT has demonstrated the ability to enhance the resilience of the electric infrastructure, which will support a more flexible infrastructure and advance grid modernization. However, the deployment of these technologies increases the overall attack surface. The MITRE ATT&CK® matrices have been developed to document an adversary's tactics and techniques based on real-world observations. The MITRE ATT&CK® matrices provide a common taxonomy for offense and defense and have become a valuable conceptual tool across multiple cybersecurity disciplines for conveying threat intelligence, performing testing through red teaming or adversary emulation, and enhancing network and system defenses against intrusions. The MITRE ATT&CK® for Industrial Control Systems (ICS) matrix was created to provide knowledge about adversary behavior in the ICS technology domain. This study analyzes the relevance of various tactics and techniques across a seven-layer DLT engineering and cybersecurity stack, known as the DLT stack, designed by the Cybersecurity Taskforce under IEEE P2418.5 - Standard for Blockchain in Energy working group sponsored by Power and Energy Systems - Smart Buildings, Loads and Customer Systems (PES/SBLC) Technical Committee. Additionally, this paper identifies specific mitigation strategies tailored to the energy ICS environment.

SECTION I. Introduction

Distributed Ledger Technology (DLT) has shown great potential in enhancing energy infrastructure's security, control, and resilience [1]¹⁵³, [2]. The ability to track and trace end-to-end energy processes and data transactions is one of the security benefits that can aid in responding to cyber-attacks and natural environmental hazards. Furthermore, transitioning from centralized to distributed architectures with redundancy can increase the

¹⁵² This section was previously published with K. Lambert as co-author with A. Lee *et al.*, *Assessment of the Distributed Ledger Technology for Energy Sector Industrial and Operational Applications Using the MITRE ATT&CK® ICS Matrix*, in *IEEE Access*, vol. 11, pp. 69854-69883, 2023, doi: 10.1109/ACCESS.2023.3288428. The published paper can be accessed at <https://ieeexplore.ieee.org/document/10159478>.

¹⁵³ Source citations for Section 8.4 are listed parenthetically at the end of the section, as they were cited in the original document.

fault tolerance and resiliency of the network while also increasing overall cyber complexity. While the opportunities of applying DLT across various sections have been examined, the literature lacks an examination of how adversaries can exploit vulnerabilities in DLT applications for energy delivery using various Tactics, Techniques, and Procedures (TTPs). Rapid DLT adoption without a comprehensive evaluation of its security and resiliency in the context of energy applications can harm the Operational Technology (OT) infrastructure. Although the inherent components of DLT, such as the database, consensus mechanism, data transmission, and smart contracts, are not novel and have existed as standalone software components for several decades, adversaries can exploit common vulnerabilities of these individual software components to compromise the DLT and impact the energy/OT infrastructure interacting with it. For example, Apache CouchDB is sometimes used as an external state database for Hyperledger Fabric DLT [3]. Still, according to the Common Vulnerabilities and Exposures (CVE) database, Apache CouchDB has 16 CVEs, with 5 having a Common Vulnerability Scoring System (CVSS) score of 9 or 10 (critical) [4]. A similar analysis can be applied to Fabric's chain code (smart contract) languages, such as Go, Node.js, and Java. Vulnerability analyses can be performed on any DLT protocols, components, and applications, as well as the deployment and management of the DLT since adversaries can exploit existing common vulnerabilities within the DLT components using known TTPs. Therefore, an adversarial TTP analysis is necessary for any DLT.

The present study builds upon the preliminary findings discussed in [5] to address the aforementioned research gaps by utilizing the MITRE ATT&CK® Industrial Control Systems (ICS) matrix [6], [7], [8] to evaluate each layer of the DLT engineering and cybersecurity stack (DLT stack), defined in [9]. The MITRE ATT&CK® ICS matrix is a tool that analyzes TTPs from real-world cyber events, with tactics representing the objectives of an adversary and techniques illustrating how an adversary achieves those objectives [10]. This matrix [6] was developed based on publicly available threats and incidents, and the techniques under the tactics were scoped according to past cyber events. One of the general uses of the MITRE ATT&CK® ICS matrix is to identify if an ICS/OT network is susceptible to threat actors that use the techniques identified in the matrix to exploit vulnerabilities. The study considers adversarial TTPs in emerging DLT architectures, where adversaries can use various techniques to exploit potential vulnerabilities in energy delivery systems. To the authors' knowledge, no relevant events or incidents related to DLT-based ICS/OT use cases have been made publicly available. Using the MITRE ATT&CK® ICS matrix to evaluate the relevance of the techniques across the DLT stack is consistent with the proposed use defined in the MITRE reference document [10]: "ATT&CK® for ICS supports many use cases, including failure scenario

development, education, and the existing ATT&CK® use cases”. This paper aims to identify potential attack tactics and techniques and mitigation strategies.

This ATT&CK® ICS matrix examines TTP from real-world cyber events. Cyber adversaries are complex, nonlinear, and evolving. These groups will continue using various TTP to exploit energy delivery systems and infrastructure vulnerabilities. This paper examines how the MITRE ATT&CK® ICS tactics and techniques may be applied to each layer of the DLT stack, from the application layer to the physical layer. This work provides information to assist in understanding and mitigating cyber threats to a deployed DLT while addressing the following questions:

- What are the potential TTPs that can be used to exploit potential vulnerabilities in the DLT stack?
- How can these potential vulnerabilities be mitigated to reduce/remove the potential impact on the other layers in the DLT stack?
- How can the proposed framework be applied in real life applications to increase the cyber-physical resilience of critical infrastructure like power systems?

This paper helps answer these questions and provides insight on leveraging the MITRE ATT&CK® ICS matrix in assessing a planned or deployed DLT and remediating/mitigating those threats in applying DLT to advance the grid modernization goals in the electricity infrastructure [11]. This research is timely as the digital transformation of the energy infrastructure is expanding the attack surface and the number of vulnerabilities that adversaries can exploit [12].

The remainder of the paper is organized as follows: Section II provides an overview of distributed ledger technology, while Section III discusses current exploratory applications of DLT in the energy sector. Section IV-A discusses the cybersecurity risk management process, while Section IV-B introduces cybersecurity resilience. After introducing the DLT engineering and cybersecurity stack in Section V, Section VI presents a detailed relationship analysis between DLT and the MITRE ATT&CK® ICS matrix. The technical details about the mapping process between the DLT stack and the MITRE ATT&CK® ICS matrix and all the mappings are presented in Sections VII and VIII. Finally, Section IX summarizes the paper’s conclusions and provides recommendations for future work.

SECTION II. DLT Overview

Distributed Ledger Technology is based on information and communication infrastructures and protocols that allow simultaneously distributed access, validation, and record updating of a distributed ledger in an immutable

manner across a network of multiple stakeholders, entities, and locations. Storage of information is done securely and accurately using keys and cryptographic signatures. The information stored in the ledger(s) becomes an immutable database, and the rules of the underlying DLT network govern any further processing. The decentralized distributed nature of the ledger makes it resilient to various cyber-attacks requiring all copies stored across the network to be compromised simultaneously for the adversary to succeed.

Distributed ledgers can be classified as public or private, permissionless or permissioned, or any combination. A variety of DLTs exist. Blockchain is the most popular, but others, such as Hashgraph, Directed Acyclic Graph (DAG), Holochain, and Tempo, are also utilized.

The *Consensus* mechanism plays a central role in the DLT architecture. It is used to have transactions verified and validated. Several consensus protocols depend on the primary objective of the specific DLT network. The widely used consensus protocols include Proof of Work (PoW), Proof of Stake (PoS), and Proof of Authority (PoA). In the PoW model, a user is rewarded [13] by being the first to solve a difficult puzzle which is computationally-intensive to solve but easy to verify by others when solved. This consensus model is used by Bitcoin but is seldom used in energy DLT applications. The PoS model is based on the premise that the more stake a user has invested into the system, the less likely they will want to subvert it. With this consensus model, there is no need to perform resource-intensive computations (involving time, processing power, and electricity). The PoA consensus model relies on the partial trust of publishing nodes. Publishing nodes must have their identities proven and verifiable within the DLT network. This algorithm only applies to permissioned DLT networks with high levels of trust. Energy DLT applications often adopt the PoS, PoA, or similar consensus models in permissioned environments.

SECTION III. DLT in the Electric Sector

In recent years, significant attention has been paid to the potential use of DLT technology in energy applications [14], [15], [16]. Examples include the use of DLT technology in peer-to-peer [17], [18] and peer-to-market energy trading [19], tracking of renewable energy credits [20], electric vehicle charge management [21], trading and record-keeping in energy trading [22], Optimal Power Flow (OPF) solutions [23], and more [24]. It can support the creation of new asset classes like crypto-tokens backed by renewable resources [25]. DLTs have also shown the potential to improve the resilience of the electric infrastructure while allowing significant increases in speed, size, and frequency of data exchange [26]. This digital transformation of energy and other critical infrastructures is increasing external connections to ICS.

However, these gains in advancing grid modernization give impetus to a more agile infrastructure incorporating distributed energy resources in ways that can better respond to all hazards, from naturally occurring environmental hazards to cyber-attacks. Therefore, the advantages of the DLT need to be assessed against potential cyber threats as organizations often lack monitoring, visibility, and defenses for their ICS environments [12]. In many energy-related DLT applications, external triggers (known as “oracles”) play an essential role where smart contracts are triggered by external metering or telemetry signals such as the delivery of a certain quantity of energy or generation from a clean source of energy. The cyber threat seeking to exploit this expanded attack surface can exploit people, processes, and technology aspects of DLT using various TTPs. Cyber threats targeting ICS that underpin the security and operations of critical infrastructure continue to increase, requiring a detailed review of the TTPs.

SECTION IV. Cybersecurity Risk Management and Resilience

A. Cybersecurity Risk Management. Comprehensive Cyber Risk Management (CRM) requires proactive cyber risk governance with an end-to-end operational and cyber risk assessment to assess the confidentiality, availability, and integrity of information and technology assets. A cyber risk management framework must include a multi-layer, preventive cyber security controls, and measures to safeguard the entire digital chain. A proper risk policy will allow all market agents with different cyber risk appetites and tolerance thresholds that differ according to the asset size and technology to assess their risk exposure according to the same set of rules.

This holistic approach can address specific security needs for the DLT by addressing the upstream and downstream interconnectors with consistent governing principles to cyber security risk management that include:

- A cybersecurity framework for the users/systems located in any downstream or upstream connection to a DLT,
- Consistent reporting mechanism for risk-based incidents and threats,
- Governance and risk oversight of internal controls with risk mitigation techniques, and
- A disclosure policy for cyber security assessments and identified threats.

Comprehensive risk identification, assessment, mitigation, and monitoring apply to all downstream entities, including those not directly connected to a DLT. Some of these assets may be more vulnerable to threats due to their simpler architecture and fewer software-based defenses. A holistic cyber risk management framework should include a reporting mechanism to the DLT and upstream entities of all incidents throughout the ecosystem. This cyber risk

framework will provide for the assessment and impact of potential risk throughout the entire ecosystem. Figure 1 exemplifies the building blocks of an integrated cyber risk framework.

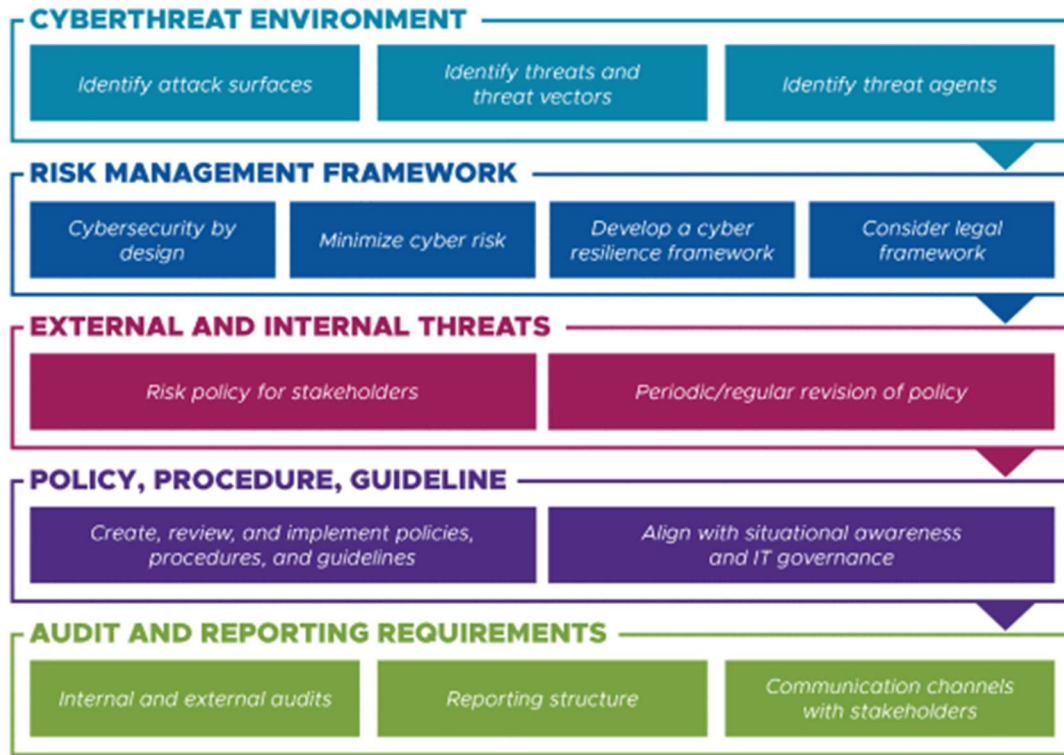


FIGURE 1. Integrated Risk Management Framework Components.

B. Cybersecurity Resilience. As defined by [8], “cybersecurity resilience is the ability of a system to continue operating under adverse conditions, stress, or an attack. The system may operate in a degraded mode while maintaining essential operational capabilities.” Cyber resiliency is critical to energy systems since an ICS/OT network generally cannot be disconnected if an adversary or attack is detected. Similarly, cyber resilience is critical for DLTs since the DLT cannot be taken off the network if adversaries or attacks are detected. For DLTs, cybersecurity resilience may be achieved by implementing fault-tolerant principles in the design of the DLT, such as Crash Fault Tolerance (CFT) or Byzantine Fault Tolerance (BFT). Although private and permissioned DLT deployment for a use case could minimize the risk of adversarial presence, a controlled environment does not eliminate cybersecurity risk. One objective of assessing DLT-based use cases and applications in the energy sector and specifically in the ICS/OT environment is to ensure that the cybersecurity and cybersecurity resilience of the energy ICS/OT system/network is not deprecated but enhanced. For example, suppose DLT is used to assist with energy operations data aggregation and analytics. In that case, the cybersecurity resilience analysis is as follows: 1)

resilient operations of the electro-mechanical systems contributing to operations (hardware perspective), 2) resilience of the DLT assisting in the operations, including executing distributed applications (software perspective), and 3) resilience analysis of the overall system (both hardware and software - combined perspective).

Ensuring cybersecurity resilience may enable the ICS/OT systems/networks to continue functioning even during a cyber-attack. In all operational states, detecting anomalies, including vulnerabilities, that may be used to exploit operations is crucial. The overall risk must be assessed to identify, select, and implement defensive and reactive response actions.

SECTION V. DLT Stack Layers

The DLT engineering and cybersecurity stack defined in [9] is a seven-layer DLT cybersecurity and engineering stack that comprises several relevant components and attributes and is designed for researchers, DLT technology developers, and end users (such as utilities). The DLT stack can be used as an architectural framework that is synergistic with the function of power grid applications. Furthermore, the DLT engineering and cybersecurity stack is designed to be used with existing cybersecurity and DLT applicability models [27], [28]. Included below are descriptions of the DLT stack layers extracted from the referenced paper:

A. Application Layer. This layer contains applications, software, scripts, and programs that the users can use (e.g., human users and nodes) to interact with the DLT. These software applications are above the DLT core and therefore do not fully belong to the DLT. Elements of the application layer can trigger rule bases and program code (such as smart contracts, chain code, atomic swaps [29], etc.) that reside in the execution layer (below). Other elements of this layer include User Interface/Graphical User Interface (UI/GUI), performance analysis applications such as Hyperledger CaliperTM, etc. [1].

B. Execution Layer. This layer contains the DLT rules and program logic, such as smart contracts, chain code, etc. The software applications from the application layer trigger the code and rules in the execution layer and instruct the code in the execution layer, which results in the execution of a transaction. In cases where the execution layer code requires data from off-chain databases, the code can trigger oracles that reside in the application layer (or between the application layer and the outside world) to fetch data/information from off-chain sources to the execution layer code.

C. Consensus Layer. The consensus layer is a critical component of DLT technologies facilitating distributed trust, ownership, and control. In this, widespread consensus-forming nodes across different geographical

and network locations work independently toward the consensus of transactions. There are two common consensus types: (1) lottery-based and (2) voting-based consensus. Consensus has two main properties: (1) indicates an agreement among the distributed nodes and synchronizes them, and (2) validates transactions and ensures reliable and fault-tolerant operations.

D. Data Model Layer. This layer handles functions and operations related to block creation and ledger maintenance tasks. This layer does not define the final ledger state, and a global consensus is required to approve the final transactions and block creations. However, the process of grouping the transactions into the block, creating a block (or appending to the ledger), maintaining a common state of the ledger, etc., are handled in this layer. Functions in this layer are primarily related to data orchestration processes but in the context of distributed databases, ledgers, etc. Examples of such processes are grouping or arranging the transactions into blocks, appending the block to the distributed ledger, and updating data-structure/ledger across the network (e.g., via replication). The block content or structure of a block depends on the blockchain/DLT technology. When transactions are submitted to a blockchain network, the transactions are ordered in a block.

E. Network Layer. The network layer corresponds to the communication infrastructure facilitating transaction, information, and data sharing between the nodes. Protocols and methods to facilitate discovery and communication between peer nodes belong in this layer. If nodes are expected to transact by digitally signing data-in-transit or engage in verification and validation of the transactions, such processes should be defined in this layer. Transport Layer Security (TLS) and other secure node-to-node handshaking mechanisms should be identified under this layer. Standard protocols are recommended instead of custom-defining new/proprietary protocols.

F. Infrastructure Layer. This layer corresponds to the virtual and physical computers or software agents participating as authorized blockchain nodes. The nodes should be capable of performing cryptographic operations (such as digital signature and hashing), maintaining, and varying the identity of other nodes, and providing identity information for authentication and authorization by the network/other nodes. Depending on the DLT, security aspects related to Membership Service Provider (MSP) and Active Directory (AD) fall within this layer. Hence, tools and processes facilitate access controls, define the identity of the nodes, and ensure permissions belong in this layer as part of the nodes. Furthermore, aspects related to on-chain and off-chain storage infrastructures are included in this layer.

G. Physical Layer. This layer may not be relevant in several use cases. The use of DLT-based industrial use cases where Internet of Things (IoT) devices and sensors play a central role is an emerging trend [30]. However, in use cases where sensors and IoT devices are expected to participate in the blockchain, these systems are expected to be part of this layer. The sensor systems may not have the capacity or capability to join as nodes in the DLT directly. In such cases, the sensors would need to interact with the middleware agents as part of the infrastructure layer to participate in the DLT network.

SECTION VI. The Mitre ICS ATT&CK® Matrix

The MITRE ATT&CK® ICS matrix uses principles from the cyber kill chain [31] and information acquired from past cyber incidents to evaluate what an adversary can do to a system and how the adversary can accomplish their goals. A simplified overview of the MITRE ATT&CK® ICS Matrix tactics shown below illustrates the 12 tactics and what the adversary is trying to achieve:

1. *Initial Access:* The adversary is trying to use entry vectors to gain an initial foothold within an ICS environment. These techniques include compromising OT assets, Information Technology (IT) resources in the OT network, and external remote services and websites. They may also target third-party entities and users with privileged access. IT resources in the OT environment are also potentially vulnerable to the same attacks as enterprise IT systems. Trusted third parties of concern may include vendors, maintenance personnel, engineers, external integrators, and other outside entities involved in expected ICS operations. Vendor-maintained assets may include physical devices, software, and operational equipment. Initial access techniques may also leverage outside devices, such as radios, controllers, or removable media, to remotely interfere with and possibly infect OT operations.
2. *Execution:* The adversary is trying to run code or manipulate system functions, parameters, and data unauthorizedly. Execution consists of techniques that result in adversary-controlled code running on a local or remote system, device, or other assets. This execution may also rely on unknowing end users or manipulating device operating modes. Adversaries may infect remote targets with programmed executables or malicious project files that operate according to specified behavior and alter expected device behavior subtly. Commands for execution may also be issued from command-line interfaces, Application Programming Interfaces (APIs), GUIs, or other

available interfaces. Techniques that run malicious code may also be paired with techniques from other tactics.

3. *Persistence*: The adversary is trying to maintain their foothold in the ICS environment. Persistence consists of techniques that adversaries use to maintain access to ICS systems and devices across restarts, changed credentials, and other interruptions that could cut off their access. Techniques used for persistence include any access, action, or configuration changes that allow them to secure their ongoing activity and keep their foothold on systems. This may include replacing or hijacking legitimate code, firmware, and other project files, adding startup code, and downloading programs onto devices.
4. *Privilege Escalation*: The adversary is trying to gain higher-level permissions. Privilege Escalation consists of techniques adversaries use to gain higher-level permissions on a system or network. Adversaries can often enter and explore a network with unprivileged access but require elevated permissions to follow through on their objectives. Common approaches are to take advantage of system weaknesses, misconfigurations, and vulnerabilities.
5. *Evasion*: The adversary is trying to avoid security defenses. Evasion consists of techniques adversaries use to avoid technical defenses throughout their campaign. Techniques used for evasion include the removal of indicators of compromise, spoofing communications, and exploiting software vulnerabilities. Adversaries may also leverage and abuse trusted devices and processes to hide their activity, possibly by masquerading as master devices or native software. Methods of defense evasion for this purpose are often more passive.
6. *Discovery*: The adversary is locating information to assess and identify their targets in the environment. Discovery consists of techniques adversaries use to survey the ICS environment and gain knowledge about the internal network, control system devices, and how their processes interact. These techniques help adversaries observe the environment and determine the next steps for target selection and Lateral Movement. They also allow adversaries to explore what they can control and gain insight into interactions between various control system processes. Adversaries may use Discovery techniques that result in Collection to determine how available resources benefit their current objective.

7. *Lateral Movement:* The adversary is trying to move through the ICS environment. Lateral Movement consists of techniques adversaries use to enter and control remote systems on a network. These techniques abuse default credentials, known accounts, and vulnerable services and may also leverage dual-homed devices and systems that reside on both the IT and OT networks. The adversary uses these techniques to pivot to their next point in the environment, positioning themselves to where they want to be or think they should be. Reaching this objective often involves pivoting through multiple systems, devices, and accounts. Adversaries may install their remote tools to accomplish Lateral Movement or leverage default tools, programs, manufacturer set, or other legitimate credentials native to the network, which may be stealthier.
8. *Collection:* The adversary is trying to gather data of interest and domain knowledge on the ICS environment to inform their goal. The collection consists of adversaries' techniques to gather domain knowledge and obtain contextual feedback in an ICS environment. This tactic is often performed as part of Discovery to compile data on control systems and targets of interest that may be used to follow through on the adversary's objective. Examples of these techniques include observing operation states, capturing screenshots, identifying unique device roles, and gathering system and diagram schematics. Collection of this data can play a key role in planning, executing, and even revising an ICS-targeted attack. Collection methods depend on the targeted data categories, including protocol-specific, device-specific, and process-specific configurations and functionality. Information collected may pertain to system, supervisory, device, and network-related data, which conceptually fall under high, medium, and low levels of plan operations.
9. *Command and Control:* The adversary is trying to communicate with, and control compromised systems, controllers, and platforms with access to the ICS environment. Command and Control consist of techniques adversaries use to communicate with and send commands to compromised systems, devices, controllers, and platforms with specialized applications in ICS environments. Adversaries often seek to use commonly available resources and mimic expected network traffic to avoid detection and suspicion. Command and Control may be established to varying degrees of stealth, often depending on the victim's network structure and defenses.

10. *Inhibit Response Function:* The adversary tries to prevent safety, protection, quality assurance, and operator intervention functions from responding to a failure, hazard, or unsafe state. Inhibit Response Function consists of adversaries' techniques to hinder the safeguards for processes and products. These techniques aim to actively deter and prevent expected alarms and responses due to statuses in the ICS environment. Adversaries may modify or update system logic or even outright prevent responses with a Denial-of-Service (DoS). They may result in the prevention, destruction, manipulation, or modification of programs, logic, devices, and communications. As prevention functions are generally dormant, reporting and processing functions can appear fine but may have been altered to prevent failure responses in dangerous scenarios.
11. *Impair Process Control:* The adversary tries to manipulate, disable, or damage physical control processes. Impair Process Control consists of techniques that adversaries use to disrupt control logic and cause detrimental effects to processes being controlled in the target environment. Targets of interest may include active procedures or parameters that manipulate the physical environment. These techniques can also include preventing or manipulating reporting elements and control logic. If an adversary has modified process functionality, they may also obfuscate the results, which are often self-revealing in their impact on the outcome of a product or the environment. The direct physical control these techniques exert may also threaten the safety of operators and downstream users, which can prompt response mechanisms.
12. *Impact:* The adversary is trying to manipulate, interrupt, or destroy the ICS systems, data, and their surrounding environment. Impact consists of techniques that adversaries use to disrupt, compromise, destroy, and manipulate the integrity and availability of control system operations, processes, devices, and data. These techniques encompass the influence and effects of adversarial efforts to attack the ICS environment or that tangentially impact it. Impact techniques can result in more instantaneous disruption to control processes and the operator or long-term damage or loss to the ICS environment and related operations.

SECTION VII. DLT Stack and Mapping Process Flow

This paper assesses the DLT stack against the ATT&CK[®] ICS matrix to understand the adversarial pivot points better. This includes identifying an adversary's tactics and techniques to compromise one or more DLT stack

layers. Tactics represent the “why” of an attack technique. It is the adversary’s tactical goal. The techniques represent “how” an adversary achieves a tactical goal by acting. The following analysis is based on the specific language and descriptions included in the MITRE ATT&CK® ICS techniques.² The IEEE SA P2418.5 working group’s cybersecurity task force investigates cyber resilience of DLT-based energy applications in the ICS/OT environment against the MITRE ATT&CK® ICS matrix through the following activities:

1. The task force has been assessing the mapping between the DLT stack against the MITRE ATT&CK® ICS matrix. Mapping the stack and the matrix can help identify the implications of various tactics and techniques that an adversary may use to compromise one or more of the seven DLT stack layers. This can provide valuable insights into possible exploits in DLTs and assist in defining security and resilience measures;
2. Identifying these tactics/techniques can also be used in a risk assessment for an ICS/OT system or network. The objective is to identify mitigation strategies to address cybersecurity resilience, including proactive and reactive cybersecurity controls;
3. The task force aims to use the mapping to provide guidance that can assist DLT designers and DLT users (application owners) in increasing the overall cyber resilience and cybersecurity for energy sector ICS/OT systems and networks — leading to a cyber resilient model;
4. A cyber resilience model should support a comprehensive cyber risk management framework with a well-defined multi-layer and preventive cyber security controls and measures to safeguard the entire digital chain. Furthermore, an acceptable risk policy should allow all market agents with different cyber risk appetites and tolerance thresholds that differ according to the asset size and technology to self-assess their risk exposure according to the same set of rules.

Figures 2 and 3 illustrate the process used to develop the mapping of the MITRE ATT&CK® ICS tactics and techniques against the layers in the DLT stack. Figure 2 illustrates the process used to identify the various tactics and techniques applicable at each layer in the DLT stack. Figure 3 expands upon Figure 2 and shows the overall process flow of the analysis as a component of a cyber resilience model process.

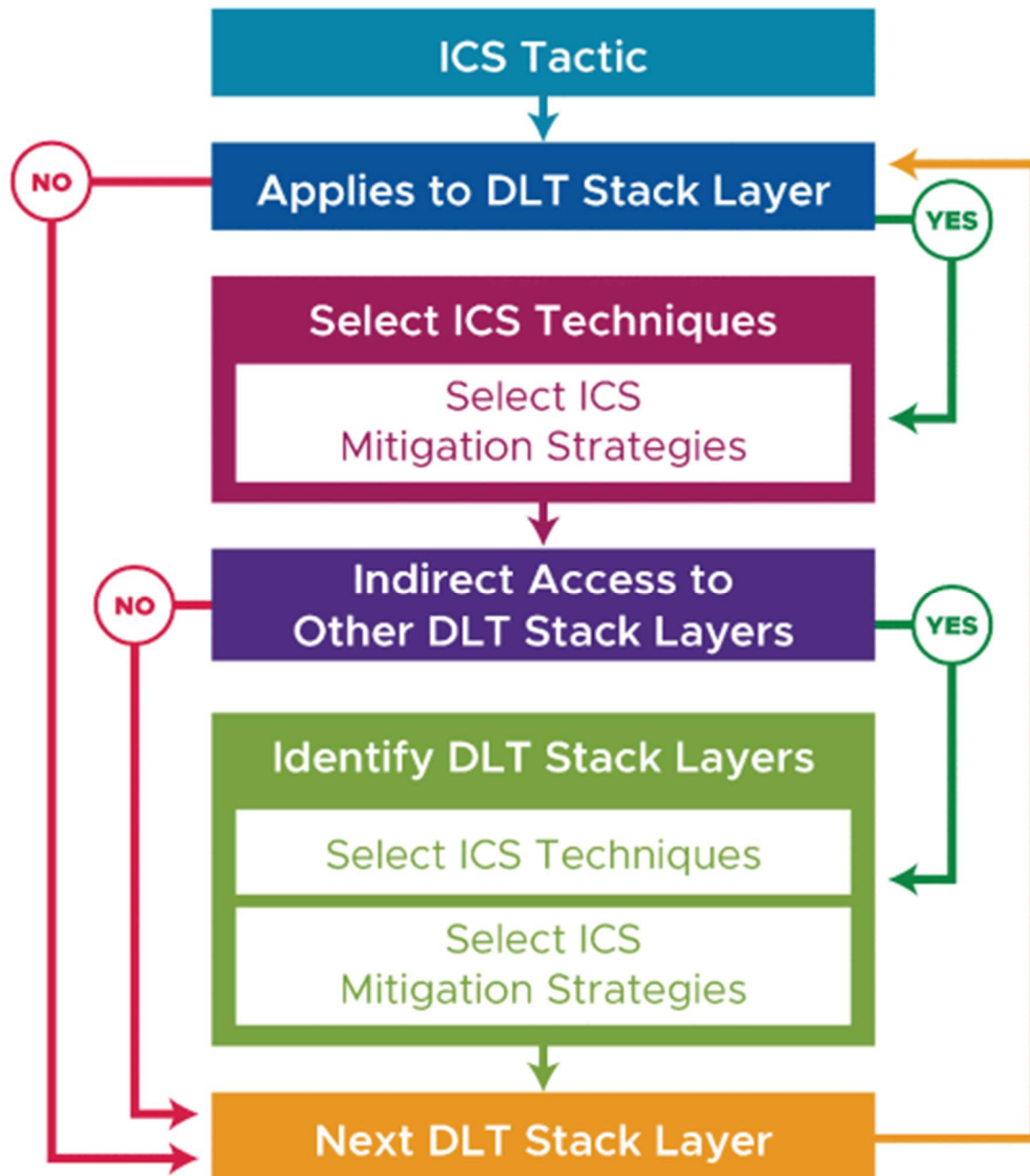


FIGURE 2. Analysis Process.

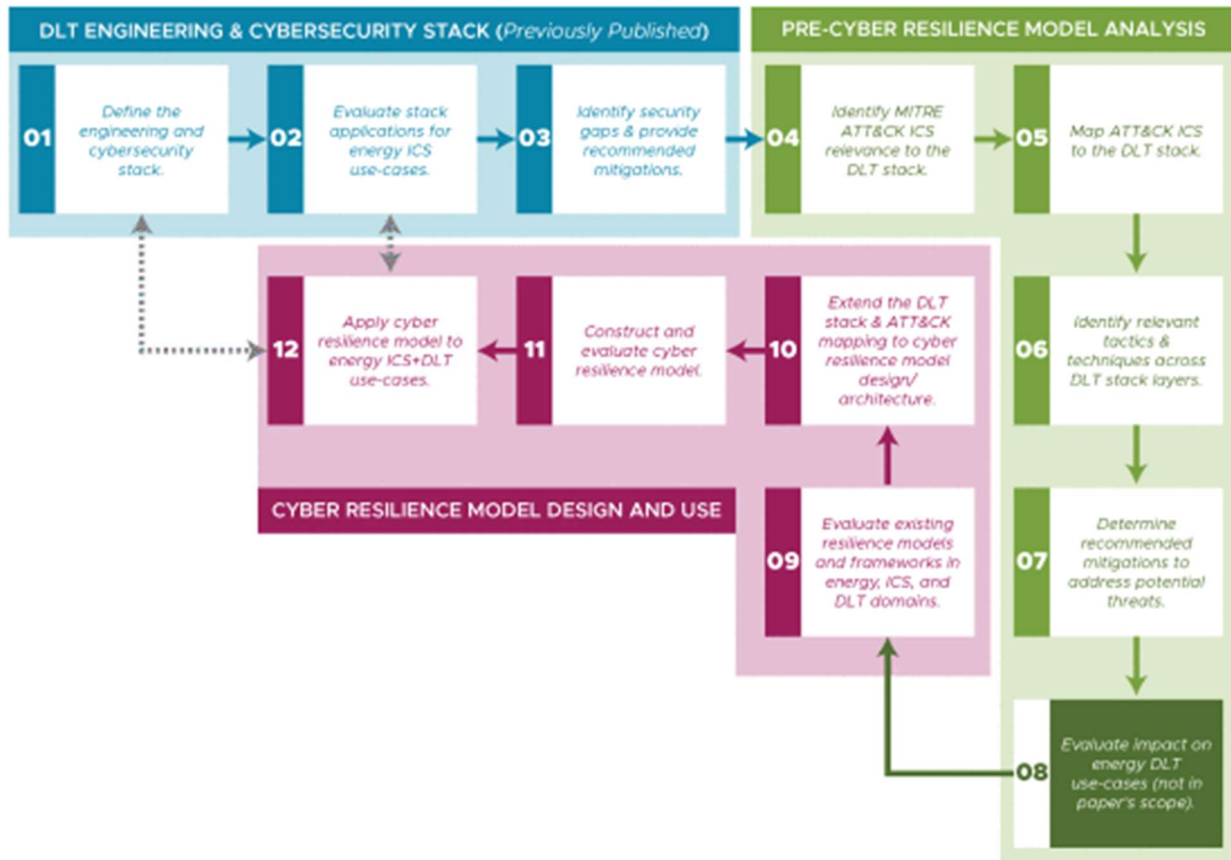


FIGURE 3. The Cyber Risk Management Process Flow, it illustrates how the team has structured the CRM flow across three staggered phases.

The first three steps involve the development builds of the DLT stack, which was completed by the task force and presented in [9]. Steps 4–8 of 3 encompass the pre- CRM analysis that evaluates the relevance of various tactics and techniques from the MITRE ATT&CK® ICS matrix against the DLT stack. Using the mapping information between the DLT stack and the MITRE ATT&CK® ICS Matrix along with the DLT stack descriptions, the final steps include the completion of the CRM phase (steps 9–12 of Figure 3).

Since the sequence on which the tactics listed by MITRE (left-to-right) is based on adversarial propagation on a kill chain or similar construct, the same sequence has been followed for this mapping. Starting with the first tactic (Initial Access), each technique has been revised to evaluate its applicability to each DLT stack layer. If the tactic applies, the relationship is further analyzed to identify the specific techniques and mitigation strategies. This process is repeated for all of the 12 tactics. The result is a detailed map between the DLT stack and the MITRE ATT&CK® ICS tactics and techniques. The mitigation recommendations (step 7 of Figure 3) are selected from

MITRE's existing mitigations for the tactics and techniques. After an initial analysis, indirect access is analyzed to determine if an adversary can compromise DLT components at other stack layers.

The determination of indirect implications was made based on the tactic that was being analyzed, the objective of that tactic, and how this would apply at each DLT stack layer. Direct vs. indirect implications was based on the following approach: 1) first, the objective of a tactic in the context of ICS environment was analyzed; then 2) a tactic's direct application/relevance to a DLT layer was evaluated. This was done by studying the techniques that are listed under a particular tactic in scope. In addition, "indirect implications" can be synonymously thought of as "indirect impact." Specifically, if an adversary's action could impact a different stack layer's component, that can be defined as "indirect access."

The analysis presented in this work is based on several assumptions, as cyber resilience and cybersecurity are broad areas for any application. The assumptions of the present work are:

Assumption 1: Every DLT architecture includes the following three components:

1. Smart contracts or similar artifacts that allow defining rules, programs, etc.
2. A distributed ledger or a database-style artifact that is immutable and captures transactional history.
3. A consensus mechanism or similar artifact that allows for fault-tolerant (crash or byzantine) group agreement and is subject to a governance model.

Assumption 2: Immutability of the ledger is achieved through cryptography. If the cryptographic algorithm is compromised, the immutability of the ledger should be assumed to be compromised.

Assumption 3: The DLT includes the seven layers defined in the DLT stack.

Assumption 4: Only permissioned/private DLT is encouraged to use in ICS/OT applications/use cases in the energy sector. This is important because the participating entities (e.g., utilities, regulators, auditors, etc.) require mechanisms to ensure non-repudiation in case of anomalous behavior.

Assumption 5: The following common security practices are implemented for the ICS/OT systems and networks: configuration management, auditing, system monitoring, a security software design and development, secure product life-cycle management, implementation of a security framework, and security patch management.

Assumption 6: Many general cybersecurity controls address data security and privacy across different vendors and manufacturers. Many of these cybersecurity controls are outside the scope of a DLT. Routine auditing of the DLT ledgers should be an ongoing process to detect malicious activity.

Assumption 7: Use case-specific threat models and attack trees are outside the scope of this analysis. The paper focuses on assessing potential compromises to a DLT and not on specific OT/ICS use cases. Specific use cases can be developed in future papers. Given this focus, false data injection from distributed IoT devices and distributed bots, etc., are outside the scope of this paper, even though they may adversely impact the DLT.

Assumption 8: No backdoors have been implemented to infiltrate an organization that develops/maintains a DLT. If backdoors have been installed, exploitation is outside the scope of the DLT.

Assumption 9: Attacks described below are limited to the DLT environment and the DLT stack. Attacks at the system level are outside the scope of this paper.

Assumption 10: The effects of the DLT making or influencing decisions in the OT environment are outside the scope.

Assumption 11: The effects of incorrect/invalid or missing data incoming from field devices to a DLT is outside the scope.

Assumption 12: Use of a DLT for command-and-control operations is outside the scope of this paper.

Assumption 13: This analysis focuses solely on the MITRE ATT&CK® ICS Matrix tactics and techniques and the specific language in the technique. Some techniques reference devices and processes in the OT environment, such as PLCs, controllers, process control, safety systems, and OT networks. This assessment did not expand upon these references. The MITRE ATT&CK® Enterprise Matrix has additional tactics and techniques that may be exploited to compromise a DLT. These additional tactics and techniques should be considered for future efforts.

SECTION VIII. Mapping Methodology

Each diagram below includes an ICS tactic and identifies whether an adversary may potentially use the tactic to compromise one (or more) layers in the DLT stack. If a tactic is selected, the applicable techniques are identified. Layers that cannot be directly compromised using the tactic are marked with an X. In addition, if a DLT

stack layer is directly compromised, an adversary may access other DLT stack layers using the same techniques or other techniques within the tactic. This is defined as indirect access, as illustrated in the diagrams with a dotted line from the initial point of compromise. In addition, some of the techniques are applicable in specific use cases. These techniques in each figure are in black letters. In each of the below sub-sections, a table of mitigations is presented after the techniques of a tactic are mapped to the DLT stack layers. The mitigations are directly exported from MITRE's guidance. DLT-tailored guidance on operationalizing the mitigations is shown in the Appendix.

The tactics cannot be assessed independently of each other. Initially, an adversary must be successful at the Initial Access and Execution tactics. If an adversary cannot access a DLT stack layer and execute techniques, the other tactics and techniques are not applicable. Compromises using Persistence, Privilege Escalation, and Lateral Movement tactics may result in a more significant effect at the different DLT stack layers.

A. Initial Access. *Initial Access* consists of techniques that adversaries may use as entry vectors to gain an initial foothold within an ICS environment. Following is an analysis of the applicability of the Initial Access tactic to the DLT stack layers and the rationale. Figure 4 illustrates the analysis.

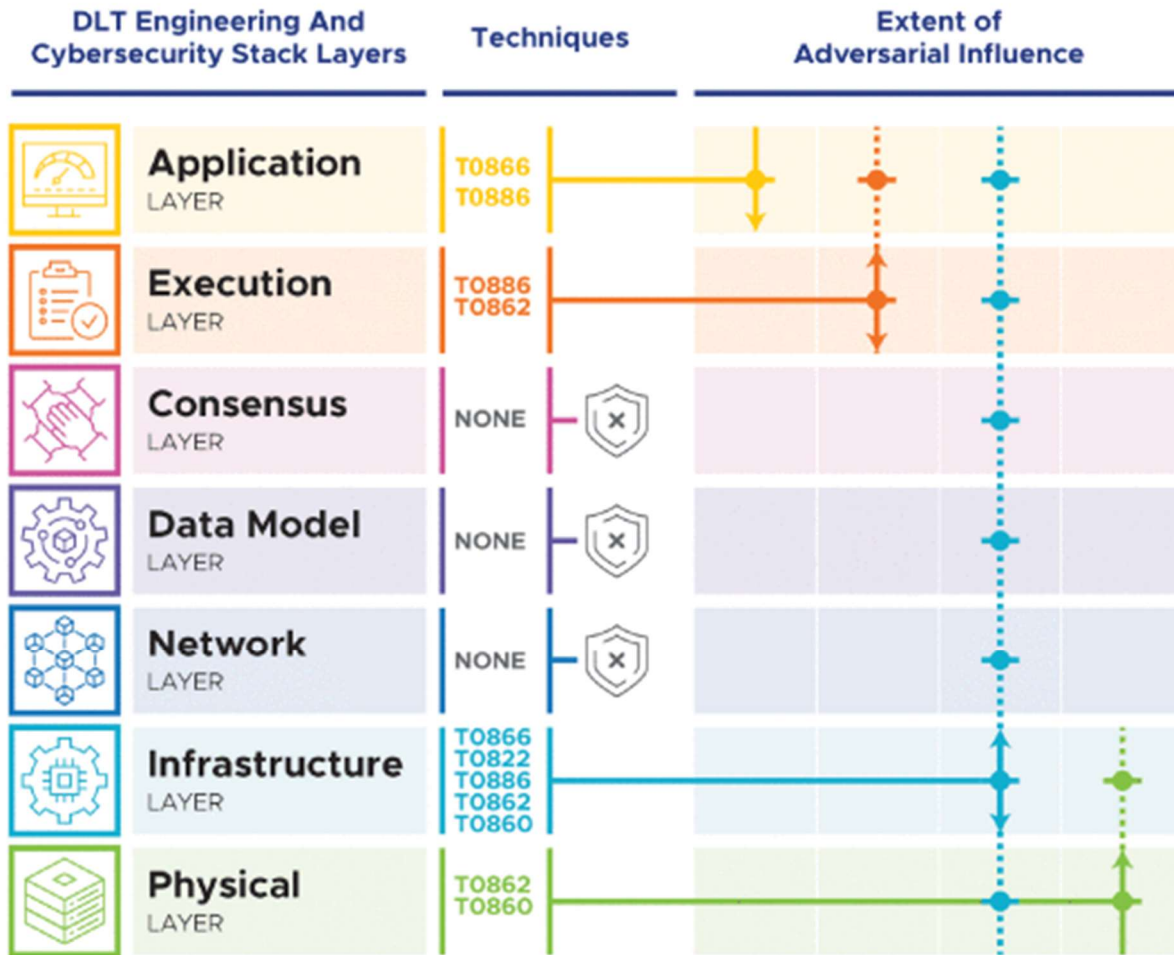


FIGURE 4. Initial access tactic analysis and mapping to the DLT stack.

1) Application Layer. This layer includes applications that trigger rule-bases and program code, APIs, UI/GUIs, Oracles, distributed applications, marketplace, monetization, etc. Because the application layer includes software and applications that interact with the DLT, this tactic applies. The specific techniques that may be used are: *Exploitation of Remote Services (T0866)* and *Remote Services (T0886)*. For most DLTs, these external applications only have limited connectivity to a DLT's components listed in *Assumption 1* above. Therefore, if any of the techniques listed under the Initial Access tactic are used to gain an initial foothold, other tactics or techniques may be used to compromise the DLT applications (e.g., corrupting the source code of the DLT application or DLT application patch through supply chain attack on libraries, etc.). Also, an adversary should be unable to modify the smart contract interacting with the application.

2) Execution Layer. This layer deals with rule-bases and program code, for example, smart contracts, chain code, atomic swaps (exchanging cryptocurrencies between the cryptographic networks), tokens, etc. Because this layer contains the DLT rules and program logic, the Initial Access tactic is relevant at this layer; the adversary may use the following techniques to gain initial access:

- *Remote Services (T0886)* for accessing the code;
- An uncommon technique would be *Supply Chain Compromise (T0862)* for unauthorized access to code developed by vendors/third parties. Because the execution layer deals with the governing program logic/code that is integral to the DLT, gaining the initial foothold at this layer's components could allow the adversary to have an impact at this layer using other tactics or techniques and traverse to the application layer (indirect access);
- If a smart contract is compromised and a transaction is approved based on the compromised smart contract, the transaction would be treated as valid, and the respective block would also be considered valid. Therefore, this would potentially have an impact on the data model layer.

3) Consensus, Data Model, and Network Layers. Unlike the components at the Application and Execution layers, these three layers have process components such as the following:

1. Process or ability to form consensus;
2. Process or ability to create blocks, and
3. Process or ability to transact over the network.

In a DLT implementation, the network layer includes the underlying blockchain platform protocols, such as Hyperledger, Ethereum, etc. The network layer also includes the communication infrastructure that supports the DLT and facilitates transactions, information, and data sharing between the nodes. Although the organization-level network infrastructure, such as routing/switching, etc., is required to host DLT nodes, compromising the network infrastructure by exploiting network system vulnerabilities can be done with or without DLT. Therefore, direct compromise of network infrastructure to pivot into the DLT environment is out of the scope of this assessment. However, if an adversary compromises a DLT node and pivots into the network infrastructure, that will be in scope (covered under the Infrastructure layer).

The Initial Access tactic does not apply to these three layers. If applicable, the primary concern would be a *Supply Chain Compromise (T0862)* technique for unauthorized modification of the protocols.

4) Infrastructure Layer. This layer deals with data storage entities, logical DLT nodes, virtual machines, clusters, and containers. An adversary may be able to gain initial access to the physical and/or virtual components (e.g., gain initial access to a virtual machine running the DLT node - transacting node, consensus node, or ordering node). The nodes should be capable of performing cryptographic operations, providing information for authentication and authorization by the network/other nodes, and configuring on-chain and off-chain storage. Depending on the compromised node(s) to which the threat actor gained initial access, the adversary may gain access to the other six layers of the DLT stack. The applicable techniques are: *Exploitation of Remote Services (T0866)*, *External Remote Services (T0822)*, *Remote Services (T0886)*, *Supply Chain Compromise (T0862)*, and *Wireless Compromise (T0860)*. Network-level hardening and host-level hardening are out of scope. Therefore, aspects of cybersecurity best practices such as network segmentation, segregation through firewall rules, Virtual Local Area Networks (VLANs), etc., are out of scope.

5) Physical Layer. This layer pertains to the systems participating on behalf of the users and lower-level ICS/OT systems such as sensors, IoT devices with Unique Identifiers (UIDs), Operating System (OS)/firmware, etc. Typically, sensors and IoT devices are external to the DLT and provide data to the DLT. If the sensors and IoT devices are external to the DLT, the Initial Access Tactic will not apply. However, if these devices are included in the DLT, techniques such as *Supply Chain Compromise (T0862)* and *Wireless Compromise (T0860)* may be used. The two techniques will allow an attacker to access the device. If a threat actor gains initial access to the physical layer, this may allow indirect access to the infrastructure layer. As described above, the other DLT stack layers may be accessed once the infrastructure layer is compromised.

6) Mitigation. Table 1 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Initial Access tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 1 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Initial Access Tactic

Technique	Mitigation
Exploitation of Remote Services (T0866)	M0948 Application Isolation and Sandboxing M0926 Privileged Account Management
Remote Services (T0886)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication M0918 User Account Management
Supply Chain Compromise (T0862)	M0945 Code Signing M0817 Supply Chain Management
External Remote Services (T0822)	M0918 User Account
Management Wireless Compromise (T0860)	M0813 Software Process and Device Authentication

B. Execution. The adversary is trying to run code or manipulate system functions, parameters, and data unauthorizedly. Following is an analysis of the applicability of the Execution tactic to the DLT stack layers and the rationale. Figure 5 illustrates this analysis.

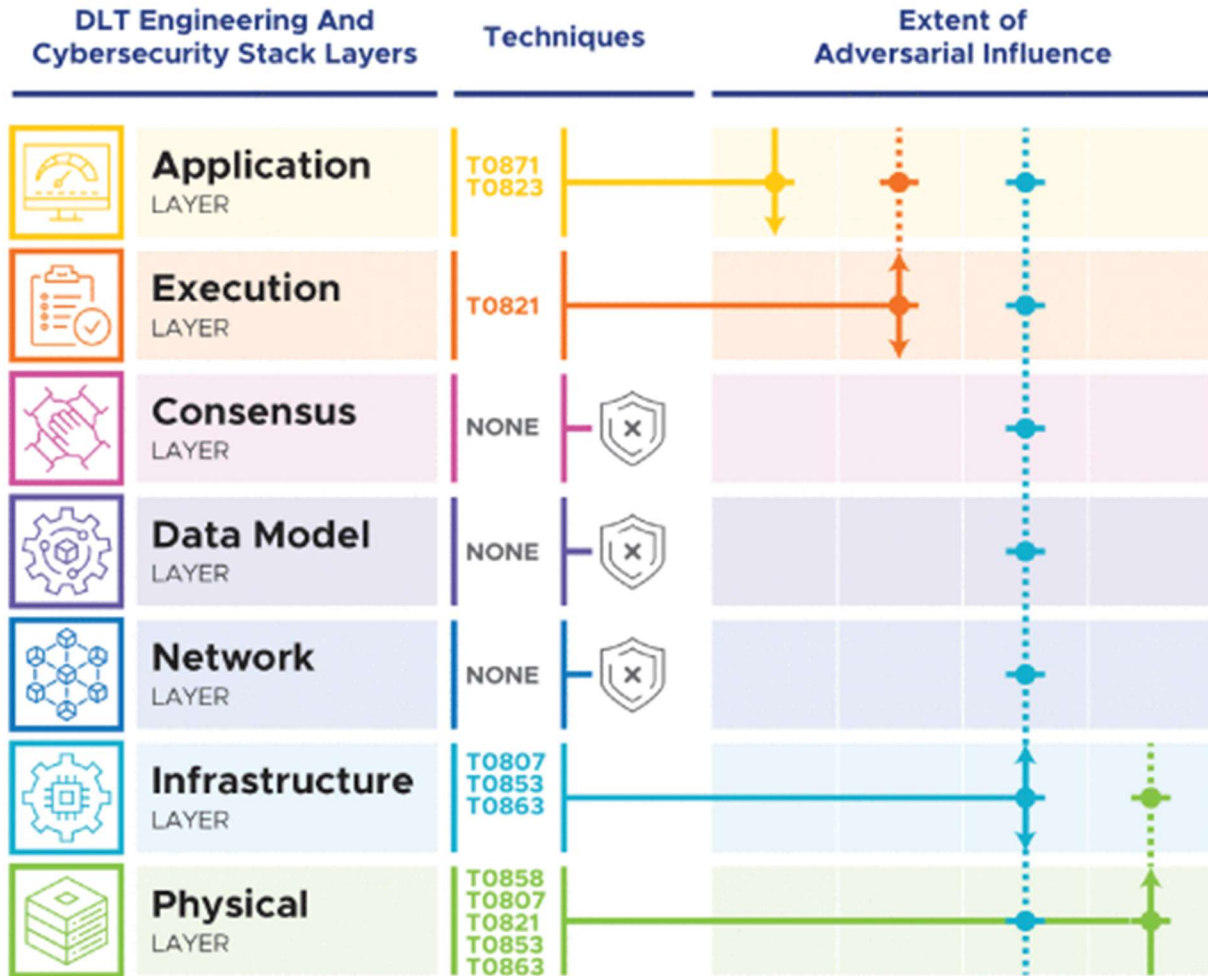


FIGURE 5. Execution tactic analysis and mapping to the DLT stack.

1) Application Layer. As the application layer contains APIs, the Execution tactic applies, particularly the *Execution through API (T0871)* and *Graphical User Interface (T0823)* techniques that would allow an adversary to alter the DLT API and UI/GUIs.

2) Execution Layer. One focus of the Execution tactic is to run adversary-controlled code. This malicious code may alter the way the DLT rule-bases and program codes operate. Therefore, the Execution tactic applies, specifically the *Modify Controller Tasking (T0821)* technique that will allow an adversary to modify the rules and code. The result could be the execution of malicious programs or the manipulation of rule-based execution flow.

3) Consensus and Data Model Layers. The Execution tactic will not apply to these two layers as it focuses on running adversary-controlled code and maliciously manipulating parameters and data. Note

that, the data model layer focuses on the model structure and associated parameters. However, any alteration of the consensus protocol should be quickly identified.

4) Network Layer. The Network layer focuses on communications and connectivity. The Execution tactic will not apply as it focuses on running adversary-controlled code and maliciously manipulating parameters and data. As mentioned, any attack that targets the network infrastructure directly instead of propagating through a DLT is out of scope, including attacks such as Distributed Denial-of-Service (DDoS), etc., that directly target the network infrastructure.

5) Infrastructure Layer. This layer comprises virtual and physical computers or software agents participating as authorized blockchain nodes. Because this layer also includes middleware and applicable tools and processes for the DLT, the Execution tactic applies. The specific techniques are: *Command-Line Interface (T0807)*, *Scripting (T0853)*, and *User Execution (T0863)*. The adversary's objective would be to maliciously alter the middleware/processes to ensure the DLT did not function correctly. An adversary may have indirect access to the other six DLT stack layers if the infrastructure layer is compromised. If an adversary moves/pivots into the network layer, the *Scripting (T0853)* and *User Execution (T0863)* techniques may be used in the network layer.

6) Physical Layer. Typically, sensors and IoT devices are external to the DLT and provide data to the DLT. The Execution tactic focuses on modifying: controller operating mode, command line interfaces, APIs/GUIs, and controller tasking. Because these are outside the scope of the DLT physical layer, the Execution tactic does not apply. However, if these devices are included in the DLT, techniques such as *Change Operating Mode (T0858)*, *Command Line Interface (T0807)*, *Modify Controller Tasking (T0821)*, *Scripting (T0853)*, and *User Execution (T0863)* may be used. All five techniques will allow an attacker to access the device. If the physical layer is compromised, an adversary may have indirect access to the infrastructure layer.

7) Mitigation. Table 2 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Execution tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 2 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Execution Tactic

Technique	Mitigation
Execution through API (T0871)	M0801 Access Management M0800 Authorization Enforcement M0938 Execution Prevention M0804 Human User Authentication
Graphical User Interface (T0823)	M0816 Mitigation Limited or Not Effective
Modify Controller Tasking (T0821)	M0945 Code Signing
Command-Line Interface (T0807)	M0942 Disable or Remove Feature or Program
Scripting (T0853)	M0948 Application Isolation and Sandboxing
User Execution (T0863)	M0945 Code Signing M0938 Execution Prevention M0921 Restrict Web-Based Content
Change Operating Mode (T0858)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication

C. Persistence. The adversary is trying to maintain a foothold in the ICS environment. Following is an analysis of the applicability of the Persistence tactic to the DLT stack layers and the rationale. Figure 6 illustrates this analysis.

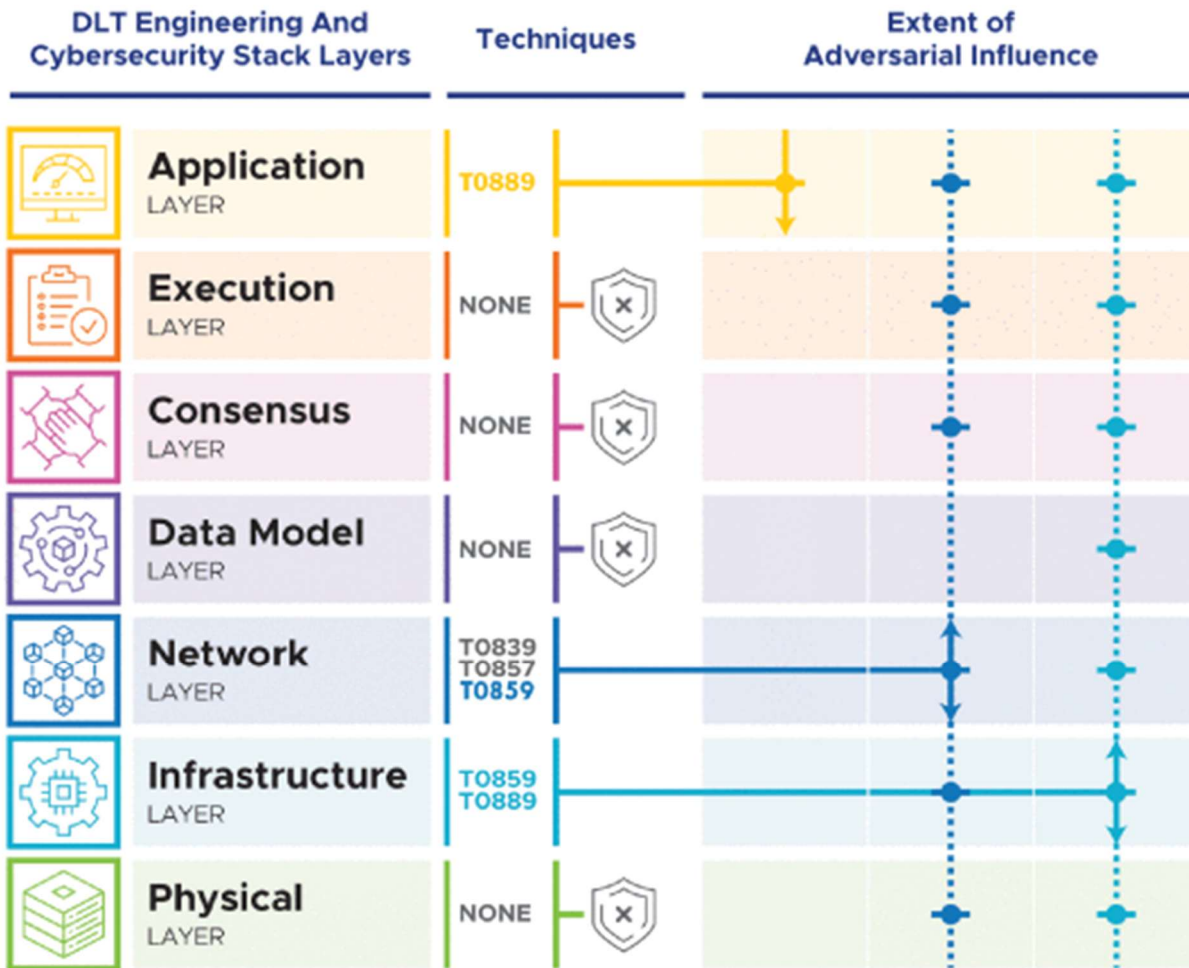


FIGURE 6. Persistence tactic analysis and mapping to the DLT stack.

1) Application Layer. The Persistence tactic may be applicable if the API under the application layer enables users to modify either 1) application-specific settings or 2) DLT platform-specific settings. The applicable technique is *Modify Program (T0889)*. For example, if a grid-based application allows users to store grid topology data, deliberate errors could cause dependent applications to fail, yield incorrect results, or help the attacker masquerade future attacks. A second example may be applicable if the API allows remote configuration of the underlying physical/network infrastructure (such as DLT node properties). Mitigation mechanisms include performing sanity checks, validations to ensure bad data is not transmitted, and ongoing checks to validate/flag previously stored data.

2) Execution Layer. This layer deals with Rule-bases and program code. Examples include smart contracts, chain code, atomic swaps, tokens, etc. Therefore, the Persistence tactic does not apply as the associated techniques target and focus on controllers and PLCs.

3) Consensus Layer. The consensus layer includes the decision rules in the DLT system. Within permissioned networks, the credential management system plays a key role in enabling consensus. Typically, the credential management system is outside the scope of the DLT system. Therefore, the Persistence tactic does not apply.

4) Data Model Layer. Direct influences to synchronization, services, block creation, chain structure, or hashing are unrealistic using privilege escalation. Based on the current knowledge and known architectures, this layer cannot be reached directly using Persistence techniques.

5) Network Layer. Based on current knowledge and known architectures, this layer can be compromised directly using the following Persistence techniques. An adversary may indirectly access the other DLT stack layers if the network layer is compromised.

- *Valid Accounts (T0859):* Within a permissioned DLT environment, peer participation, and thus access permissions, are tied to a credential management system. Since such a system is a key component, it must be engineered to:
 1. Prevent unauthorized systems from obtaining valid credentials;
 2. Identify and handle credential theft (from the server side);
 3. Limit the risks and scenarios in which the credential management system can restrict or limit valid transactions.

In addition, peers must be diligent in preventing and reporting credential theft. This could be done proactively (e.g., periodic credential renewals) or actively (e.g., analyzing ledger operations vs. submitted transactions).

Although the following techniques cannot be directly exploited/used through the DLT, exploits can be used to disrupt DLT operations

- *Module Firmware (T0839):* Adversaries may install malicious or vulnerable firmware onto network hardware devices;

- *System Firmware (T0857)*: An attacker may install malicious or out-of-date firmware that could be used to disrupt the DLT.

6) Infrastructure Layer. Data storage entities and logical blockchain nodes may include traditional virtual machines, virtual containers (e.g., Kubernetes), and cloud-managed infrastructures. The Persistence tactic is applicable. If the infrastructure Layer is compromised, an adversary may have indirect access to the other DLT stack Layers. Similar to the Consensus layer, the following technique should be considered in the system's design.

- *Valid Accounts (T0859)*: The infrastructure Layer can be compromised by abusing or exploiting the management accounts. This could lead to the exploitation of additional tactics, such as a system stop, data extraction, or rootkit installation, which may lead to a full system compromise. Therefore, infrastructure-related accounts must be protected using industry best practices, such as multi-factor authentication, periodic password changes, and least privilege approaches;
- *Modify Program (T0889)*: The infrastructure Layer can be compromised by modifying a program that manages the DLT infrastructure.

7) Physical Layer. This Layer pertains to the systems participating on behalf of the users and lower-level ICS/OT systems such as sensors, IoT devices with UIDs, OS/firmware, etc. Although a wide variety of persistence techniques can be applied, these are generic and do not necessarily reflect the vulnerabilities introduced by DLT platforms.

8) Mitigation. Table 3 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Persistence tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 3 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Persistence Tactic

Technique	Mitigation
Modify Program (T0889)	M0945 Code Signing
Module Firmware (T0839)	M0801 Access Management M0945 Code Signing M0813 Software Process and Device Authentication
System Firmware (T0857)	M0801 Access Management M0945 Code Signing M0804 Human User Authentication M0813 Software Process and Device Authentication
Valid Accounts (T0859)	M0801 Access Management M0926 Privileged Account Management M0918 User Account Management

D. Privilege Escalation. The adversary is trying to gain higher-level permissions. Regarding DLT systems, privilege escalation [32] may not significantly affect unless the adversary can gain higher-level permissions to most DLT nodes. This will also depend on the specific DLT architecture. Following is an analysis of the applicability of the Privilege Escalation tactic to the DLT stack layers and the rationale. Figure 7 illustrates the analysis.

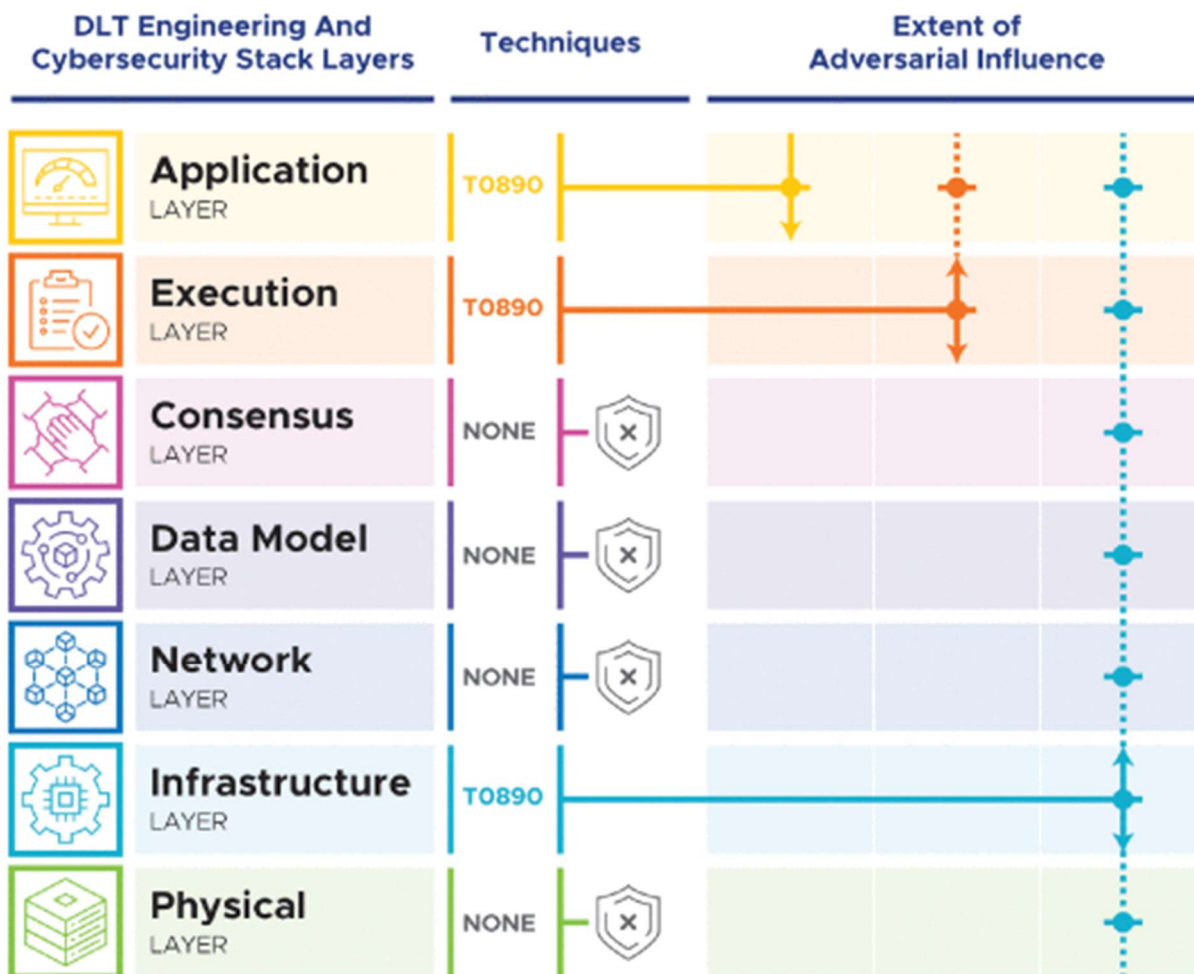


FIGURE 7. Privilege Escalation tactic analysis and mapping to the DLT stack.

1) Application Layer. An adversary successfully exploiting the privilege escalation tactic may use other tactics and techniques to compromise the DLT application layer nodes. The applicable technique is *Exploitation for Privilege Escalation (T0890)*.

Although *Hooking (T0874)* may not be applicable in this layer, the risk of an attacker exploiting a low-level API management call to gain access to underlying levels is a possibility. For example, DLT Software Development Kits (SDKs) may expose configurable endpoints to define execution or consensus options that may be abused to inject malicious hooks. This vulnerability should be identified and addressed in the secure system design.

2) Execution Layer. This layer deals with Rule-bases and program code. Examples include smart contracts, chain code, atomic swaps, tokens, etc. Smart contracts' vulnerabilities and errors (intentional or accidental) can lead to privilege escalation. Compromising or usurpation of tokens can cause malfunctions in the DLT system. The applicable technique is *Exploitation for Privilege Escalation (T0890)* within an application user space context.

3) Consensus Layer. The consensus layer includes the decision rules in the DLT system. Since there are consensus algorithms (e.g., PoS and PoA) that are based on a node's reputation, this tactic is applicable. An attacker who has gained access to a node using the Initial Access tactic can behave like an honest network member, waiting for the opportune moment after having accumulated enough reputation to try to influence the result of the decision in a particular situation. However, the current version of the ICS matrix does not have a directly applicable technique but could be considered an extension of *Exploitation of Privilege Escalation (T0890)*.

4) Data Model Layer. Direct influences on synchronization, services, block creation, chain structure, or hashing are unrealistic using Privilege Escalation techniques.

5) Network Layer. The Network Layer cannot be reached directly using Privilege Escalation techniques.

6) Infrastructure Layer. This is the most critical layer. Once an adversary successfully implements the Initial Access tactic and obtains access to a DLT node, the Privilege Escalation tactic may be used to gain control over the data storage entity or local blockchain node. After the other DLT stack layers are compromised using the Initial Access tactic, an adversary may exploit software vulnerabilities to elevate privileges. The level of compromise will depend on the security architecture. The applicable technique is *Exploitation for Privilege Escalation (T0890)*. The applicability is by compromising software systems in the Application and Execution layers through the compromised infrastructure. Once the infrastructure layer is compromised, an adversary may have indirect access to the other six DLT stack layers.

7) Physical Layer. Privilege escalation on this layer in DLT systems is not applicable because of the distributed structure. Alternatively, the physical layer could be indirectly accessed through the infrastructure layer.

8) Mitigation. Table 4 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Privilege Escalation tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 4 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Privilege Escalation Tactic

Technique	Mitigation
Exploitation for Privilege Escalation (T0890)	M0948 Application Isolation and Sandboxing

E. Evasion. An adversary uses evasion techniques to avoid security controls. Evasion’s techniques can only be implemented after an adversary has reached a DLT node and its specific functionality. This tactic is applicable only at three layers: application, infrastructure, and physical. Figure 8 illustrates the analysis.

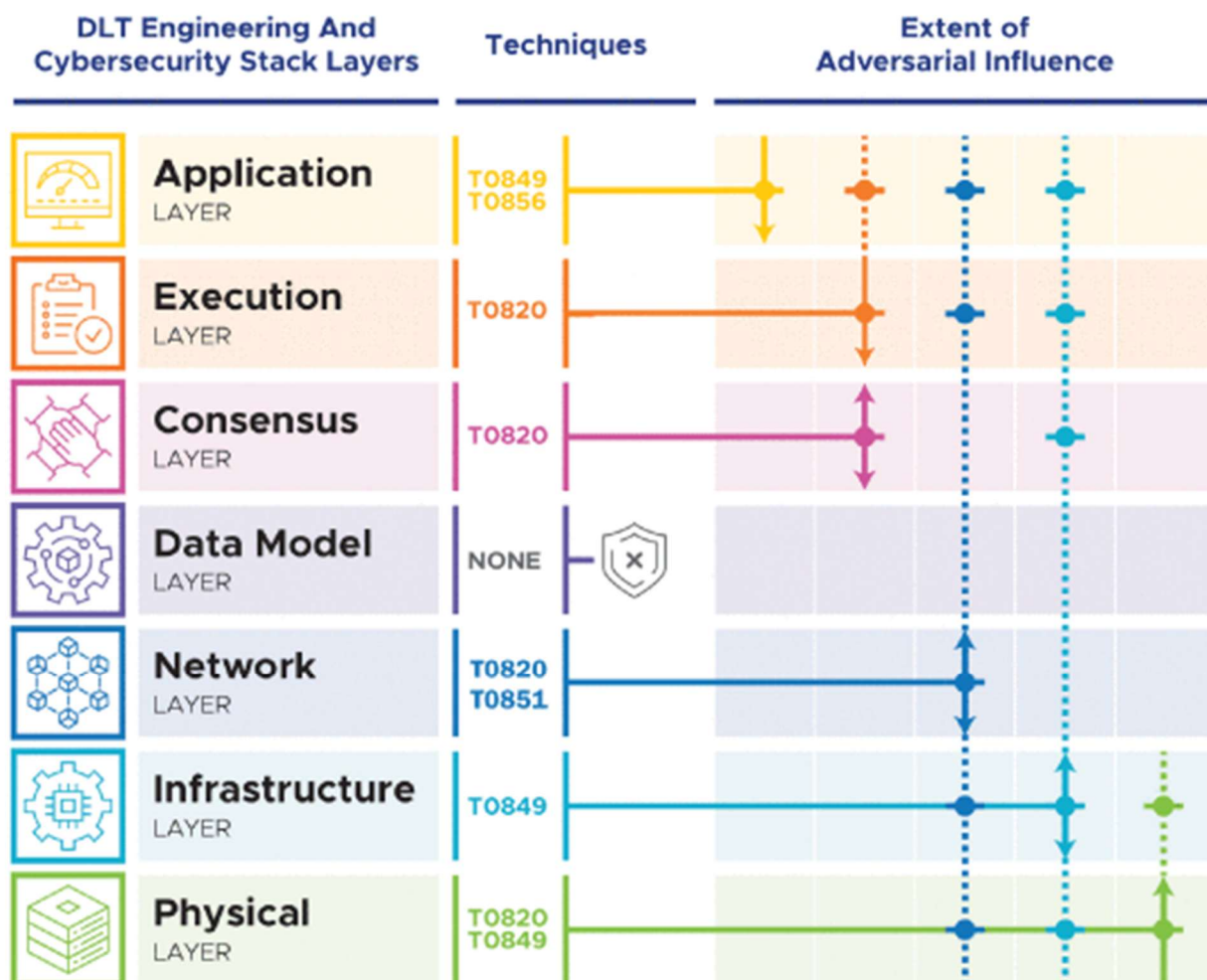


FIGURE 8. Evasion tactic analysis and mapping to the DLT stack.

1) Application Layer. An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The applicable techniques are:

- *Masquerading (T0849):* An adversary may use masquerading to disguise a malicious application or executable as another file, which can be achieved by duplicating an API front end, leading to deceit or impersonation of a given state;
- *Spoof Reporting Message (T0856):* An adversary could spoof the interactions between the Application layer components and the database that the application may be interacting with. Furthermore, the same technique can be used to fake a DLT system's health system/messages.

2) Execution Layer. The only applicable technique is *Exploitation for Evasion (T0820)*, which may be used to exploit potential program bugs/errors in the smart contract code. Such exploitation may require the adversary to gain access to the infrastructure layer to access the smart contract program.

3) Consensus Layer.

- The only applicable technique is *Exploitation for Evasion (T0820)*, which may be applied to this layer if the adversary could compromise inherent vulnerabilities of the consensus mechanism itself;
- Evasion occurs when the consensus-forming nodes are compromised and form a consensus that is against the rules defined in the smart contract. If this happens, the applications and smart contracts will not have any context around what happened in the consensus layer. The adversary could execute an attack by targeting the consensus nodes.

4) Data Model Layer. The Data Model Layer cannot be compromised using the Evasion tactic. The nature of DLT systems explains this. Every adversary's attempt to use evasion techniques on this layer should be identified. However, these layers may be compromised through the infrastructure layer as indirect access.

5) Network Layer. The following technique can be applied:

- *Exploitation for Evasion (T0820)*: By taking advantage of programming errors and erroneously deployed configurations, an attacker may perform exploitations that affect or compromise a DLT's ability to operate;
- *Rootkit (T0851)*: Attackers may inject malicious code to disrupt a DLT's ability to operate, potentially leading to service outages and/or the application of other tactics.

Note that impacts can be observed at higher levels once the network infrastructure is compromised due to disrupted network traffic patterns. For example, smart contracts deployed within the execution layer could experience communication timeouts, while API end-points may experience demand surges due to replay attacks or DoS attacks. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

6) Infrastructure Layer. An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

An adversary may use masquerading to disguise a malicious application or executable as another file. The applicable technique is *Masquerading (T0849)*. A supervisory agent may be deceived by impersonating services, and/or devices, leading to an incorrect system evaluation.

7) Physical Layer. An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The following techniques may be applied:

- *Masquerading (T0849)*: an adversary may disguise a malicious application or executable as another file;
- *Exploitation for Evasion (T0820)*: this is applicable if there are programming errors and erroneously deployed configurations. An attacker could perform exploitations that affect or compromise DLT's operating ability.

Change Operating Mode (T0858) and *Spoof Reporting Message (T0856)* are valid techniques that could be implemented at the controller level. However, they do not have a direct relationship to the DLT or

its use in the OT/ICS network. The adversarial compromise and use of these techniques are not specific to the DLT and can occur independently of the deployment of a DLT.

8) Mitigation. Table 5 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Evasion tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 5 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Evasion Tactic

Technique	Mitigation
Exploitation for Evasion (T0820)	M0948 Application Isolation and Sandboxing
Masquerading (T0849)	M0945 Code Signing M0938 Execution Prevention
Spoof Reporting Message (T0856)	M0802 Communication Authenticity
Rootkit (T0851)	M0945 Code Signing

F. Discovery. The Discovery tactic includes techniques that adversaries may use to survey (recognize) the ICS environment and gain knowledge about the internal network, control system devices, and how the processes interact. The specific techniques help adversaries observe the environment and determine the next steps for subsequent target selection and lateral movement. The techniques allow adversaries to explore what they can control and gain insight into interactions between various control system processes [11]. Figure 9 illustrates the analysis.

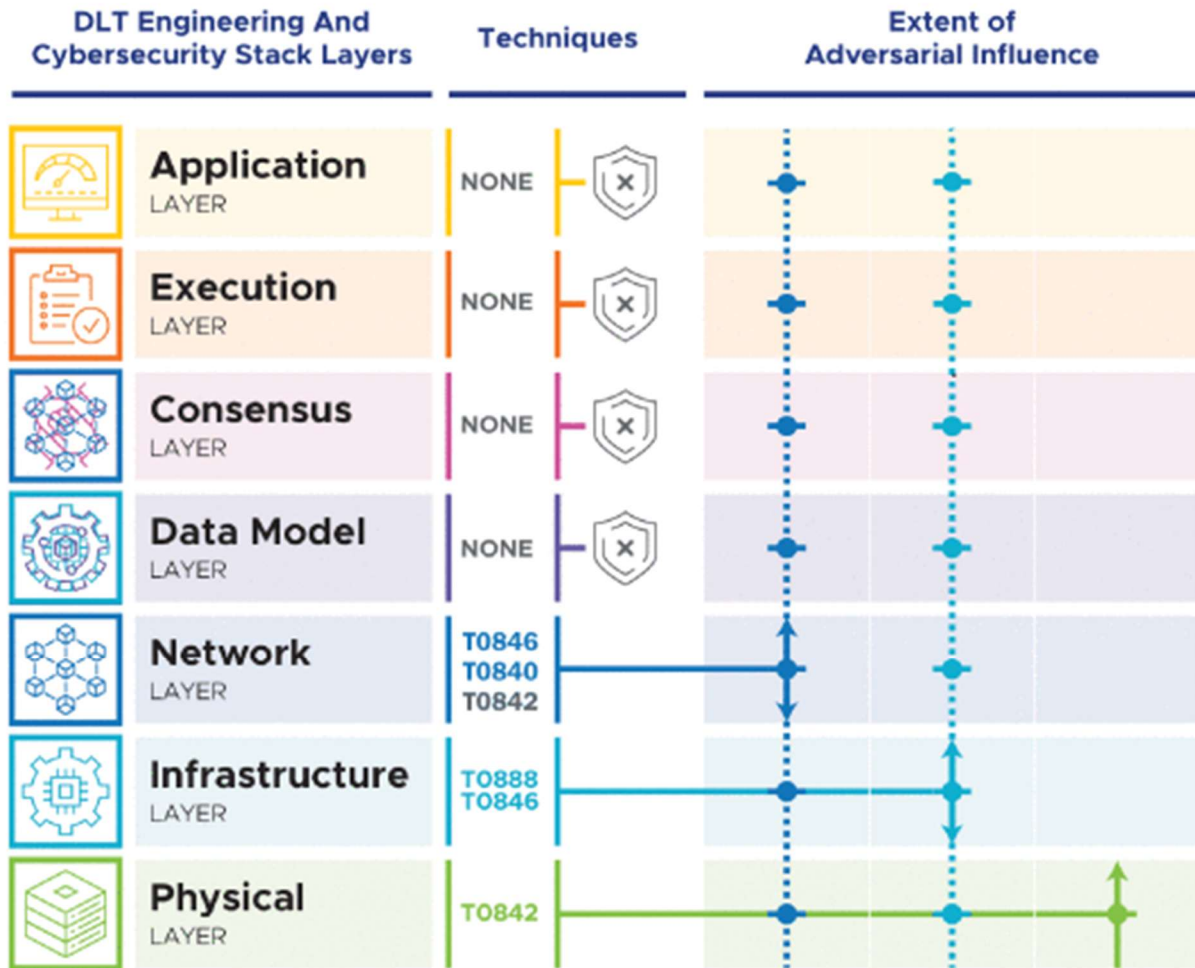


FIGURE 9. Discovery tactic analysis and mapping to the DLT stack.

1) **Application Layer.** The tactic is not applicable at the application layer.

2) **Execution Layer.** Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The tactic is not applicable at this layer.

3) **Consensus Layer.** Consensus protocols include lottery-based, voting-based, etc. The tactic is not applicable at this layer.

4) **Data Model Layer.** Data (and time) synchronization. Ordering services, block creation, chain structure, hashing, etc. The tactic is not applicable at this layer.

5) **Network Layer.** Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The following techniques apply in the network layer:

- *Network Connection Enumeration (T0840)*: This technique may be used to acquire connection profiles used by clients to connect to the DLT and the neighbor peers' connection addresses to perform targeted network disruptions;
- *Remote System Discovery (T0846)*: This technique may be used to acquire the neighbor peers' connection addresses that may be used with other techniques;
- *Network Sniffing (T0842)*: From a local node, local network visibility can be attained to launch eclipse-like attacks. Network sniffing may be possible depending on the network topology, particularly when traffic crosses public/non-trusted networks or unprotected devices. Some of the network sniffings would be outside the scope of the DLT.

Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

6) Infrastructure Layer. Data storage entities. Logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The following techniques are applicable.

- *Remote System Information Discovery (T0888)*: This technique may be used to acquire connection profiles used to connect to the DLT and to discover the oracle or off-chain database used by the smart contract to make decisions;
- *Remote System Discovery (T0846)*: This technique may acquire the neighbor peers' connection addresses to perform targeted network disruptions.

Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

7) Physical Layer. Systems participating on behalf of the users. Examples: sensors, IoT devices with UID, OS, etc. The tactic is not directly applicable to this DLT stack layer. However, depending on the OS and the available system libraries, then network sniffing may be possible at a local level (via *T0842*). If running a DLT node, they may learn more about the application.

8) Mitigation. Table 6 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Discovery tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 6 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Discovery Tactic

Technique	Mitigation
Network Connection Enumeration (T0840)	M0816 Mitigation Limited of Not Effective
Remote System Discovery (T0846)	M0814 Static Network Configuration
Remote System Information Discovery (T0888)	M0814 Static Network Configuration
Network Sniffing (T0842)	M0808 Encrypt Network Traffic M0930 Network Segmentation M0926 Privileged Account Management M0814 Static Network Configuration

G. Lateral Movement. In the Lateral Movement tactic, the adversary tries to move through the ICS environment. In this assessment, the focus is on movement through the various DLT stack layers. Lateral movement can only be successful after the Initial Access and Execution tactics are effective against a specific DLT stack layer, as described above. If the Privilege Escalation tactic is successful, the effect of the Lateral Movement tactic may be more significant. Figure 10 illustrates the analysis.

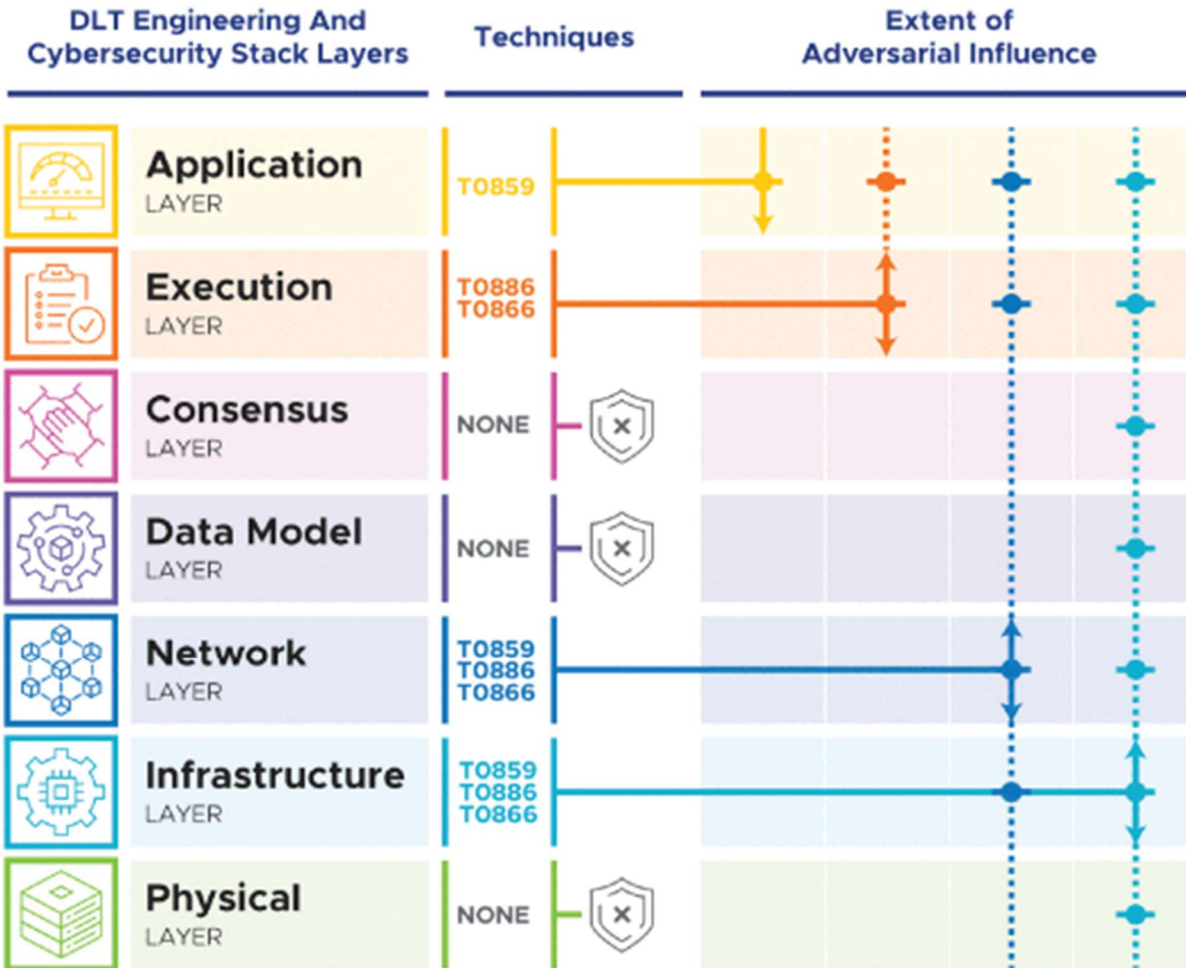


FIGURE 10. Lateral Movement tactic analysis and mapping to the DLT stack.

1) Application Layer. An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The only applicable technique is the *Valid Accounts (T0859)*. In this case, compromised credentials may be used to bypass access controls placed on the API layer and may be used to maintain persistent access to remote systems. Compromised and default credentials may also grant an adversary increased privilege to application-specific functionalities.

2) Execution Layer. Rule-bases and program code. The Lateral Movement tactic is applicable, including the *Remote Services (T0886)* and *Exploitation of Remote Services (T0866)* techniques that are applicable at the execution layer, which could result in application-level compromises if a vulnerable piece

of code within a smart contract or a library *node.js* library was used to write a smart contract. These techniques may allow an adversary to move between DLT components.

3) Consensus Layer and Data Model Layer. Lateral Movement techniques will not be effective at these layers because any attacker's attempt can be easily traced. This is achieved through the distributed nature of the network and the use of cryptography.

4) Network Layer. The Lateral Movement tactic is applicable, including the following techniques. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Valid Accounts (T0859):* An adversary may steal the credentials of a specific user or account. Compromised credentials may be used to bypass access controls;
- *Remote Services (T0886):* This technique may be used to move between DLT assets and network components;
- *Exploitation of Remote Services (T0866):* This may allow an adversary lateral movement between the DLT components.

5) Infrastructure Layer. If attacker software was installed on the validator node (and this node did not establish the fourth level of authentication, i.e., storing private keys on a secure hardware token), an attacker could gain access to the validator node's private key by reading files/while processing the private key in the device memory. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Valid Accounts (T0859):* An adversary may steal the credentials of a specific user or account. Compromised credentials may be used to bypass access controls;
- *Remote Services (T0886):* This technique may be used to move between DLT assets;
- *Exploitation of Remote Services (T0866):* This may allow an adversary lateral movement between the DLT components.

6) Physical Layer. Purdue Level 0 systems (controllers, PLC, safety systems) should not be directly connected or part of a DLT; therefore, techniques such as *Program Download (T0843)* do not apply. Lateral Movement techniques should not be effective at these layers because any attacker's attempt can be easily traced. This is achieved through the distributed nature of the network and the use of cryptography.

7) Mitigation. Table 7 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Lateral Movement tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 7 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Lateral Movement Tactic

Technique	Mitigation
Remote Services (T0886)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication M0918 User Account Management
Exploitation of Remote Services (T0866)	M0942 Disable or Remove Feature or Program M0926 Privileged Account Management
Valid Accounts (T0859)	M0801 Access Management M0926 Privileged Account Management M0918 User Account Management

H. Collection. The Collection tactic consists of techniques that allow attackers to collect information that would later be used for illegal purposes. The adversary tries to collect data of interest and information about the domain, thus obtaining feedback on the ICS environment. Figure 11 illustrates the analysis.

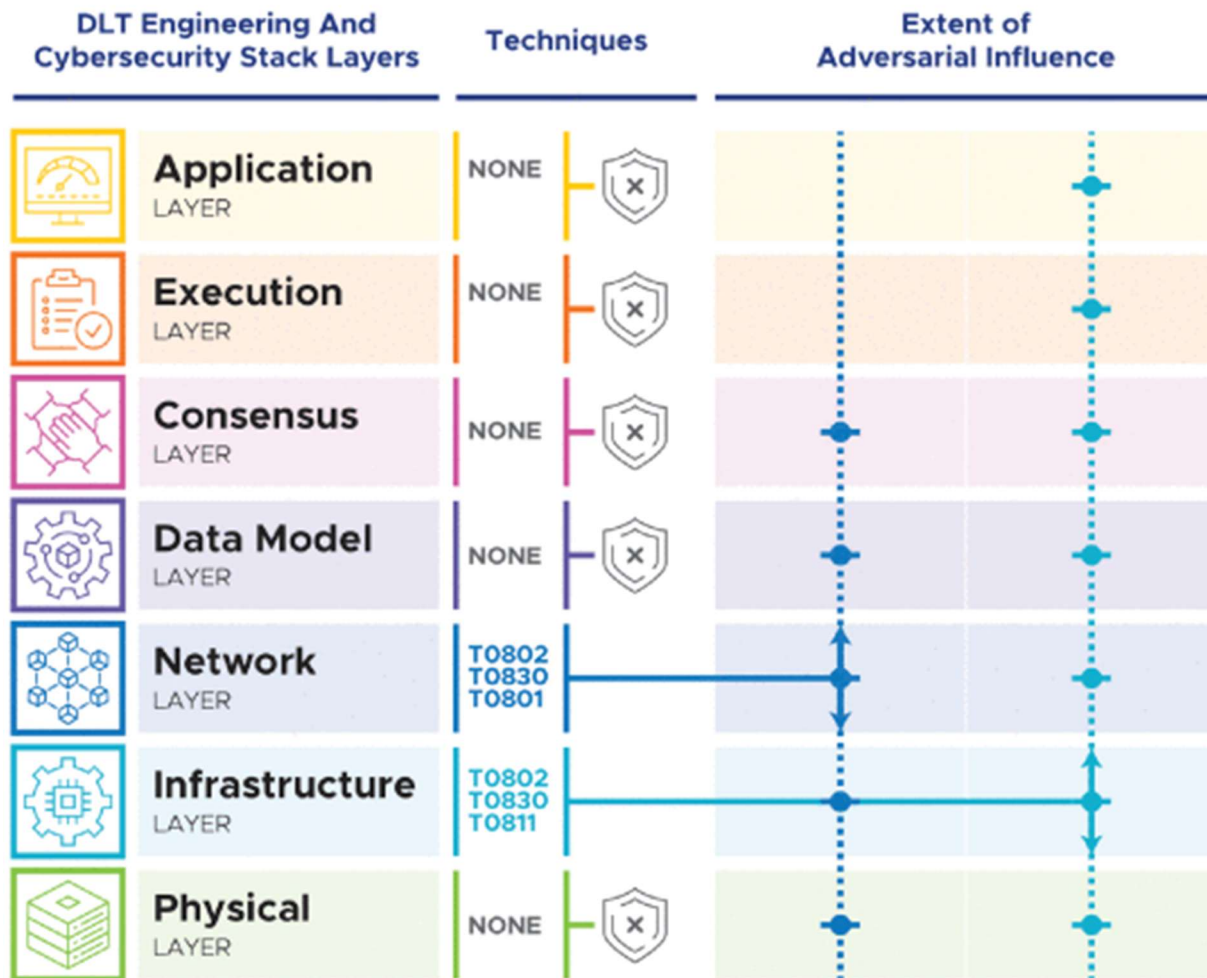


FIGURE 11. Collection tactic analysis and mapping to the DLT stack.

1) Application Layer.

- Applications that trigger rule-bases and program code, such as APIs, UI/GUIs, Oracles, distributed applications, marketplace and monetization, exchange, etc. The Collection tactic does not apply at this DLT stack layer;
- An adversary could compromise APIs to collect information on the DLT and about the DLT network of peers. Currently, the techniques included in the Collection tactic do not apply. There may be other techniques not included in the MITRE ATT&CK® ICS Matrix that could be used.

2) Execution Layer. This layer deals with Rule-bases and program code. Examples include Smart Contracts, Chain code, atomic swaps, Tokens, etc. The Collection tactic does not apply at this DLT stack layer. An adversary could gain access to the smart contract and reverse engineer the functions, etc., to enumerate inner workings. Currently, the techniques included in the Collection tactic do not apply. Other techniques not included in the MITRE ATT&CK® ICS Matrix could be used.

3) Consensus Layer. Consensus protocols include proof-based, voting-based, etc. The Collection tactic does not apply at this DLT stack layer.

4) Data Model Layer. Data (and time) synchronization, ordering services, block creation, chain structure, hashing, etc. The Collection tactic applies and the following is the *Automated Collection (T0802)* technique that may be used. The adversary could use data collection strategies in network traffic or script execution automatically in the DLT environment. This may allow an adversary to more accurately determine the structure of the data model layer.

5) Network Layer. Peer-to-peer transaction broadcast/discovery. Connectivity, runtime, telecommunications, and network parameters. The Collection tactic applies, and the following techniques may be used. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Automated Collection (T0802)*: The adversary could use data collection strategies in network traffic or script execution automatically in the control environment. This may allow an adversary to identify the DLT network architecture;
- *Adversary-in-the-Middle (T0830)*: The adversary with certain privileges could alter the network traffic in search of benefiting their illegal activities. Specifically, an adversary could intercept traffic associated with the various DLT nodes and potentially block or modify the traffic;
- *Monitor Process State (T0801)*: The adversary may use the information of the processes to cancel or allow actions that help him in his fraud purposes. Once this layer is compromised, this technique will allow an adversary to map the DLT more accurately.

Compromising the network infrastructure will allow an adversary to collect data-in-transit, such as the communication between consensus-forming nodes and the block-creation nodes. Similarly, the adversary would be able to intercept the traffic between nodes. The knowledge obtained from this collection process could then be used to launch targeted attacks.

6) Infrastructure Layer. Data storage entities. Logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Collection tactic applies, and the following techniques may be used. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Automated Collection (T0802):* The adversary could use data collection strategies in network traffic or script execution automatically in the control environment. This may allow an adversary to identify the DLT network architecture;
- *Adversary-in-the-Middle (T0830):* The adversary with certain privileges could alter the network traffic in search of benefiting their illegal activities. Specifically, an adversary could intercept traffic associated with the various DLT nodes and potentially block or modify the traffic;
- *Data from Information Repositories (T0811):* The adversary seeks to obtain available information about the system parts, whether in parts of used codes, version history, or any information about system or environment components. An attacker may gather information about the DLT off-chain and on-chain storage devices, including the content and structure.

7) Physical Layer. System participating on behalf of the users, for example, Sensors, IoT devices with UID, OS, etc. Traditional DLT systems do not have a physical layer in themselves.

8) Mitigation. Table 8 provides a list of suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Collection tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 8 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Collection Tactic

Technique	Mitigation
Automated Collection (T0802)	M0807 Network Allowlists
Adversary-in-the-Middle (T0830)	M0802 Communication Authenticity M0813 Software Process and Device Authentication
Monitor Process State (T0801)	M0816 Mitigation Limited or Not Effective
Data from Information Repositories (T0811)	M0926 Privileged Account Management M0922 Restrict File and Directory Permissions M0918 User Account Management

I. Command and Control. Command and Control techniques are typically used to communicate between affected parts of the system. In the case of DLT systems, this might have a place in Network and Infrastructure layers. Figure 12 illustrates the analysis.

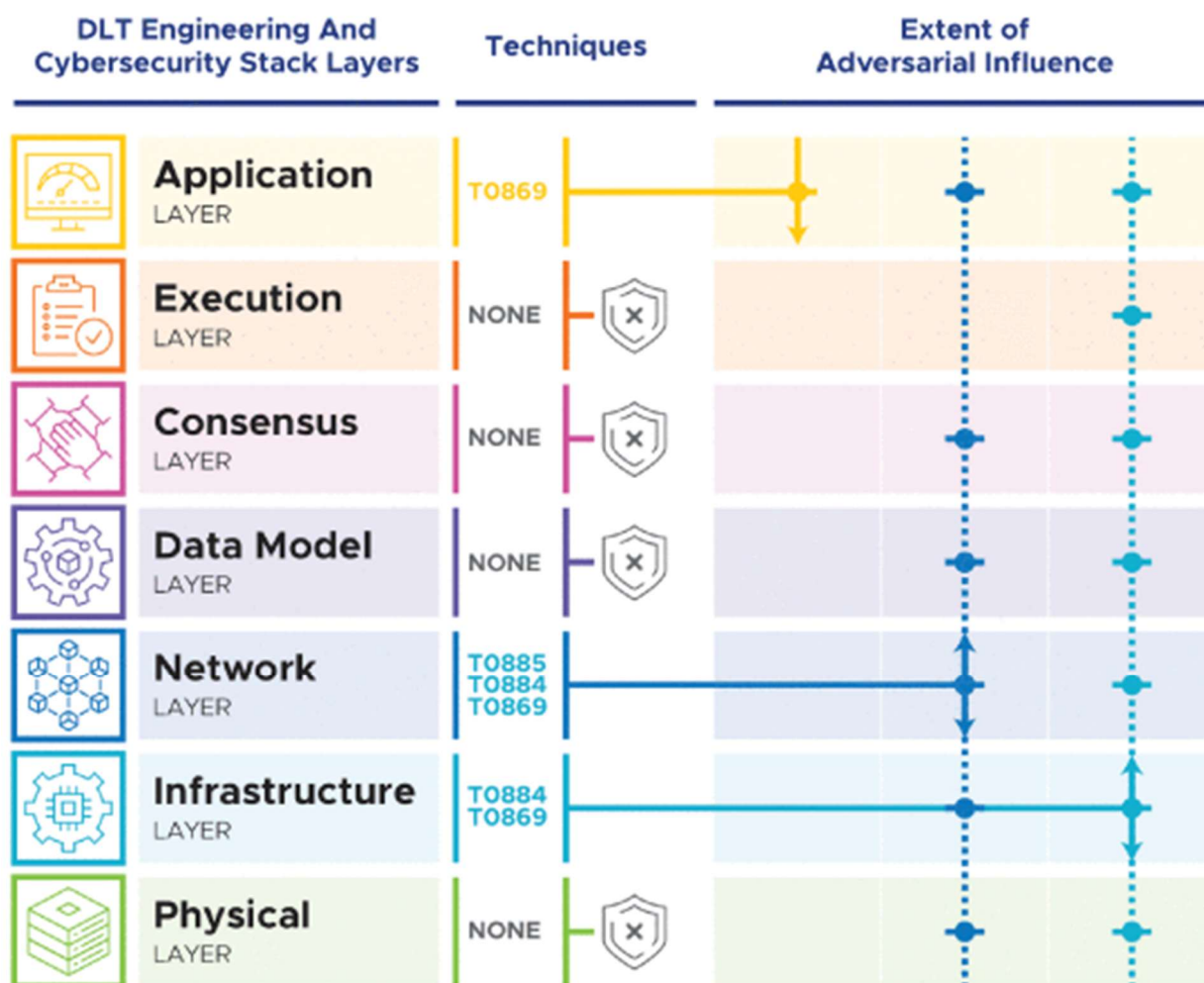


FIGURE 12. Command and control tactic analysis and mapping to the DLT stack.

1) Application Layer. The Command-and-Control tactic does not directly apply at this layer.

However, The *Standard Application Layer Protocol (T0869)* technique may indirectly apply if the Infrastructure and/or Network layers are compromised.

- The application layer components, such as the APIs, software, etc., may be programmed to initiate network connections over protocols, such as Hypertext Transfer Protocol Secure (HTTPS), etc., to reach outside the network. The equipment in the network and infrastructure layers facilitate such connections. Therefore, if an adversary aims to compromise communications over protocols such as HTTPS, etc., the adversary would have to go through the network and infrastructure layers. This can then affect the

application itself because the application may accept an adversarial connection as a legitimate connection;

- This technique may be indirectly applicable by disguising non-DLT traffic as DLT legitimate traffic. This would enable an attacker to control/disrupt services without supervisory agents. For example, protocols such as gRPC may be used to encode various API calls ranging from transaction submission to peer-level management. This would allow supervisory agents to differentiate using simple network protection rules.

Special attention should be devoted to systems that rely on common or shared endpoints to manage or operate the system. For example, suppose a given port is used to configure, monitor, and propose to commit transactions. In that case, then deep-packet inspection tools and firewalls may need to be deployed to ensure these endpoints do not process low-level API calls from untrusted systems.

2) Execution, Consensus, and Data Model Layer. The adversary cannot initiate a Command-and-Control tactic attack in these layers. Therefore, the Command-and-Control tactic does not apply.

3) Network Layer. Peer-to-peer transaction broadcast or discovery, including connectivity, runtime, telecommunications, and network parameters. Using a secure local network environment, an adversary can try to reach other points or devices in the network. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers. The Command-and-Control tactic applies, and the following techniques may be used.

- *Commonly Used Port (T0885):* an adversary may communicate over a commonly used port to blend in with normal network activity associated with the DLT to execute attacks such as brute force, DDoS, port denial, etc;
- *Connection Proxy (T0884):* an adversary may use a connection proxy to direct the DLT traffic among the various devices and alter the DLT communications;
- *Standard Application Layer Protocol (T0869):* this technique directly applies. An adversary could gain logical or physical port-level access to the network infrastructure through a compromised DLT node and/or DLT application.

4) Infrastructure Layer. Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. DLT nodes running on local or virtual machines or clusters might be subject to

standard attacks. The attacks at the infrastructure layer can enable indirect access to other layers in the DLT stack. The Command-and-Control tactic applies, and the following techniques may be used.

- *Connection Proxy (T0884)*: the adversaries may use a connection proxy to redirect the DLT traffic among the various devices and alter the DLT communications. The impact may be more significant because an attack at this DLT stack layer will allow indirect access to the other DLT stack layers. For example, by modifying a Virtual Machine (VM) or Docker image launch parameters, an attacker can set up additional ports that may later be used to take control of the infrastructure layer;
- *Standard Application Layer Protocol (T0869)*: an adversary could gain logical or physical port level access to the network infrastructure through a compromised DLT node and/or DLT application.

5) Physical Layer. The adversary cannot initiate a Command-and-Control tactic attack in this layer. Therefore, the Command-and-Control tactic does not apply because of the limited effects of compromising a single unit or field device.

6) Mitigation. Table 9 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Command-and-Control tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 9 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Command-and-Control Tactic

Technique	Mitigation
Standard App Layer Protocol (T0869)	M0807 Network Allowlists
Commonly Used Port (T0885)	M0942 Disable or Remove Feature or Program M0804 Human User Authentication
Connection Proxy (T0884)	M0937 Filter Network Traffic M0807 Network Allowlists

J. Inhibit Response Function. Adversaries use this tactic to circumvent a system's health monitoring or inherent self-protection mechanisms. This may result in a system losing its safety guarantees or preventing human (or automated) monitors from performing service and quality evaluations. Depending on the systems involvement,

these attacks could result in human loss of life, equipment destruction, and production interruption. Figure 13 illustrates the analysis.

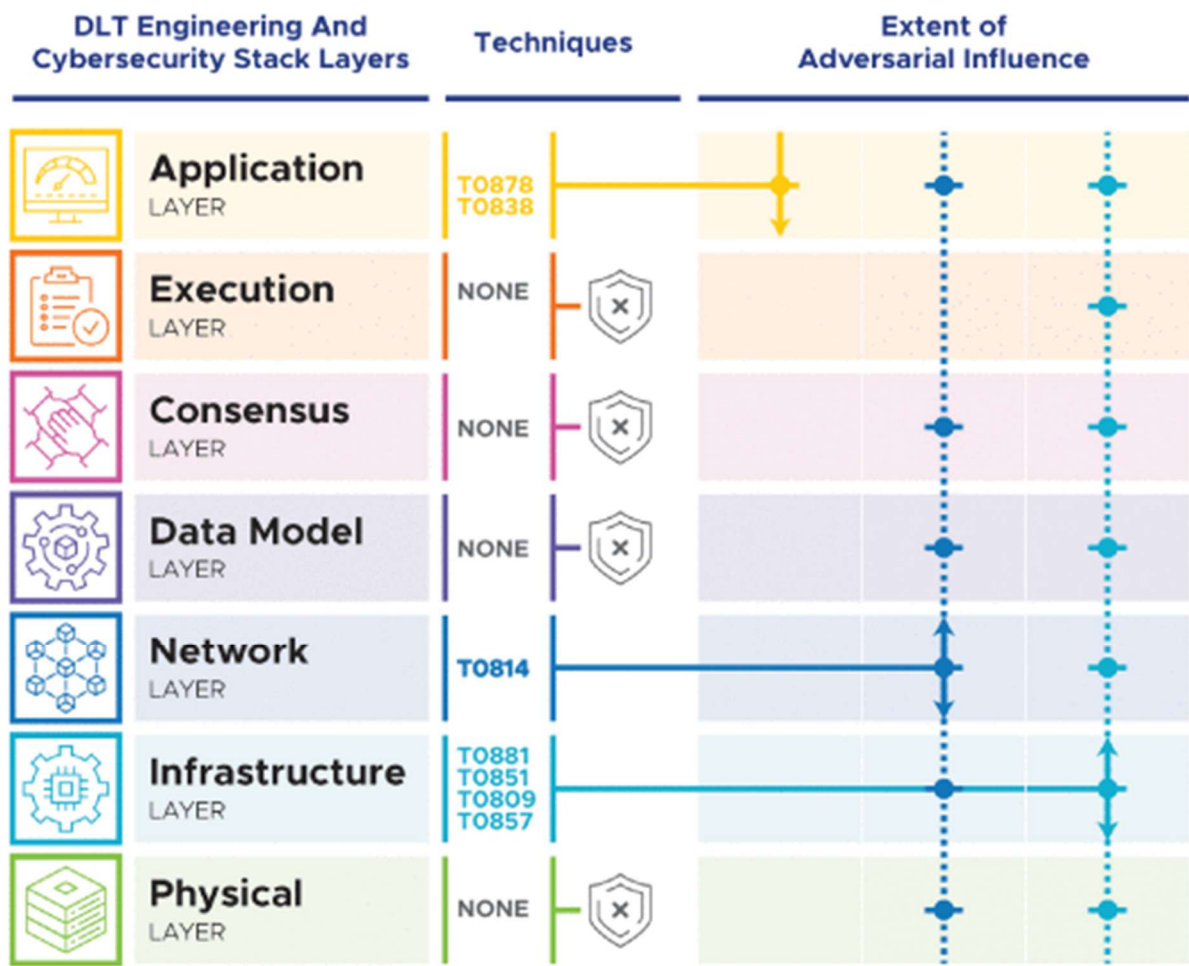


FIGURE 13. Inhibit response function tactic analysis and mapping to the DLT stack.

1) **Application Layer.** The Inhibit Response tactic applies at this DLT stack layer, and the following techniques may be used.

- *Alarm Suppression (T0878):* tools that are used to monitor the DLT health, such as Hyperledger Caliper and Eth-netstats, rely on DLT-provided logs and endpoints to enable end users to interpret DLT operations. These logs do not always inherit the same protection mechanisms in a DLT; therefore, mechanisms that ensure valid data are presented to the supervisor agent must be devised;

- *Modify Alarm Settings (T0838)*: similarly to the alarm suppression attack, mechanisms to ensure that correct logs are being used and tools to interpret them can be trusted must be in place to prevent the end user from receiving erroneous alarms.

2) Execution Layer. Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The tactic is not applicable at this DLT stack layer.

3) Consensus Layer. Consensus protocols: proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. This layer should be considered a “transient” layer where attacks will go through. For example, a DoS or data destruction attack will require some knowledge of this layer to be successful, but it is not the primary target or entry point.

4) Data Model Layer. Data (and time) synchronization. Ordering services, block creation, chain structure, hashing, etc. The tactic is not applicable at this DLT stack layer.

5) Network Layer. Peer-to-peer transaction broadcast/discovery. Connectivity, runtime, telecommunications, and network parameters. The attacks at the network layer can enable indirect access to other layers in the DLT stack. The only applicable technique is the *Denial of Service (T0814)*. An adversary may perform a DoS attack to disrupt the operation of the DLT (e.g., at the network switch to affect DLT nodes).

6) Infrastructure Layer. Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Inhibit Response tactic applies at this DLT stack layer, and the following techniques may be used. The attacks that happen at the infrastructure layer can enable indirect access to other layers in the DLT stack.

- *Service Stop (T0881)*: If attackers gain access to the underlying infrastructure hosting the DLT nodes, there exists a risk of having a service/stop the attack. Host-level diversity and distinct host networks may limit the ability of an attacker to succeed (e.g., use a mixture of cloud vendors, operating systems, etc.);
- *Data Destruction (T0809)*: An adversary may perform data destruction to disrupt the DLT from reporting potential adversarial actions, such as data modification. This technique may then be implemented through the infrastructure layer at the other DLT stack layers;

- *Rootkit (T0851)*: Attackers may inject malicious code to disrupt a DLT's ability to prevent expected alarms and response mechanisms;
- *System Firmware (T0857)*: An attacker may install malicious or out-of-date firmware that could be used to disrupt the DLT.

7) Physical Layer. System participating on behalf of the users. Examples: sensors, IoT devices with UID, OS, etc. The tactic is not applicable at this DLT stack layer because traditional DLT systems do not have a physical layer.

8) Mitigation. Table 10 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Inhibit Response Function tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 10 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Inhibit Response Function Tactic

Technique	Mitigation
Alarm Suppression (T0878)	M0807 Network Allowlists M0814 Static Network Configuration
Modify Alarm Settings (T0838)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0807 Network Allowlists M0813 Software Process and Device Authentication M0918 User Account Management
Denial of Service (T0814)	M0815 Watchdog Timers
Service Stop (T0881)	M0922 Restrict File and Directory Permissions M0918 User Account Management
Data Destruction (T0809)	M0926 Privileged Account Management M0922 Restrict File and Directory Permissions
Rootkit (T0851)	M0945 Code Signing
System Firmware (T0857)	M0801 Access Management M0945 Code Signing M0937 Filter Network Traffic M0804 Human User Authentication M0813 Software Process and Device Authentication

K. Impair Process Control. Field devices rely on API calls to reach the DLT infrastructure (e.g., via a web service). The execution layer (e.g., the smart contract) may rely on the application layer to access external oracles or off-chain storage databases. Figure 14 illustrates the analysis.

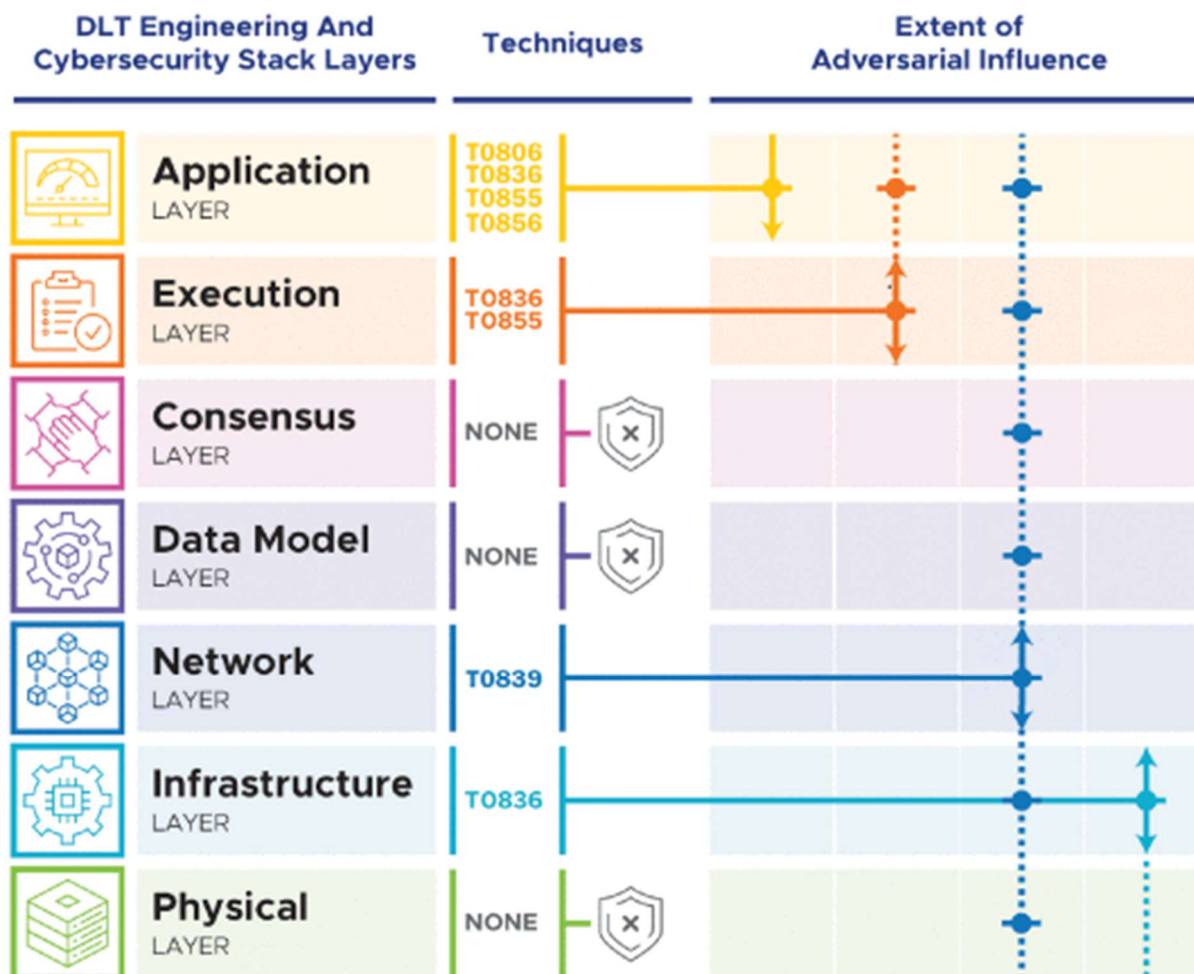


FIGURE 14. Impair process control tactic analysis and mapping to the DLT stack.

1) **Application Layer.** The Impair Process Control tactic applies at this DLT stack layer, and the following are the techniques that may be used:

- *Brute Force I/O (T0806):* a brute force attack may base its functionality on issuing malformed or random commands. Such attacks will likely affect API entry points, reducing their ability to serve legitimate traffic. To reduce the risk, APIs should be protected by brute force detection tools and gracefully handling malformed requests;
- *Modify Parameter (T0836):* similar to the Brute Force I/O attack, service APIs may be vulnerable to maliciously crafted API calls (e.g., via a web service). Therefore,

mechanisms to authenticate and secure a request's validity should be deployed (e.g., to avoid replay or modification);

- *Unauthorized Command Message (T0855)*: this technique should be considered in the design of the system. API entry points must also check the identity or role of an agent submitting an API request to reduce the risk of bypassing internal control mechanisms, for example, in a system with 1) a measurement device, 2) a control algorithm, and 3) an actuator. The measurement device should not be allowed to issue control commands to the actuator;
- *Spoof Reporting Message (T0856)*: a DLT can secure the integrity of a message once the transaction hits the consensus layer. This implies that DLT's API entry points must be capable of serving as gatekeepers and detecting message spoofing (e.g., via nonces and signatures). Failure to do so may result in incorrect messages reaching the consensus layer, thus leading to incorrect states at the receiver end. APIs used to access external oracles or storage services must be equally secure, specifically when the overall solution assumes a high-level of trust towards these external systems.

2) Execution Layer. Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The Impair Process Control tactic applies at this DLT stack layer, and the following are the techniques that may be used:

- *Unauthorized Command Message (T0855)*: similar to the API entry points, smart contracts must verify the identity/role and effective permissions over the target request to reduce the risk of bypassing internal control mechanisms. For example, in a system with 1) a measurement device; 2) a control algorithm; and 3) an actuator, the measurement device should not be allowed to issue direct control commands to the actuator;
- *Modify Parameter (T0836)*: similar to service APIs, smart contracts may be vulnerable to maliciously crafted invocations. Therefore, mechanisms to handle incorrect parameters or data formats should be developed.

3) Consensus Layer. Consensus protocols include proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. This should be considered a “transient” layer

used by an attacker. For example, a DoS or data destruction attack will require some knowledge of this layer to be successful, but it is not the primary entry point. Therefore, this tactic does not apply at this DLT stack layer.

4) Data Model Layer. Data and time synchronization include ordering services, block creation, chain structure, and hashing. The tactic does not apply at this DLT stack layer.

In specific application use cases, incorrectly-handled application and execution layer behaviors could lead to inconsistent or incorrect data states. Such inconsistencies could lead to incorrect decisions (e.g., clearing a market that appears to have not received bids). This issue can be compounded by the ledger's immutability, making it difficult for DLT applications to correct the error.

5) Network Layer. Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The Impair Process Control tactic applies at this DLT stack layer, and the following technique may be used. Once the network layer is compromised, an adversary may access the other six DLT stack layers indirectly. The only applicable technique is the *Module Firmware (T0839)*. Adversaries may install malicious or vulnerable firmware onto network hardware devices.

6) Infrastructure Layer. Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Impair Process Control tactic applies at this DLT stack layer, and the following technique may be used. The only applicable technique is the *Modify Parameter (T0836)*. An adversary may maliciously modify DLT values to ensure integrity in the OT/ICS environment. If these DLT values are modified, the resulting OT/ICS parameters may be invalid.

7) Physical Layer

Systems participating on behalf of the users, for example, sensors, IoT devices with UID, OS, etc. This tactic does not apply at this DLT stack layer.

Unauthorized or invalid command messages may reach the physical layer. It is difficult for end/edge devices to differentiate between genuine/fake commands if they appear to originate from the DLT because the end/edge devices do not often have the built-in capability to contextualize commands and decide if a command is legitimate or adversarial. Therefore, protection mechanisms will be limited to ensuring the validity of the commands (e.g., is the command within the device capabilities? Is the action consistent with the local observation?). The protection mechanisms should be implemented in the other DLT stack layers.

One mitigation strategy is implementing device-level whitelisting to ensure participants are positively identified before access is granted.

8) Mitigation. Table 11 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Impair Process Control tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 11 Suggested MITRE ATT&CK® ICS Mitigations for the Cited Techniques in the Impair Process Control Tactic

Technique	Mitigation
Brute Force I/O (T0806)	M0937 Filter Network Traffic M0807 Network Allowlists
Modify Parameter (T0836)	M0800 Authorization Enforcement
Module Firmware (T0839)	M0801 Access Management M0945 Code Signing M0804 Human User Authentication M0813 Software Process and Device Authentication
Spoof Reporting Message (T0856)	M0802 Communication Authenticity M0937 Filter Network Traffic
Unauthorized Command Message (T0855)	M0802 Communication Authenticity M0937 Filter Network Traffic M0813 Software Process and Device Authentication

L. Impact. For this analysis, an assumption is that the DLT is not an integral part of the OT/ICS environment. This means that the DLT cannot monitor and control physical devices. Any interaction with the OT/ICS environment is secondary to the impact on the DLT stack layers. OT/ICS devices should have autonomous, fail-safe operational schemes that can guarantee end-devices will continue to operate securely and functionally. Loss-of-control and loss-of-visibility due to DLT-related failures are valid concerns. Therefore, mechanisms to avoid, detect, and mitigate these scenarios should be engineered from the beginning. Figure 15 illustrates the analysis.

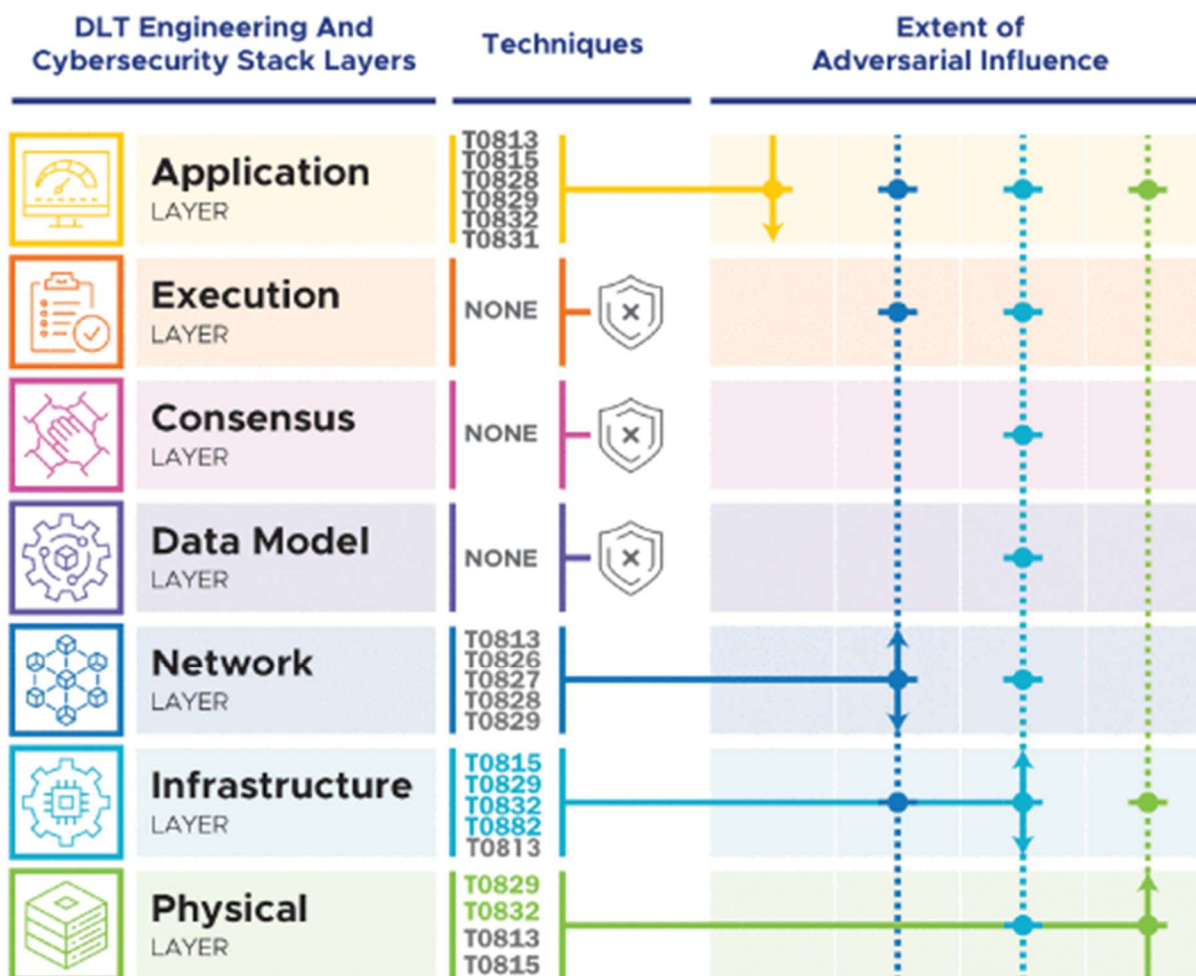


FIGURE 15. Impact tactic analysis and mapping to the DLT stack.

1) Application Layer. Applications that trigger rule-bases and program code, including APIs, UI/GUIs, Oracles, distributed applications, marketplace, monetization, etc. The Impact tactic does not apply at the application layer.

The following techniques may be applicable in application-specific use cases in the OT/ICS environment. However, these are outside the scope of the DLT.

- *Denial of Control (T0813):* Attackers could compromise applications and deny user control of DLT applications. This technique may be used in applications that enable control actions via a DLT system;

- *Denial of View (T0815)*: Attackers could disrupt operator oversight on the status of the ICS environment;
- *Loss of Productivity and Revenue (T0828)*: Attackers could disrupt or damage the availability and integrity of control system operations, devices, and related processes;
- *Loss of View (T0829) and Manipulation of View (T0832)*: Incorrectly protected telemetry APIs could be exploited, leading to incorrect health or operational state records. This is particularly applicable when a “translator or gateway” service is used to interact with the external world, and this interface can be abused. Mitigation strategies include providing operators with redundant, out-of-band communications to support monitoring and control of the operational processes;
- *Manipulation of Control (T0831)*: If devices are dependent on control signals transmitted through a DLT, there is a risk of causing Stuxnet/Industroyer type attacks if the device communication channel is not adequately protected by digital signatures or authentication.

2) Execution Layer. Rule-bases and program code, examples include smart contracts, chaincode, atomic swaps, tokens, etc. This layer would play a role, but other layers would reflect the impact. For other tactics and techniques, this should be considered a “transient” layer. For example, although manipulating the data input will lead to an incorrect decision, the impact will appear on other DLT stack layers. Therefore, the Impact tactic does not apply at this layer.

3) Consensus Layer. Consensus protocols include proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. For other tactics and techniques, this should be considered a “transient” layer. For example, although manipulating the data input will lead to an incorrect decision, the impact will appear on other DLT stack layers. Therefore, the Impact tactic does not apply at this layer.

4) Data Model Layer. Data and time synchronization include ordering services, block creation, chain structure, and hashing. This tactic does not apply at the data model layer.

5) Network Layer. Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The Impact tactic does not apply at this layer.

For the OT/ICS environment, the following techniques are applicable. However, these are outside the scope of the DLT.

- *Denial of Control (T0813)*: If devices are dependent on control signals transmitted through a DLT, there is a risk of causing communication disruptions that may lead to a loss of control if the remote device does not include an autonomous, fail-safe controller;
- *Loss of Availability (T0826)*: An adversary may disrupt components and services;
- *Loss of Control (T0827)*: This includes sustained loss of control and/or the inability to recover from a malicious event;
- *Loss of Productivity and Revenue (T0828)*: Attackers could disrupt or damage the availability and integrity of control system operations, devices, and related processes;
- *Loss of View (T0829)*: An adversary may cause a temporary or permanent loss of view of the DLT.

6) Infrastructure Layer. Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. This tactic applies to the infrastructure layer using the following techniques. Once the infrastructure layer is compromised, an adversary may have indirect access to the other six DLT stack layers.

- *Denial of View (T0815)*: An adversary may modify the DLT information to disrupt operations;
- *Loss of View (T0829)*: An adversary may cause a temporary or permanent loss of view of the DLT;
- *Manipulation of View (T0832)*: An adversary may manipulate DLT status and information sent to the DLT nodes.
- *Theft of Operational Information (T0882)*: Public, encrypted ledgers may be left exposed during an attack, leading to unwanted data exfiltration. Similarly, compromised peers could be used to extract all the data that is accessible to the peer.

For the OT/ICS environment, the *Denial of Control (T0813)* technique is applicable at the Infrastructure layer. Suppose devices depend on control signals transmitted through a DLT. In that case,

there is a risk of causing communication disruptions that lead to losing control if the remote device does not include an autonomous, fail-safe controller. However, this is outside the scope of the DLT.

7) Physical Layer. Systems participating on behalf of the users, examples include sensors, IoT devices with UID, OS, etc. This tactic applies to the physical layer using the following techniques.

- *Loss of View (T0829):* An adversary may cause a temporary or permanent loss of view of the DLT;
- *Manipulation of View (T0832):* An adversary may manipulate DLT status and information sent to the DLT nodes.

For the OT/ICS environment, the following techniques are applicable. However, these are outside the scope of the DLT.

- *Denial of Control (T0813):* If devices are dependent on control signals transmitted through a DLT, there is a risk of causing communication disruptions that could lead to a loss of control if the remote device does not include an autonomous, fail-safe controller;
- *Denial of View (T0815):* An adversary may modify the DLT information to disrupt operations.

8) Mitigation. Table 12 provides a list of the suggested MITRE ATT&CK[®] ICS mitigations for the cited techniques in the Impact tactic. Each mitigation strategy may need to be tailored for specific DLT architectures.

TABLE 12 Suggested MITRE ATT&CK[®] ICS Mitigations for the Cited Techniques in the Impact Tactic

Technique	Mitigation
Manipulation of Control (T0831)	M0802 Communication Authenticity
Manipulation of View (T0832)	M0802 Communication Authenticity
Theft of Operational Information (T0882)	M0922 Restrict File and Directory Permissions
Manipulation of View (T0832)	M0802 Communication Authenticity

SECTION IX. Conclusion and Potential Future Work

This work uses the MITRE ATT&CK[®] ICS matrix to explore the security aspects of using DLT for the energy sector. Building upon the previously published DLT technology stack, this paper provides an in-depth analysis of potential adversarial techniques and tactics potentially relevant to the layers of the DLT stack. Because DLT is not largely used in real-world energy sector use cases and deployments, particularly in the OT/ICS environment, the adversarial threat analysis was approached by allocating adversarial tactics and techniques to the DLT stack layers. Future work will propose a comprehensive approach to address security needs for the entire DLT for upstream and downstream applications. This may include:

1. Governing principles to cyber security risk management;
2. A cybersecurity framework for the agents in any downstream or upstream device connected to a DLT;
3. Principles for cyber risk assessment by technology type such as remotely controlled Distributed Energy Resources (DERs);
4. Reporting mechanisms for risk-based incidents and threats;
5. Governance and oversight of internal controls and risk mitigation techniques;
6. Disclosure policy concerning cyber security assessment and identified threats;
7. In-depth attack propagation analysis through threat modeling driven by specific scenarios.

Examples are as follows:

a. If an adversary uses a worm or malicious code to target the components of the execution layer, the adversary could laterally move and impact components in other DLT layers. If malware is designed to target components of a DLT layer, what level of propagation and impact can the malware have on components in other layers?

b. In a large DLT ecosystem, when the execution layer processes are distributed across multiple edges/nodes, the execution layer spans across different nodes executing in parallel. In such scenarios, qualitative estimates of the attack propagation and possible attack trees could be investigated;

c. T0889 (Modify Program), T0839 (Module Firmware), T0873 (Project File Infection), and T0857 (System Firmware) are all related to the upgrade, update, or maintenance of

operating systems or applications running on Intelligent Electronic Devices (IEDs), so they are typically outside the domain of a DLT. However, if a DLT-based package manager is used to track or validate changes or updates to IED software or firmware, incorrect information could cause failures or security bugs. Any analysis would be use case specific.

8. Perform an analysis using the MITRE Enterprise ATT&CK[®] Enterprise matrix. Although there is some commonality across both MITRE matrixes, there are some unique differences across tactics and techniques;
9. Evaluate if the proposed mapping process can be extended to the attacks such as exploitation for Privilege Escalation in proof-based consensus such as PoS consensus. Such analysis will lead to understanding the adversarial ability to manipulate privileges of components across the DLT stack layers of the nodes that have more weight in consensus mechanisms.
10. Evaluate and apply to distributed multi-agents systems: A longer-term noteworthy future opportunity of the presented work is in the space of distributed multi-agents systems research:
 - a. Use and integration of other global information sources and threat intelligence sources that could potentially inform the severity of the TTPs across DLT layers;
 - b. Ability to use the stack to explore the opportunities in the multi-agent systems to reduce resource wastage through event-triggered mechanisms. For instance, exploring the role of DLT in cooperative fault-tolerant output regulation of linear heterogeneous multi-agent systems via an adaptive dynamic event-triggered mechanism and robust adaptive event-triggered fault-tolerant consensus control of multi-agent systems with a positive minimum inter-event time. We believe that the presented stack can be useful in architecting the above solution space.

During the execution of the presented work, the authors faced three noteworthy challenges:

1. Maintain consistency across the mappings presented due to the lack of automated tools to assist with the effort;
2. Lack of extensive scientific literature (conference and journal publications) that is similar to the presented work; and
3. Inadvertent confusion between cybersecurity and cyber resiliency.

The team addressed those challenges by defining the rules of execution/mapping (discussed in Section VII) that were consistently followed across all mapping exercises. The team then used MITRE's documentation and NIST definitions/standards extensively in parallel and used limited but relevant conference/journal publications.

The present work involved a manual process of mapping and analysis. Therefore, the future work opportunity here is to automate the rules of execution/mapping discussed in this paper to accommodate any future changes to the ATT&CK[®] matrix and the DLT stack. In the OT environment, performing analysis while the system is operational may have a serious impact, such as bricking devices or impacting latency. Digital models could be used, but they would need to be developed and assessed in the IT environment. Also, if a digital model is to be useful, it must match the operational architecture.

Future studies will focus on the cyber risk assessment, cybersecurity, and cyber resiliency analysis of DLTs applied in the power energy sector for the remote monitoring and control of DERs, particularly in the case of scenarios involving the tokenization of energy flows between prosumers and the activation of smart contracts among energy aggregators/prosumers and market operators or power utilities.

Considering the interdependent structure of power systems as previously expounded, it is plausible that susceptibilities within the DLT stack could be exploited by means of malware injection, DoS attacks, or man-in-the-middle attacks. Furthermore, owing to the decentralized structure of DLTs, power systems may be vulnerable to double-spending, 51% and similar attacks, in which an actor acquires control over the majority of the network's computational resources to influence transactions. These attacks depend on the specific consensus protocol that is implemented.

The utilization of this framework, in conjunction with the inherent benefits of DLT such as transparency and immutability, has the potential to enhance the robustness and resilience of power systems in the face of cyber threats.

Businesses and industry should adhere to cybersecurity regulations and legislative standards, such as the Directive on Security of Network and Information System (NIS and NIS 2.0 Directives) within the European Union and the guidelines set forth by the National Institute of Standards and Technology (NIST) in the United States.

ACKNOWLEDGMENT

The content presented in this article documents the activities of the IEEE Standards Association (SA) P2418.5 Working Group and IEEE Energy and Blockchain Technical Community members. This includes the IEEE

SA, IEEE Power and Energy Society (PES) Smart Building, Loads and Customer System (SBLC) Technical Committee, and the IEEE Blockchain Initiative. We acknowledge these organizations for sponsoring and promoting some of these initiatives.

Appendix. Tailored Mitigation Strategies

See Tables 13–15

TABLE 13 Tailored Mitigation Strategies. Part 1 of 3

Mitigation	DLT-specific guidance
M0948 Application Isolation and Sandboxing	This applies to all programmable components, including smart contracts, software in the application layer, and components of ICS/OT systems participating in the DLT.
M0926 Privileged Account Management	The DLT-based use case environment will involve multiple account classes, and the privileged users may change depending on the DLT layer. For instance, applications running on non-utility systems may have the system owner as the privileged user for the application. A consortium of utilities and other administrative entities responsible for DLT operations, such as consensus, block formation, Verification and Validation (V&V) at the DLT level, etc., may involve privileged users across multiple administrative entities. At the infrastructure and physical layer level, the assets may have utility-based and non-utility-based owners. Across these varying user classes, privileged users and their controls should be defined. Specific best practices around this mitigation are beyond the scope of this paper - generally known best practices should be used.
M0801 Access Management	The general guidance discussed for M0926 applies here but to all users, not just the privileged ones. The DLT nodes within the network premises of an organization could take advantage of their AD to define access management to the nodes. Similarly, access controls between the systems and the nodes can be restricted through routing Access Control Lists (ACL) and firewall rules. If an OT/ICS asset directly participates in the DLT network without an intermediate node (infrastructure layer), the device may or may not synchronize with the AD. Instead, the device may have local authentication mechanisms. In such cases, the use case owner should decide if the asset can directly participate in the DLT or through an intermediate DLT node that runs on a virtual/physical machine that can be connected to AD. Evaluating device-level security hardening will be crucial in making the decision. Principles of least privilege and need-to-know should be used in user access management and system access management. Also, principles of least functionality should be implemented.
M0800 Authorization Enforcement	As suggested by MITRE, Role-based Access Controls (RBAC) should be used within the DLT and interactions of the systems/users with the DLT.
M0804 Human User Authentication	As discussed under M0926, various user groups and classes may exist. Some users may be responsible for the nodes forming consensus, and some may be the owners of smart contracts. Also, some users may be OT/ICS asset owners, application owners, or application users, etc. Shared/same credentials and default credentials should not be implemented in the ecosystem. Multi-factor authentication should be implemented across all aspects of user interaction with the DLT components and DLT connected/peripheral components, including the ICS/OT devices that participate in the DLT.
M0813 Software Process and Device Authentication	The authentication between the application and smart contract, DLT node and corresponding ICS/OT asset, etc. may rely on APIs. Secure design, verification, and monitoring of the APIs and their use is critical. Furthermore, DLT's peripheral network monitoring solutions can be used to monitor all traffic associated with the DLT within an organization and between organizations in an agreed-upon consortium that are ultimately responsible for fault-tolerant DLT operations.
M0918 User Account Management	The scope of this may be limited to the users within the organization. A layer of administrative users should enforce the user access management controls for those users. The permissions will vary based on the user access to the DLT nodes and participating ICS/OT or software assets. For the non-organizational users (owners of devices that run applications that interact with the smart contracts and owners of external ICS/OT systems such as DERs), an authorized/whitelisted user list should be maintained. Furthermore, monitoring solutions, such as Intrusion Detection System (IDS) and Intrusion Prevention System (IPS), should be in place for deep packet inspection of traffic in and out of the DLT network.
M0945 Code Signing	Binary and application integrity with digital signature verification should be enforced for all software components, including the application code, smart contracts, code that runs on the DLT node, and the code that runs on the ICS/OT systems that participate in the DLT through the respective DLT nodes.
M0817 Supply Chain Management	The supply chain management policies should be enforced on the DLT ecosystem (network, nodes, connected assets, etc.). Software Bill of Materials (SBOM) of all code should be performed to identify malicious libraries in the applications, smart contracts, etc. If hardware DLT nodes are used, Hardware Bill of Materials (HBOM) of the nodes and associated ICS/OT should be performed, and discovered vulnerabilities should be mitigated. To prioritize mitigations, the scope of vulnerabilities and threats should be evaluated in the context of DLT operations that support the use cases. Suppose vulnerabilities and threats are identified in the DLT (or its components). In that case, the risk should be evaluated in the context of the consequences of continuing/operating the DLT components and the potential impact on the ICS/OT systems. Principles of Safety, Reliability, and Productivity (SRP) should be used in the risk-mitigation decision-making process.
M0938 Execution Prevention	Following MITRE's guidance, execution of all unauthorized code should be prevented in all virtual and physical systems in the DLT ecosystem.

TABLE 14 Tailored Mitigation Strategies. Part 2 of 3

Mitigation	DLT-specific guidance
M0816 Mitigation Limited or Not Effective	Unused native features and services of all components/assets in the DLT should be disabled. For instance, all DLT nodes can be evaluated with the Security Technical Implementation Guides (STIG) and necessary configuration settings to harden the OS of the systems that host the DLT nodes, run smart contracts, form consensus, etc. Similarly, the ICS/OT participating in the DLT should be thoroughly evaluated to identify features and services that could be abused. In such cases, those vulnerabilities should be mitigated. If that is not possible, the use case owner should strongly consider the risks of connecting the ICS/OT devices directly or indirectly to the DLT.
M0942 Disable or Remove Feature or Program	The guidance discussed under M0816, M0945 is applicable.
M0921 Restrict Web-Based Content	The guidance discussed under M0816 applies here. In addition, the DLT nodes may converse with other DLT nodes outside the organization and over the internet. As much as possible, the external firewall should be used to granularly define the rules for the nodes' interaction. This includes specifying the ports, protocols over expected communications, etc. web-based browsing access should not be needed for DLT operations. Therefore, these services should be restricted at the host/network level and monitored for abuse or anomalous behavior.
M0802 Communication Authenticity	When possible, all interactions between the systems and their respective DLT nodes and all interactions between the DLT nodes (such as consensus forming or block creation nodes, if any interactions exist) should be performed on a trusted network. In architectures with untrusted network segments, MITRE's guidance to utilize secure protocols to authenticate the message and verify the integrity of the message should be followed. This can be achieved through digital signatures or Message Authentication Codes (MACs). Network and endpoint detection should be positioned to detect spoofed messages, unauthorized connections, and communication from and to the DLT nodes.
	M0814 Static Network Configuration The guidance provided by MITRE applies to all computing sources (e.g., VMs and physical systems with DLT nodes, connected storage, etc.) utilized in the DLT environment.
M0808 Encrypt Network Traffic	All DLT transactions or interactions should be encrypted to the maximum possible extent. This includes any API-driven interactions between the application and execution layers and interactions between the infrastructure, physical, execution, and application layers. All interactions will leverage network layer components, and therefore end-to-end encryption of all traffic should be strongly considered. All interactions about the data-model and consensus layers should be encrypted to ensure that the independent decisions by the nodes in these layers are confidential. An expansion to this mitigation strategy is to ensure that data-at-rest is encrypted, including the ledger, all static or at-rest data records on systems hosting the nodes, etc., any ICS/OT cannot inherently support encryption. The infrastructure hosting the DLT nodes could most likely support encryption, but if the respective ICS/OT systems cannot support encryption, then in-line peripheral encryption solutions may be considered. The primary concern is the impact on latency and availability.
M0930 Network Segmentation	Physical and logical network and perimeter segmentation and segregation should be used. Included is some tailored guidance: depending on the DLT architecture (single organization and single network vs. multi-organizational/multi-network architecture), the DLT nodes will most likely communicate over the internet. Depending on the use case, external non-utility applications may interact with the DLT (e.g., a non-utility node submits a bid to the DLT that runs the market logic). Irrespective of the use case, it is important not to compromise the organization's SRP. Therefore, the ICS/OT that supports the SRP triad should not be disrupted. Evaluate each use case architecture with and without DLT and compare the attack surface analysis. Perform an attack surface analysis and threat modeling of a use case with non- DLT-based traditional methods and compare against the attack surface analysis with DLT. Based on the findings, define network and device level segmentation and segregation. For instance, to limit any lateral movement through compromised DLT node systems, ensure that the infrastructure with DLT nodes is on a separate VLAN compared to the respective ICS/OT systems. Such design will ensure layer-2 segmentation, which is insufficient. Ensure that explicit firewall rules are defined to allow and deny communication/traffic between the DLT nodes and other systems. Such design will ensure layer 3-7 segmentation. Deploy IDS/IPS with deep packet inspection on communications with the DLT nodes. Suppose DLT has been used in ICS/OT use cases with some level of command and control aspects. In that case, the network and endpoint security tools can be deployed in monitor mode not to disrupt the interactions and minimize the risk of false positives. Ensure to do log collection and analysis from the DLT nodes (physical and virtual). When possible, use technologies such as Extended Detection and Response (XDR) / Endpoint Detection and Response (EDR) / Network Detection and Response (NDR) agents and antivirus/antimalware agents on the DLT nodes (physical and virtual). Leverage the guidance from IEC 62443 and implement the zoning and conduit model that includes the DLT infrastructure in the environment. Furthermore, leverage Demilitarized Zone (DMZ) for internet-facing communications from the DLT nodes.

TABLE 15 Tailored Mitigation Strategies. Part 3 of 3

Mitigation	DLT-specific guidance
M0807 Network Allowlists	Most of the guidance from M0930 applies here. The use case owners should leverage firewalls and granularly define allow lists. However, this can be challenging for some DLTs if the DLT network is configured with dynamic Internet Protocol addresses (IPs) assigned to the nodes. In such a case, whitelisting/ allows listing the IPs is impossible. If dynamic IP addressing is used for the DLT nodes across participating organizations, note that the Security Information and Event Management (SIEM) in the respective organizations may falsely categorize the inter-organization DLT node traffic as malicious. Ensure that the organizations define firewall rules to allow specific IPs of the DLT nodes to communicate, which may even involve static IP addressing. Some challenges can be addressed at the governance stage of using a permissioned DLT by agreeing upon ports, IPs, and other network details to build Allowlists. For some architectures, technologies such as VPN, Internet Protocol Security (IPSec), etc., can establish predetermined connections between the DLT nodes.
M0922 Restrict File and Directory Permissions	Guidance discussed in M0801, M0816, and M0800 apply here. Since the DLT nodes might communicate between zones and to outside nodes through the internet, evaluate the adversarial impact if the infrastructure hosting the nodes and supporting network components is compromised. Use the inferences to define file and directory-level access to assets (human, hardware, and software assets).
M0937 Filter Network Traffic	Most of the network and end-point security and network traffic analysis guidance discussed in the above mitigations is applicable here. DLT-tailored guidance under M0807 and M0930 is relevant here. For more details, MITRE's baseline guidance applies to the DLT supporting and hosting infrastructure and pertinent network traffic.
M0815 Watchdog Timers	Use of watchdog timers will help address denial of service attacks. If the DLT-based use case architecture is not configured/built based on the guidance discussed in the above mitigations, a compromised node can be used by the adversary to start a DoS attack on the ICS/OT core operations. Watchdog timers will add another layer of detection measures to check for system responsiveness. This mitigation may have little to do with the core DLT infrastructure itself but provides advantages to protect/defend the operations infrastructure that communicates with the DLT nodes.

References

1. S. Saha, N. Ravi, K. Hreinsson, J. Baek, A. Scaglione and N. G. Johnson, "A secure distributed ledger for transactive energy: The electron volt exchange (EVE) blockchain", *Appl. Energy*, vol. 282, Jan. 2021.
2. Ü. Cali, M. Kuzlu, M. Pipattanasomporn, J. Kempf and L. Bai, *Digitalization of Power Markets and Systems Using Energy Informatics*, New York, NY, USA: Springer, 2021.
3. E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains", *Proc. 13th EuroSys Conf.*, pp. 1-15, 2018.
4. *CVE Security Vulnerability Database: Apache Couchdb Security Vulnerabilities*, 2022, [online] Available: https://www.cvedetails.com/vulnerability-list/vendor_id=45/product_id=19046/Apache-Couchdb.html.
5. S. N. G. Gourisetti, A. Lee, R. Reddi, K. Isirova, M. Touhiduzzaman, D. J. Sebastian-Cardenas, et al., "Assessing cybersecurity resilience of distributed ledger technology in energy sector using the MITRE ATT&CK ICS framework", *Proc. IEEE 1st Global Emerg. Technol. Blockchain Forum Blockchain Beyond (iGETblockchain)*, pp. 1-5, Nov. 2022.
6. *ICS Matrix*, Aug. 2022, [online] Available: <https://attack.mitre.org/matrices/ics/>.
7. R. Ross, V. Pillitteri, R. Graubart, D. Bodeau and R. McQuaid, *Developing cyber-resilient systems: A systems security engineering approach*, Gaithersburg, MD, USA, Dec. 2021, [online] Available: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160v2r1.pdf>.
8. Joint Task Force Transformation Initiative of the National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, Mar. 2011, [online] Available: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-39.pdf>.
9. S. N. G. Gourisetti, Ü. Cali, K.-K.-R. Choo, E. Escobar, C. Gorog, A. Lee, et al., "Standardization of the distributed ledger technology cybersecurity stack for power and energy applications", *Sustain. Energy Grids Netw.*, vol. 28, Dec. 2021.
10. B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and philosophy", Jul. 2018.

11. O. Alexander, M. Belisle and J. Steele, "MITRE ATT&CK for industrial control systems: Design and philosophy", Mar. 2020.
12. *Dragos ICS/OT Cybersecurity Year in Review (YIR) Report*, Aug. 2022, [online] Available: <https://www.dragos.com/resource/dragos-2021-industrial-cybersecurity-year-in-review-announcement/>.
13. A. Meneghetti, M. Sala and D. Taufer, "A survey on pow-based consensus", *Ann. Emerg. Technol. Comput.*, vol. 4, no. 1, pp. 8-18, 2020.
14. A. Hrga, T. Capuder and I. P. Žarko, "Demystifying distributed ledger technologies: Limits challenges and potentials in the energy sector", *IEEE Access*, vol. 8, pp. 126149-126163, 2020.
15. T. Roth, M. Utz, F. Baumgarte, A. Rieger, J. Sedlmeir and J. Strüker, "Electricity powered by blockchain: A review with a European perspective", *Appl. Energy*, vol. 325, Nov. 2022.
16. K. Y. Yap, H. H. Chin, and J. J. Klemes, "Blockchain technology for distributed generation: A review of current development challenges and future prospect", *Renew. Sustain. Energy Rev.*, vol. 175, Apr. 2023.
17. P. R. Padghan, S. A. Daniel and R. Pitchaimuthu, "Grid-tied energy cooperative trading framework between prosumer to prosumer based on ethereum smart contracts", *Sustain. Energy Grids Netw.*, vol. 32, Dec. 2022.
18. A. E. Soto, B. L. Bosman, E. Wollega and D. W. Leon-Salas, "Peer-to-peer energy trading: A review of the literature", *Appl. Energy*, vol. 283, Feb. 2021.
19. J. Ping, Z. Yan and S. Chen, "A privacy-preserving blockchain-based method to optimize energy trading", *IEEE Trans. Smart Grid*, vol. 14, no. 2, pp. 1148-1157, Mar. 2023.
20. U. Cali, M. Kuzlu, D. J. Sebastian-Cardenas, O. Elma, M. Pipattanasomporn and R. Reddi, "Cybersecure and scalable token-based renewable energy certificate framework using blockchain-enabled trading platform", *Electr. Eng.*.
21. T. R. Alsenani, "The participation of electric vehicles in a peer-to-peer energy-backed token market", *Int. J. Electr. Power Energy Syst.*, vol. 148, Jun. 2023.
22. Y. Wu, X. Zhang and H. Sun, "A multi-time-scale autonomous energy trading framework within distribution networks based on blockchain", *Appl. Energy*, vol. 287, Apr. 2021.
23. M. Foti, C. Mavromatis and M. Vavalis, "Decentralized blockchain-based consensus for optimal power flow solutions", *Appl. Energy*, vol. 283, Feb. 2021.
24. U. Cali, M. Deveci, S. S. Saha, U. Halden and F. Smarandache, "Prioritizing energy blockchain use cases using type-2 neutrosophic number-based EDAS", *IEEE Access*, vol. 10, pp. 34260-34276, 2022.
25. K. Zhou, J. Chong, X. Lu and S. Yang, "Credit-based peer-to-peer electricity trading in energy blockchain environment", *IEEE Trans. Smart Grid*, vol. 13, no. 1, pp. 678-687, Jan. 2022.
26. D. Bose, C. K. Chanda and A. Chakrabarti, "Blockchain insisted resilience enhancement of power electricity markets using distributed energy trading", *Int. J. Emerg. Electr. Power Syst.*, vol. 23, no. 5, pp. 663-671, Oct. 2022.
27. S. N. G. Gourisetti, M. Mylrea and H. Patangia, "Evaluation and demonstration of blockchain applicability framework", *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1142-1156, Nov. 2020.
28. Y. J. Song, "Design of binary sequences with optimal frame synchronization property", *Proc. IEEE Int. Symp. Inf. Theory*, pp. 353, Jun. 2000.
29. F. Schär, "Decentralized finance: On blockchain- and smart contract-based financial markets", *Review*, vol. 103, no. 2, pp. 1-8, 2021.
30. J. Wang, W. Chen, L. Wang, Y. Ren and R. Simon Sherratt, "Blockchain-based data storage mechanism for industrial Internet of Things", *Intell. Automat. Soft Comput.*, vol. 26, no. 5, pp. 1157-1172, 2020.
31. J. M. Assante and M. R. Lee, *The Industrial Control System Cyber Kill Chain*, 2021, [online] Available: <https://sansorg.egnyte.com/dl/HHa9fCekmc>.
32. Y. A. Min, "A study on privilege elevation attack management for smart transaction security on blockchain Ethereum based system", *J. Korea Soc. Comput. Inf.*, vol. 24, no. 4, pp. 65-71, 2019.

8.4. Digital Privacy: Technical, Legal, and Legislative Aspects¹⁵⁴

Abstract

Digital services have revolutionized consumer and business quality, but often at the expense of individual privacy and security. Harvesting consumer data has become routine across many industries as organizations increasingly rely on consumer data within their operations and often extract valuable information for financial gain. As the costs of collecting, processing, and managing massive amounts of consumer data go down, consumer privacy has been largely ignored or has not been a priority. A recent rule-making effort by the United States Federal Trade Commission (FTC) to set guard rails for the collection, analysis, and monetization of consumer data is a recognition of the growing need for a renewed focus on consumer privacy protections. In this paper, we outline the technical, legal, and legislative barriers that currently inhibit control over the access and use of consumer data collected with or without consumers' knowledge or respecting their expectations of privacy. We also give a synopsis of Digital Privacy policies in the European Union, and its trend-setting legislative actions that shape privacy discussions around the world. We conclude with suggestions for new areas of research to quantify/measure consumers' expectation of privacy, improvements in technical solutions for consumers to assert control over their expected Digital Privacy, regulatory thought leadership to account for harms to consumer privacy, and a call for privacy professionals to add consumers' priorities into their designs for sustainable Digital Privacy frameworks.

1 Introduction

The continuous collection, creation, processing, aggregation, transmission, storage, sharing, interpretation, and monetization of digitized information about consumers has given rise to modern “surveillance capitalism” [1]¹⁵⁵, creating a clear and present danger for consumer privacy. An FTC fact sheet on Commercial Surveillance and Data Security Rulemaking states, “Technologies essential to everyday life also enable near-constant surveillance of people’s private lives” [2]. Essential technologies such as high-speed internet access and mobile services come with built-in capabilities to track everything from what websites users visit, what they are searching for, what stores they visit, and what routes they take to drive around their community. These technologies have enabled any social or

¹⁵⁴ This conference white paper was co-authored by K. Lambert, S. Rao, Dr Ümit Cali, Dr. Christopher Gorog, and Sin-Kuen Hawkins, but was not published.

¹⁵⁵ Source citations for Section 8.4 are listed parenthetically at the end of the section, as they were cited in the original document.

economic activity undertaken by a consumer¹⁵⁶ (individual) with persons¹⁵⁷ (individual, partnership, corporation, trust, estate, cooperative, association, government or governmental subdivision or agency, or other entity)¹⁵⁸ to be captured as information in a digital form.

Most consumers are confused and unsure of what their privacy rights entail [3]. At the same time, lured by convenience, consumers have shown a willingness to exchange data about their activities without a clear understanding of the volume of data collected or the myriad ways it gets shared and monetized by persons [4]. The gap between consumers' desires for privacy and their inability to comprehend the vulnerability of the data they are unwittingly sharing with persons in the "App Economy" [5], presents technical, legal, and legislative challenges for policymakers. Consumers need to recognize that privacy in the information age is intertwined with digitized data about themselves.

In this paper, we propose to use the term Digital Privacy to redefine privacy from consumers' points of view, as seen by socio-political structures where individuals have sovereign rights to make decisions about their own data and identities. Paraphrasing "Digital Privacy" as akin to "privacy" will help consumers visualize the role of data in terms of their individual privacy expectations. At the same time, we think it will serve as a reminder of the important correlation between digitized consumer data that organizations collect and the implicit expectation of privacy that data carries.

We define Digital Privacy as consumers' expectations of privacy of digitized information about themselves that they implicitly or explicitly share with persons in any social or economic interactions. Digital Privacy provides consumers with the power of self-sovereign control over access to digitized information about themselves by persons. It comes with the expectation that the persons will adequately maintain the confidentiality, security, and integrity of the consumer data they have collected, created, accessed, or shared.

We understand that individuals' expectations of privacy are vastly influenced by their social, cultural, and community norms and etiquette [6]. However, framing Digital Privacy as a preservation of individuals' privacy rights in the digital realm, in which individuals are free to make choices about their personal information without interference or coercion from other persons, supports the basic human rights principle of self-determination. We believe Digital

¹⁵⁶ We use consumer and individual interchangeably in this paper.

¹⁵⁷ We use persons and organizations interchangeably in this paper.

¹⁵⁸ For the purposes of this paper, we use the broad definition of 'consumer' and 'persons' originally outlined within the Fair Credit Reporting Act (FCRA) (FCRA, 1970)

Privacy is about more than just protecting personal information from unauthorized access or misuse, it is also about giving individuals control over their data and allowing them to exercise their rights to privacy, autonomy, and self-determination in the digital sphere. Likewise, this definition recognizes that people have different requirements and unique preferences about their own Digital Privacy. As such, it emphasizes the significance of individual agency and gives people meaningful choices and opportunities to exercise their Digital Privacy rights.

Upholding Digital Privacy as defined above will require a framework of policies, procedures, regulations, and standards that respect consumers' rights to privacy and their expectations of privacy as they interact with persons in any physical, virtual, or online environment.

Consumers need to be educated to understand their Digital Privacy rights, especially as organizations (persons) adopt cloud and artificial intelligence/machine learning (AI/ML) technologies to create immersive digital experiences in a ubiquitously connected world that could put their privacy at greater risk. Organizations also need to become more privacy-conscious in their design choices. At the same time, technology professionals must harness the convergence of rapid communications and computing technologies to build products and services that respect consumers' expectations of privacy.

This article presents the following contributions:

- Definitions of foundational terminologies related to Digital Privacy in the international landscape.
- Motivation for privacy professionals to design long-term interdisciplinary Digital Privacy frameworks and processes.
- Investigation of the effects of digital services and operations on individual privacy, particularly the gathering and commercialization of consumer data.
- Elevation of Digital Privacy expectations in terms of technological, legal, regulatory, and judiciary impacts on consumer data access and use from a US perspective.
- Summarizing international Digital Privacy regulations and identifying ongoing legislative actions that affect privacy deliberations.
- Calling for the study of new solutions that promote Digital Privacy and incorporate regulatory thought leadership to identify consumer privacy expectations and harms.

2 Technical Perspectives of Digital Privacy in the United States

The breakneck speed of technological advances to find new ways to extract information and value from consumer data necessitates urgent action for consumer representation in ongoing technical, legal, and legislative conversations about consumer privacy. Digital Privacy has a multitude of stakeholders that require effective cross-disciplinary interactions. Understanding the interactions between these stakeholders serves as a good reference point to set the stage for effective management of consumers' Digital Privacy. As shown in Figure-1, placing individuals and consumers as the primary drivers to set the expectations of privacy helps lay a strong foundation for a Digital Privacy framework that truly empowers consumers in their desired privacy outcomes. The directional flows of consumer data collection and the associated harms from privacy violations clearly show how consumers are the primary source of data to organizations while bearing the brunt of potential harms from privacy violations.

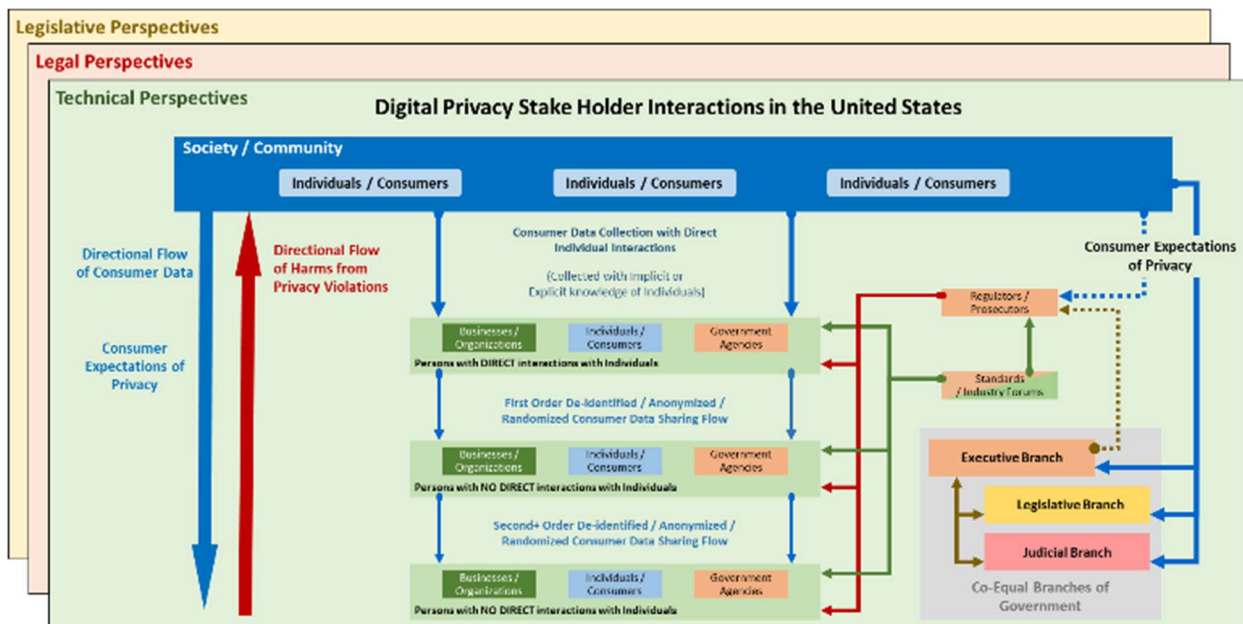


Figure-1: Digital Privacy Stakeholder Interactions

Since the digital domain is inherently technical, solutions for Digital Privacy should examine how privacy problems in information systems can be solved by better engineering designs. Understanding the data that organizations have collected about consumers is an important first step for technical teams and policymakers to support Digital Privacy. One of the earliest benefits of Fair Information Practice Principles (FIPPs), introduced in the mid-1970s [7], was the categorization of consumer data as Personal Information (PI), Personally Identifiable Information (PII), and Personal Health Information (PHI), which were subsequently defined by various legislative actions. These provided a starting point for product owners and solution designers to identify consumer data and select the use of appropriate security and privacy controls. However, PI, PII, and PHI do not adequately cover the vast

amounts of meta-data (data about data) contained in modern data-driven architectures. Consumer data collected directly from them undergo several iterations of modifications creating first-order, second-order, or even third-order derivatives (Figure-1) that are used by persons in their quest for monetization. Data derivatives of the PI, PII, or PHI from consumers are incredibly valuable, and despite anonymization techniques still carry sensitive private information when aggregated across multiple sources of data sets about consumers. Even if datasets do not directly include PI, PII, or PHI information, current advances in data analysis techniques and AI/ML tools have allowed organizations to build mosaics of intrusive information about consumers that paint detailed physical, psychological, and economic profiles of them as individuals.

Table-1: The era FIPP-based legislative actions across the world 1960s -1990s

Table-1: The era of FIPP-based legislative actions across the world 1960s - 1990s
1974 FIPPs
8 Principles
1. Transparency
2. Individual Participation
3. Purpose Specification
4. Data Minimization
5. Use Limitation
6. Data Quality and Integrity
7. Security
8. Accountability and Auditing

Static FIPPs-informed categorizations of data are no longer sufficient. Designing consumer-focused Digital Privacy solutions, therefore, needs better and more descriptive classifications of data and the information they carry. Product managers and policymakers will benefit from granular views of data to properly design solutions to address the Digital Privacy needs of consumers to suit specific use cases. Table-1, Table-2, and Table-3 provide a high-level summary of the evolution of FIPPs into privacy engineering best practices suitable for DevOps or DevSecOps of modern organizations.

Table-2: The era of Design thinking by Privacy Professionals.

1990s - 2010s	
1995 Privacy by Design	2006 FIPs / GPS
7 Principles	10 Principles
1. Proactive not Reactive; Preventative not Remedial	
2. Privacy as the Default	3. Purpose Specification
	4. Collection Limitations, Data Minimization
	5. Use, Retention, and Disclosure Limitation
3. Privacy Embedded into Design	
4. Full Functionality Positive-Sum, not Zero-Sum	
5. End to End Lifecycle Protection	7. Security
6. Visibility and Transparency	2. Accountability
	8. Openness
	10. Compliance
7. Respect for User Privacy	1. Consent
	2. Accuracy
	3. Access

Limitations of FIPPs

Organizations have commonly used the three principles of data minimization, purpose specification, and use limitations as their vehicle to design their privacy policies. Technical realizations of these policies mostly relied on Privacy Impact Assessments (PIAs) to implement and test the chosen privacy controls. As PIAs became common, consumer privacy practices were predominantly reduced to a measure of compliance with FIPPs [8]. Many of the technical designs for privacy-aware implementations of purpose specifications and use limitations based on FIPPs predominantly consist of privacy notices to inform consumers of the organization's privacy policies. Unfortunately, increasingly complex privacy notices with detailed legal language have not resulted in better-informed consumers as they acknowledge or agree to them [9]. Privacy notices have also failed to provide consumers any choice or power to negotiate such agreements, even though they have allowed organizations to be FIPPs compliant, shielding them against possible regulatory compliance and liability risks. As data became more valuable to organizations, privacy notices with even more vague purpose specifications and watered-down use limitations fueled an arms race to justify collecting increasing amounts of data swept up by automated tools. The Cambridge Analytica saga [10] with Facebook in 2018 is just one example that exposed the limitations of relying on FIPPs to prevent the inappropriate collection and misuse of data.

Table-3: The era of DevOps and DevSecOps approach to privacy engineering

2010s – Present		
2010 FTC FIPPs	2016 OMB FIPPs	2020 NIST Privacy Framework
8 Principles	9 Principles	5 Functions
7. Purpose Specification and Use Limitation	3. Purpose Specification	IDENTIFY-P (ID-P)
2. Accountability	8. Accountability and Auditing	GOVERN-P (GV-P)
7. Purpose Specification and Use Limitation	5. Use Limitation	
3. Authority		CONTROL-P (CT-P)
7. Purpose Specification and Use Limitation	3. Purpose Specification	
	5. Use Limitation	
6. Individual Participation	2. Individual Participation	
4. Minimization	4. Data Minimization	COMMUNICATE-P (CM-P)
1. Access and Amendment	2. Individual Participation	
9. Transparency	1. Transparency	
7. Purpose Specification and Use Limitation	3. Purpose Specification	
	5. Use Limitation	PROTECT-P (PR-P)
7. Purpose Specification and Use Limitation	3. Purpose Specification	
	5. Use Limitation	
5. Quality and Integrity	6. Data Quality and Integrity	
8. Security	7. Security	

Engineering Digital Privacy by Design

Introduced in 1995, Privacy by Design (PbD) [11] expanded FIPPs ideas with a set of seven foundational principles to provide a technical framework for product designers. These went beyond the FIPPs framework and provided a more holistic approach to privacy for implementation teams. However, the aspirational nature of the PbD principles did not gain mainstream adoption by the technical community in the absence of policy standards by regulators and legislative bodies around the world. The advent of Privacy Engineering [12] eventually paved the way for implementation teams to translate their PbD plans into reality. Applying systems engineering methods to incorporate privacy within the designs of IT systems and business operations allowed organizations to proactively manage privacy as legal and regulatory compliance risks instead of merely being reactive to incidents of privacy breaches.

In early 2020, the National Institute of Science and Technology (NIST) published its very first Privacy Framework [13], which became the most comprehensive set of tools for organizations to measure their privacy readiness to date. Organizations can use it to plan, design, execute, operate, and measure their privacy-aware

processes. Although it primarily views privacy as an enterprise risk, it is a practical methodology for privacy practitioners to adopt. It provides an industry-agnostic approach to incorporating PbD into rigorous and repeatable privacy engineering best practices. Organizations can use NIST Privacy Framework to build resilience into their operations, lower potential risks from privacy breaches, and build trust with consumers.

Need for Consumer-centric Privacy Engineering

Using enterprise risk management as a vantage point to uphold consumers' Digital Privacy fails to fully support consumer-centric design choices. This is because the primary benefactor of such risk calculations remains the persons who handle consumer data and might have a conflict of interest. A nearly 2% rise in Facebook's share price increased its market value by \$10B on the same day the FTC imposed a blockbuster \$5B fine as a result of the privacy violations from the Cambridge Analytica episode [14], showing the limitations of managing privacy as an enterprise risk and illustrates how markets have started to factor potential liability for privacy violations by organizations into their stock valuations. The rise in the market cap also reflected the collective relief from the investors about the level of fines imposed on Facebook for their privacy violations. Focusing on enterprise risk also pushes technical implementations to conflate Digital Privacy with Data Security and Data Protection. Data Security is the application of security controls to prevent unauthorized access, whereas Data Protection deals with ensuring data availability and preventing data loss under adverse conditions.

Bringing consumers' voices into privacy engineering best practices will require new techniques to quantify consumer harms as part of enterprise risk. New research may be needed to explore methods for representing consumers' expectations of privacy as measurable metrics for technical implementations. Privacy-preserving design patterns may be needed to address industry-specific challenges such that consumers remain the beneficiary of these designs. At the same time, technical designers may need to find ways to use new advances in observability within IT/Network operations to provide consumers some measure of control over how and where their data are being used. Increasing use of cloud computing and data centralization (Data Lakes, Big Data tools) along with the speed of data transactions can adversely affect consumer privacy and require new engineering practices to limit consumer harms caused by privacy breaches. Technologies to mitigate these harms, such as end-to-end data encryption, immutability, and non-repudiation of transactions, present many opportunities for further study within academia and technology institutes.

3 Legal Perspectives of Digital Privacy in the United States

Privacy has been part of accepted norms within human settlements for ages and found acceptance in legal proceedings in the form of common law or tort law for several centuries. As technology evolved and institutional structures emerged, formal declarations of privacy found their way into constitutions or legislation that codified privacy protections for citizens. Privacy, however, remains one of the most complicated legal questions today. Within the context of the stakeholder interactions in Figure-1, from a legal perspective, one should observe that privacy is not explicitly mentioned as a right directly enshrined in the US Constitution. It is nevertheless a right acknowledged by courts via the application of various doctrines, statutes, and precedents over the last two-hundred-plus years that collectively serve to uphold and extend, or sometimes even reduce, the privacy rights of individuals. Each decision made by different courts has added a gradual expansion of the meaning and scope of the corpus of law concerning the privacy rights of individuals. The existing legal infrastructure within the United States is therefore a combination of constitutional rights, tort law, statutory law, and judicial rulings that acknowledge individuals' "right to privacy" and "expectations of privacy."

Many of the privacy rights individuals have today are based on precedents set by case law and court decisions rendered after incidents of privacy violations rather than as preemptive efforts designed to protect against such violations from occurring. Justice Louis Brandeis postulated that the US Constitution granted privacy which he interpreted as the "right to be left alone" [15]. In the modern digital ecosystem, this may also extend to the "right to forget" or expunge personal data that may otherwise permanently exist in the public domain. William Prosser [16] explained the four tort laws that help provide the legal basis for individuals' privacy rights. The privacy rights derived from the US Constitution are:

- privacy of beliefs [17]
- privacy of the home [18]
- privacy of persons, houses, papers, and effects [19]
- more general protection of privacy [20]
- due process of law [21]

Legal doctrines developed by lawyers and justices have been used to test individuals' reasonable expectations of privacy and to expand their privacy rights derived from the Constitution. At the same time, technological advances have created new privacy concerns raised by consumers, which in turn push the legal community to innovate and find new ways to define and test the boundaries of individuals' privacy rights. Devising tests for legal doctrines follow

more subjective processes than those used by technical professionals to test precise engineering solutions. Evaluating doctrines for expectations of privacy often rely on examples of physically tangible privacy that society and legal communities have accepted as reasonable over time. As new technologies are introduced, developing such doctrines and tests for proving or disproving individuals' claims to privacy frequently involve the time-consuming process of building case law precedents.

In 1928, the "trespass doctrine," for example, was used to test individuals' privacy claims against Fourth Amendment protections of persons, houses, papers, and effects within a "constitutionally protected area" [22]. That view of privacy essentially allowed for individuals' privacy expectations to be limited only to a physical space within which Fourth Amendment protections were recognized. Any actions infringing on privacy performed outside individuals' "constitutionally protected areas" were not considered sufficient to prove an expectation of privacy and failed the "trespass doctrine." Before the invention of electronic communications and digital devices, such a physically constrained view of privacy expectations might have seemed reasonable. However, Justice Louis Brandeis questioned the validity of this view when applied to wiretapping of individuals' telephone conversations, even when the wiretaps were physically installed by government agents on a public street, which was technically declared to be outside the constitutionally-protected area of the individual in question. In a dissenting opinion to the *Olmstead v. United States* [23] decision, Justice Brandeis said the actions by government agents were no different from opening a sealed letter and violated the "right to be let alone" protected by the Fourth Amendment. Although warrantless wiretapping was illegal in Washington State, the *Olmstead* decision still ruled that the government had not violated individuals' Fourth Amendment rights since there was no trespass. The legally recognized privacy protections and the doctrines used to test violations, in this case, failed to uphold individuals' changing expectations of privacy when using new technologies.

In the following years, rapid technological changes led the courts to recognize the "public-observation and third-party doctrine" which was used by Justice Stewart to write that, "the Fourth Amendment protects people, not places," in the landmark *Katz v. United States* decision that reversed the *Olmstead* decision and agreed with the basic premise of the previous dissent by Justice Brandeis [24]. From *Olmstead* to *Katz*, it took nearly 40 years for a legal theory to devise a way to advance Fourth Amendment protections for people, not places. The *Katz* decision also established a test for reasonable expectations of privacy [25]. The ruling clarified that when individuals knowingly disclose things in public, they do not have Fourth Amendment privacy protections, even if they do so in their homes

(constitutionally protected areas). But if individuals demonstrate reasonable expectations of privacy even in public areas (public observation), then their privacy is constitutionally protected under the Fourth Amendment. This continues to be relevant today. When individuals exercise their speech in public forums such as social media or other outlets, they cannot expect privacy protections. A literal application of the “third party doctrine” could easily mean that in the current data-driven economy, individuals may only rarely be able to exercise Fourth Amendment privacy protections if they are willing to share information with other persons in public. Despite pushing the boundary of legal privacy protections in an innovative manner, it took another 45 years for the limitations inherent in the Katz decision to be addressed by the innovative “mosaic theory doctrine” [26].

First used in *United States v. Jones* [27], the “mosaic theory doctrine” established that a series of information communications collected over a period amounted to a search under Fourth Amendment protection definitions. Using this new theory, the courts ruled that the collection of GPS data by the government over an extended period of a month for the purposes of tracking individuals’ movements constitutes an unreasonable search under the Fourth Amendment and failed the test for reasonable expectations of privacy. This expanded the privacy protections for individuals as digital technologies created new ways to track individuals. It also challenged the assertion by the third-party doctrine that individuals did not have a reasonable expectation of privacy when they volunteered information to third parties. The subsequent *Carpenter v. United States* decision [28] expanded on this theory by considering the nature of the information being collected. In that case, a question was raised whether a suspect’s cell phone location information data, aggregated over time, violated the suspect’s reasonable expectation of privacy [29]. These court decisions show that individuals can claim a reasonable expectation of privacy even when they share data with persons or third parties in the course of their social and economic interactions using digital services. While all of these forays into establishing privacy in the eyes of the law and judiciary have had success in clarifying the law, the capabilities of technologies such as cloud computing, 5G, AI/ML, the metaverse, and public surveillance by police and intelligence organizations threaten to further complicate contemporary legal doctrines.

Most of the case law affirming individuals’ right to privacy, and the tests to determine an expectation of privacy in various situations, have been when the individuals challenged government actions. As non-governmental persons collect individuals’ data and trade them in private or public marketplaces there are no easy legal mechanisms for individuals to assert their expectations of privacy despite the precedents set by case law. The existing legislative and regulatory frameworks in most cases do not provide individuals with a “private right of action” to sue

organizations in the event of privacy violations. The FTC has used its authority to regulate trade and protect consumers against unfair business practices and has become the de facto privacy ombudsman at the federal level in the United States. At the state level, attorneys general usually have the executive authority to enforce consumer protection laws. Relying heavily on the legal doctrines and interpretations of privacy rights via case law for new technologies however has its pitfalls, like the “public observation and third-party doctrine” used to reinterpret “expectations of privacy” defined by the “trespass doctrine.” Now more than ever, there is a need to develop conceptual and analytical structures to provide legal protections for individuals' Digital Privacy that transcends the individual as a physical being and extends to their virtual self as a digital twin or an avatar in the metaverse. Establishing case law precedents for such complex legal issues will be slow and time-consuming, probably taking decades for consumers to gain adequate legal protections. A more comprehensive and deliberate process can often be found through the legislative process.

4 Legislative Perspectives of Digital Privacy in the United States

In 1973, a US Department of Health, Education, and Welfare (HEW) Report recognized the increasing use of electronic data stores and articulated a set of basic principles for protecting personal data about individuals. Those principles have become a standard model for legislation around the world [30]. The original principles defined by the HEW Report were codified into legislation in the Privacy Act [31] in the form of Fair Information Practice Principles (FIPPs). Since then, FIPPs have had a first-mover advantage in their influence on legislative thought leadership, including adoption by the Organisation for Economic Co-operation and Development (OECD) [32], and then by the European Union (EU) Directive 95/46/EC [33]. FIPPs-like thinking has continued to dominate legislation and privacy processes around the world. Modified forms of FIPPs are part of legislation for privacy and data protection in China [34], Australia [35], Brazil [36], New Zealand [37], and several other countries. Starting as one report in one executive branch in the United States, the migration of FIPPs as “best practices” through several legislative bodies is a good example of how a technical solution became a legal solution, and eventually through a series of deliberative legislative solutions became the basis for law in several countries.

Although the Privacy Act of 1974 started the ball rolling, privacy protections in the United States have typically covered specific industries as reactions to court cases and rulings. States have sometimes taken more aggressive stances on privacy and enacted their own laws. This has contributed to legislative and regulatory complexity for consumers as well as organizations, and sometimes with conflicting policies as shown in Table-4. Regulatory policies might be expedient, but do not have the weight of the law. Judicial decisions on the other hand

might be narrowly focused within the scope of existing statutes, previous case law, and single issues, without the benefit of a long-term study, legislative deliberation, public debate, and open participation by the public. If done well, statutory law can provide more transparent and authoritative guidance on issues like privacy, which can evolve as popular sentiment and politics shape the debates and identify new principles.

Despite their widespread use, FIPPs-enabled laws have not been completely successful in providing consumers with true Digital Privacy. In practice, most regulations based on FIPPs have been limited to flurries of consent requests and privacy notices to consumers. FIPPs-inspired legislative efforts also tend to equate consumers' Digital Privacy to their Data Privacy and Data Protection. Consumers do not yet have any practical control over government or business policies, the amount of data being collected, the purpose for its collection, its ownership, or consent for its use. Privacy policies and consent notices lack any measure of consumers' ability to negotiate in their interactions with the persons seeking their acknowledgment for collecting data about them.

Data Privacy focuses on the de-identification and anonymization of consumers' identities from the data collected about them, whereas Data Protection, as noted before, is focused on managing the availability of collected data and data loss prevention. Digital Privacy goes beyond both of these capabilities and covers the expectation of privacy by individuals in exchange for sharing their data during any interaction with organizations. The scope of legislative thinking may need to include ways to define and quantify consumers' expectations of privacy in ways that consumers can understand. At the same time, legislation may need to acknowledge guidelines from the judicial and legal communities that test expected privacy under specific situations. Legislation may also need to have a view toward new technologies to allow for innovative solutions to emerge while setting boundaries for consumers' privacy expectations when using them.

Table-4: Sectoral/State Privacy Regulations in the USA

1960s - 1990s	Covered Industries
Fair Credit Reporting Act, 1970 [38]	Credit Reporting Companies, Financial Industry
Family Educational Rights and Privacy Act (FERPA), 1974 [39]	Educational Institutions
Privacy Act of 1974 [31]	US Federal Agencies
1990s - 2010s	Covered Industries
Drivers Privacy Protection Act (DPPA), 1994 [40]	State Bureau of Motor Vehicle (BMV) Agencies
Health Insurance Portability and Accountability Act (HIPAA), 1996 [41]	Healthcare Industry
Children's Online Privacy Protection Act (COPPA), 1998 [42]	All online systems
Gramm-Leach-Bliley Act (GLBA), 1999 [43]	Financial Industry
Federal Information Security Management Act (FISMA), 2002 [44]	US Federal Agencies
2010s - Present	Covered Industries
California Consumer Privacy Act (CCPA), 2018 [45]	All industries handling PII for California residents
California Privacy Rights Act (CPRA), 2020 [46]	All industries handling PII for California residents
Colorado Privacy Act, 2021 [47]	All industries handling PII for Colorado residents
Virginia Consumer Data Protection Act, 2021 [48]	All industries handling PII for Virginia residents
Connecticut Data Privacy Act, 2022 [49]	All industries handling PII for Connecticut residents
Utah Consumer Privacy Act, 2022 [50]	All industries handling PII for Utah residents

Another complicated dynamic is that every jurisdiction, every individual, and every culture may base their privacy expectations on vastly different subjective perceptions, which might require a great deal of flexibility in formulating common policies. Quality attribute metrics, possibly discovered by quantitative surveys of consumer opinions over time, might be useful to help determine the amount of convergence or divergence in subjective consumer priorities and desires, the degree of central tendencies, and the statistical sensitivity to differing privacy-related practices. Processes like that could discover and implement more rigorous and politically acceptable nonpartisan legislative outcomes.

Conceived at a time when monetization of consumer data was not the norm, legislation needs to evolve beyond FIPPs thinking to new ideas for defining and managing Digital Privacy in the complex environments of ubiquitous financial technologies, exploitive Big Data, intrusive data breaches, artificial intelligence, and the

metaverse. A taxonomy of consumer Digital Privacy needs to consider societal as well as personal (physical, psychological, and economic) harms inflicted on individuals when organizations fail to adequately account for their individual privacy concerns. Little or none of the privacy legislation currently on the table in the United States and around the world adequately define consumer harm or provide effective provisions to address it. This remains an area that needs attention across the board. Research within academia, policy think tanks, and standards bodies may need to expand their privacy discourse to bring consumers into focus, and to offer improved policy options for legislation. Legislation also needs to guard against creating unintended moral hazards, or arbitrary, unfair, or inconsistent punishments. For example, if fines are levied for data breaches, are they imposed for each breach, each compromised data element, each harm inflicted, or over a spectrum of harms from a single breach? This might illustrate some extremely complex problems that make simple solutions difficult.

5 International Perspectives on Digital Privacy

According to the European Union's Charter of Fundamental Rights [51], the right to personal data protection is a fundamental right. Article 8 of the Charter states, "Everyone has the right to the protection of personal data concerning him or her." Moreover, the "Right to Respect for Private Life" is addressed under Article 7 of the same Charter. Although both rights concern privacy and personal data, they are distinct and serve different purposes. The same rights are mentioned in Article 8 of the European Convention on Human Rights (ECHR) [52], but only acknowledge a broad right to privacy. In terms of respect to fundamental rights, the following four important interests are protected: private life, family life, home, and correspondence.

Data protection is not necessarily synonymous with privacy, particularly in jurisdictions such as the European Union (EU). As a result, personal data privacy is strictly guaranteed in the EU legislative framework, most notably via the General Data Protection Regulation (GDPR) [53], which was based on prior legal standards such as Directive 95/46/EC [33] and others. The creation of a balance between data protection regulations as essential human rights and free data movement to have an effective single market in the EU is a difficult undertaking to complete.

The EU has stated unequivocally that the right to personal data protection is a fundamental right, and that Digital Privacy aspects have significant ramifications for the interpretation and implementation of the GDPR. The provisions of the GDPR are intended to give effect to this right. Article 8(1) of the European Union's Charter of Fundamental Rights (the "Charter") [51] and Article 16(1) of the Treaty on the Functioning of the European Union

(TFEU) [54] are directly linked to the GDPR to provide a solid foundation for protecting personal data as a critical aspect of fundamental human rights. According to Article 16 of the TFEU, "everyone has the right to the protection of personal data relating to them."

Additionally, as emphasized in C-203/15 [55], Articles 7 and 8 of the Charter are unique in terms of basic human rights interpretation. As a final note, the GDPR is a better legal standard in contrast to its worldwide equivalents since, it is a law that can be applied directly to real-life implementations more successfully, providing a firm foundation for future legal standards in the EU or other territories.

The EU has taken a leading role among major global economies in safeguarding individuals' personal data with the introduction of the GDPR in May 2018 with a large Digital Single Market in EU territory. GDPR has protected the information of EU citizens by imposing strict rules for the acquisition and processing of personal data. GDPR also requires companies to get permission from consumers before collecting and using their data, and it allows consumers the opportunity to see, modify, and remove their information.

Table-5: Overview of Prominent International Privacy Regulations (US and EU Region)

Law/Agreement	Country/Region	Definition
General Data Protection Regulation (GDPR) [56]	European Union (EU)	A regulation aimed at protecting people's privacy and personal data within the EU. It applies to every organization that handles the personal data of EU people, regardless of its location. The GDPR requires enterprises to get consent for data gathering, maintain data accuracy, and provide people access to and control over their data.
Directive on ePrivacy [57]	European Union (EU)	A directive intended to protect individuals' privacy while utilizing electronic communications. It requires businesses to acquire permission for the use of cookies, to notify customers about data breaches, and to restrict the use of unsolicited electronic communications (such as spam). The GDPR-compliant guideline is presently being modified.
California Consumer Privacy Act (CCPA) [45]	State of California, United States	Legislation that provides California citizens the right to know what personal data is being gathered about them, to have it destroyed, and to opt out of having their data sold to enhance privacy rights and consumer protection. The CCPA is applicable to businesses that gather data from California citizens and satisfy specific criteria which have specific characteristics in terms of annual revenues and the number of individuals whose data is processed.
Children's Online Privacy Protection Act (COPPA) [42]	United States of America (USA)	A US legislation that governs the acquisition of personally identifiable information from minors under the age of 13. It requires businesses to get a parental agreement before collecting data from children, to notify parents about data collection methods, and to allow parents the ability to access and remove their child's information.
EU-US Privacy Shield [58]	European Union (EU) and United States of America (USA)	The EU-US Privacy Shield was an agreement between the European Union and the United States that permitted personal data to be transferred between the two countries. The deal required US firms to self-certify that they met EU data protection rules, such as the General Data Protection Regulation (GDPR). The Privacy Shield also granted EU individuals some rights, including the ability to view and update their data and register complaints against US corporations. Nevertheless, the Privacy Shield was declared invalid by the European Union's Court of Justice in 2020 because of concerns about US monitoring methods and a lack of effective protections for EU individuals' data.

Table-6: Overview of Prominent International Privacy Regulations (APEC, India, China).

Law/Agreement	Country/Region	Definition
OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data [32]	Organization for Economic Cooperation and Development (OECD)	Guidelines outlining principles for preserving individuals' personal data privacy in the context of transborder data transfers. The rules underscore the necessity of gaining permission, giving notifications, and restricting personal data usage and dissemination.
Asia-Pacific Economic Cooperation (APEC) [59]	Asia-Pacific Economic Cooperation (APEC)	A framework for preserving individuals' personal data privacy in the context of cross-border data transfers among APEC member countries. The framework highlights the significance of gaining permission, giving notice, and putting security measures in place.
Digital Personal Data Protection, 2022 [60]	India	The Digital Personal Data Protection introduced in 2022 aims to recognize individuals' right to protect their personal data while also creating a process for access to the individual's personal data for lawful purposes. It seeks to build toward a vision for a comprehensive Digital India Act that is also currently in the works. This is a draft policy document that seeks to get public input. If passed this will establish rules of engagement for the cross-border transfer of data about Indian citizens while also adding data fiduciary responsibilities to the organizations that handle the individuals' personal data.
Personal Information Protection Law (PIPL) [34]	China	China's Personal Information Protection Law (PIPL) governs the gathering, use, and storage of people's personal information with the goal of preserving their right to privacy. Organizations are required by law to get people's express permission before collecting any of their personal information and to use it only for those specified, authorized reasons. Moreover, the legislation grants people the right to see, update, and delete their personal information and mandates that businesses take steps to protect it from being accessed, used, or disclosed without authorization. The PIPL also specifies guidelines for the cross-border transfer of personal data, mandating that enterprises seek people's express agreement before sending their personal data outside of China and adhere to certain data localization standards. Overall, the PIPL offers a thorough legal framework for defending people's privacy rights in China.

Privacy and the protection of data are increasingly essential considerations for individuals and businesses alike in today's international society. To address these issues, different laws and agreements governing the collection, use, and sharing of personal data have been enacted in the global and international context. In addition to the GDPR, the ePrivacy Directive of the European Union [57] provides rules for data protection and privacy in the digital domain of the EU. Even though the EU has some state-of-the-art legal standards in this field, it is essential to look at the global landscape to gain a wider perspective. The United States has its own set of rules, such as the California Consumer Privacy Act (CCPA) [45], the Children's Online Privacy Protection Act (COPPA) [61], and the Health Insurance

Portability and Accountability Act (HIPAA) [41]. International treaties such as Convention 108 [62], the OECD Guidelines [32], and the APEC Privacy Framework [59] all give guidelines on how to safeguard personal data in cross-border scenarios. Although these procedures give some degree of protection, concerns regarding data privacy remain, as proven by the invalidation of the EU-US Privacy Shield in 2020 [58]. As technology advances, data security and privacy are likely to be critical issues for many years to come.

6 Conclusion

Digital Privacy is an area of intense interest that will shape consumer experiences and public policy in a hyper-connected world. While consumer data continues to be a key driver for economic and social activity across the world, consumers themselves are generally unaware of the overall value and vulnerability of their data within the emerging digital ecosystem. Ethical and effective ideas can be advanced in the form of model legislation, supported by interested legislators and advocacy groups, and widely proliferated for legislative action across jurisdictions. Within most states or countries, technical, legal, and legislative perspectives should ideally not be competitive, but designed to inform and support process improvements, and generate healthy communications among all the stakeholders in evaluating consumers' Digital Privacy outcomes from all three perspectives.

A consumer-centric Digital Privacy Framework goes beyond Data Protection and Data Privacy and should cover the expectation of privacy by individuals in exchange for sharing their data during any interaction with organizations.

- Technology professionals might need to consciously adopt Privacy by Design (PbD) as part of their engineering best practices that translate consumer expectations of privacy into detailed requirements for functional capabilities within their solutions. Solution designers may need to find mechanisms to quantify and/or qualitatively measure consumers' expectations of privacy in the products and services they build for organizations.
- Legislators might need to modernize their existing FIPPs framework to regulate and enforce evolving consumer expectations of privacy. Legislation might need to consider a taxonomy of societal as well as personal (physical, psychological, and economic) harms inflicted on individuals when organizations fail to adequately account for consumers' privacy expectations. An industry-agnostic approach might be needed to ensure that cross-industry consumer data collection practices

do not slip through regulatory oversight. A minimum standard for security and privacy controls might need to be part of the legislative repertoire relative to the harms inflicted on consumers when privacy violations occur.

- Regulatory agencies and prosecutors might need to reevaluate the effectiveness of their enforcement methods for Digital Privacy protection and create appropriate incentives and disincentives for adequate and fair enforcement.
- Legal and judicial matters for Digital Privacy might need to account for privacy harms from new technologies and devise methods to protect and serve the needs of consumers in their quest to gain Digital Privacy.
- Organizations collecting, processing, storing, and sharing consumer data might need to reconsider their overall risk profiles to account for consumer harms as new abuses of consumer data for monetization are discovered.
- Consumers might also need to become more aware and educated of the value and risks their data represents to organizations they are willing to share it with or those that have incentives to take it without their informed consent.

Ongoing multi-disciplinary collaborations and activities such as those organized by the IEEE Digital Privacy Initiative [63] are vital to encourage communications among individual and cross-industry stakeholders to shape the Digital Privacy landscape. These efforts will foster a common language for discussion, encourage knowledge sharing on unique cultural perspectives, promote standardization and best practices, drive advanced research, and create innovative solutions to foster respect for individuals' Digital Privacy.

References

- [1] S. Zuboff, "Big other: Surveillance Capitalism and the Prospects of an Information Civilization," 1 March 2015. [Online]. Available: <https://journals.sagepub.com/doi/10.1057/jit.2015.5>.
- [2] S. T. Nguyen, "Commercial Surveillance and Data Security Rulemaking," 11 August 2022. [Online]. Available: <https://www.ftc.gov/legal-library/browse/federal-register-notices/commercial-surveillance-data-security-rulemaking>. [Accessed 28 February 2023].
- [3] B. Auxier, L. Rainie, M. Anderson, A. Perrin, M. Kumar, and E. Turner, "Americans and Privacy: Concerned, Confused and Feeling Lack of Control Over Their Personal Information," 15 November 2019. [Online]. Available: <https://www.pewresearch.org/internet/2019/11/15/americans-and-privacy-concerned-confused-and-feeling-lack-of-control-over-their-personal-information/>. [Accessed 28 February 2023].
- [4] FTC, "Data Brokers: A Call For Transparency and Accountability: A Report of the Federal Trade Commission (May 2014)," 19 May 2014. [Online]. Available: <https://www.ftc.gov/system/files/documents/reports/data-brokers-call-transparency-accountability->

- report-federal-trade-commission-may-2014/140527databrokerreport.pdf. [Accessed 28 February 2023].
- [5] M. Mandel and J. Shapiro, "U.S. App Economy Update, 2022," 25 May 2022. [Online]. Available: https://www.progressivepolicy.org/wp-content/uploads/2022/05/PPI_US-App-Economy-Update-2022_V4.pdf. [Accessed 28 February 2023].
 - [6] P. Fleming, A. P., Bayliss, S. G. Edwards and C. R. Seger, "The role of personal data value, culture and self-construal in online privacy behaviour," 16 July 2021. [Online]. Available: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0253568#pone.0253568.ref035>.
 - [7] FPC, "Fair Information Practice Principles (FIPPs) | FPC.gov," 1973. [Online]. Available: <https://www.fpc.gov/resources/fipps/>. [Accessed 28 February 2023].
 - [8] S. S. Shapiro, "Time to Modernize Privacy Risk Assessment," *Issues in Science and Technology*, vol. 1, no. 38, p. 20–22, Fall 2021.
 - [9] J. R. Reidenberg, N. C. Russell, A. J. Callen, S. Qasir and T. B. Norton, "Privacy Harms and the Effectiveness of the Notice and Choice Framework," 3 March 2014. [Online]. Available: <https://core.ac.uk/download/pdf/159586048.pdf>. [Accessed 28 February 2023].
 - [10] I. ur Rehman, "Facebook-Cambridge Analytica data harvesting: What you need to know," 1 January 2019. [Online]. Available: <https://core.ac.uk/download/pdf/220153793.pdf>. [Accessed 28 February 2023].
 - [11] A. Cavoukian, "Privacy by Design: The 7 Foundational Principles Implementation and Mapping of Fair Information Practices," May 2010. [Online]. Available: <https://gpsbydesigncentre.com/portfolio-item/privacy-by-designthe-7-foundational-principlesimplementation-and-mapping-of-fair-information-practices/>. [Accessed 1 March 2023].
 - [12] A. Cavoukian, S. Shapiro, and R. J. Cronk, "Privacy Engineering: Proactively Embedding Privacy by Design," 6 January 2014. [Online]. Available: <https://www.ipc.on.ca/wp-content/uploads/Resources/pbd-priv-engineering.pdf>. [Accessed 1 March 2023].
 - [13] NIST, "Privacy Framework | NIST," 8 January 2020. [Online]. Available: <https://www.nist.gov/privacy-framework/privacy-framework>. [Accessed 1 March 2023].
 - [14] C. Jee, "Facebook is actually worth more thanks to news of the FTC's \$5 billion fine," 15 July 2019. [Online]. Available: <https://www.technologyreview.com/2019/07/15/134196/facebook-is-actually-richer-thanks-to-news-of-the-ftcs-5-billion-fine/>. [Accessed 1 March 2023].
 - [15] S. D. Warren and L. D. Brandeis, "The Right to Privacy," 15 December 1890. [Online]. Available: <https://www.cs.cornell.edu/~shmat/courses/cs5436/warren-brandeis.pdf>. [Accessed 1 March 2023].
 - [16] W. L. Prosser, "Privacy," August 1960. [Online]. Available: https://canvas.cornell.edu/courses/21938/files/1919488/download?download_frd=1. [Accessed 4 March 2023].
 - [17] U. 1789 Congress, "Freedom of Religion, Speech, Press, Assembly, and Petition (Bill of Rights)," 15 December 1791. [Online]. Available: <https://constitutioncenter.org/the-constitution/amendments/amendment-i>.
 - [18] U. 1789 Congress, "Quartering of Soldiers," 15 December 1791. [Online]. Available: <https://constitutioncenter.org/the-constitution/amendments/amendment-iii>.
 - [19] U. 1789 Congress, "Search and Seizure," 15 December 1791. [Online]. Available: <https://constitutioncenter.org/the-constitution/amendments/amendment-iv>.
 - [20] "Non-Enumerated Rights Retained by People," 15 December 1791. [Online]. Available: <https://constitutioncenter.org/the-constitution/amendments/amendment-ix>.
 - [21] "14th Amendment - Citizenship Rights, Equal Protection, Apportionment, Civil War Debt | Constitution Center," 13 June 1866. [Online]. Available: <https://constitutioncenter.org/the-constitution/amendments/amendment-xiv>. [Accessed 1 March 2023].
 - [22] W. S. Doenges, "Search and Seizure: The Physical Trespass Doctrine and the Adaption of the Fourth Amendment to Modern Technology," 1965. [Online]. Available: <https://digitalcommons.law.utulsa.edu/cgi/viewcontent.cgi?article=1038&context=tlr>. [Accessed 4 March 2023].

- [23] W. H. Taft, "Olmstead v. United States," 4 June 1928. [Online]. Available: <https://www.oyez.org/cases/1900-1940/277us438>. [Accessed 4 March 2023].
- [24] P. Stewart, "Katz v. United States," 18 December 1967. [Online]. Available: <https://www.oyez.org/cases/1967/35>. [Accessed 4 March 2023].
- [25] O. S. Kerr, "The Case for the Third-Party Doctrine," 2009. [Online]. Available: <https://repository.law.umich.edu/mlr/vol1107/iss4/1/>. [Accessed 4 March 2023].
- [26] O. S. Kerr, "The Mosaic Theory of the Fourth Amendment," 2012. [Online]. Available: <https://repository.law.umich.edu/mlr/vol111/iss3/1/>. [Accessed 4 March 2023].
- [27] A. Scalia, "United States v. Jones," 23 January 2012. [Online]. Available: <https://www.oyez.org/cases/2011/10-1259>. [Accessed 4 March 2023].
- [28] J. G. Roberts, "Carpenter v. United States," 22 June 2018. [Online]. Available: <https://www.oyez.org/cases/2017/16-402>. [Accessed 4 March 2023].
- [29] T. H. Wilson, "The Mosaic Theory's Two Steps: Surveying Carpenter in the Lower Courts," 2023. [Online]. Available: <https://texaslawreview.org/the-mosaic-theorys-two-steps-surveying-carpenter-in-the-lower-courts/>. [Accessed 4 March 2023].
- [30] "Records, Computers, and the Rights of Citizens: Report of the Secretary's Advisory Committee on Automated Personal Data System," 1 July 1973. [Online]. Available: <https://www.justice.gov/opcl/docs/rec-com-rights.pdf>. [Accessed 5 March 2023].
- [31] "Overview of the Privacy Act of 1974, 2020 Edition," 1974. [Online]. Available: <https://www.justice.gov/media/1122281/dl?inline>. [Accessed 5 March 2023].
- [32] "Guidelines Governing the Protection of Privacy and Transborder Flows of Personal Data," 22 September 1980. [Online]. Available: <https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0188>. [Accessed 5 March 2023].
- [33] "Directive 95/46/EC," 24 October 1995. [Online]. Available: https://edps.europa.eu/sites/default/files/publication/dir_1995_46_en.pdf. [Accessed 5 March 2023].
- [34] "Translation: Personal Information Protection Law of the People's Republic of China – Effective Nov. 1, 2021," 20 August 2021. [Online]. Available: <https://digichina.stanford.edu/work/translation-personal-information-protection-law-of-the-peoples-republic-of-china-effective-nov-1-2021/>. [Accessed 5 March 2023].
- [35] "The Privacy Act - Australia," 14 December 1988. [Online]. Available: <https://www.oaic.gov.au/privacy/the-privacy-act>. [Accessed 5 March 2023].
- [36] "Brazilian General Data Protection Law," 2018. [Online]. Available: https://iapp.org/media/pdf/resource_center/Brazilian_General_Data_Protection_Law.pdf. [Accessed 5 March 2023].
- [37] "Privacy Act 2020 No 31 (as of 30 November 2022), Public Act Contents – New Zealand Legislation," 30 June 2020. [Online]. Available: <https://www.legislation.govt.nz/act/public/2020/0031/latest/LMS23223.html>. [Accessed 5 March 2023].
- [38] "1128 US Public Law 91-508: Fair Credit Reporting Act," 26 October 1970. [Online]. Available: <https://www.congress.gov/bill/91st-congress/house-bill/15073/text/pl?overview=closed>. [Accessed 28 February 2023].
- [39] "Family Educational Rights and Privacy Act (FERPA)," 21 August 1974. [Online]. Available: <https://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html>.
- [40] "Drivers Privacy Protection Act (DPPA)," 1994. [Online]. Available: <https://www.law.cornell.edu/uscode/text/18/2721>.
- [41] U. 1996 Congress, "Health Insurance Portability and Accountability Act of 1996 (HIPAA)," 21 August 1996. [Online]. Available: <https://www.cdc.gov/php/publications/topic/hipaa.html>.
- [42] U. 1998 Congress, "Children's Online Privacy Protection Act of 1998 (COPPA)," 21 October 1998. [Online]. Available: <https://www.ftc.gov/legal-library/browse/rules/childrens-online-privacy-protection-rule-coppa>.

- [43] U. 1999 Congress, "Gramm-Leach-Bliley Act," 12 November 1999. [Online]. Available: <https://www.ftc.gov/legal-library/browse/statutes/gramm-leach-bliley-act>.
- [44] "H.R.3844 - Federal Information Security Management Act of 2002," 17 December 2002. [Online]. Available: <https://www.congress.gov/bill/107th-congress/house-bill/3844>.
- [45] "California Consumer Privacy Act (CCPA)," 28 June 2018. [Online]. Available: <https://oag.ca.gov/privacy/ccpa>. [Accessed 24 February 2023].
- [46] "California Consumer Privacy Rights (Proposition-24)," 01 January 2023. [Online]. Available: <https://oag.ca.gov/privacy/ccpa>.
- [47] "Colorado Privacy Act 2021," 2021. [Online]. Available: <https://leg.colorado.gov/bills/sb21-190>. [Accessed 10 March 2023].
- [48] "Virginia Consumer Data Protection Act," 2021. [Online]. Available: <https://law.lis.virginia.gov/vacode/title59.1/chapter53/>. [Accessed 10 March 2023].
- [49] "Connecticut Data Privacy Act," 10 May 2022. [Online]. Available: <https://portal.ct.gov/AG/Sections/Privacy/The-Connecticut-Data-Privacy-Act#:~:text=The%20CTDPA%20takes%20effect%20on,controllers%20that%20process%20personal%20data>. [Accessed 10 March 2023].
- [50] "Utah Consumer Privacy Act," 24 March 2022. [Online]. Available: <https://le.utah.gov/~2022/bills/static/SB0227.html>. [Accessed 10 March 2023].
- [51] "Charter of Fundamental Rights of the European Union. European Parliament," 18 December 2000. [Online]. Available: https://www.europarl.europa.eu/charter/pdf/text_en.pdf. [Accessed 24 February 2023].
- [52] "European Convention on Human Rights," 3 September 1953. [Online]. Available: https://www.echr.coe.int/Documents/Convention_ENG.pdf. [Accessed 24 February].
- [53] "General Data Protection Regulation (GDPR)," 25 May 2018. [Online]. Available: <https://gdpr.eu/tag/gdpr/>. [Accessed 24 February 2023].
- [54] "Consolidated Version of the Treaty on the Functioning of the European Union.," EUR-Lex., 25 March 1957. [Online]. Available: https://eur-lex.europa.eu/resource.html?uri=cellar:41f89a28-1fc6-4c92-b1c8-03327d1b1ecc.0007.02/DOC_1&format=PDF. [Accessed 24 February 2023].
- [55] "Tele2 Sverige AB v post-Och Telestyrelsen 62015CJ0203 - EN.," 21 December 2016. [Online]. Available: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:62015CJ0203>. [Accessed 24 February 2023].
- [56] "Regulation (EU) 2016/679 of the European Parliament and of the Council, on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC," 27 April 2016. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2016/679/oj>.
- [57] "Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications)," 12 July 2002. [Online]. Available: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32002L0058>.
- [58] "EU-U.S. and Swiss-U.S. Privacy Shield Framework," 12 July 2016. [Online]. Available: <https://www.privacyshield.gov/welcome>. [Accessed 10 March 2023].
- [59] "Asia Pacific Economic Cooperation Privacy Framework," December 2005. [Online]. Available: https://www.apec.org/docs/default-source/Publications/2005/12/APEC-Privacy-Framework/05_ecsg_privacyframewk.pdf. [Accessed 10 March 2023].
- [60] "Digital Personal Data Protection Bill, 2022," 2022. [Online]. Available: <https://www.meity.gov.in/writereaddata/files/The%20Digital%20Personal%20Data%20Protection%20Bill%2C%202022.pdf>.
- [61] "Children's Online Privacy Protection Rule ("COPPA")," FTC, 21 October 1998. [Online]. Available: <https://www.ftc.gov/legal-library/browse/rules/childrens-online-privacy-protection-rule-coppa>. [Accessed 24 February 2023].

- [62] "Convention 108 and Protocols," 15 June 1999. [Online]. Available: <https://www.coe.int/en/web/data-protection/convention108-and-protocol#>. [Accessed 10 March 2023].
- [63] DPI, "About, IEEE Digital Privacy Initiative," 2022. [Online]. Available: <https://digitalprivacy.ieee.org/about>.

8.5. IEEE Digital Privacy Model^{© 159}

As the Chair and a member of the IEEE Digital Privacy Initiative Policy and Legislative Subgroup, I co-created and co-wrote the IEEE Digital Privacy Model. The Digital Privacy Model is highlighted on the IEEE Digital Privacy Initiative website at <https://digitalprivacy.ieee.org/about/digital-privacy-model>, and is further explained in detail in Section 8.6, through my IEEE article, “Bridging the Technologist/Policymaker Divide in Digital Privacy.”

¹⁵⁹ IEEE. “IEEE Digital Privacy Model.” <https://digitalprivacy.ieee.org/about/digital-privacy-model>.

Achieving Privacy Expectations in the Digital World

Technology and privacy professionals need simple and effective ways to represent individuals' expectations of privacy within any environment. The **IEEE Digital Privacy Model** is an evolving easy-to-understand visual representation that covers the broad and dynamic aspects of digital privacy. The model focuses attention on individuals and their expectations of privacy and brings together the factors that influence digital privacy in any physical or virtual environment.



EXPECTATIONS OF PRIVACY

The **IEEE Digital Privacy Model** depicts individuals' expectations of privacy as a set of six characteristics that collectively represent digital privacy for individuals. The data and meta-data that reflect the six characteristics of expectations of privacy are:

Identities: Any identifiers uniquely associated with individuals

Behaviors: Patterns of behaviors by individuals in social or economic settings

Inferences: Inferences attributed to individuals by human or AI/ML algorithms (profiling or abstract representations)

Transactions: Transactions by individuals in any social or economic context within physical or virtual environments

Confidentiality and Integrity: Confidentiality and integrity of individuals' identities, behaviors, inferences, and transactions

Access and Observability: Access to and observability of their identities, behaviors, inferences, and transactions by individuals



INFLUENCES ON PRIVACY

The **IEEE Digital Privacy Model** identifies seven environmental influences on the actions of various actors to achieve digital privacy and manage individuals' expectations of privacy.

Technical influences form the basis to establish technical standards and industry best practices for solutions that support expectations of privacy.

Regulatory influences drive rules for industries, government oversight, and enforcement of privacy regulations.

Economic influences shape organizational and individuals' decisions to achieve appropriate digital privacy outcomes.

Legislative influences help recognize issues and codify laws that affect expectations of privacy.

Legal influences set up legal standards for checks and balances to protect expectations of privacy.

Individual influences create privacy boundaries with individuals as the primary subject of data and metadata in any digital ecosystem.

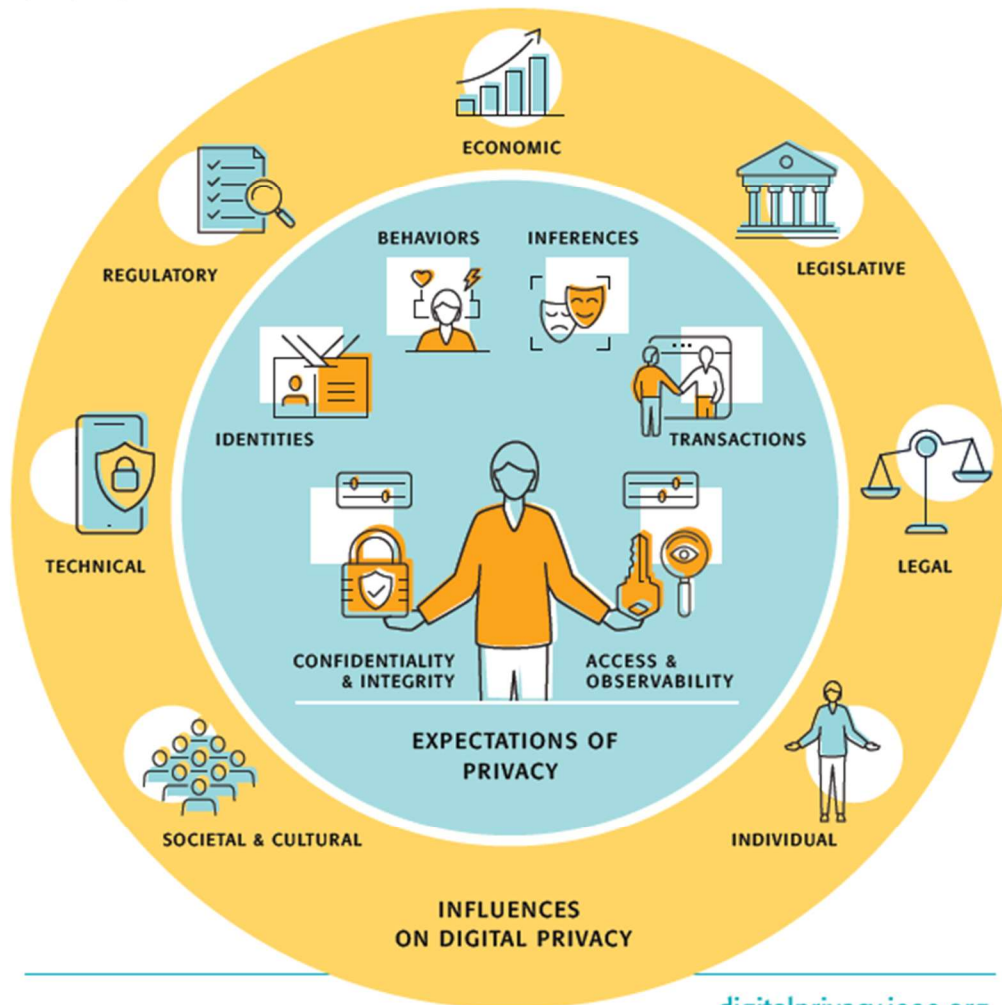
Societal and Cultural influences represent the values that recognize and shape individuals expectations of privacy.



This evolving model makes Technical, Regulatory, Economic, Legislative, Legal, Individual, and Societal & Cultural influences on individuals' expectations of privacy more attainable by using Confidentiality & Integrity, and Access & Observability of individuals' Identities, Behaviors, Inferences, and Transactions in any digital ecosystem.

IEEE DIGITAL PRIVACY MODEL

The **IEEE Digital Privacy Model** is a work in progress that combines the six characteristics of individuals' expectations of privacy—identities, behaviors, inferences, transactions, confidentiality & integrity, access & observability and seven primary influences—technical, regulatory, economic, legislative, legal, individual, and societal & cultural that impact the implementation of a robust digital privacy infrastructure. It is industry- and country-agnostic and recasts digital privacy as expressions of individuals' expectations of privacy. Using the **IEEE Digital Privacy Model**, businesses, government organizations, or other stakeholder entities can systematically map the gaps in their people, process, and technology capabilities to implement solutions that support individuals' expectations of privacy. The model encourages cross-functional exchange of ideas among environmental influences while ensuring a common understanding of individuals' expectations of privacy among all the actors engaged in building a digital privacy ecosystem.



digitalprivacy.ieee.org

© IEEE 2023. All rights reserved.

8.6. Bridging the Technologist/Policymaker Divide in Digital Privacy¹⁶⁰

As our lives become increasingly intertwined in a cyber-physical world, discussions about digital privacy are growing louder and more complicated by the day. The conversation takes place in a variety of public and private organizations and venues globally, but until now it has been mostly framed as a corporate compliance issue. The needs of and interests of individuals have been largely overlooked.

The technologists and policymakers who are engaged in the conversation are normally speaking from very different viewpoints, and with very little shared understanding of one another. That is problematic because the inputs, experiences, and expertise of separate viewpoints are absolutely critical to creating an environment in which the capability exists to enable individuals to privately maintain presence, data, identity, and dignity online.

There are social and public policy implications of creating technology. New code or new inventions can either hurt people or benefit people, depending on how they are conceived, written, and applied. On the other hand, policymakers and engineers are naturally enthusiastic about their own ideas and might not initially consider substantial technical, financial, or social barriers that could complicate their best intentions. The IEEE Digital Privacy Model shows that many ideas and viewpoints must be considered to create truly robust technical designs and social policies. Before making any real progress to identify and preserve the digital privacy of individuals, we need to bridge the gaps and to enable meaningful back-and-forth discussions between “techno geeks,” “policy wonks,” and other stakeholders with something like a *Star Trek* “universal translator” function to allow better communication.

The IEEE Digital Privacy Model illustrates the broad, dynamic, and sometimes conflicting aspects of digital privacy from a high-level visual perspective. Using the IEEE Digital Privacy Model, businesses, government organizations or other stakeholder entities can systematically map the gaps in their people, processes, and technology capabilities to implement solutions that support individuals’ expectations of privacy. The model encourages cross-functional exchange of ideas among environmental influences, while ensuring a common understanding of individuals’ expectations of privacy among all the actors engaged in building a digital privacy ecosystem.

A Broad, High-Level Examination of the Challenge

The IEEE Digital Privacy Model is a comprehensive, 30,000-foot view of the complex influencers on privacy and why is it important to reframe the global digital-privacy discussion from the perspective of individuals.

¹⁶⁰ Lambert, Kent. *Bridging the Technologist Policymaker Divide in Digital Privacy*. Published in IEEE-USA InSight; (accessed Feb 26, 2024).

The model was created by volunteers under the sponsorship of the IEEE Digital Privacy Initiative, which was formed under IEEE Future Directions in January 2022. The initiative focuses on the digital-privacy needs of individuals beyond just the technological issues of the security of data, products, and organizations, and works to incorporate a holistic and cross-disciplinary collaboration to address privacy for the benefit of humanity.

On 13 November 2023, the IEEE Digital Privacy Initiative convened with 25 technology and policy professionals at the U.S. Capitol Building in Washington, D.C., for the *Reframing Privacy in the Digital World* workshop. The event included a diverse roster of scheduled speakers, including Mason C. Clutter, Chief Privacy Officer, U.S. Department of Homeland Security; Chris Irwin, Program Manager, Office of Electricity, U.S. Department of Energy; Nick Napp, co-founder, Xmark Lab; and Sreedhar Rao, Field CTO Telecommunications, Snowflake.

The IEEE Digital Privacy Model Goes to Washington

The Washington, D.C., workshop served to socialize the IEEE Digital Privacy Model firsthand among professionals with significant influence on how decisions about digital privacy will be made in the years to come in both the United States and other parts of the world.

The IEEE Digital Privacy Model provides a common starting point to drive privacy solutions that are predicated on individuals. This approach is unique. Until this point, discussions among either policy or technical proponents of digital privacy have rarely been focused on needs and expectations of individuals.

The model depicts six characteristics of individuals that together represent digital privacy:

- Identities
- Behaviors
- Inferences
- Transactions
- Confidentiality and integrity
- Access and observability

The model also identifies seven influences on the actions of various actors to achieve digital privacy and manage individuals' expectations of privacy:

- Technical
- Regulatory

- Economic
- Legislative
- Legal
- Individual
- Societal and cultural

The goal is to show the synergistic effects of all of the digital privacy expectations and influences that need to work in concert for the benefit of individuals. Because all of the defined areas must be considered as participants, the IEEE Digital Privacy Model does not narrowly focus on any single aspect, or attempt to solve any particular problem within the digital privacy sphere. Furthermore, the IEEE Digital Privacy Initiative volunteers who wrote the model conscientiously tried to make it very broad and applicable across all global regions and industries.

Next Steps Toward Achieving Concrete Progress

The loss of privacy is often rooted in technical breaches where personally identifiable information (PII) or other sensitive information such as industrial data is gathered illegally or unethically and used for something other than its intended purposes. It is very difficult to safeguard digital privacy unless features and functionality have been predesigned into systems across a generally unsecured architecture like the internet.

One thing that the IEEE Digital Privacy Model does *not* attempt to do is explore, study or discuss the technological solutions to problems that have occurred in digital privacy breaches globally. Indeed, while there may be many different methods to address those issues, this model is intentionally not meant to be prescriptive.

Rather, the important next step is for both technology- and policy-oriented readers of the model to consider its implications that are the most important and relevant within their contexts. For example, what might be the most likely obstructions to achieve the expectations of privacy in a given environment? Which of the model's documented influences most prominently participate in that environment?

The IEEE Digital Privacy Initiative is undertaking surveys of international subject-matter experts about what people in different regions realistically expect in terms of privacy. It is already clear that there are widely divergent public policies about what should and what should not be private. IEEE Digital Privacy Initiative volunteers are also continuing to solicit feedback on the model itself and planning to develop next-level details to help companies operationalize digital privacy for individuals. As the first line of the document states, "The IEEE Digital Privacy Model is a work in progress ..."

Digital privacy is a topic of increasing scrutiny across global political, geographic, and technological boundaries. The IEEE Digital Privacy Model has been designed to fuel a more informed, individual-centric conversation. The model is an intentionally generic, high-level, easy-to-understand visual representation that is easily downloadable in several languages and portable for any venue. To learn more about how you can engage in the work of the IEEE Digital Privacy Initiative and/or to keep apprised of future events such as the *Reframing Privacy in the Digital World* workshop, please visit <https://digitalprivacy.ieee.org>.

8.7. Briefing to the Colorado County Clerks and Records Association and Colorado Department of State

On January 24, 2024, I presented a briefing to the Colorado County Clerks Association and representatives of the Colorado Department of State on applications of distributed ledger technologies at the request of the Association, and the University of Colorado Colorado Springs. Due to its size and redundancy with much of the material in this dissertation, the briefing slides will not be included here.

Blockchain Technology

**Making Distributed Ledger A Reality In Colorado
(Edited from a Presentation with Colorado OIT)**

Presenter

**Kent Lambert
BlockFrame®, Inc**

8.8. IEEE Region 5 Training

On April 17, 2024, I presented a live lecture to the IEEE Region V training seminar, “A Call-to-Action Uniting Tech Innovators, Policy Leaders, and Businesses to Safeguard Digital Privacy,” at their request. Due to its format, size, and redundancy with much of the material in this dissertation, the briefing slides will not be included here. The lecture was published on IEEE.TV, and can be accessed and downloaded at: <https://ieeetv.ieee.org/video/a-call-to-action-uniting-tech-innovators-policy-leaders-businesses-to-safeguard-digital-privacy>.

8.9. Participation to revise the IEEE-USA Digital Privacy Policy Statement

In early 2024, IEEE-USA began a review process to replace their Digital Privacy Policy Statement, which was due to expire in November 2024. At the request of the IEEE Digital Privacy Initiative, and as Chair of that Policy and Legislative Subcommittee, I was invited to work on the team to review and update that policy statement and have co-written inputs for the new document. This is an ongoing effort.

This Section 8 in total addresses my practical research conducted with professional and public policy organizations and satisfies the goals of Research Question 4 (RQ4).

9.0. Representative Practical Use Cases

This Chapter addresses and satisfies Research Question 5 (RQ5) by defining and assessing examples of practical applications and use cases. Restated, Research Question 5 (RQ5) is, “What are examples of practical applications of this research for selected industrial use case examples?” This enables applied and translational research on the effectiveness and impacts of BlockFrame® technologies to practical use cases.

The BlockFrame® core system of systems can be considered as a set of reference models that can be adapted and applied to selected industrial use cases, using either individual systems, or using the entire system of systems collectively.

BlockFrame®’s corporate selections of the most promising initial industrial verticals for production and application are education, distributed transactive energy, medical devices, and cybersecurity/privacy. Many other use cases and applications have been developed in parallel to these corporate priorities, including digital supply chain security, digital privacy, financial technologies, and data center operations and security.

Financial analysis was applied when necessary to research cost comparisons, financial risk management, resilience, analysis of capitalization, grants, investor marketing, return on investment, and lessons learned. Various forms of quantitative business analyses have been used to calculate performance and non-performance system metrics for RQ5 that are included in other chapters and appendices. Some other market research contained elements that are considered proprietary and will not be included here. A major consideration for prioritizing research, development, and production is the expected marketability for different industrial verticals. Figure 9-1 shows three estimates of the Total Addressable Market (TAM)¹⁶¹ and the Serviceable Obtainable Market (SOM) of three verticals of high interest: education systems, IoT systems, and secure supply chain systems.¹⁶² While these priorities for pursuing use case applications, changing demands, and available funding will change in the future, these initial assessments provide information for definition of marketing, investments, and feasible rates of corporate growth.

¹⁶¹ TAM (Total Addressable Market) is the total revenue opportunity available for a product or service if a company has 100% market share. SAM (Serviceable Available Market) is the portion of TAM that a company can realistically target and serve, while SOM (Serviceable Obtainable Market) is the specific share of SAM that a company can capture in the near term. Source of definitions: seerinteractive.com 
thebusinessplanshop.com.

¹⁶² C. Gorog, et al, BlockFrame Investor Deck, Unpublished. June 4, 2025.

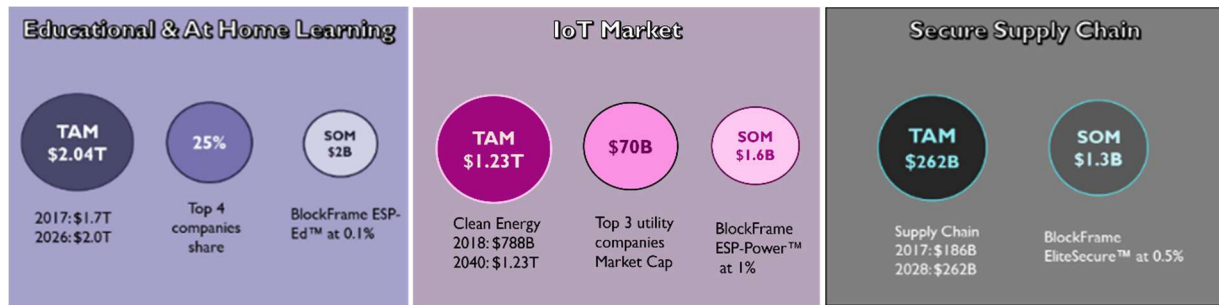


Figure 9.1. Market Opportunities

BlockFrame® technologies are designed to operate globally across different industries as a system of systems using common components within each system. Individual systems within the BlockFrame® suite of systems may also operate independently with commercial systems that do not require the full system-of-systems suite. Since the BlockFrame® architecture has those characteristics and flexibility, it should also be viewed as a reference architecture that can be applied to numerous industrial verticals without significant changes to its core components. Depending on the specific use cases, any unique interfaces, required capabilities, and/or any unique software required by users will require customization.

State Level Use Cases Identified by BlockFrame® and the Colorado Office of Information Technology (OIT)

Since the passage of Colorado Senate Bill SB18-086 in 2018 [28], the State of Colorado identified more than 75 state level distributed ledger/blockchain use cases, primarily for the security and safety of state records, data privacy, data security, nonrepudiation, accountability, and cross jurisdictional data. Table 9.1 shows a partial list of suggested use cases for application by state departments. Once applied in one state, it is likely that similar use cases will proliferate to other states and to similar federal programs.

The examples in the next sections explain three other use cases in more detail: distributed energy resources, military supply chains, and remote educational trust and security. These three selected examples show the breadth of applications that are possible with the BlockFrame® systems of systems.

Table 9.1. Proposed Colorado State Government Use Cases for Blockchains/Distributed Ledgers

State Department	Proposed Use Cases
Agriculture	Brand board/cattle ownership and transfer; market logistics; organic certifications; food tracking
Corrections	Sentencing/Probation/eligibility records; health and prescription drug records
Education	Longitudinal student record security; grade reports
HCPF	Electronic Visit Verification (EVV) records; prescription drug certifications and records (opioids); Specialized/individualized food certification (allergies, diabetic diets, like Angel Heart)
Higher Education	New demand for cybersecurity education/training; secure academic degrees/grades, student data; new R&D; teach public policy, NIST security standards
Human Services	Welfare payment/SNAP accountability; CBMS security; Child welfare/County Admin financing; mental health/medical/prescription drug records
Judiciary	eDiscovery; Court, sentencing, and probation records; Jury records; Forensic chains of custody
OIT	Hardware/software inventory and configurations; CISO Secure Colorado to mitigate cyber threats; structured project management and IT contracting
OEDIT	New economic developments in cyber technologies; secure interstate/international trade
Personnel/Administration	Cyber Insurance (to reduce civil liability risk); Transmittal of tax records (for DOR); Employee personnel records; Controller records (CORE); Address protection system; Digital Rights Management
Public Health	Vital Statistics; medical records; food safety logistics tracking; medical device protection
Public Safety	CBI/CIAC: Protected investigation information; witness protection programs
Regulatory Agencies	Business, real estate, and securities ownership records “Stock Ledgers”; Professional licenses and credentials
Revenue	DMV ownership/transfers; signature security; marijuana identification/enforcement
State	Voter signature data/registration/results databases and transfers; remote notarization; business registration information; Apostilles; Go Code app developments
Utilities	Critical infrastructure security (IOT); Energy safety controls; rail, traffic, autonomous vehicles

9.1. Case 1. Distributed Energy Resources

BlockFrame®’s initial development priorities were for educational applications, transactive energy security and management, medical devices, digital supply chain cybersecurity, and digital privacy. However, as of this writing transactive energy management emerged as the most urgent use case of interest. The white paper, *Carbon-Neutral Distributed Ledger*¹⁶³, attracted attention not just to the technical mechanisms of secure data collection from digital transactions managed by blockchains, but also economic savings from eliminating expensive proof-of-stake mining. BlockFrame® has also used several other design principles to eliminate the limitations of other blockchains, especially the inability to scale to serve large systems. With common cyber-physical components like the ESP™ that can be proliferated throughout complex systems, and cryptographically secure components like the BlockFrame® system container, system monitoring and security components can be either designed into new systems or can interface with legacy systems until they can be more conveniently upgraded. As one practical example, a demonstration to measure and record energy transactions with an energy co-op was partially funded and research began in the Summer of 2022. Philos™ uses a “green” design with no energy mining required as a “proof of stake” for the distributed ledger. An IEEE grant was awarded to BlockFrame® to begin demonstrations with a Colorado energy cooperative on practical transactive energy use case applications of these technologies. BlockFrame® applied some of that funding to the study of transactive energy with a Colorado energy distributor. For that purpose, BlockFrame® also gained agreements with the US Department of Energy, a Colorado-based power distribution company, and two national energy laboratories, NREL and PNNL, to conduct research and testing with new products to securely collect digital data, facilitate digital transaction contracts, and demonstrate proof of transactions for use by power utilities and the calculation of carbon credits. Heretofore, energy distributors have not been able to monitor and accurately record transactions to either bill members, or to calculate carbon credits accurately. Following this demonstration, the technology can be operationalized and expanded to a broader scope within and among energy distribution systems. Other use cases like education, medical devices, and cybersecurity/privacy applications will be initiated as resources become available. A combination of grants, investments, and direct contracts for services are actively being pursued to help proliferate those efforts. Figure 9-2 compares the characteristics of the low carbon “green” Philos™ blockchain/distributed ledger design to other blockchain designs. Economic benefits and cost savings using a carbon-

¹⁶³ Gorog, Christopher, Pam Russell, and Terrance E. Boulton. *Carbon Neutral Distributed Ledger*. In *IEEE Innovative Smart Grid Technologies (ISGT)*. New Orleans, LA: IEEE Pre-Release, 2022.

neutral blockchain design were compared to electrical energy mining alternatives that are addressed in the white paper, *Carbon-Neutral Distributed Ledger*¹⁶⁴. Although that research has been completed, other economic benefits will be sought and documented in the future. The potential for expansion of the use of these BlockFrame products is currently part of ongoing discussions with IEEE and with Michael Pesin, the Deputy Assistant Secretary of Advanced Grid Research and Development of the US Department of Energy (DoE) Office of Electricity Delivery and Energy Reliability, and several others. As of this writing, contractual Statements of Work are being reviewed by DoE that include the use of Philos™ and the BlockFrame® Industrial Utility Registrar, which include large-scale stress and speed tests.

For energy regulators and companies, two primary advantages of a “green” carbon-neutral blockchain are to eliminate unnecessary energy consumption, and techniques to manage secure systems to accurately and securely calculate costs for rate payers. Among some users, the calculation and use of carbon tax credits is also considered desirable.

Generations of Blockchain and Capabilities	First Generation			Second Generation			Third Generation
	Bitcoin	Crypto-currency	Stable Coins	Ethereum	Hyper-ledger	Stellar	Philos
No 3rd Party needed for Transaction	x	x	x	x	x	x	x
Content Publicly Verifiable	x	x	x	x	x	x	x
Volatile Speculation	x	x		x			
Publicly Accessible	x	x	x	x			x
Cost Effective			x			x	x
Operation of Smart Contracts	Smart Operations & Application Enhancement			x	x	x	x
Control and Vetting Participants					x	x	x
Minimize Data Usage						x	x
Expandable To Large Scale					x	x	x
Separation of Multiple Industries	Governance, Cross Industry Support, & Organizational Management Controls						x
Tracing Trust of Individuals							x
Separation of Private Data							x
Control of Multiple Jurisdictions							x
Licensing of Transaction Contracts							x
Economic Indicator Controls							x
Indefinite Modular Scalability							x
Zero-Trust Ledgers							x
No Wasted Energy Consensus							x

Figure 9-2. Generations of Blockchain Capabilities¹⁶⁵

¹⁶⁴ Gorog, Christopher, Pam Russell, and Terrance E. Boulton. *Carbon Neutral Distributed Ledger*. In *IEEE Innovative Smart Grid Technologies (ISGT)*. New Orleans, LA: IEEE Pre-Release, 2022.

¹⁶⁵ Gorog, Christopher, and K. Lambert. *BlockFrame, Inc., Technology -- Use Cases for Blockchain Technologies*. PowerPoint. Unpublished. July 2021.

9.2. Case 2. Military Supply Chain Use Case

In 2024, BlockFrame® and Colorado State University collaborated to research an approach to satisfy Lockheed Martin Space Division (LMCO) requirements to improve supply chain transactions by applying and demonstrating capabilities from the BlockFrame® technology suite as a reference architecture to LMCO's industrial supply chain process.¹⁶⁶ BlockFrame® considers the use of cryptography and the secure delivery of cryptographic keys to support digital supply chains as a core corporate competency. In this case, those principles were extended to consider physical supply chains as well.

Current LMCO supply chain

LMCO provided a generic representation of their supply chain process to describe typical transactions that might be used for a major defense contractor. That process is shown in Figure 9.3. The key to abbreviations is contained in Figure 9.4.

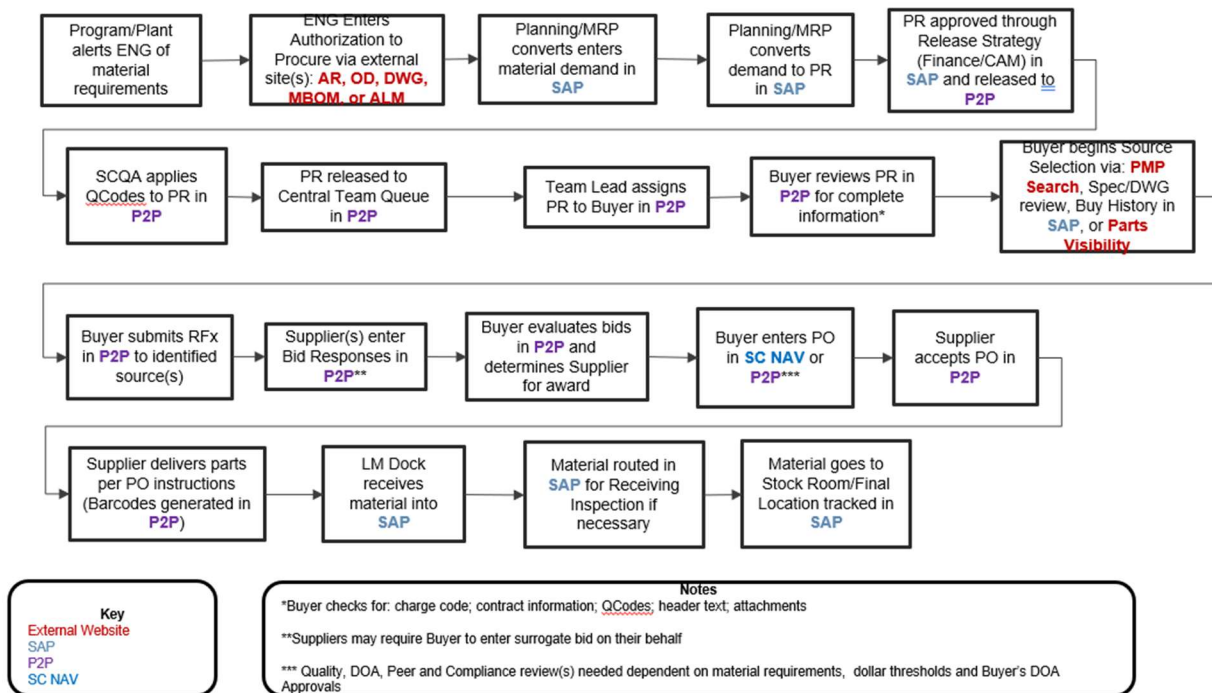


Figure 9.3. Representation of a Typical Major Military Contractor Supply Chain.

¹⁶⁶ The study team was comprised of BlockFrame® Principal Investigator Kent Lambert, CSU Principal Investigator Dr Thomas Bradley, and CSU doctoral student Ethan Muhs. Dr Christopher Gorog and Dr Steve Simske provided project support.

- ALM – Advanced List of Materials (released in EPDM)
 - AR – Acquisition Request
 - CAM - Cost Account Manager
 - DOA – Delegation of Authority
 - DWG – Drawing (released in EPDM)
 - ENG – Engineering
 - EPDM - Enterprise Product Data Management
 - MBOM – Material Bill of Materials (Parts List) (released in EPDM)
 - MRP - Material Requirements Planning
 - OD – Operations Directive
 - P2P – Procure to Pay
 - PO – Purchase Order
 - PR – Purchase Request
 - QA – Quality Assurance
 - QCodes – Quality Codes
 - RFx – Request for (x): proposal, quote, information, etc.
 - SC NAV – Supply Chain Navigator
 - SCQA – Supply Chain Quality Assurance
- PMP – project management professional?
- SAP – company

Figure 9-4. Supply Chain Terms and Definitions

Mapping the Supply Chain

The first step considered was to apply BlockFrame[®]'s uniformity principles to the LMCO-supplied supply chain description. Figure 9.5 shows a typical BlockFrame[®] description of an enterprise marketplace but using labels of specific potential transaction points from the LMCO example, along with specific insertion points where BlockFrame[®] technologies could be effectively applied to provide various functions such as recording secure transactions or secure storage.¹⁶⁷ When appropriate, these transactions can be securely recorded on the Philos[™] distributed ledger with appropriate metadata, or in some cases data could be securely stored offline.

¹⁶⁷ For convenience in discussions, this was nicknamed the “horseshoe diagram”.

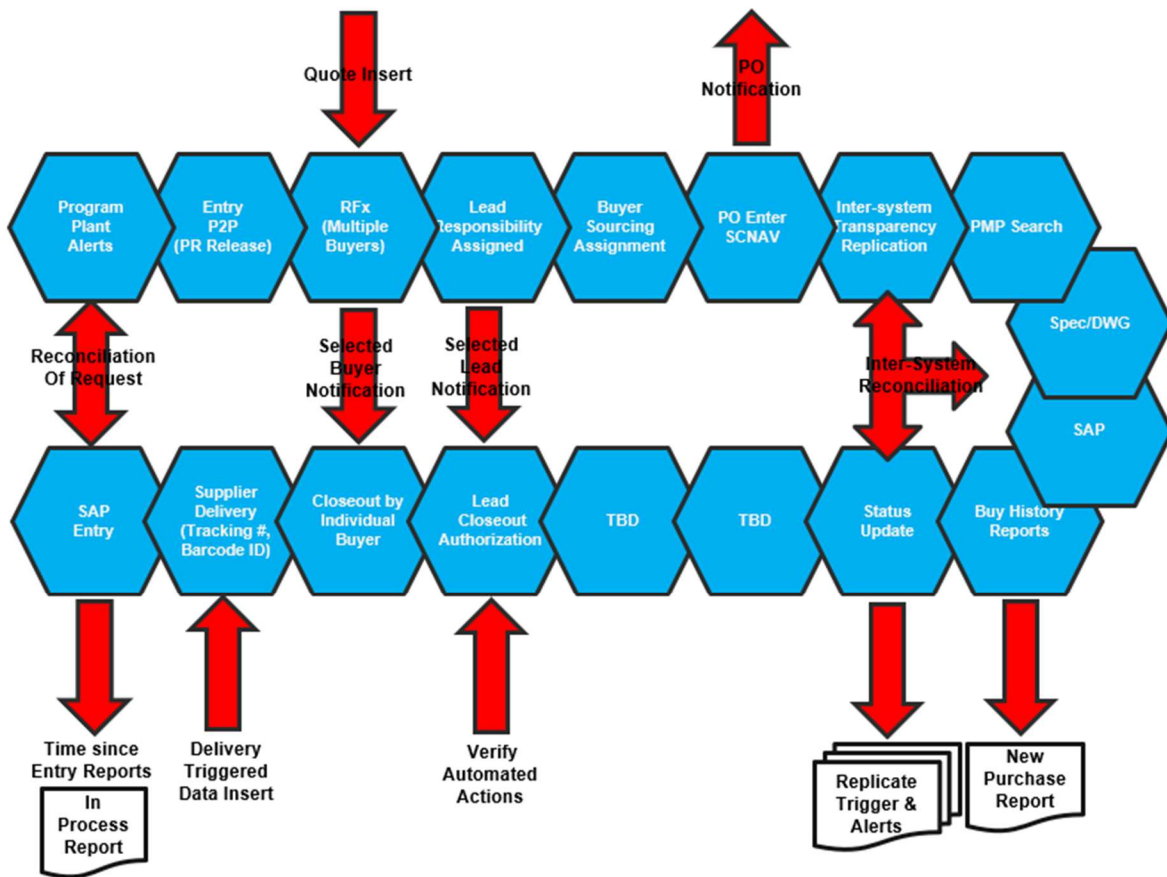


Figure 9.5. BlockFrame Supply Chain “Horseshoe” map

Research survey questions by supply chain action and researcher

From previous research, experience, and commercial applications, BlockFrame® developed a questionnaire to collect specific descriptive data about each step of the supply chain process.¹⁶⁸ These descriptive data help enforce a uniform view of the systems being described – in this case a military supply chain – but also a structured data model that not only describes each of the nodes in system behaviors, but also the system with more consistency to allow better governance of the system as a whole. Figure 9.6 shows the questions posed by the questionnaire to populate the descriptive data about each step in the supply chain which is used to form the data model for the whole system.

¹⁶⁸ T. Rossi and Dr. C. Gorog. *System Details Gathering Questionnaire Guidelines for Block Chain Governance Applications*. BlockFrame, Inc. 2004.

Estimated Number of Users Performing action	Year over Year growth Rate of Users	Estimate users to meet TAM	Number of Locations for Users Base	Attestation of Users Required	Intifcation of Users, or user system required	Peer to peer user negotiations from individual accounts	Volume of Transactions per/time	Historical Growth rate of Transactions	Average Number Of Transactions Per User	Estimated Transactions meeting TAM	Transaction Validation of Verification	Public Transaction display required
Number	Percentage	Number/Percent	Number	Yes/No	Number	Yes/No	Number	Percentage	Number	Number/Percent	Yes/No	Yes/No

Current Number of distribute Locations	Historical Growth Rate of Distributed Locations	Geographical diversity of distributed	Estimated Locations to Meet TAM	Current Number of Servers, Service points, cloud	Needs for Servers, Service points, cloud location for TAM	Attestation of Location, platform, Identification, or data Point of Origin Required	Number of Distributed Organizations performing Operation	List of All Participating Organizations	Number of Organization for TAM	Attestation of Organization, platform, Identification, or data Point of Origin
Number	Percentage	Text	Number/Percent	Number	Number/Percent	Each Required Analysis (text)	Number	List (Text)	Number/Percent	Yes/No

Peer to peer negotiate Transactions required	Public Data Items required for transactions	Analysis of public requirements for Industry, User, Org, Status, Category, NAICS, Loc, etc.	Private Protected Data Items for operations	Privacy Analysis for Each Protected Data Item (Allowed Access, Required Notification, PII Set, Sensitivity Level, Industry Designation, Compliance Requirements, Handling, visibility an access	Analysis of Smart Contract, Enforcement operations, sequences, relationships, account values, Status, etc.	Revenue Model Service base BDC (POS or Tokenized)	Prepaid Service Sales or Consumption as a service
Yes/No	Yes/No	Text	Yes/No	Text	Text	Text	Text

Revenue model P2P (tokenized, wallet, third party POS, Contractual, etc)	Distributed Application Analysis, Handheld Device, Mobile App, Service, Web	LM Most Valuable Analysis Items	Risk Level	SWOT Analysis
Text	Text	Low/Med/High/Crit	Low/Med/High/Crit	Text

Figure 9.6. Questionnaire for Blockchain Governance Applications

Integration with CONOPS and Data: Mapping the LMCO Supply Chain

As it was not possible or necessary to analyze the entire LMCO supply chain within existing resources, discussions with LMCO discovered that the third through the eighth steps in their supply chain process were of highest importance to them. Therefore, research was limited to those steps, questionnaire data were collected for those steps, and an abbreviated model of the supply chain was constructed. In some cases, there appear to be enough data to determine which supply chain steps could benefit from BlockFrame® and Philos™ distributed ledger technologies, and where they might not be necessary. Further research could determine more specific recommendations, but from the limited study conducted to date, the team observed that significant efficiencies and cost savings would be likely if development continued. While this research primarily considered organizational efficiency and cost reductions above cybersecurity requirements, the inherent BlockFrame® capabilities to improve information integrity and non-repudiation can also help guard against some malicious cyber penetration threats. Those cybersecurity capabilities could be expanded in the future. Figure 9.7 shows the scope of the research to date.

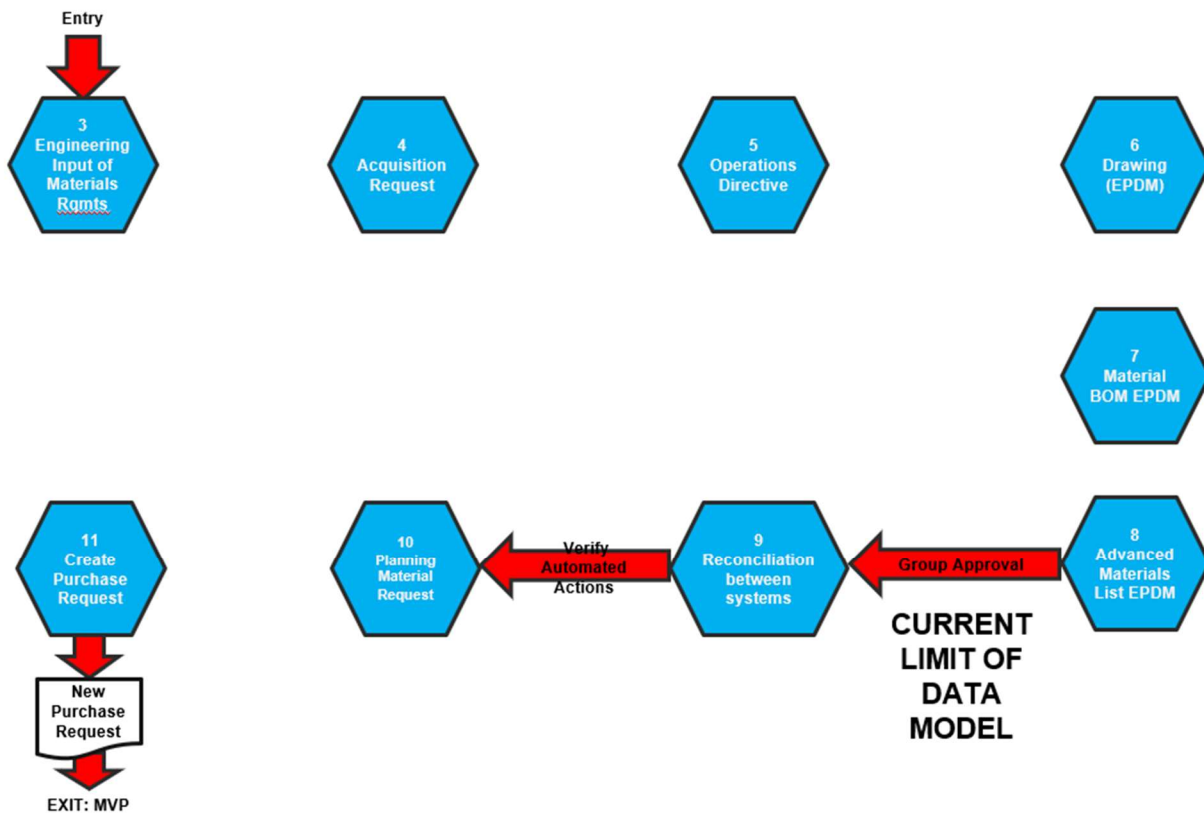


Figure 9.7. Scope of Research

GS1 Supply Chain Data Communication Standard: Electronic Product Code Information Services (EPCIS)¹⁶⁹

As an important part of the LMCO research, Dr Simske and Ethan Muhs provided important insights into applying information standards from the Electronic Product Code Information Services (EPCIS) used as part of the GS1 Supply Chain Data Communication Standard¹⁷⁰. As described in Figure 9.8, the EPCIS standard already exists, and can add essential information about the structure of the supply chain, its behaviors, and the products being handled. In concert with BlockFrame uniformity, security, and scalability, all of these capabilities if integrated could significantly improve the management of current supply chains systems.

Although the immediate goal of this research was to examine and provide insight for the LMCO customer, the process is an example of methods that can be applied to many generic supply chains to improve security and reduce cost.

¹⁶⁹ Source: GS1 US®. *Applying Standards for Supply Chain Visibility in Blockchain Applications*, Release 1.1. January 30, 2020.

¹⁷⁰ GS-1. EPCIS & CBV. <https://www.gs1.org/standards/epcis>.

Sharing Physical Event Data With EPCIS

Movements or "events" comprise four dimensions

1 What products are impacted?	Global Trade Item Number* (GTIN*) (urn:epc:id:sgtin:0614141.000001.2)
2 When did this time-stamped event occur?	Date and Time Stamp (2017-10-02 10:00:00)
3 Where was the product and where is it now?	Global Location Number (GLN) (urn:epc:id:sgin:0614141.00300.0)
4 Why was this observed and at which step?	Business Step (urn:epcglobal:cbv:bizstep:receiving)

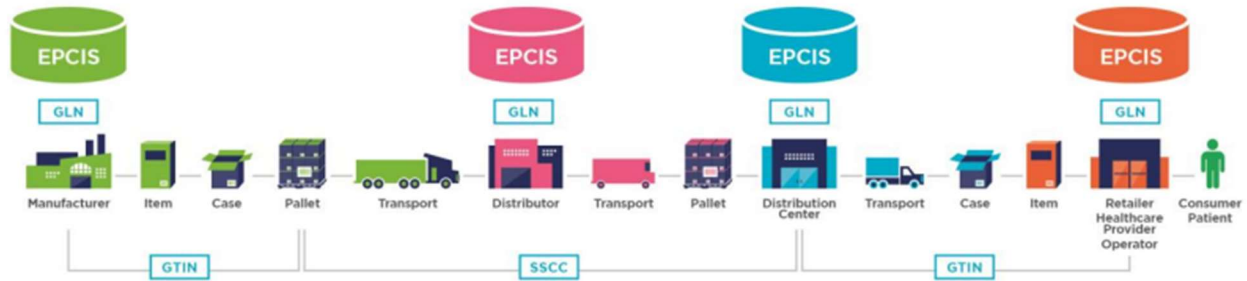


Figure 9.8. GS1 Supply Chain Data Communication Standard

9.3. Case 3. Remote Education Trust and Security

BlockFrame, Inc, RANDA Corporation, IDRamp, Fluree, Velocity, Evernym, and UPD Consulting entered a joint pursuit of grant funding under the American Council of Education (ACE) Blockchain Innovation Challenge (BIC) in cooperation with the US Department of Education and were notified of a successful award of the grant in January 2021. The BIC grant award was intended to provide funding to support a multistate Lifelong Learner Project, to include integration work for an education credential tracking platform which was intended to be used by the Colorado Department of Education. The states of North Dakota, Utah, and Ohio also contributed to the project.¹⁷¹

Although the BlockFrame® Philos B/DL™ was the primary blockchain contributor to the Blockchain Innovation Challenge (BIC), no funding was extended to BlockFrame® for actual blockchain development for the project. However, the ACE/BIC experience was useful for BlockFrame®'s experience and understanding of remote secure distance learning. As this occurred during the height of the COVID epidemic, the interest for improved capabilities for secure, reliable, and trusted platforms became a growing requirement for educators. Even after the stabilization of the epidemic, the same capabilities appear to maintain excellent potential to support remote learning, testing, improved security and trust, and greatly reduced costs. Many states and schools, in both urban and rural areas, are actively pursuing new and more flexible options to access digital learning opportunities across local, state, or international boundaries to improve their quality of education at reasonable costs.

The BlockFrame® Eco-Secure Provisioning System for Education (ESP-Ed™) is a use case under development to meet all those requirements. Figure 9.9 shows the relationships of the essential functional components. Using existing equipment that is available or easily accessible to almost any household – a TV/monitor, a keyboard, and a mouse – BlockFrame® would add an HDMI link, a camera, an ESP-Ed™ integrated circuit, and a WIFI integration device. Device and smart data handling and transaction costs through the ESP-Ed™ would be far less than in-person proctored testing. Because each ESP-Ed™ device has its own unique identifiers, educational content curated by specific instructors to specific students are both highly secure and highly trusted. Data collected from biometrics can securely prove the identity of the student without being highly intrusive or compromising privacy. All data collected would be encrypted, stored under the management of the blockchain, and delivered securely to the instructors

¹⁷¹ C. Gorog. Draft BlockFrame SOW to Support the Lifelong Learner Project in support of American Council on Education Blockchain Innovation Challenge.

or proctoring officials. While some internal details of this process are proprietary to protect the integrity of the process and the quality of trust within the system, the ESP-Ed™ provides an inexpensive and trustworthy improvement in the quality of remote learning.

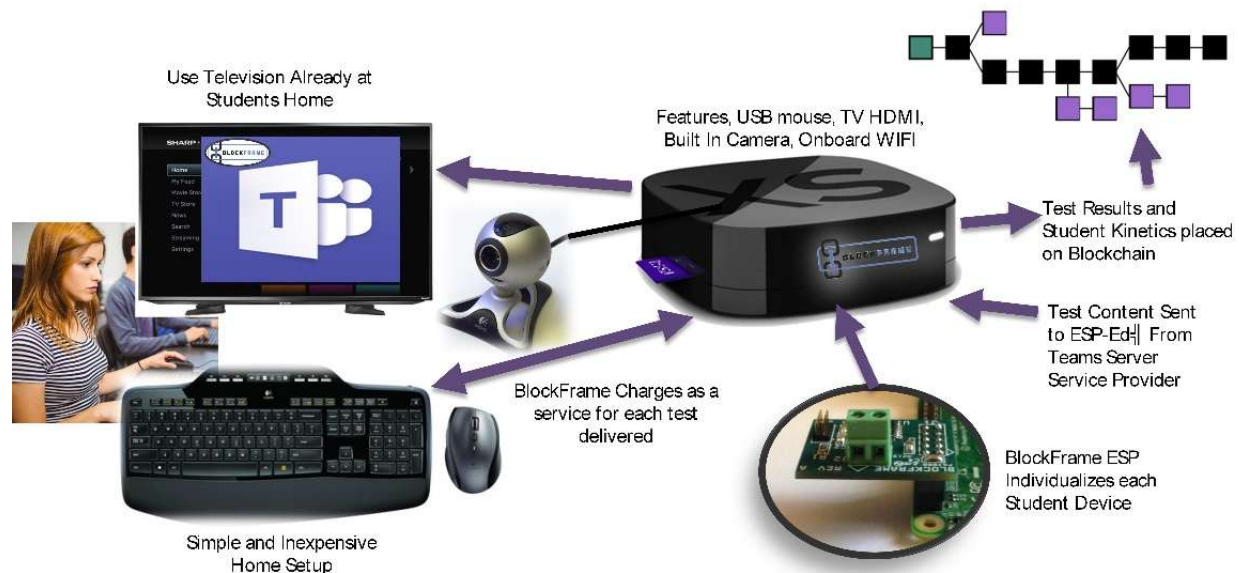


Figure 9.9. BlockFrame® Concept for ESP-Ed™ System for Secure Remote Learning
The relative simplicity of the ESP-Ed™ and the potential for a revolutionary improvement in remote learning

is an important example of how the flexibility of the BlockFrame® suite of systems can be adapted to a relatively small-scale industry with a broad scope of quality improvement.

10.0. Research Contributions

10.1. Systematic Research Process. This research has used an applied and translational approach to develop a deeper understanding of the state of systems engineering knowledge and applications to leading-edge industrial and professional practices. The Colorado State University Systems Engineering Department Doctor of Engineering seeks to prepare candidates for innovation, impact on industry, and the practice of systems engineering. Its practicum and dissertation process seeks to connect modern systems engineering principles, business practices, and practical engineering experience to observe the operations of advanced industries and a broad range of stakeholders and professional organizations. This dissertation therefore derived its research contributions from those synergies to develop a set of contributions to the field of systems engineering and industrial practices. Emphasis was placed on the translation of Emergence Theory to cybersecurity and distributed ledger-based digital supply chain management. Systems Modeling Language (SysML), Model-Based Systems Engineering (MBSE), and Model-Based Systems Architecture Process (MBSAP) were used extensively to study the BlockFrame[®] suite of technologies and behaviors across many potential applications, use cases, and industrial verticals.

10.2. New Contributions and Results from this Research. Based on the research questions and tasks, the contributions of this research include:

10.2.1. This research has applied Emergence Theory to digital cybersecurity environments, thereby defining, modeling, and demonstrating security as a product of positive and negative emergence in this application.

10.2.2. This research has created a Monterey Phoenix/Gryphon standardized and documented Schema and reference model (in collaboration with the US Naval Postgraduate School) to model emergent behaviors of cyber-insecure and secure systems for educational and research purposes. This contribution was published as *Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments*, in the book, *Emergent Behavior in System of Systems Engineering*, which compares the contributions of different authors to exemplify many model-based examples of Emergence Theory¹⁷².

¹⁷² Lambert, Kent. *Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments*. In *Emergent Behavior in System of Systems Engineering*. Pre-Release. CRC Press, 2022.

10.2.3. This research has modeled and characterized the behaviors of novel industrial development of blockchain technologies from BlockFrame, Inc., to advance digital supply chain security products, to advance control management and security actions, and to address industrial and national needs in the areas of cybersecurity, transactive energy management, digital privacy, and other use cases. Systems Engineering-relevant models and behaviors captured the main functions of the BlockFrame® system of systems, at the Operational and Logical/Functional Levels.

10.2.4. In terms of applied and translational research activities, this work applied SysML, MBSE, and MBSAP to create models and reference architectures for selected high priority use cases and applications. These include several baseline examples of distributed energy resources, and several energy-related components that were researched for the US Department of Energy, PNNL, and NREL. Supply chain processes were analyzed with Lockheed-Martin Space.

10.2.5. In terms of applied and translational research activities, this work has developed education and training videos and products for users and students of BlockFrame® applications using examples of supply chain security and blockchain management systems. Live video training for IEEE Region 5 on, *A Call-to-Action Uniting Tech Innovators, Policy Leaders, and Businesses to Safeguard Digital Privacy*, was conducted as a direct result of innovations in this dissertation.

10.2.6. Researched the translational capabilities of the BlockFrame® Enterprise to commercial business processes to demonstrate and manage the provision of a “green” carbon-neutral, cryptographically secure and scalable blockchain to commercialize effective and accurate transactive energy capabilities, reduce energy consumption and costs, allow carbon credit calculations, and to commoditize cybersecurity capabilities.

10.2.7. In terms of the applied and translational research activities, this work has contributed to and led public digital privacy policies and cybersecurity standards, defining best practices, and fostering mutual information exchanges in conjunction with new organizations in the IEEE Digital Privacy Initiative, the IEEE Standards Society, and the NCSL Cybersecurity Task Force/Privacy Working Group with the intent to translate and bridge technology into informed public policy and useful legislation.

10.2.8. The author has moderated an international IEEE Future Directions Digital Privacy Workshop on “Policies and Frameworks around the World,” which compared digital privacy approaches

from the United States (NIST), the European Union General Data Protection Regulation (GDPR), and the People's Republic of China Personal Information Protection Law (PIPL)¹⁷³.

10.2.9. In terms of the applied and translational research activities, the author directly co-created the IEEE Digital Privacy Initiative Digital Privacy Model¹⁷⁴ and co-authored published articles on digital privacy.

¹⁷³ IEEE Future Directions. Digital Privacy Workshop. "Policies and Frameworks around the World." Oct 7, 2021. Accessed July 22, 2022. <https://cmte.ieee.org/futuredirections/events/digital-privacy-workshop/>.

¹⁷⁴ IEEE Digital Privacy Model. IEEE Digital Privacy Initiative. <https://digitalprivacy.ieee.org/about/digital-privacy-model>.

11.0. Conclusions

The Doctor of Engineering in Systems Engineering is an opportunity to observe advanced, complex, industrial systems, to research architectures, behaviors, and capabilities, and to articulate their real-world and commercial applications through a robust academic process. The origin of the idea behind the Doctor of Engineering process might be attributed to the Greek term *πρακτικός* (practicos), from which the English terms practicum, practical, and practice are all derived, and the verb *πράσσω* (prasso), meaning “to do.” A common definition of “translational” in scientific fields is, “*of or relating to the transfer of scientific knowledge into practical applications*”¹⁷⁵. One aspirational goal of the Doctor of Engineering should be to “reverse engineer” that flow of information by examining practical industrial applications and translating those concepts to expand scientific knowledge. This doctoral practicum and dissertation observed new advanced technologies and processes being developed by the BlockFrame® Corporation and many other partners in the areas of emergence theory, digital supply chain management, cybersecurity, and digital privacy. Those capabilities were examined using the Systems Modeling Language Framework, Model-Based Systems Engineering, and the Model-Based Systems Architecture Process in mind to satisfy industry requirements, interact with subject matter experts from industry, academia, government, and professional organizations, and design new functional capabilities to meet those requirements. Articles, white papers, models, projects, and this dissertation help articulate actionable systems engineering principles that have been used to research, document, and advance the state of the art and knowledge in these areas. This research program followed a set of pre-selected research questions and tasks which were fulfilled in this dissertation and advanced of the practice of systems engineering by applying systems engineering to distributed-ledger-based cybersecurity management. Starting with an understanding that negative emergence in cybersecurity can be improved through better uniformity in system architectures and behaviors, there are almost innumerable opportunities for future research and applications in a vast number of academic and industrial applications.

Many use cases and applications of BlockFrame® technologies have already been listed and can proceed as resources allow. Using the basic structures as reference architectures from this dissertation, near-term efforts are already transitioning from logical/functional levels of abstraction to commercial physical products and processes, especially for distributed energy systems, data security for data center transactions, and capabilities to expand to

¹⁷⁵ “Translational — definition, examples, related words and more at Wordnik,” *Wordnik.com*. <https://www.wordnik.com/words/translational> (accessed Jul. 22, 2022).

national and international operations. High levels of interest within IEEE for research to improve digital privacy is still largely unresearched, but the application of BlockFrame® technologies could offer excellent technical solutions to many digital privacy requirements. Hundreds of use cases can be extended to other industrial verticals and government security functions, especially telecom, medical, and education are fertile fields for more academic research and development. Most recently, BlockFrame® has initiated new research to improve artificial intelligence applications using attested data using BlockFrame® technologies. New discussions with a foreign government economic development office¹⁷⁶ expressed their high interest in BlockFrame® applications and data security for supply chains for green energy, manufacturing, real estate transactions, transportation and storage, information and telecom, “precision agriculture,” healthcare, and supply chains for critical minerals. That government was also very interested in including institutions of higher education, and applications that supported economic development for blockchain research and services in rural areas. That seems to validate that many of the use cases that have already been reported in this dissertation are also of the highest interest in other international markets, and that useful and exciting opportunities for more academic research and commercial development will continue to expand. Many of the topics addressed from this research with industry have shown that there are many other opportunities for more valuable academic research with industry and professional partners, especially in the areas of emergence theory, rigorous ontologies and definitions of new concepts, the application of new algorithms for digital trust and more optimized blockchain consensus, the application of reference architectures of BlockFrame® technologies, and more technical solutions for digital privacy and security. All of the research questions addressed in this dissertation can and should be viewed as starting points for ongoing research and inquiry.

¹⁷⁶ BlockFrame, Inc., Board of Directors, September 4, 2025.

Appendix A: Monterey Phoenix Script Chain Example

File name: BlockFrame_Chain_Sequence v6.mp

/* Practice Programming Example of BlockFrame Content Sequence*/

SCHEMA BlockFrame_Chain_Sequence

ROOT Chain:

/* NESTED APPROACH

```
    (+ Broker_Agent_Gets_Current_Content
      ( Create_Primary_Block
        ( Create_Primary_Sync_List_Chain
          (Create_Bridge_Block
            (Create_Global_Block
              | Defenses_Fail_Attack_Successful )
            | Defenses_Fail_Attack_Successful )
          | Defenses_Fail_Attack_Successful )
        | Defenses_Fail_Attack_Successful )
      | Defenses_Fail_Attack_Successful )
  */

      (+ <1> Broker_Agent_Gets_Current_Content          +)
      (+ <1> Create_Primary_Block                        +)
      (+ <1> Create_Primary_Sync_List_Chain              +)
      (+ <1> Create_Bridge_Block
    +)
      (+ <1> Create_Global_Block
    +)
    ;

Broker_Agent_Gets_Current_Content:
  Cyber_Attack_on_Current_Content
  Deny_Attack_on_Current_Content
  Get_Current_Content /* returns content block number where content pointer is
currently pointing */
  ( Return_Content | /* returns block at current pointer or */
    Set_Content ) /* set content to most recently made content block then return
block */
  Last_Content /* sets content to most recently made content block then returns block
*/
  Index_Content /* starting at the content pointer, return current pointer content block
for selected index */
  Index_Page_Content /* starting at the content pointer, return current pointer content
block and all content blocks back to last sync primary block */
  Primary_Content /* return that number content and set pointer to that primary */
  ;
```

Create_Primary_Block:

```
  Cyber_Attack_on_Primary_Block
  Deny_Attack_on_Primary_Block
  Get_Current_Primary /* returns primary "block number" where primary pointer is
pointing currently */
```

```

        ( Return_Primary | /* Return block at primary pointer */
          Set_Primary ) /* (or set pointer to most recently made primary block then
return block) */
        Last_Primary /* Sets pointer to most recently made primary block then returns
block */
        Index_Primary /* starting at the primary pointer return current pointer "primary
block" for selected index */
        Index_Page_Primary /* starting at the primary pointer return current pointer "Primary
block" and all primary blocks back to last sync block */
        Primary_Primary /* return that number primary and set pointer to that primary */
;

```

Create_Primary_Sync_List_Chain:

```

        Cyber_Attack_on_PSL_Chain
        Deny_Attack_on_PSL_Chain
        Get_Current_PSL_Chain /* returns psl "block number" where psl pointer is pointing
currently */
        ( Return_PSL_Chain | /* Return "block" at (psl pointer) */
          Set_PSL_Chain ) /* (or set psl pointer to most recently made psl (under current
selected bridge??) then return "block") */
        Last_PSL_Chain /* Sets psl pointer to most recently made psl (under current selected
bridge??) then returns "block" */
        Index_PSL_Chain /* (starting at psl pointer) return "Sync, Link numbers" and all
"Sync, Link numbers" in the same bridge */
        Index_Page_PSL_Chain /* (starting at the psl pointer) return current pointer Sync, Link
block and all primary sync list included in the same bridge */
        Bridge_PSL_Chain /* return that number "Sync (or ??) Link blocks" and set pointer to
that bridge */
;

```

Create_Bridge_Block:

```

        Cyber_Attack_on_Bridge_Block
        Deny_Attack_on_Bridge_Block
        Get_Current_Bridges /* returns "block number" where bridge pointer is pointing
currently */
        (Return_Bridges | /* Return "block" at (bridge index) */
          Set_Bridges) /* (or set bridge pointer to most recently made bridge block then return
"block") */
        Last_Bridges /* Sets bridge pointer to most recently made bridge block then return
"block" */
        Index_Bridges /* (starting at bridge pointer) return "bridge block number" and all
previous "bridge block numbers" back to last global */
        Index_Page_Bridges /* (starting at bridge pointer) return "bridge block" and previous
10 "bridge blocks" */
        Index_Page_Full_Bridges /* (starting at bridge index) return "bridge block" and all
previous back to last global "bridge blocks" ??? */
        Bridge_Block_Bridges /* return single "bridge block" and set global pointer to that
(number) bridge */
;

```

Create_Global_Block:

```

        Cyber_Attack_on_Global_Block

```

```

    Deny_Attack_on_Global_Block
    Get_Current_Global    /* returns "block number" where global pointer is pointing
currently */
    (Return_Global |      /* Return "block" at (global index) */
     Set_Global)          /* (or set global pointer to most recently made global then return
"block") */
    Last_Global           /* Sets global pointer to most recently made global then returns "block"
*/
    Index_Global          /* (starting at global pointer) return "global block number" and
previous 10 "global block numbers" */
    Index_Page_Global     /* (starting at glob_al pointer) return "global block" and previous
10 "global blocks" */
    Global_Block_Global   /* return single "global block" and set global pointer to that
(number) global */
;

```

```

ROOT Threat:      (* Cyber_Attack_on_Current_Content  Deny_Attack_on_Current_Content
*)
                  (* Cyber_Attack_on_Primary_Block
Deny_Attack_on_Primary_Block      *)
                  (* Cyber_Attack_on_PSL_Chain
Deny_Attack_on_PSL_Chain          *)
                  (* Cyber_Attack_on_Bridge_Block
Deny_Attack_on_Bridge_Block       *)
                  (* Cyber_Attack_on_Global_Block
Deny_Attack_on_Global_Block       *)
;

```

```

Cyber_Attack_on_Current_Content:      /* If attack successful, STOP; if unsuscessful */
    (Attack_Content_Software_and_Hardware |
     Attack_Content_Personnel)
;

```

```

Deny_Attack_on_Current_Content:
    ( System_Container_Defend_Content | Attack_Successful )
;

```

```

Cyber_Attack_on_Primary_Block:
    Attack_Primary_Software_and_Hardware
;

```

```

Deny_Attack_on_Primary_Block:
    ( System_Container_Defend_Primary | Attack_Successful )
;

```

```

Cyber_Attack_on_PSL_Chain:
    Attack_Chain_Software_and_Hardware
;

```

```

Deny_Attack_on_PSL_Chain:
    ( System_Container_Defend_Chain | Attack_Successful )
;

```

```

Cyber_Attack_on_Bridge_Block:
    Attack_Bridge_Software_and_Hardware
    ;

Deny_Attack_on_Bridge_Block:
    ( System_Container_Defend_Bridge | Attack_Successful )
    ;

Cyber_Attack_on_Global_Block:
    (Attack_Global_Software_and_Hardware |
     Attack_Global_Personnel)
    ;

Deny_Attack_on_Global_Block:
    ( System_Container_Defend_Global | Attack_Successful )
    ;

Attack_Successful:
    ;

Threat, Chain SHARE ALL
    Cyber_Attack_on_Current_Content,
    Deny_Attack_on_Current_Content,
    Cyber_Attack_on_Primary_Block,
    Deny_Attack_on_Primary_Block,
    Cyber_Attack_on_PSL_Chain,
    Deny_Attack_on_PSL_Chain,
    Cyber_Attack_on_Bridge_Block,
    Deny_Attack_on_Bridge_Block,
    Cyber_Attack_on_Global_Block,
    Deny_Attack_on_Global_Block,
    Attack_Successful
    ;

COORDINATE $f: Attack_Successful
    DO
        IF ( #$$EVENT BEFORE $f == 0 AND
            #$$EVENT AFTER $f == 0 )
            THEN REJECT;
        FI;
    OD;

    IF #Attack_Successful >0 THEN
        SAY ("Attack Successful; Service Denied");
    FI;

    IF #Create_Global_Block >0 AND #Attack_Successful == 0 THEN
        SAY ("Complete Chain successfully constructed!");
    FI;

/* GLOBAL OR LOCAL REPORTS SEE EXAMPLES

```

*/

File name: Attack_Stopping_Construction_v2.mp
SCHEMA Attack_Stopping_Construction

```
ROOT Chain: (+ Content
              ( Primary
                ( PSL_Chain
                  ( Bridge
                    ( Global
                      | Defenses_Fail Attack_Successful )
                    | Defenses_Fail Attack_Successful )
                  | Defenses_Fail Attack_Successful )
                | Defenses_Fail Attack_Successful )
              +);
```

ROOT Threat: [Attack_Successful];

Chain,Threat SHARE ALL Attack_Successful;

```
COORDINATE $f: Attack_Successful
DO  IF ( #$$EVENT BEFORE $f == 0 AND
        #$$EVENT AFTER $f == 0 )
    THEN REJECT;
FI;
OD;
```

```
IF #Global >0 AND #Attack_Successful == 0 THEN
    SAY ("Complete chain successfully constructed!"); FI;
```

```
/*  UN-COMMENT the following documented assumption and constraint if
    iteration is added. */
/*ENSURE (#Attack_Successful<=1);
*/
```

Appendix B: Philos™ Neutral Governance Architecture Overview and Summary Information

Kent D. Lambert, BlockFrame, Inc.

February 9, 2020

Executive Summary

This is intended to be a “living” document prepared to consider the application of model-based systems engineering (MBSE) and model-based system architecture processes (MBSAP) (Borky & Bradley, 2019¹⁷⁷) to a Neutral Governance Organization (NGO)¹⁷⁸ formed to provide policy governance for the Philos Blockchain/Distributed Ledger (B/DL) Enterprise™ or other similar systems. NGO policy governance will also connect to the operations of technical System governance. This document was also prepared for partial satisfaction of the author’s research at Colorado State University for Engr 667 and Engr 587. Philos™ could provide B/DL Services to a wide range of state government use cases, and/or services to many other applications, including the BlockFrame® Eco-Secure Security Provisioning Enterprise™, ESP™. To date, the State of Colorado has identified 71 additional use cases for B/DL applications (per Sri Garimella, Deputy Colorado CIO). This document is expected to help define, refine, document, and eventually support a transparent development and business plan for the Philos™ Neutral Governance Organization (NGO). All of the use cases mentioned are currently beyond the scope of the NGO itself but should be considered tangentially as the ultimate end users for the capabilities that will be required and provided by the NGO. Following sections will consider the architecture of the NGO, with the goals of identifying its structure, behaviors, data, services, and contextual artifacts. Initially, the focus will be on the highest-level operational viewpoint (OV), including process/workflow, the concept of operations (CONOPS), program management, and organizational and stakeholder identification, relationships, and roles. In the future, as more specificity is discovered, this document is expected to evolve more detailed functional/logical (LV) and physical (PV) viewpoints of the NGO System. Ideally, and perhaps most importantly, this Architecture Overview and Summary is intended to act as a voluntary “living” contract, a transparent pattern to organize, visualize, and guide discussions concerning the system architecture, and as a set of organizing principles among policy leaders, program and technical leaders, the System

¹⁷⁷ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

¹⁷⁸ As used here, “NGO” refers to a “Neutral Governance Organization,” not to a “Non-Governmental Organization.”

Architectural Team, and stakeholders, based on establishing a mutual understanding and commitment to the overall program and system. That process began on February 7, 2020, as reflected in Ref (BCDC¹⁷⁹).

Architecture Identification and Description

General

The NGO is a necessary but separate system to provide neutral policy and technical governance support for the Philos™ Blockchain/Distribute Ledger (B/DL). The neutrality of the NGO is important to avoid potential conflicts of interest, to provide public transparency and scrutiny, to manage the oversight for the technical processes of providing governance services to Philos™, and to operate the final consensus process of each new blockchain block within the Philos™ B/DL Enterprise.

Assumptions and Constraints

This document is not intended to be a legal charter for the NGO, which is being planned separately. It is intended to support the NGO's organizational decision-making processes by anticipating the system architecture, capabilities, stakeholder goals, and documenting those into a structured design process to increase understanding of every aspect of the system. Although there should ideally be a one-to-one association and dependency between the Philos™ Enterprise and the NGO system, we assume that system engineering best practices of object-orientation, modularity, encapsulation, and loose coupling should still be enforced to assure that future upgrades or replacements can be made with minimal disruption or loss of services.

Architecture Development Resources

Architecture development for the NGO is being initiated by the author, using MBSE and MBSAP processes described in Borky & Bradley, 2019.¹⁸⁰ Figure B.1, from the same source shows a top-level overview of the MBSAP methodology.

¹⁷⁹ Blockchain Development Community (BCDC) Policy & Governance Team, "Meeting Notes: Policy and Governance for Philos Distributed Ledger."

¹⁸⁰ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

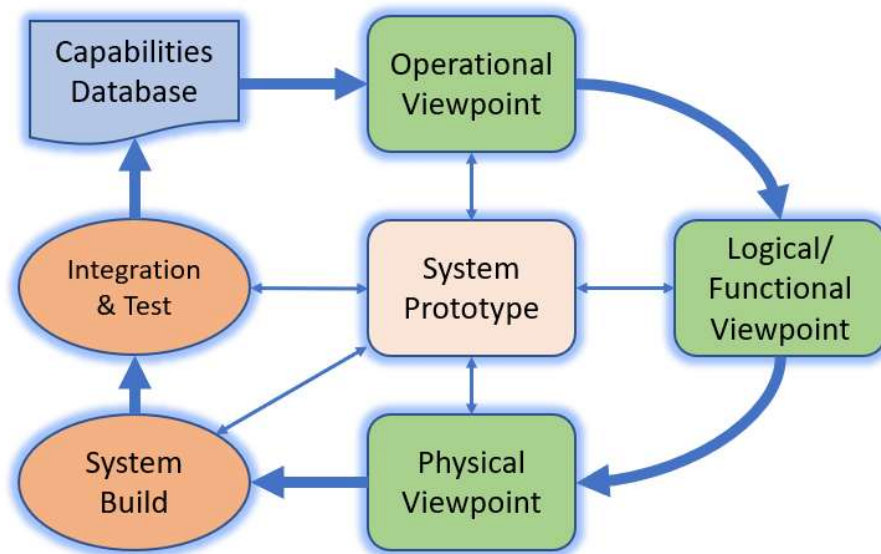


Figure B.1. Top-level overview of MBSAP Methodology

Scope

The initial scope of this Architecture Overview is to study and develop an operational architecture of the OV for the NGO. That would initially involve identification and cataloging high-level capabilities into the Capabilities Database, modeling artifacts in the OV, and possibly include some limited prototyping and testing. In the future, the scope should be extended to an LV and a PV of a worldwide blockchain governance mechanism, involving both human policy interactions and real-time technical automation, to support processes, perhaps involving millions of applications on billions of devices.

Architecture Model and Artifacts

Computer-aided SysML architectural modeling was accomplished using the NoMagic Cameo Model, under license to Colorado State University for academic research. The goal of this modeling is to use MBSE and MBSAP to create a digital twin of the actual NGO system to assist in system design and management. Modeling and artifacts will initially be limited to focus on the OV and will include the primary artifacts below.

Primary Artifacts

Operational Viewpoint (OV) containing the following Perspectives:

- Structural Perspective

- A notional Block Definition Diagrams of the NGO is shown in Figure B.2. (This and all subsequent diagrams are borrowed from (Lambert. 2019¹⁸¹). Initial internal vetting within BlockFrame, Inc., will begin with a higher degree of specification for Values, Operations, and Parts.
- A notional Internal Block Diagram of the NGO is shown in Figure B.3. Initial internal vetting within BlockFrame, Inc., will begin with a higher degree of specification for Ports, Flows, and User Roles. As development continues, more specific technical definitions of Ports, Flows, and Constraints would be expanded within these models.

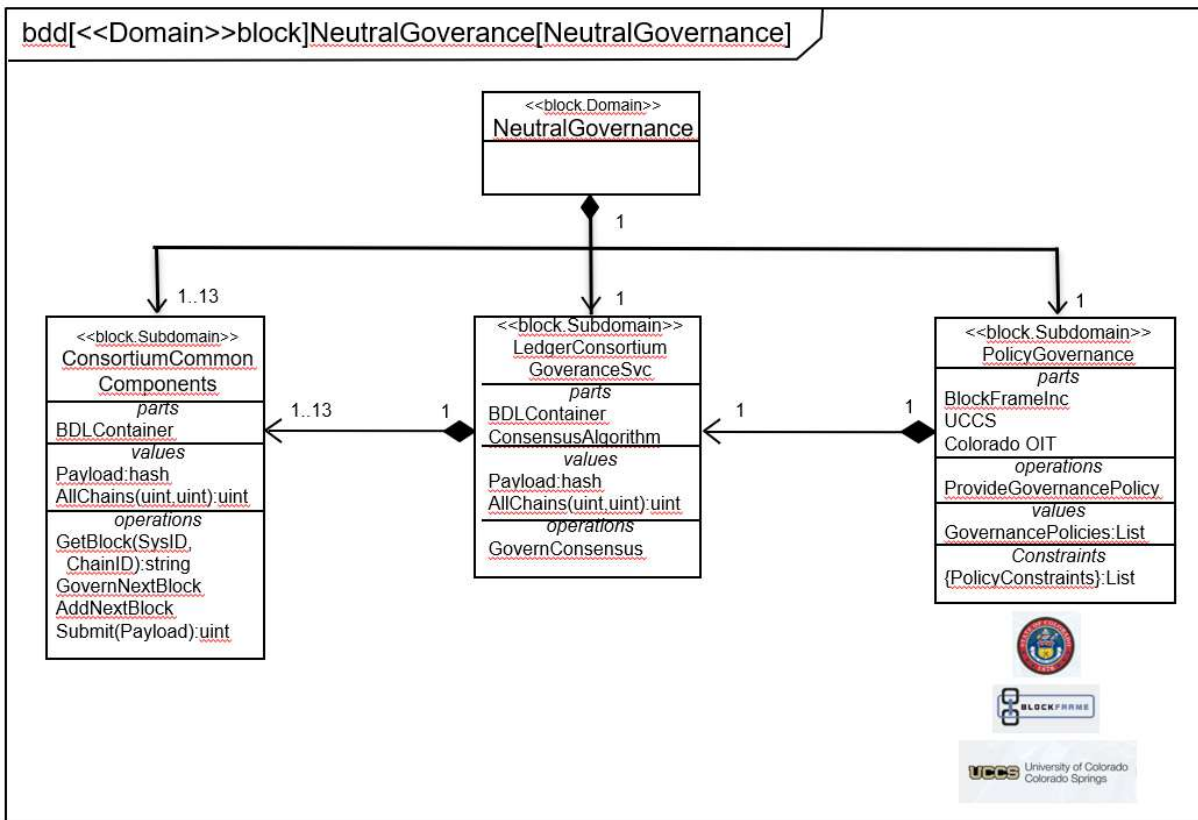


Figure B.2. Notional Neutral Governance Domain with Parts, Values, Operations, and Constraints

¹⁸¹ Lambert, "Final Project Report for Engr 567."

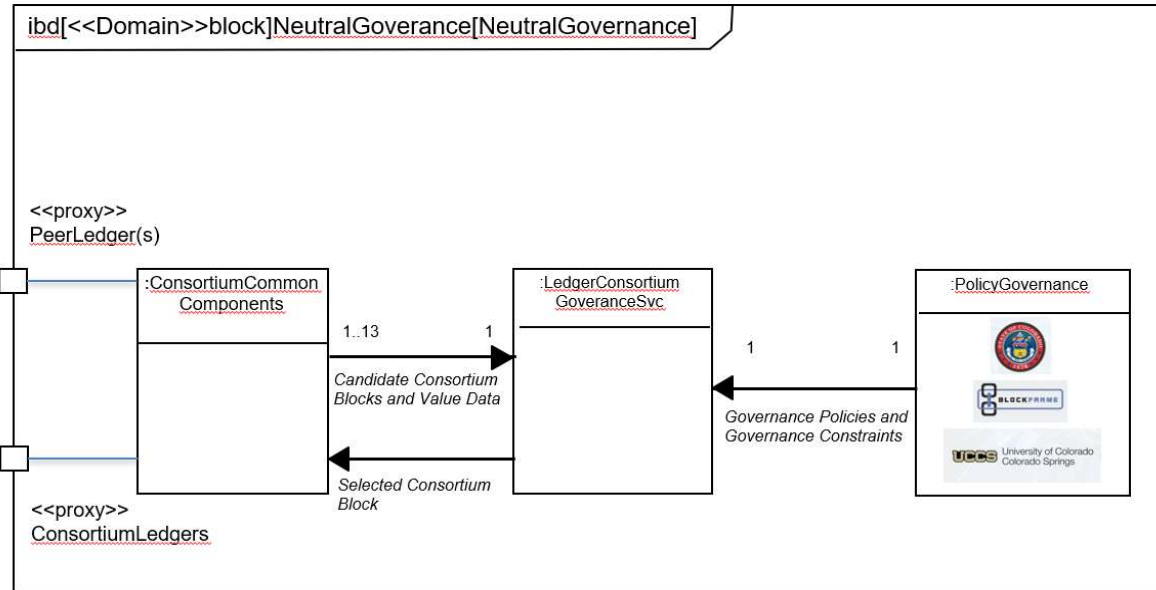


Figure B.3. Notional Neutral Governance Internal Block Diagram

- Behavioral Perspective
 - The Behavioral Perspective describes the key high-level behaviors within the OV. For the NGO, the most important diagrammatic visualizations that should be considered are case diagrams to help specify the diverse Actors within the NGO, and external Actors that they will need to interface with. Activity and Sequence Diagrams that describe and model the behaviors of blockchain formation, blockchain consensus processes, and the processes used for governance decision making.
 - Figure B.4 gives a simple example of a technical activity diagram to show hypothetical behavior to provide proof of a transaction posted by an energy company. The only transactions of importance to the NGO in this example are the data flowing to and from the Philos™ B/DL. The primary purpose of diagrams like this would be to associate the behavioral activities of the NGO System with the Philos™ Enterprise and thoroughly describe the detailed behaviors of each.

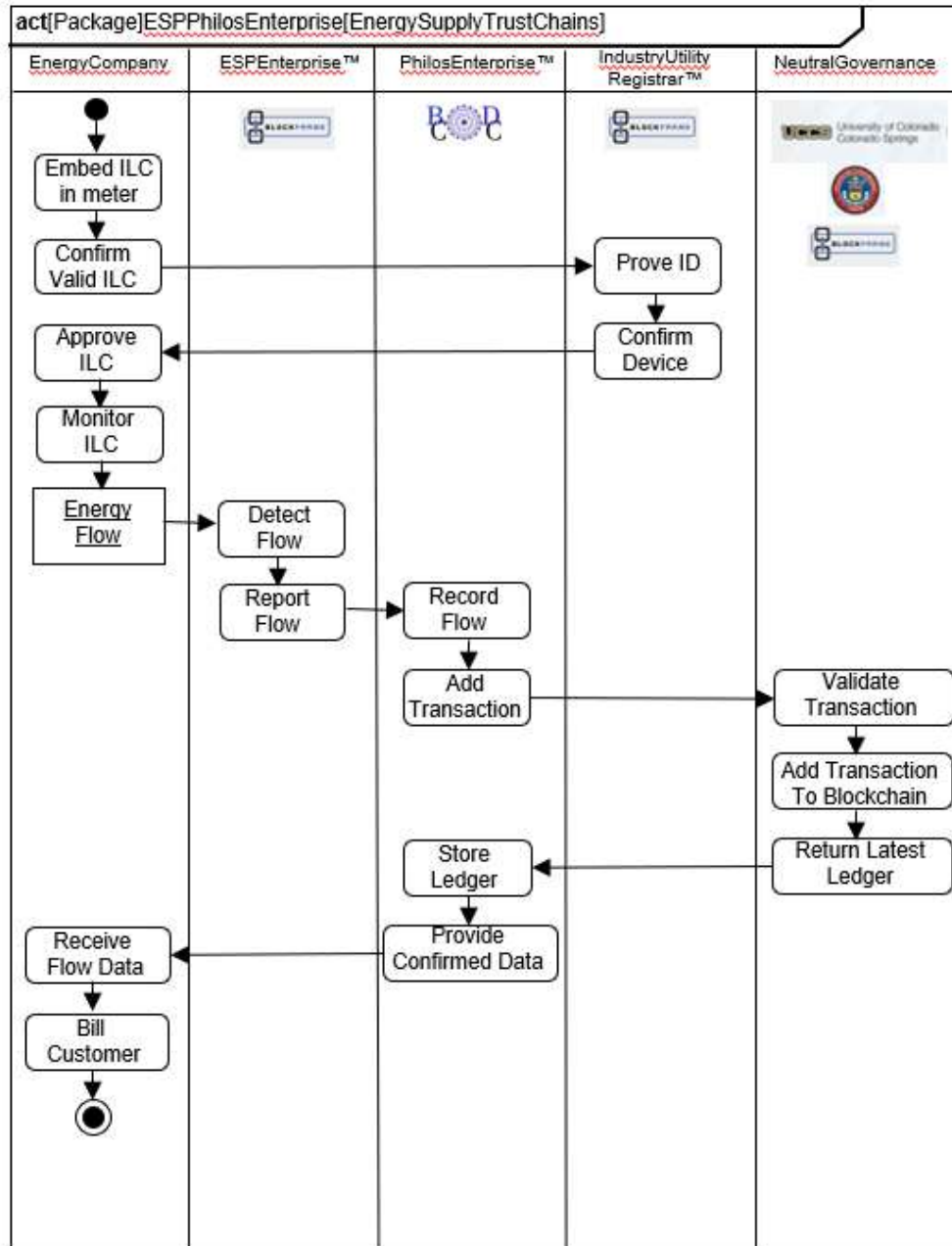


Figure B.4. Notional Activity Diagram to Provide Proof of Energy Company Transactions

- Specific descriptions of forming new blocks within the Philos™ B/DL are essential to the success of not just the NGO, but the Enterprises that use its Services. Complex Sequence and State Machine Diagrams should be used to specify how highly distributed Philos™ Peer Ledgers and other Consortium elements interact. Due to the criticality of successfully meeting Consensus deadlines within a system using heavily encrypted data processes, these

processes could be considered as Real Time processes or should at least be analyzed as such.

- Other behaviors that should be considered are policy interactions among domestic and foreign governments and organizations such as the Institute of Electrical and Electronic Engineers (IEEE), the International Council on Systems Engineering (INCOSE), the National Institute of Standards and Technology (NIST), the International Organization for Standardization (ISO), and others.
- Behaviors of the Philos™ Enterprise and the NGO will use models and simulations – initially using interactive dashboards – for decision support to NGO governing members. The behaviors of that governing process, with specific descriptions of any constraints, could be important descriptions provided by the Model.

- Data Perspective

As specific data requirements are discovered and/or defined for each component of within the NGO, and for Ports and Interfaces with the Philos™ Enterprise, those specific data definitions will be embedded within all appropriate structural diagrams. Data consistency throughout the architecture is important to identify early so help reduce time and costs as the process moves through the LV and PV Viewpoints, and for engineering design specifications. Explicitly, the custom design of blockchain components, encryption data and artifacts, and other specialized data formats are also essential for successful NGO operations.

- Services Perspective

The input/output of the NGO is essentially – and maybe exclusively – a governance service provided to the Philos™ Blockchain to collect, merge, and process blockchain data components from a worldwide Enterprise of Philos™ users, and to re-distribute newly-created blockchain blocks back to each user. As such, this suggests that a Service-Oriented Architecture (SOA) should be considered and designed from the start. For security purposes, interfaces should consider the best practices of encapsulation, loose coupling, and modularity. Although normal operations might not need to “pull” data from many sources, regulatory and/or forensic audits

of blockchain data may be required to be processed upon demand, which again suggests that an SOA should be an inherent part of the System design.

- Contextual Perspective
 - Figure B.5 shows an overview of the primary contextual marketplace environment of the Philos™ Enterprise, the Eco-secure Provisioning Enterprise. Within the current concept, the NGO would interface only with the Philos™ B/DL Enterprise. All other applications using Philos™ would be separated from the NGO, although the Requirements for other Enterprises and Systems could drive some NGO Requirements, and therefore need to be considered as part of the overall System-of-Enterprise Marketplace.
 - Non-Functional Requirements could be very important to identify for the NGO, especially for various business processes such as connections with State licensing and payment processes with the Colorado Department of Regulatory Agencies, tracking business costs, and analyzing fee schedules for economic sustainability.
 - Tracking Reliability, Maintainability, and Availability (RMA) metrics of important Non-Functional Requirements should be a very important consideration. Hardware RMA may typically be less important than the RMA of software components, software versions, security procedures, and compliance with rapidly evolving standards.
 - International law and possible dispute arbitration processes are an important consideration for the NGO and will be considered throughout the system design and operations.
 - Within the Contextual Perspective, key documents and definitional dictionaries associated with the NGO should be documented within the model as the “single source of technical truth,” to avoid misunderstanding that could occur through multiple divergent definitions from multiple sources.

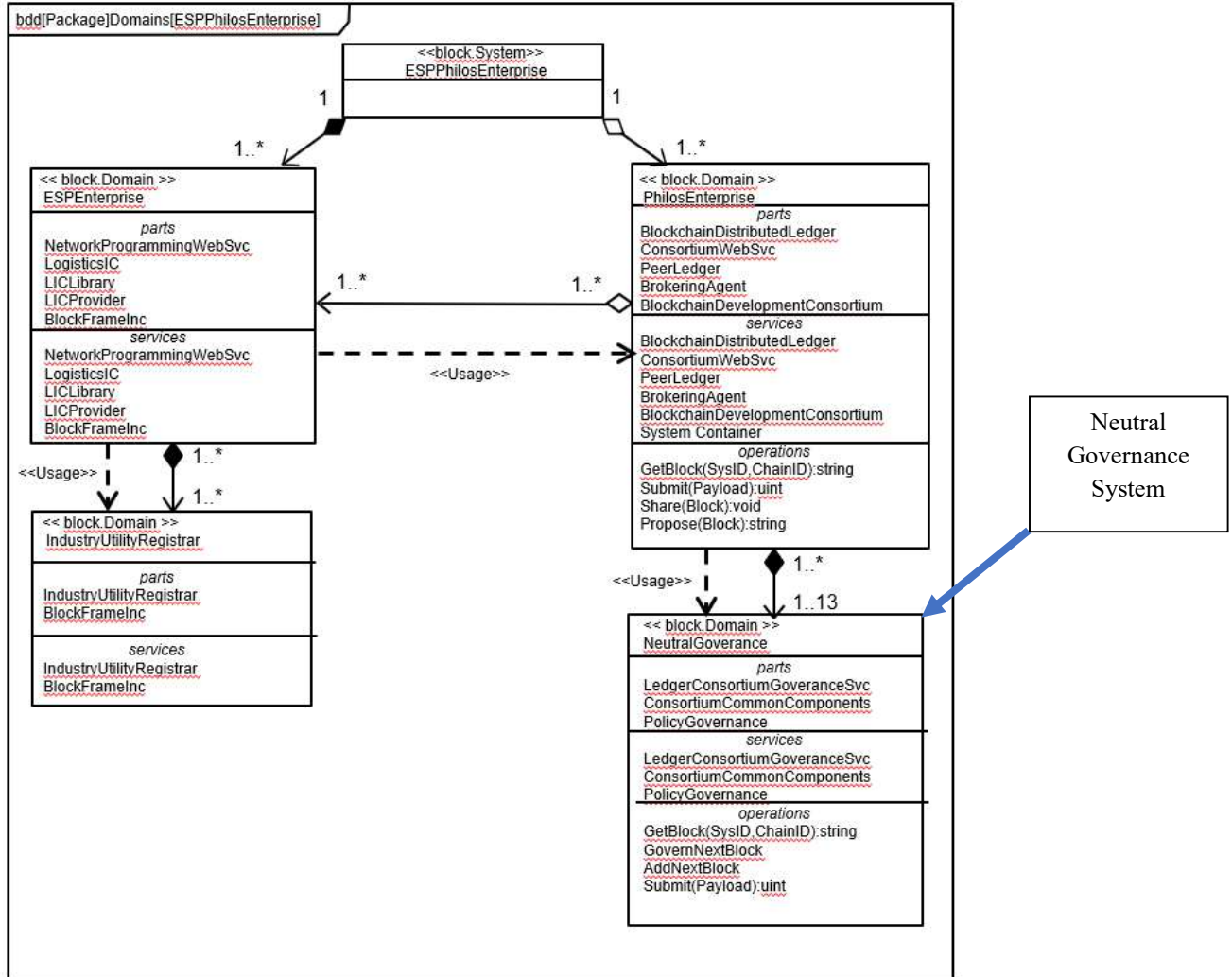


Figure B.5. Combined ESP™ and Philos™ Enterprises with Allocated Services

Logical/Functional and Physical Viewpoints will be added later as more detailed information and decisions about the NGO process evolve.

Focused Viewpoints:

- Security Viewpoint
 - Very complex and aggressive security considerations and requirements inherent within the Philos™ B/DL Enterprise require that an identical level of security be applied to the NGO System. However, since those Requirements and Capabilities will not originate within the NGO System, they will be considered outside the scope of the NGO organizational issues and development. However, technical design of NGO components will need to comply

with all Philos™ security components and procedures and must inherently be considered within the NGO technical design.

- Compliance with NIST, ISO, and Cybersecurity Maturity Model Compliance (CMMC) will be considered in System design.

- Systems Management Viewpoint

Technical Systems Management will be performed, at least initially, through the BCDC, with support from partners Bees Computing, Inc., and BlockFrame, Inc.

Architecture Timeline and Evolution

Extensive academic research on NGO System technical and organizational consensus theory is conducted by Christopher Gorog, UCCS, and BlockFrame, Inc. (Gorog, 2020¹⁸²). Formal NGO organizational and architecture discussions began on February 7, 2020 (BCDC¹⁸³). NGO evolution is expected to parallel the development of the Philos™ B/DL. Both are expected to be developed as a function of resources available, but initial development and productization could be realized in late 2020 or shortly thereafter.

Organizations Involved

The NGO organization is initially expected to be comprised of members and support members from BlockFrame, Inc., the Blockchain Development Community (BCDC), the University of Colorado Colorado Springs, Colorado State University, the University of Denver, and the Colorado Office of Information Technology. Future participants are expected to include national, international, and professional organizations, and include national users and government representatives, the Internet Corporation for Assigned Names and Numbers (ICANN), IEEE, INCOSE, and other professional organizations. Government organizations, especially within the State of Colorado, will initially include participation with the Colorado Office of Information Technology and the Colorado Department of Regulatory Agencies.

Purpose and Viewpoint

Architecture Purposes and Uses

The NGO organization and architecture will be designed as a neutral governance system to support the Philos™ B/DL, as previously described.

¹⁸² Gorog and et. al., “Blockchain Synchronous Trust Consensus Model.”

¹⁸³ Blockchain Development Community (BCDC) Policy & Governance Team, “Meeting Notes: Policy and Governance for Philos Distributed Ledger.”

Architecture Analysis

Initial analysis will be a product of modeling and simulations of components shared between the Philos™ B/DL and the NGO. The main focus is currently to verify the shared blockchain structural and behavioral operational processes. As of February 2020, live demonstrations of those processes have been conducted across a limited number of components, with initial visualizations and analysis displayed on custom dashboards. Initial validation with system stakeholders started with displays of representative dashboard reports. Future expansions of those processes will eventually evolve to complex decision displays and tools for NGO governance participants, and to some extent for public transparency and oversight.

Technical algorithms and processes are currently being considered to optimize and prioritize Philos™ blockchain operations. When candidate algorithms are delivered, they will be simulated and tested within the physical distributed blockchain environment, and possibly virtually within the Cameo Systems Engineering Model as well. It is likely that multiple algorithmic alternatives could be developed, which could lead to an analysis of alternatives as they become ready for testing.

Stakeholder Perspectives

An initial organizational meeting found a strong willingness for stakeholders to contribute to the NGO process. This report has not been presented yet to NGO participants, and specific MBSE and MBSAP capabilities have not yet been discussed. Future stakeholder perspectives, issues, and concerns will be included in this section.

Findings

Analysis Results

Initial live simulations of technical Philos™ blockchain propagation have been successful in demonstrating that it can meet specific goals. That suggests that the formation of the Neutral Governance Organization should proceed.

Recommendations

The author recommends that this Architecture Overview and Summary be scheduled for distribution to current NGO members and selected project support members for initial markup, feedback, and further discussions.

It is also recommended that the NGO accept this Architecture Overview and Summary process as an interactive structure and roadmap for future discussions, architectural design, and decision support.

Addendum

Since this AOS document was prepared in February 2020, several assumed characteristics of the NGO have changed. A working group was formed to discuss policy details, roles and missions, bylaws, a charter (which are currently under legal review), the probable formation of the NGO as a non-profit corporation, and the formation of at least one for-profit corporation to perform assigned tasks that could avoid conflicts of interest or tax liability concerns. Extensive participation in these issues has involved UCCS, with new inclusion with the University of Denver. As a result, it is possible that this AOS could be split into two or more documents to specific the roles and missions of more organizations affiliated with the Philos™ Blockchain.

Appendix C: Modeling Examples using SysML, MBSE, and MBSAP

This Appendix C extends research reported in Chapter 6 with more detail and addresses Research Question 2 (RQ2) by demonstrating how Model-Based Systems Engineering (MBSE), Model-Based System Architecture Process (MBSAP), and the Systems Modeling Language (SysML) Framework can be applied to the design of the BlockFrame® system-of-systems for supply chain and cybersecurity applications. Restated, Research Question 2 (RQ2) is: “How can Model-Based Systems Engineering (MBSE) and Model-Based System Architecture Process (MBSAP) be used to support the design of cybersecurity systems?” Because most fundamental MBSE and MBSAP principles have already been adequately discussed in Chapter 6, the examples shown here will focus more on graphic examples using automated tools for analysis and visualization. The Dassault Cameo Systems Modeler¹⁸⁴ tool is used here because of its availability, but other automated tools could be used that follow the same SysML Framework standards. In most cases, Cameo and other SysML tools provide automated capabilities for more rapid architectural development, and a highly structured environment to enforce uniformity of definition, eliminate duplication of terms throughout architectures, and many other specialized capabilities to map and in some cases to simulate detailed behaviors that are not possible through manual approaches.

Requirements examples: Functional and Non-functional Requirements

While it is not necessary or mandatory to start systems engineering projects with detailed lists and explanations of requirements, many systems engineering processes begin by specifically defining functional and non-functional requirements. The Cameo tool has good capabilities to create organized hierarchies of requirements and allows the import and export of those data in spreadsheet formats. Once defined, the Cameo tool allows users to associate requirements with those system components, parts, and behaviors that satisfy those requirements, as well as test and evaluation procedures that can help prove those capabilities. The Cameo tool can also be used to add context by extensively documenting comments and attaching entire relevant documents. When done following MBSAP concepts, the ideal end product of a SysML architecture may be to stand as the “single source of technical truth” for the system being modeled.

¹⁸⁴ “Cameo Systems Modeler - CATIA - Dassault Systèmes®.” Accessed July 21, 2022. <https://www.3ds.com/products-services/catia/products/no-magic/comeo-systems-modeler/>

Table C.1 shows Functional Requirements (FRs) input into a Requirements Table from Excel using the Cameo tool add-in import and synchronization commands.¹⁸⁵

Table C.1. Functional Requirements

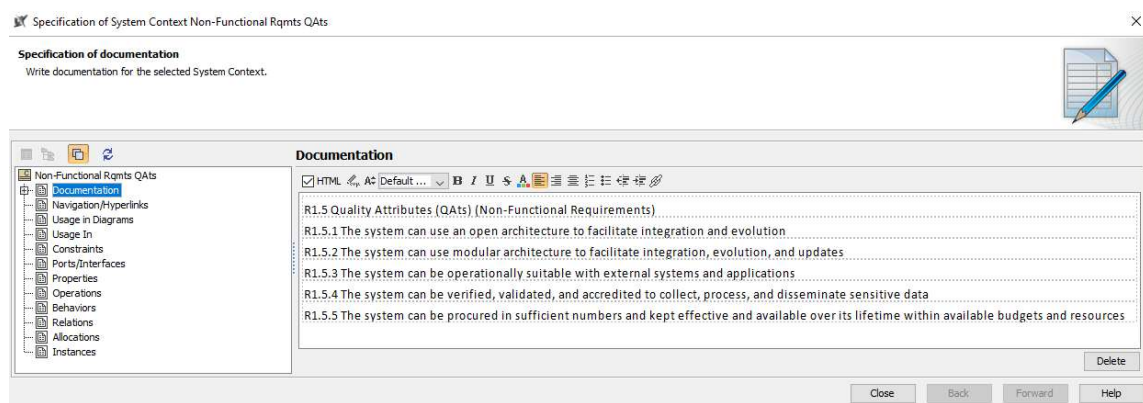
#	Name	Text
1	R.1.1	The system shall provide an Eco-Secure Provisioning Framework for IoT Devices (Project Domain Nickname ESP)
2	R.1.1.1	The system shall provide identity features for supply and trust chains
3	R.1.1.2	The system shall be scalable to many use cases in large marketplaces across any industry
4	R.1.1.3	The system shall be able connect to individual IoT devices through embedded logistics devices
5	R.1.1.4	The system shall program mass distributed IoT devices in network
6	R.1.1.5	The system shall provide cryptologically-secure protection for IoT devices
7	R.1.1.6	The system shall provide cryptologically-secure protection for system components
8	R.1.1.7	The system shall provide security management and control services at the organization/business level
9	R.1.1.8	The system shall provide global services across jurisdictional boundaries
10	R.1.1.9	The system shall provide verification and validation of user/actor/device identities
11	R.1.1.10	The system shall operationally increase measurable trust levels among users (and reduce risk) over time
12	R.1.1.11	The system shall use open architectures of modular components
13	R.1.1.12	The system shall design components to accept expandable technologies
14	R.1.1.13	The system shall maintain and provide standardized libraries for embedded logistics devices
15	R.1.1.14	The system shall have the capability to prove the origin of data -- Proof of Origin
16	R.1.1.15	The system shall coordinate, manage, and synchronize all operational ecosystem components
17	R.1.1.16	The system shall have the ability to wrap legacy equipment by tools/APIs to support cyber security
18	R.1.1.17	The system shall manage the use and delivery of cryptographic tools and secret codes
19	R.1.1.18	The system shall assure the security of the handling and delivery of products
20	R.1.1.19	The system shall conduct capabilities and security testing and evaluation
21	R.1.1.20	The system shall (after initial stability) be economically self-sustaining by user fees
22	R.1.1.21	The system shall provide education and training to developers and users
23	R.1.1.22	The system shall manage cryptographic key management throughout the IoT life cycle
24	R.1.1.23	The system shall detect, authenticate, and update any IoT devices within the trust chain
25	R.1.1.24	The system shall manage the secure distribution and installation of software to IoT devices
26	R.1.1.25	The system shall facilitate media control or digital rights management (DRM)
27	R.1.1.26	The system shall manage "New Supply Chain Industry Creation" initiatives and modifications

¹⁸⁵ Dassault Systèmes®. "Cameo Systems Modeler - CATIA - Dassault Systèmes®." Accessed July 21, 2022. <https://www.3ds.com/products-services/catia/products/no-magic/cameo-systems-modeler/>.

28	☐ R1.2	The system shall provide Records, Documents, and Transactions management (Project Domain Nickname Philos)
29	☐ R1.2.1	The system shall provide immutable records of data transactions
30	☐ R1.2.2	The system shall provide privacy of individual records
31	☐ R1.2.3	The system shall minimize the use of energy for management
32	☐ R1.2.4	The system shall transact at speeds to support large-scale operations
33	☐ R1.2.5	The system shall incentivize cooperative distributed sharing of managed data
34	☐ R1.2.6	The system shall limit the growth of managed data to a controllable size
35	☐ R1.2.7	The system shall support both continuous and non-continuous operations
36	☐ R1.2.8	The system shall manage data phase-out and end-of-life
37	☐ R1.2.9	The system shall support managed data blocks of different sizes
38	☐ R1.2.10	The system shall balance transparency and visibility with privacy
39	☐ R1.2.11	The system shall manage transactions to allow indefinite number of large-scale smart contracts
40	☐ R1.2.12	The system shall use independent subcomponents that allow indefinite expandability and operational consensus of trust
41	☐ R1.2.13	The system shall have an open architecture to allow developers and general public to make their own contracts
42	☐ R1.2.14	The system shall have the capability to add and remove users and organizations
43	☐ R1.3	The system shall have usable controls --like web interfaces, mobile apps, Zero-Trust
44	☐ R1.2.16	The system shall manage management subcomponents to provide service to multiple organizations and industry types
45	☐ R1.2.17	The system shall provide end-to-end transaction security for participating transaction users on supply chains
46	☐ R1.3	The system shall provide an Industry Utility Registrar
47	☐ R1.3.1	The system shall use expandable industry identification registers of users/devices on multiple networks
48	☐ R1.4	The system shall Manage Neutral Governance
49	☐ R1.4.1	The system shall use neutral governance infrastructure to manage and optimize the (DXG-like) transaction engine
50	☐ R1.4.2	The system shall use a consortium of common components for all indefinitely scalable governance

The Non-Functional Requirements (NFRs) in Table C.2 were created as a Context Package in an Excel table under the Specification Documentation field for Non-Functional Requirements.

Table C.2. Non-Functional Requirements



Domain Diagram examples

Figure C.1 shows the Specification of four ESP™/Philos™ Domains. For consistency, the Eco-Secure Provisioning Framework (ESP)™ and the Philos™ Blockchain/Distributed Ledger are usually defined as modular Enterprises, making the combined model a “System of Enterprises.” For similar reasons, the IndustryUtilityRegistrar™ and the NeutralGovernance Domain can be described as separate Systems. Note that Reference Properties here represent Services in SysML.

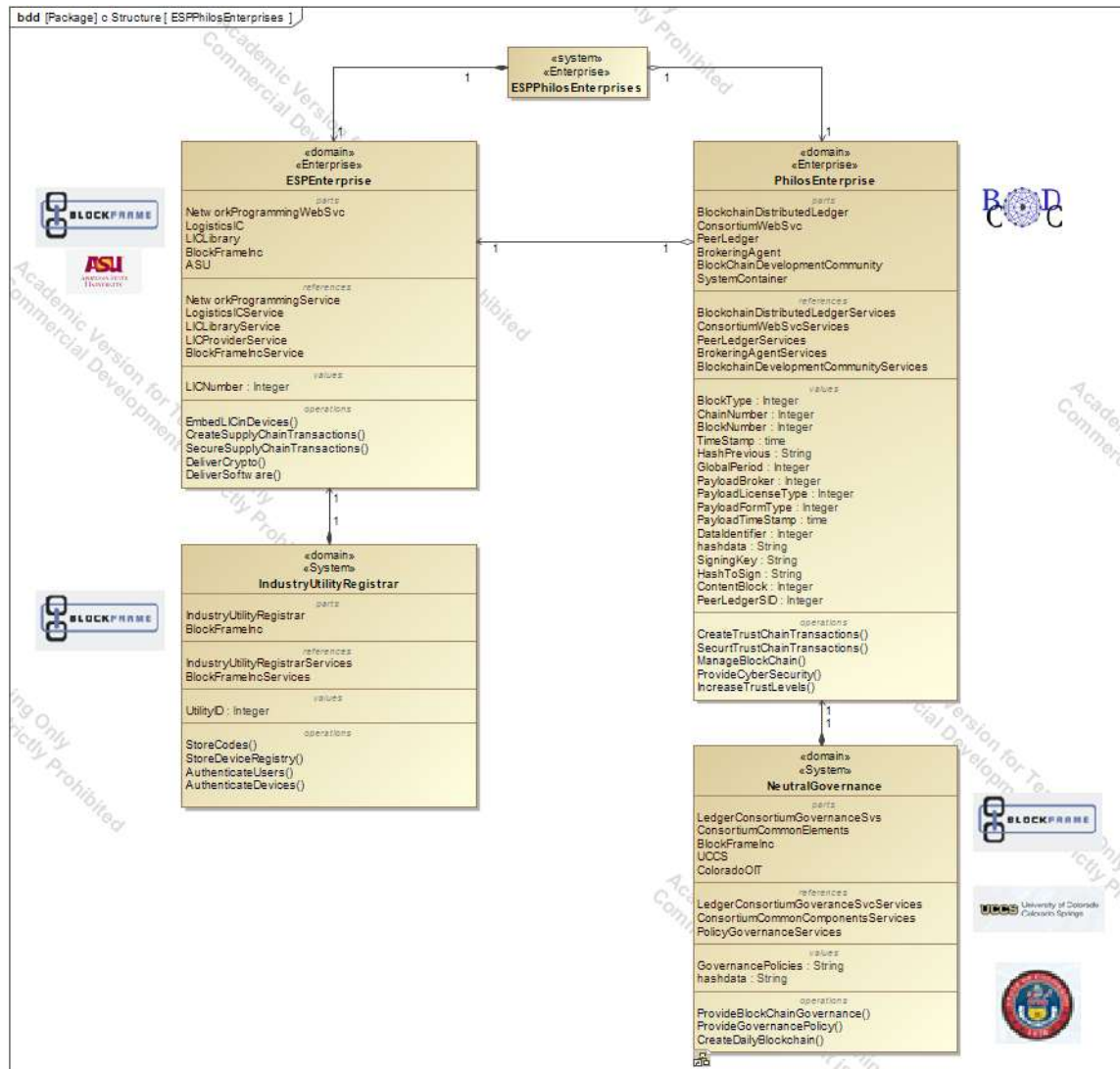


Figure C.1. Domain Diagrams for the Four ESP™/Philos™ Enterprise Domains

Use Case Diagram with Specifications

Figure C.2 is an example of four representative high-level Use Cases of the ESP™ and Philos™ Enterprises. It is different from the representation in Chapter 6 in that it is more focused on flows of provisioning of products from the ESP™ and Philos™ as Actors, and the Roles of 3rdPartyContractors and State entities rather than a flow originating from the User's point-of-view.

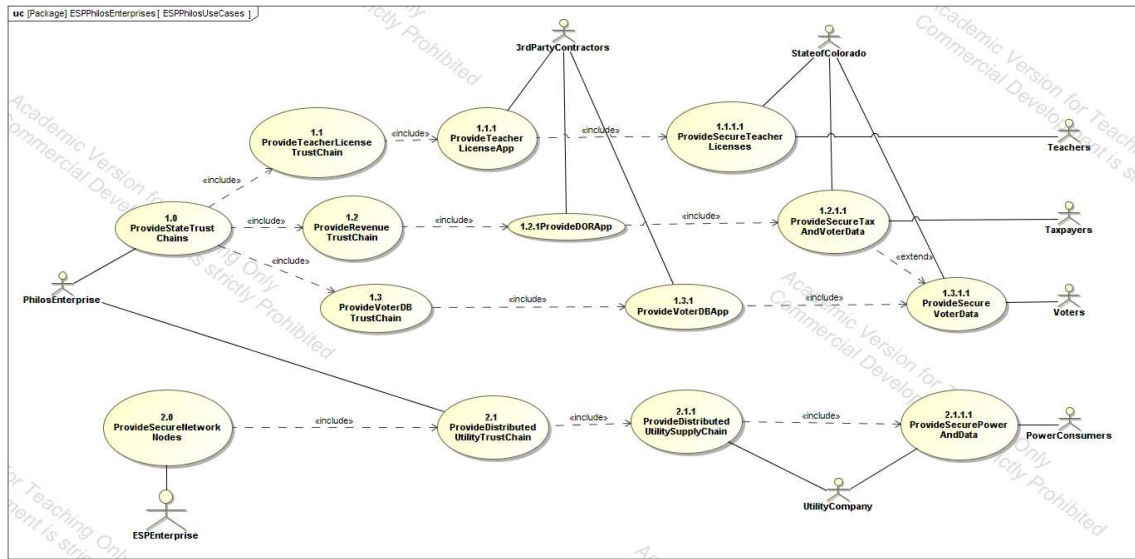


Figure C.2. Representational ESP™ and Philos™ Use Cases

Activity Diagrams

Two Activity Diagrams (AD) below show an end-to-end major mission thread. The first AD, Figure C.3, shows high-level Activities associated with the lower flow in the Use Case Diagram above. In that AD, it is postulated that the ESP™ Enterprise provisions Logistics Integrated Circuits into each appropriate Node of a Network; then the Philos™ Enterprise provisions a TrustChain blockchain Capability into the UtilityCompany's SupplyChain, thereby providing secure power and data with immutability and non-repudiability to the utility customers. The effect would be to prove that the Utility Company is providing accurate meter readings to their customers, and that the customers cannot cheat if they sell energy back to the Company. The Rake Specifies the 2.1.1.1 Behavior, "ProvideSecurePowerAndData," and when double-clicked in the Cameo tool displays that AD, which is shown in Figure C.4.

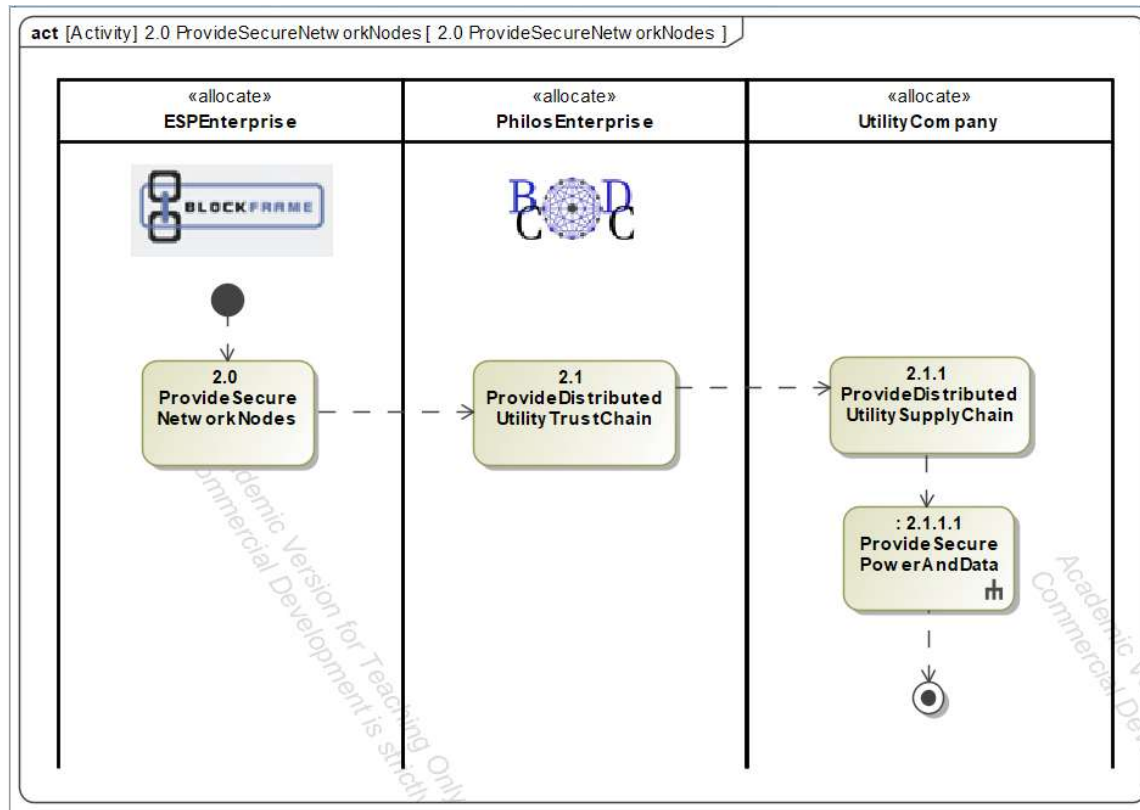
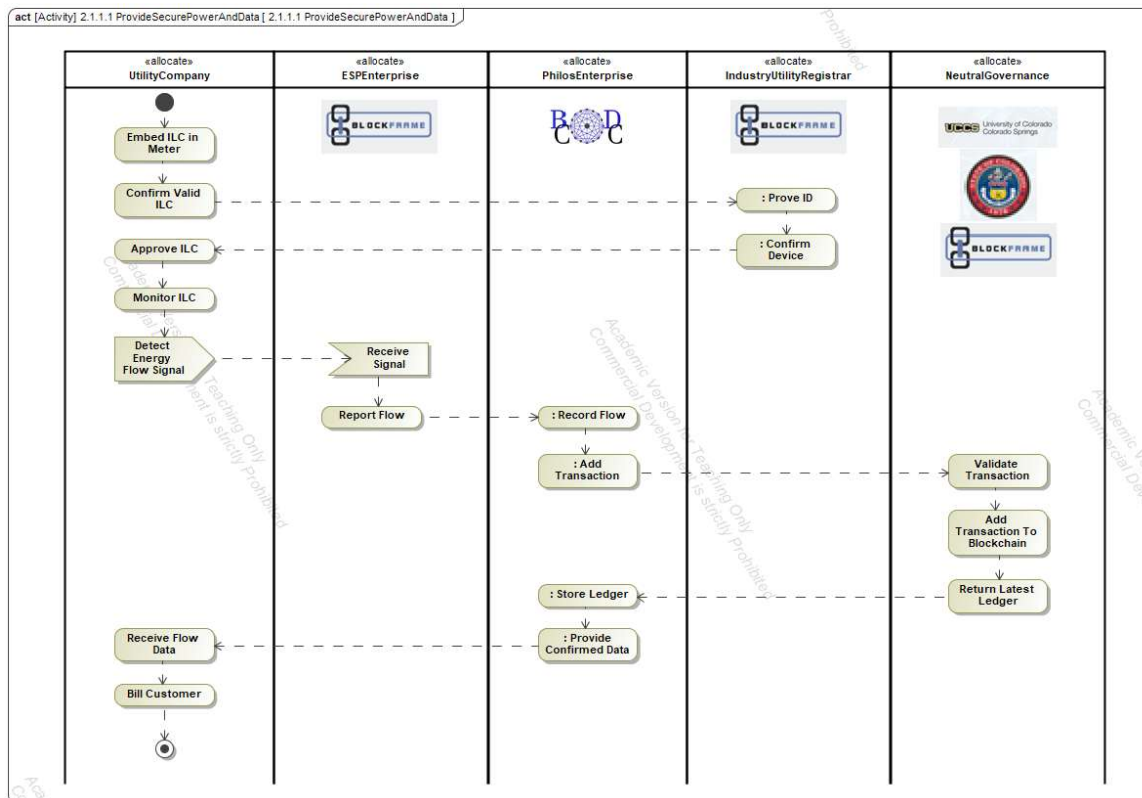


Figure C.3. Representational ESPTM and PhilosTM High-Level Activity Diagram of the Decomposition of Behavior 2.0 “ProvideSecureNetworkNodes”



C.4. Representative ESPTM and PhilosTM Detailed AD of Behavior 2.1.1.1 “ProvideSecurePowerAndData”

Figure

Conceptual Data Model (CDM)

Figure C.5. shows a representative Conceptual Data Model with Foundation Data Classes for Philos™. The blocks show the primary data entities discovered in the course of requirements analysis and modeling, identifies a few Foundation Classes, and creates BDDs for them.

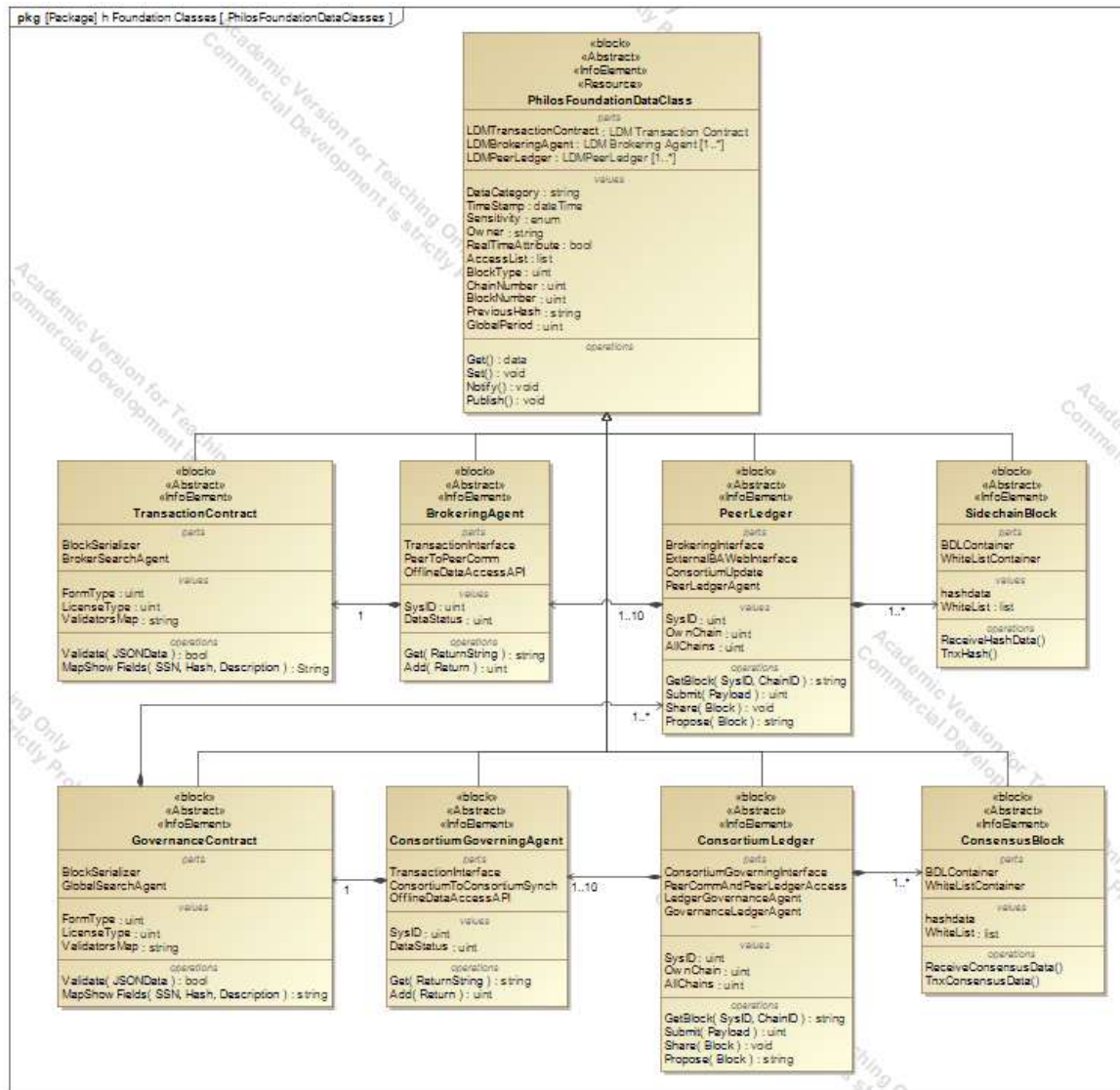


Figure C.5. Representative Conceptual Data Model for Philos™

Services Taxonomy

Table C.3 shows a Services Taxonomy using an Excel Spreadsheet.

Table C.3. Domain Services Taxonomy Among Blocks

Block:	Services Provided:	Service Type:	Interface:	Provides Service To:
<<block.Domain>> ESPEnterprise	NetworkProgrammingWebSvc	External	Yes	<<block.Domain>> ESPEnterprise
	LogisticsIC	External	Yes	<<block.Domain>> ESPEnterprise
	LICLibrary	External	Yes	LogisticsIC
	LICProvider	Internal	No	LogisticsIC
	BlockFrameInc	Internal	No	LICLibrary
<<block.Domain>> PhilosEnterprise	BlockchainDistributedLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	ConsortiumWebService	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises; Neutral Governance
	PeerLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	BrokeringAgent	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises
	BlockchainDevelopmentConsortium	Internal	No	<<block.Domain>> PhilosEnterprise
<<block.Domain>> NeutralGovernance	LedgerConsortiumGovernanceSvc	External	Yes	<<block.Domain>> PhilosEnterprise
	ConsortiumCommonComponents	Internal	No	Internal
	PolicyGovernance	Internal	No	Internal
<<block.Domain>> IndustryIdentificationRegistry	IndustryIDRegistry	External	Yes	<<block.Domain>> ESPEnterprise
	BlockFrameInc	External	Yes	<<block.Domain>> IndustryIdentificationRegistry

Figure C.6 uses *references* compartments as surrogates to list Services within the BDD, specifically showing each element as a Service.

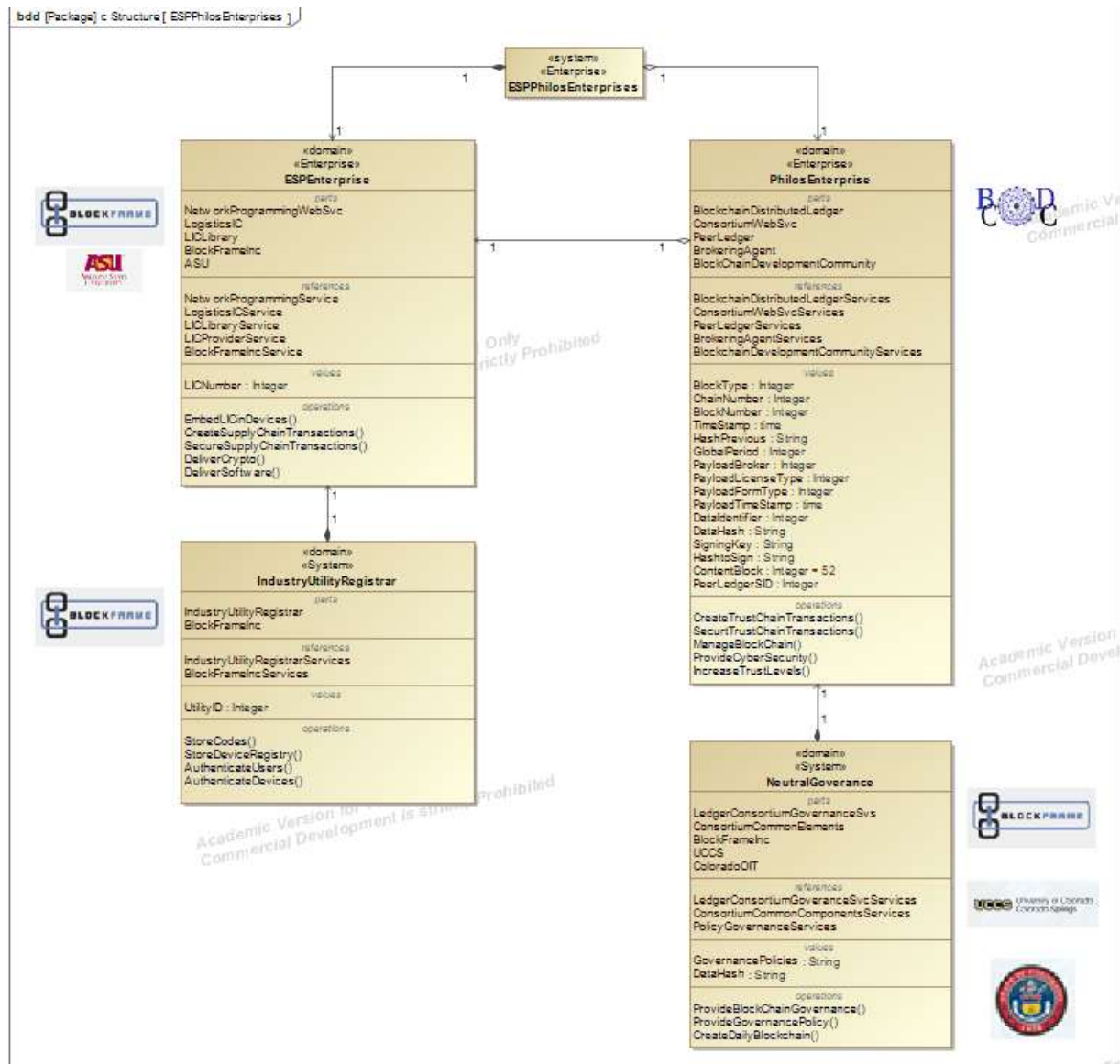


Figure C.6. BDD with Added Services Taxonomy for ESP™ and Philos™ Enterprises

Contextual Perspective

Figure C.7 shows a basic Logical/Functional Viewpoint (LV) structural decomposition of the Philos™ NeutralGovernance Domain/System. Diagrams decompose the Domain to the level of Blocks/Parts, and include Attributes/Values and Operations for Blocks

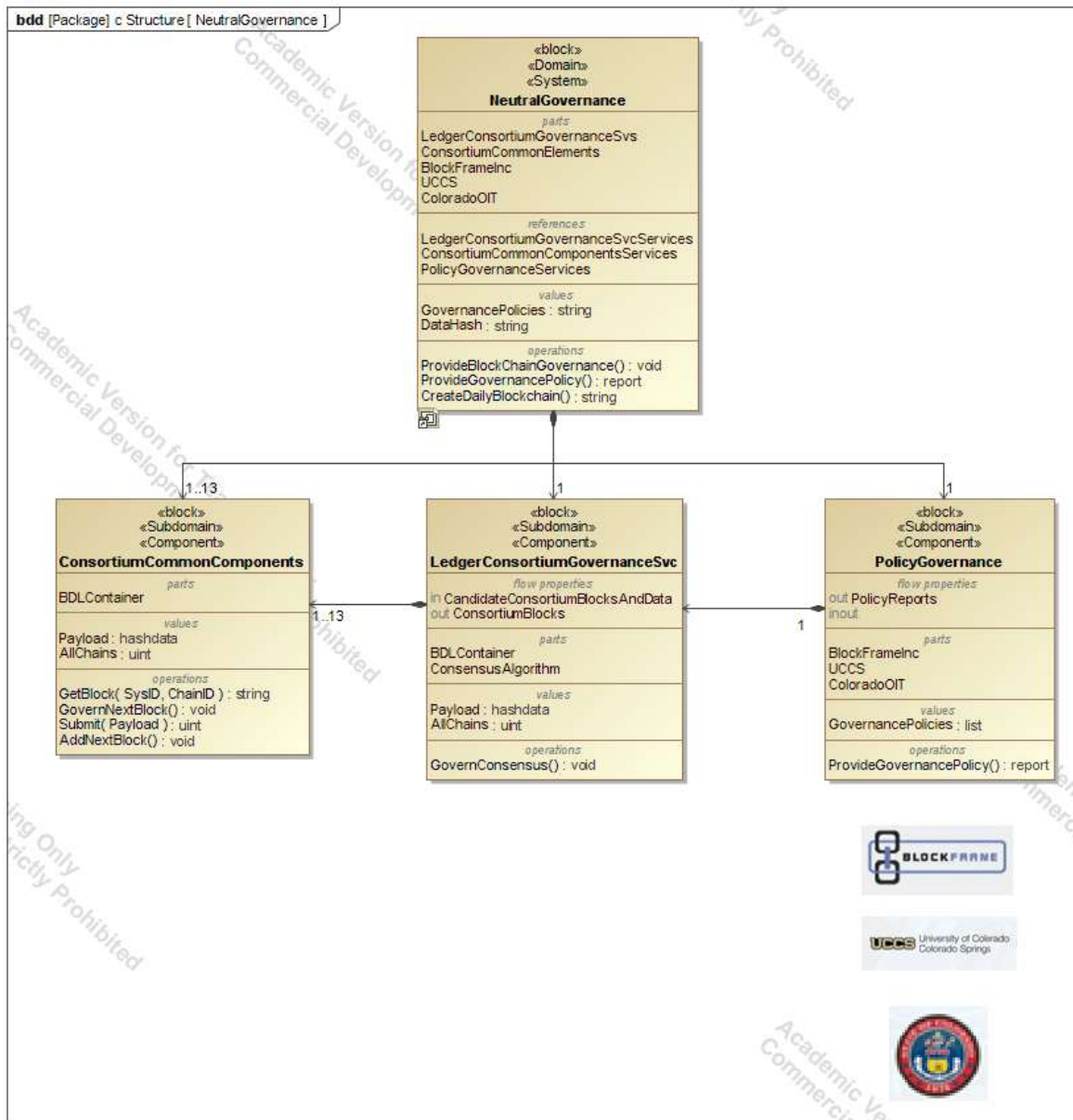


Figure C.7. BDD of the Philos™ NeutralGovernance Domain/System

Figure C.8 shows an IBD decomposition of the NeutralGovernance with internal information flows.

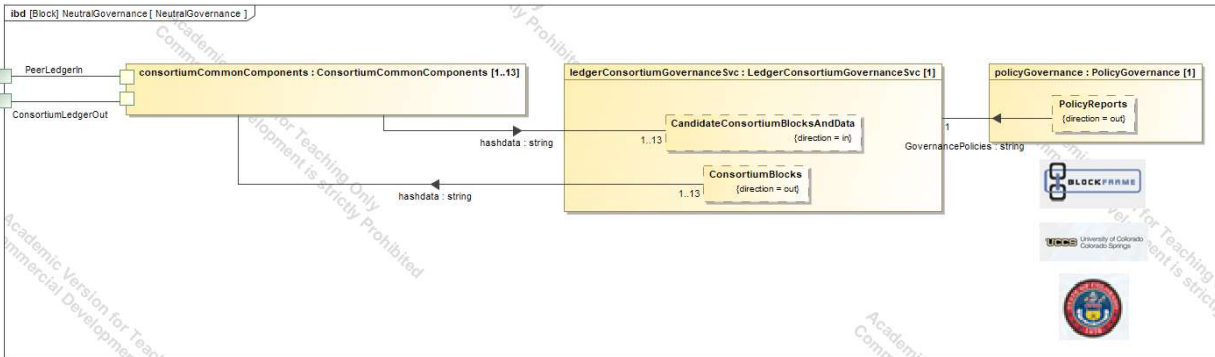


Figure C.8. IBD of the Philos™ NeutralGovernance Domain/System

Stateful behavior

Figure C.9 shows some stateful behaviors using a State Machine Diagram (Statechart) for provisioning cyphers into ESP™ LogisticsIntegratedCircuits (LICs).

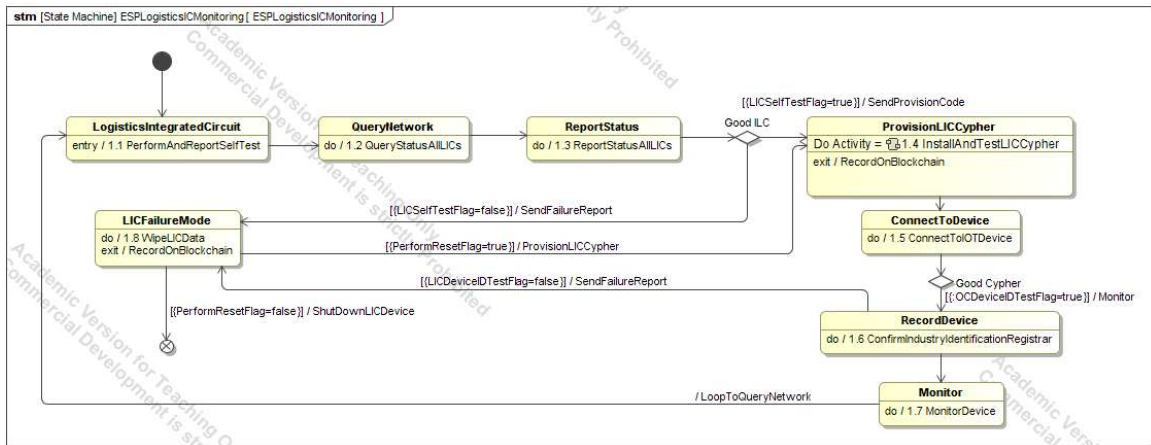


Figure C.9. State Machine Diagram for Provisioning Cyphers into LICs

Non-stateful behavior¹⁸⁶

Figure C.10 shows a Sequence Diagram to summarize non-stateful behavior of a group of Blocks using a Sequence Diagram; includes timing intervals for several parts of the behavior and illustrates the Philos™ blockchain consensus process.

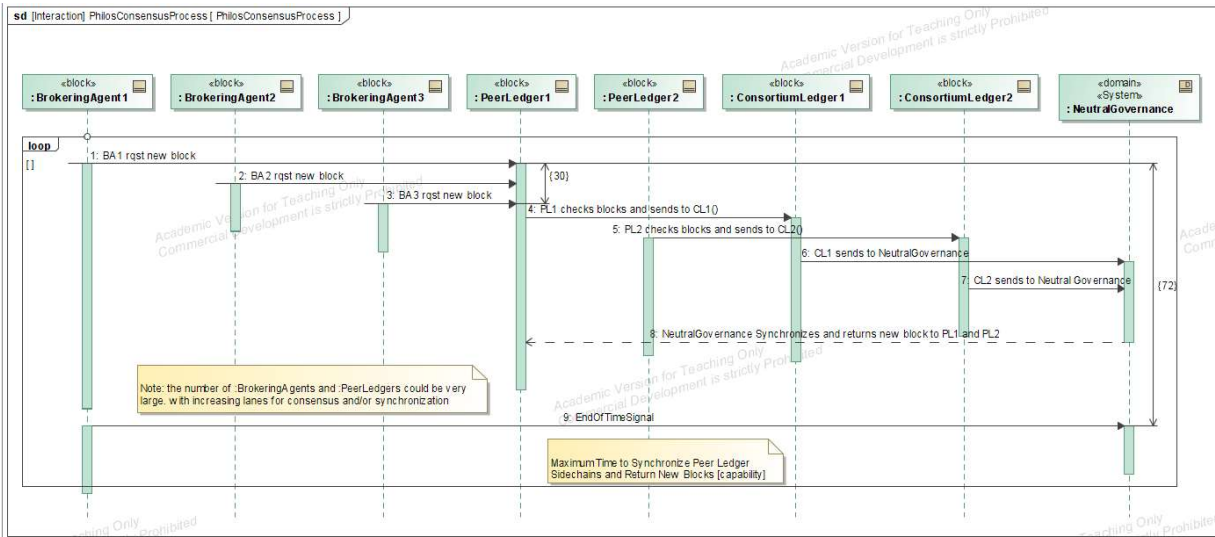


Figure C.10. Sequence Diagram for Philos™ Blockchain Synchronization

¹⁸⁶ Borky, Dr John M., and Dr Thomas H. Bradley. *Effective Model-Based Systems Engineering*. 1st ed. Colorado State University: Springer, 2019.2019. Accessed June 22, 2022. <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

Logical Data Model

As an example of a Logical Data Model, three Foundation Classes from the Conceptual Data Model are specialized to a set of System Classes, actually modeled as Blocks. Attributes, and Values, and Operations are then added. Figure C.11 specializes the Philos™ FoundationDataClass into a Logical Data Model.

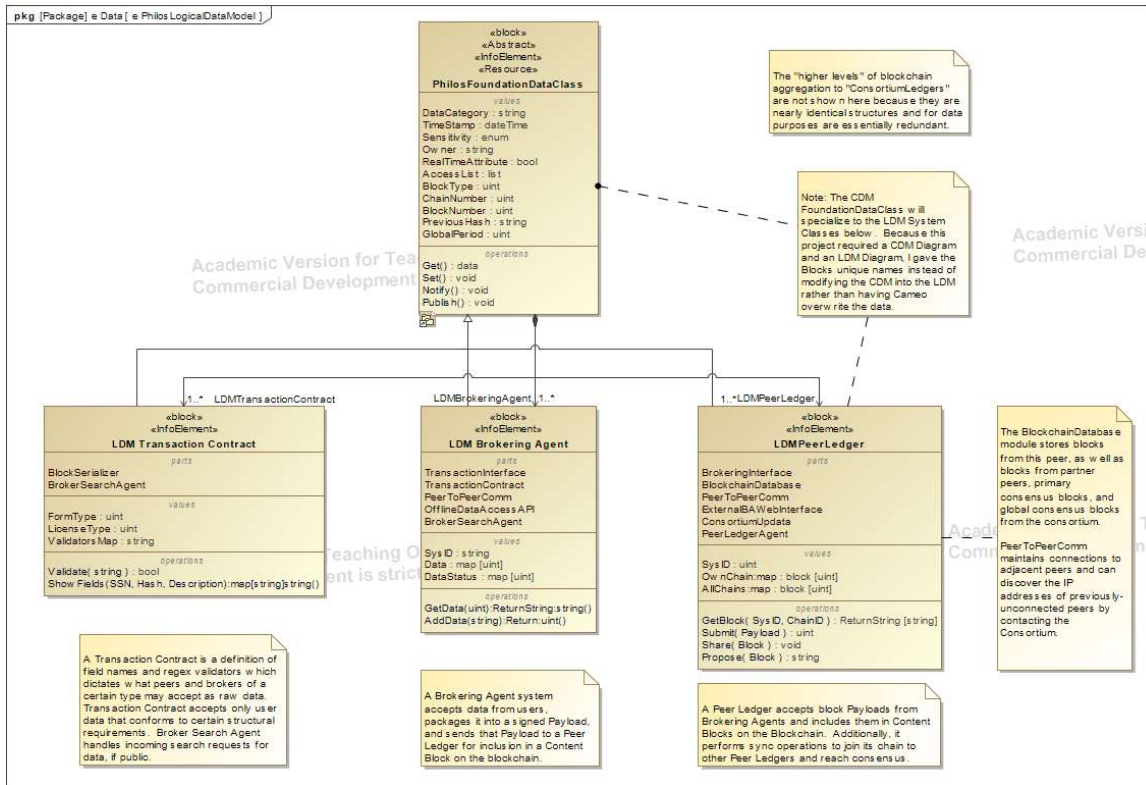


Figure C.11. Philos™ Logical Data Model

Services

A Domain Service is decomposed down the level of Block Operations and allocated to the appropriate Blocks. Figure C.12 shows a Package Diagram of ESP™ and Philos™ Service Architectures. Within the model, clicking on each icon will connect to BDDs for each <<Block>> <<Domain>> <<Services>> Diagram.

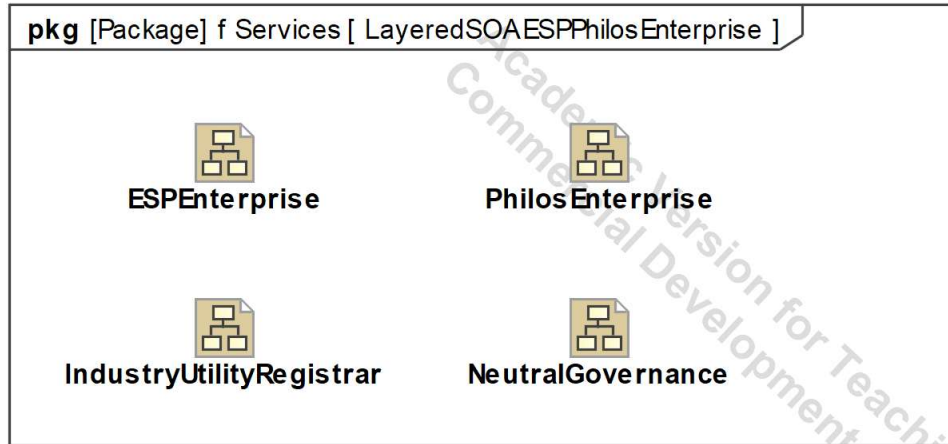


Figure C.12. Layered Package of ESP™ and Philos™ Service Architectures

Figures C.13, C.14, C.15, and C.16 are BDDs for Services for the ESPEnterprise™, PhilosEnterprise™, IndustryUtilityRegistrar™, and NeutralGovernance Domains, respectively.

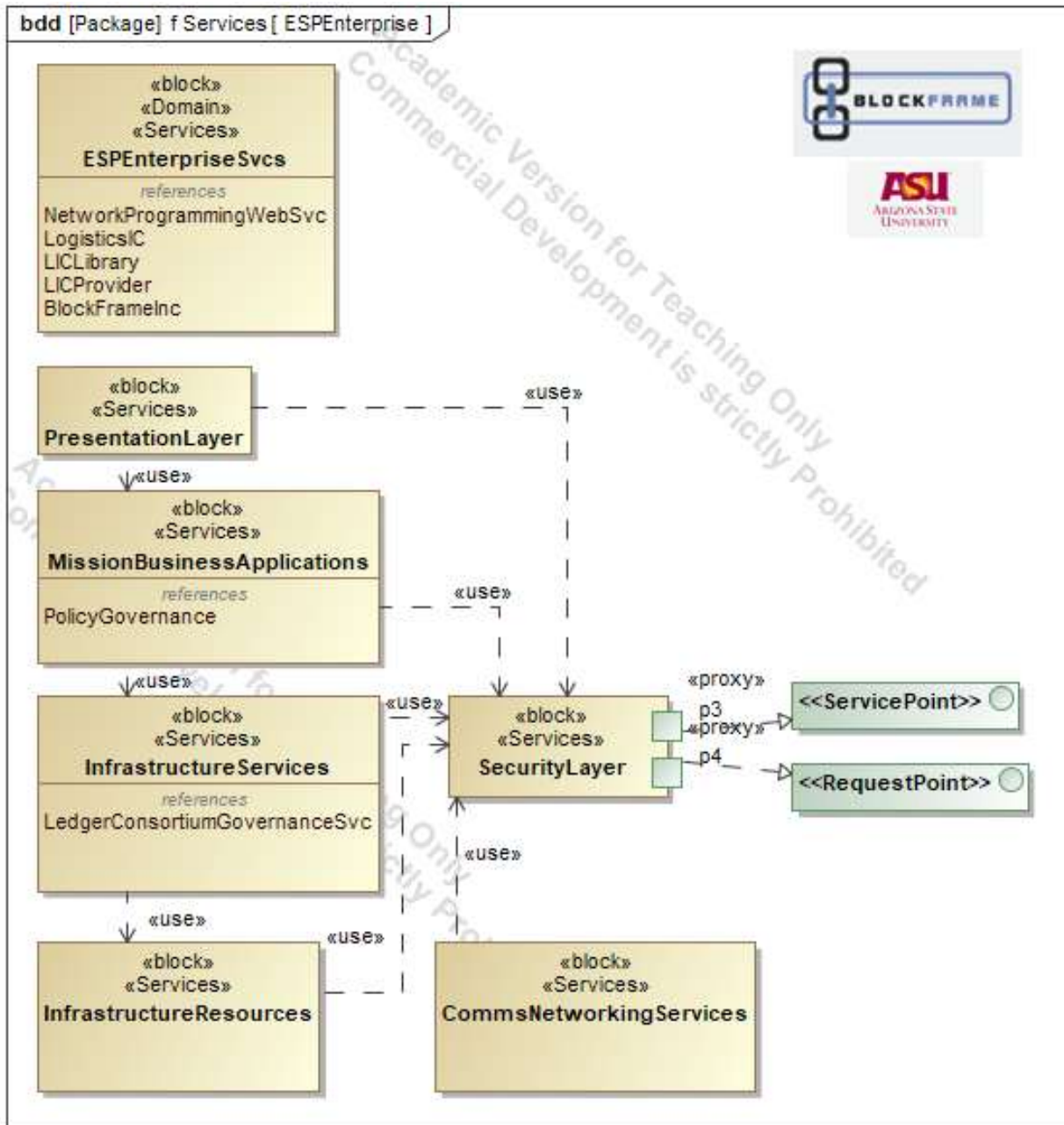


Figure C.13. Layered Package of the ESPEnterprise™ Service Architecture

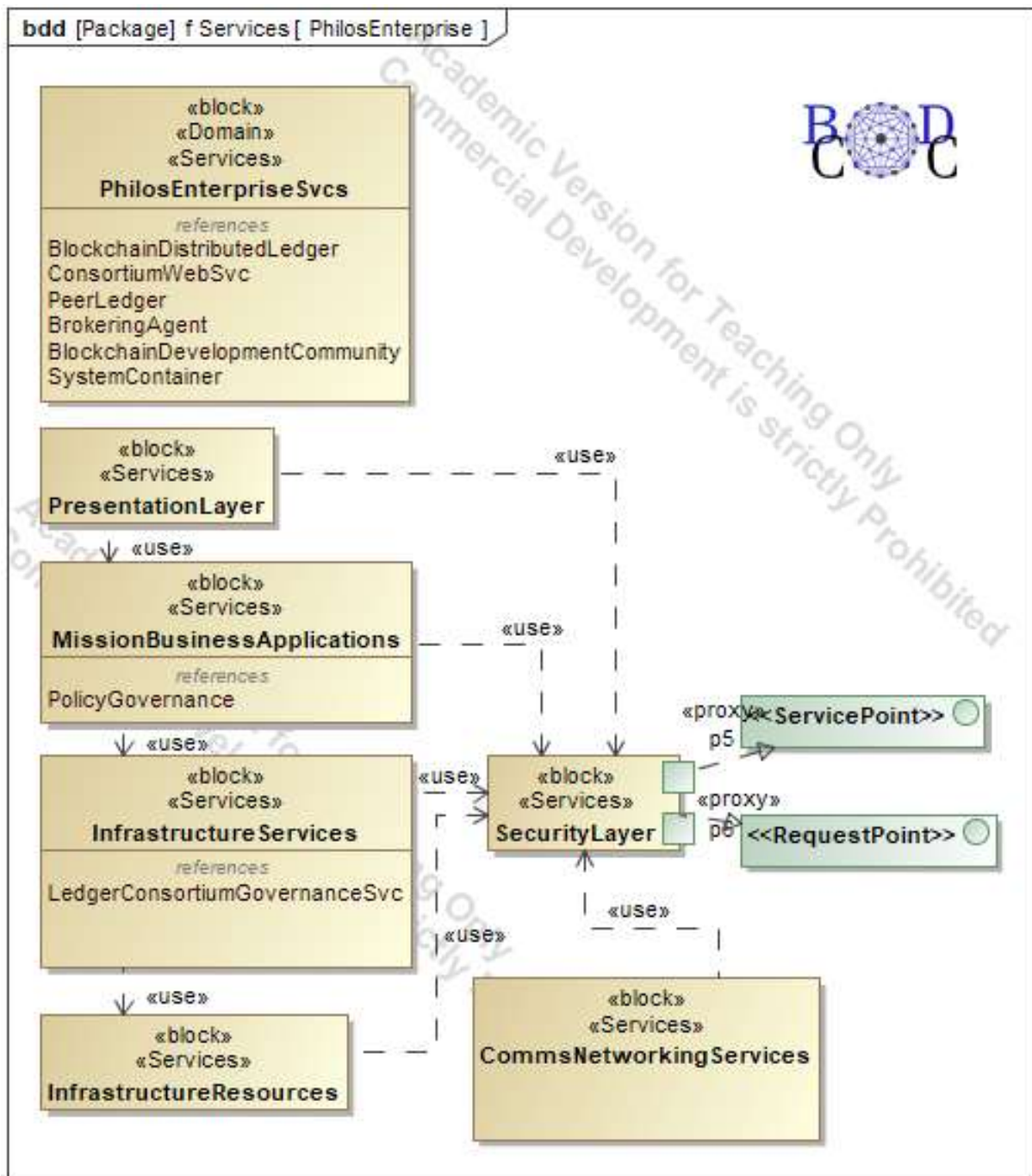


Figure C.14. Layered Package of the Philos™ Service Architecture

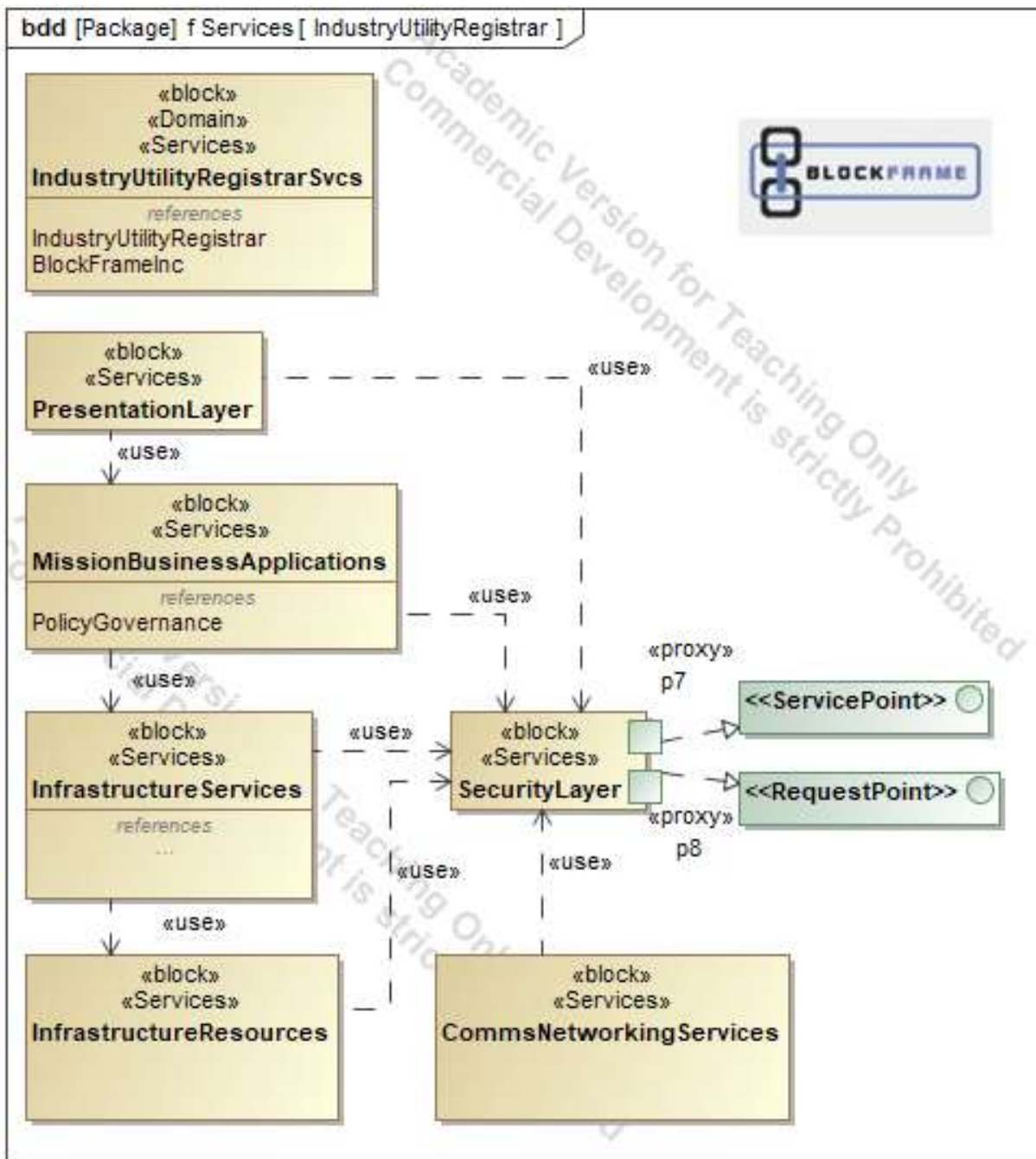


Figure C.15. Layered Package of the IndustryUtilityRegistrar Service Architecture

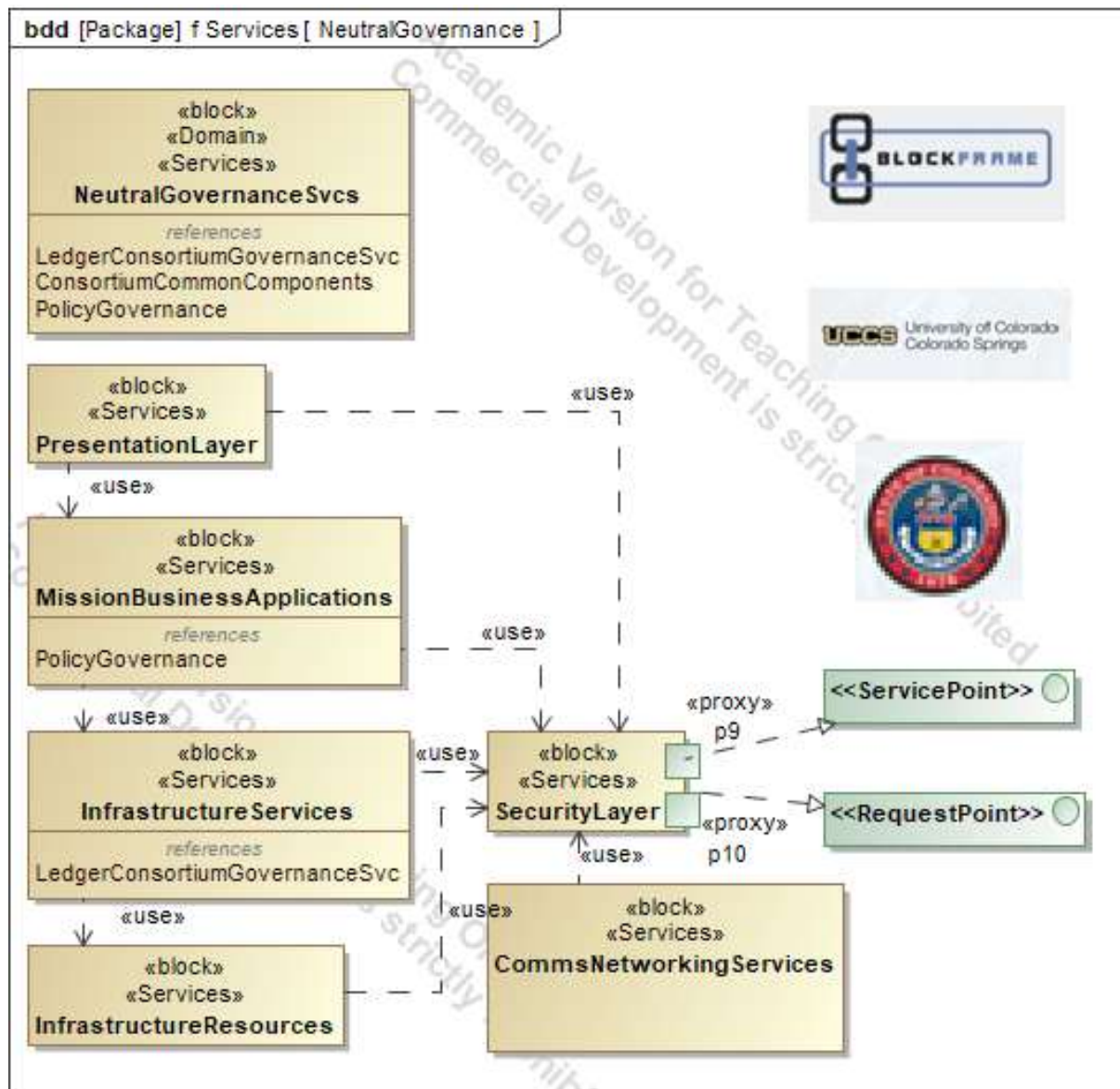


Figure C.16. Layered Package of the NeutralGovernance Service Architecture

Expanded Services Taxonomy from the OV with additional columns for this content

Table C.3 showed the Domain Services Taxonomy in tabular form. Figures C.13, C.14, C.15, and C.16 graphically show the locations within each Domain and Structural BDD, and within each SystemContainer variety. Within the ESP™ and Philos™ Enterprise Systems, almost all functional activities should occur within components of SystemContainers, which are intended to be distributed to highly-dispersed locations, and to take their Service provision to each location. Since all the Services BDDs exist within the Services Folder, they are separate from the “traditional” Structural and elements of the model, and Requirements or other activities can be assigned to them separately or laterally. This represents an “expansion” of Services into a different process. By creating a separate

Services architecture within the model, some Requirements could be satisfied by the “traditional” Structure, and some could be satisfied by the Services Structure. Modeling Behaviors can also apply to “traditional” Activities or be modeled separately within the “Services Domain.” Because not all “traditional” architectures involve Services, Those Activities could be separated and modeled separately, which might be more “transparent” on how they would be actually used. As these Enterprises evolve, it appears that some “traditional” components like physical security, software, or physical components might be better modeled from a “traditional” sense, but many of the business processes may really develop better as Services, like “Trust as a Service,” quantifying a process to move from “Zero Trust” to “Full Trust,” applying those “Services” to either developing or existing platforms. A ZT to FT process could use Emergence Theory thinking to help quantify “stochastic uncertainty,” and attempt to move to more “deterministic certainty” by linking particular “Service Requirements” to the specific Components or Activities that satisfying them.

Document Block Specifications in the model

Figures C.5, C.13, C.14, C.15, and C.16 represent many Block Specifications, with some caveats. Within some “traditional” Specifications – like dimensionality of packet switching attributes, cryptological hashes, etc. – the Cameo tool does not seem to internally define those special attributes but does allow them to exist as documentation. Multi-dimensional parameters within “Operations” definitions might be more limited for things like multi-dimensional matrices, but for now are probably flexible enough at least for documentation. The downside might be that the capabilities of the Cameo tool might not be as flexible for “cross-walking” those definitions into other Cameo tool capabilities like animation, defining complex behaviors, control algorithms, etc., that could be very useful. Although it could be very useful to have a “Services” Compartment in BDDs, using the “Reference Compartment” to identify Services might work fine, as long as it is understood and used consistently.

Layered Architecture

This depicts a layered SOA architecture as a separate “layer” from the “traditional” Structural architecture, within the same model, but not necessarily within the same Diagrams. This might cause some confusion by having two BDDs or IBDs of the same system, but if they are used for different functional purposes, they may not.

Contextual Perspective Content

As shown in the OV, a Context folder contains comments under the Documentation sub-folder. It currently contains an example list of Non-Functional Requirements, and several folders with some extensive background information on this project that should cover contexts for all Viewpoints.

Figure C.10 showed a simple theoretical Sequence Diagram for Philos™ Blockchain Synchronization, for BrokeringAgents, PeerLedgers, ConsortiumLedgers, and NeutralGovernance components. It was also annotated with timing information. Figure C.17 below shows some more detailed information related to “an overall deadline to the individual steps of the process.” However, it only focuses on timing from PeerLedgers through NeutralGovernance. As a hypothetical timing scheme, it is assumed that all Peer Ledgers must Synchronize all of their Brokering Agent Sidechain transaction requests within XXX minutes, and the entire Philos™ Blockchain must finish YYY percent of consensus by governance standards on all Peer Ledger Sidechains in the Enterprise within 24 hours. Due to design sensitivities, the actual capability values of XXX and YYY are not revealed here. The new 24-hour Blockchain is then transmitted for distributed distribution back to all the Peer Ledgers in the Enterprise for storage.

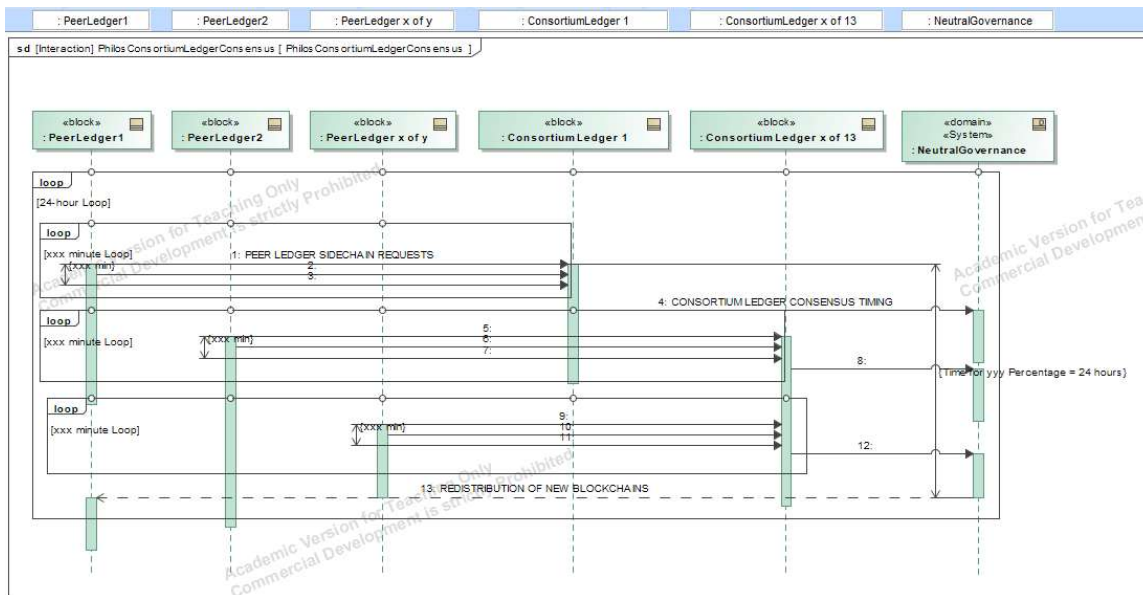


Figure C.17. Sequence Diagram of Philos™ Consortium Ledger Consensus Interactions

Table C.4 shows Domain Services Taxonomy Among Blocks with an initial Standards Profile with typical standards applied to various elements of the model. Table 4. Domain Services Taxonomy Among Blocks With Standards

Block:	Services Provided:	Service Type:	Interface:	Provides Service To:	Typical Standards:
<<block.Domain>> ESPEnterprise	NetworkProgrammingWebSvc	External	Yes	<<block.Domain>> ESPEnterprise	For Transport: PKI, TLS, https; AES-256
	LogisticsIC	External	Yes	<<block.Domain>> ESPEnterprise	For Transport: PKI, TLS, https; AES-256
	UCLibrary	External	Yes	LogisticsIC	For Transport: PKI, TLS, https; AES-256; For data/storage: XML, AES-256, SQL, Java
	UCProvider	Internal	No	LogisticsIC	Defined by 3rd Party contract/service agreement
	BlockFrameInc	Internal	No	UCLibrary	No parameters, consulting service only
<<block.Domain>> PhiloEnterprise	BlockchainDistributedLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	Largely custom design with few recognized standards, attempting compliance OMS and cyber best practices
	ConsortiumWebService	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises; Neutral Governance	For Transport: PKI, TLS, https; AES-256
	PeerLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	Largely custom design with few recognized standards, attempting compliance OMS and cyber best practices
	BrokerAgent	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	Largely custom design with few recognized standards, attempting compliance OMS and cyber best practices
	BlockchainDevelopmentConsortium	Internal	No	<<block.Domain>> PhiloEnterprise	No parameters, development service only
<<block.Domain>> NeutralGovernance	LedgerConsortiumGovernanceSvc	External	Yes	<<block.Domain>> PhiloEnterprise	For Transport: PKI, TLS, https; AES-256
	ConsortiumCommonComponents	Internal	No	Internal	No parameters, policy service only
	PolicyGovernance	Internal	No	Internal	No parameters, policy service only
<<block.Domain>> IndustryIdentificationRegistry	IndustryIDRegistry	External	Yes	<<block.Domain>> ESPEnterprise	For Transport: PKI, TLS, https; AES-256; For data/storage: XML, AES-256, SQL, Java
	BlockFrameInc	External	Yes	<<block.Domain>> IndustryIdentificationRegistry	No parameters, consulting service only

Table C.4 and Figures C.16, C.17, C.18, and C.19 show specific Services allocated to Blocks. Table 5 shows discrete numerical parameters that would be used for items like Systems IDs (SysID), Certificate Numbers, Company IDs, hashdata, Device IDs, etc. Although these may guide Services, they would be almost exclusively using robotic APIs, with little or no human intervention after installation, and data formats would be self-defined by the entities using the Services. “Parameters” invoking the Services for items like CryptoKeys, hashdata, and DataStorage would largely be huge binary numbers that would appear pseudo-random and might defy categorization as “parameters” in a management sense, and the “parameters” are probably best described as unknown hexadecimal “numbers.” If they can’t be “defined” they are also probably more secure.

Table C.5. Domain Services Taxonomy Among Blocks Typical Parameters

Block:	Services Provided:	Service Type:	Interface:	Provides Service To:	Typical Parameters:
<<block.Domain>> ESPEnterprise	NetworkProgrammingWebSvc	External	Yes	<<block.Domain>> ESPEnterprise	CryptoKey, WebSvcAddress, SysID
	LogisticsIC	External	Yes	<<block.Domain>> ESPEnterprise	UCSysID, DeviceID, CertificateNumber
	UCLibrary	External	Yes	LogisticsIC	DeviceID, CompanyID, CertificateNumber
	UCProvider	Internal	No	LogisticsIC	UCRootKit, ProviderID, CertificateNumber
	BlockFrameInc	Internal	No	UCLibrary	No parameters, consulting service only
<<block.Domain>> PhiloEnterprise	BlockchainDistributedLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	SystemsContainer, hashdata, DataStorage
	ConsortiumWebService	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises; Neutral Governance	CryptoKey, WebSvcAddress, SysID
	PeerLedger	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	SystemsContainer, hashdata, DataStorage
	BrokerAgent	External/Internal	Yes	<<block.Domain>> ESPPhilosEnterprises	SystemsContainer, hashdata, DataStorage
	BlockchainDevelopmentConsortium	Internal	No	<<block.Domain>> PhiloEnterprise	No parameters, development service only
<<block.Domain>> NeutralGovernance	LedgerConsortiumGovernanceSvc	External	Yes	<<block.Domain>> PhiloEnterprise	SystemsContainer, hashdata, BlockchainCreationEngine
	ConsortiumCommonComponents	Internal	No	Internal	SystemsContainer, hashdata, DataStorage
	PolicyGovernance	Internal	No	Internal	No parameters, policy service only
<<block.Domain>> IndustryIdentificationRegistry	IndustryIDRegistry	External	Yes	<<block.Domain>> ESPEnterprise	DeviceID, CompanyID, CertificateNumber
	BlockFrameInc	External	Yes	<<block.Domain>> IndustryIdentificationRegistry	No parameters, consulting service only

Allocating at Requirements to Blocks

Table C.1 showed that all project Functional Requirements were entered into a Requirements Table. Figure C.18 shows that Requirement R1.4 “The System shall Manage Neutral Governance,” is <<Satisfied>> by :NeutralGovernance:PolicyGovernance, shown in Figure C.19.

R1.4

Properties: Standard

Requirement	
Name	R1.4
Id	
Text	The system shall Manage Neutral Governance
Source	
Verify Method	
Risk	
Traceability	
Owner	15 ESPPhilosRequirements [a Requirements]
Refined By	
Traced To	
Satisfied By	GovernancePolicies : e Data::list [c Structure::NeutralGovernance::PolicyGovernance]
Master	
Derived	
Derived From	
Verified By	

Figure C.18. Requirement R1.4

policyGovernance

Properties: Standard

Part Property	
Name	policyGovernance
Type	PolicyGovernance [c Structure::NeutralGovernance]
Default Value	
Owned By	NeutralGovernance [c Structure::NeutralGovernance]
Applied Stereotype	PartProperty [Property] [MD Customization for SysML::additional_stereotypes]
Multiplicity	1
Is Read Only	☐ false
Is Static	☐ false
Aggregation	composite
Is Derived	☐ false
To Do	

Figure C.19. Requirement R1.4 Satisfied by :NeutralGovernance:PolicyGovernance

Figure C.20 shows that Requirement R1.2.17 “The System shall provide end-to-end transaction security for participating transaction users on supply chains,” is <<Satisfied>> by :PhilosEnterprise:SystemContainer, shown in Figure C.21.

R1.2.17

Properties: Standard

Requirement	
Name	R1.2.17
Id	
Text	The system shall provide end-to-end transaction security for participating transaction users on supply chains
Source	
Verify Method	
Risk	
Traceability	
Owner	15 ESPPhilosRequirements [a Requirements]
Refined By	
Traced To	
Satisfied By	SystemContainer [c Structure::PhilosEnterprise]
Master	
Derived	
Derived From	
Verified By	

Figure C.20. Requirement R1.2.17

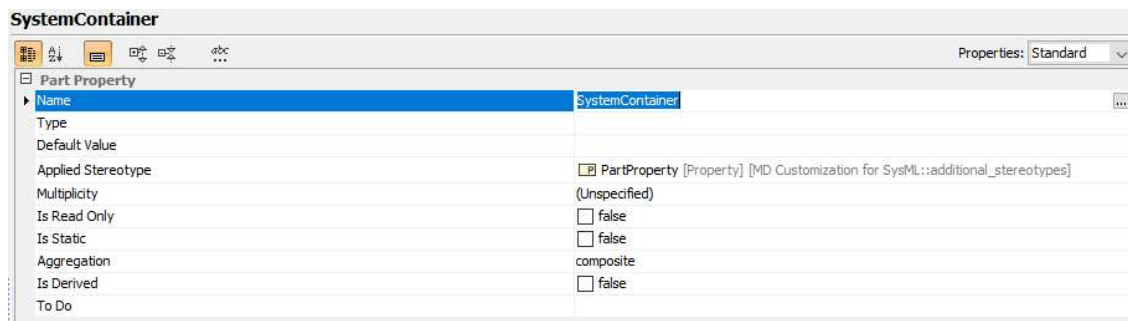


Figure C.21. Requirement R1.2.17 Satisfied by :PhilosEnterprise:SystemContainer.

Figure C.22 shows that Requirement R1.1.17 “The System shall manage the use and delivery of cryptographic tools and secret codes,” is <<Satisfied>> by ESPEnterprise:NetworkProgrammingWebSvc, shown in Figure C.23.

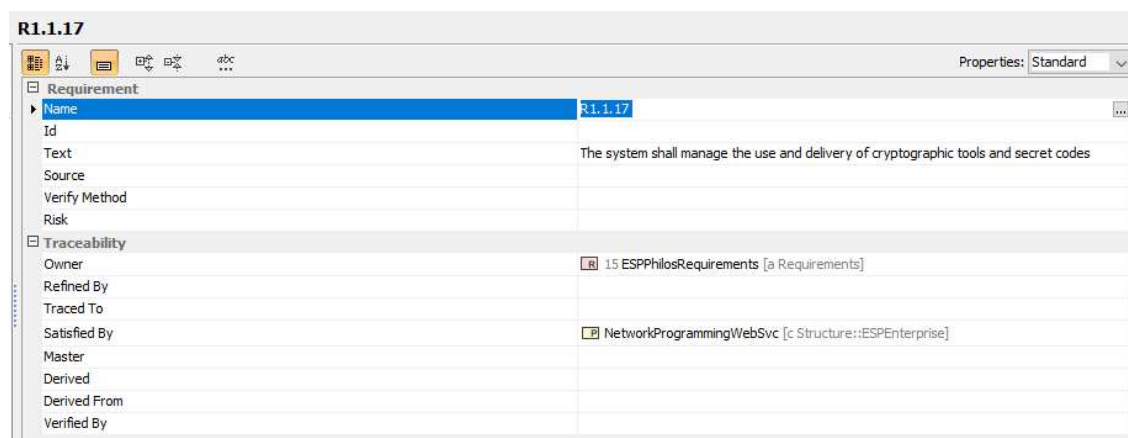


Figure C.22. Requirement R1.1.17

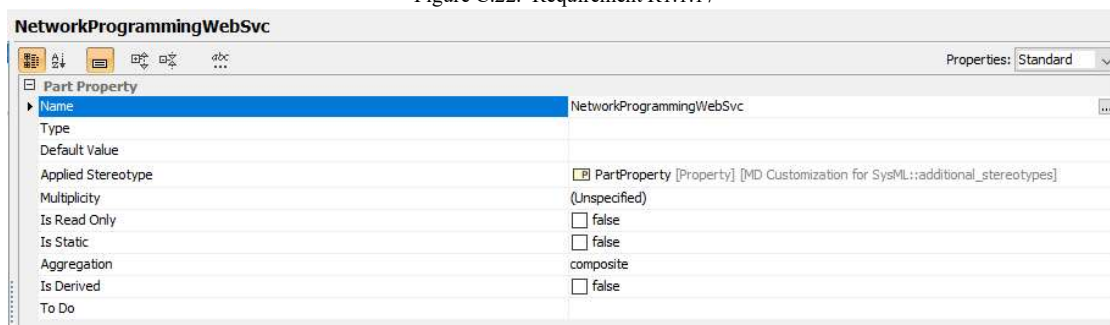


Figure C.23. Requirement R1.1.17 Satisfied by :ESPEnterprise:NetworkProgrammingWebSvc

Specialized Stereotypes

The ESPEnterprise™, the PhilosEnterprise™, and their Components are elusive to classify because the various elements have unique characteristics that in some cases can be simultaneously described by different Stereotypes. Figure C.24 shows an ESP™ Philos™ Profile with “Non-Cameo” Stereotypes added to more accurately describe different elements. Figure C.25 shows one of several BDDs that use the User-defined Stereotypes.

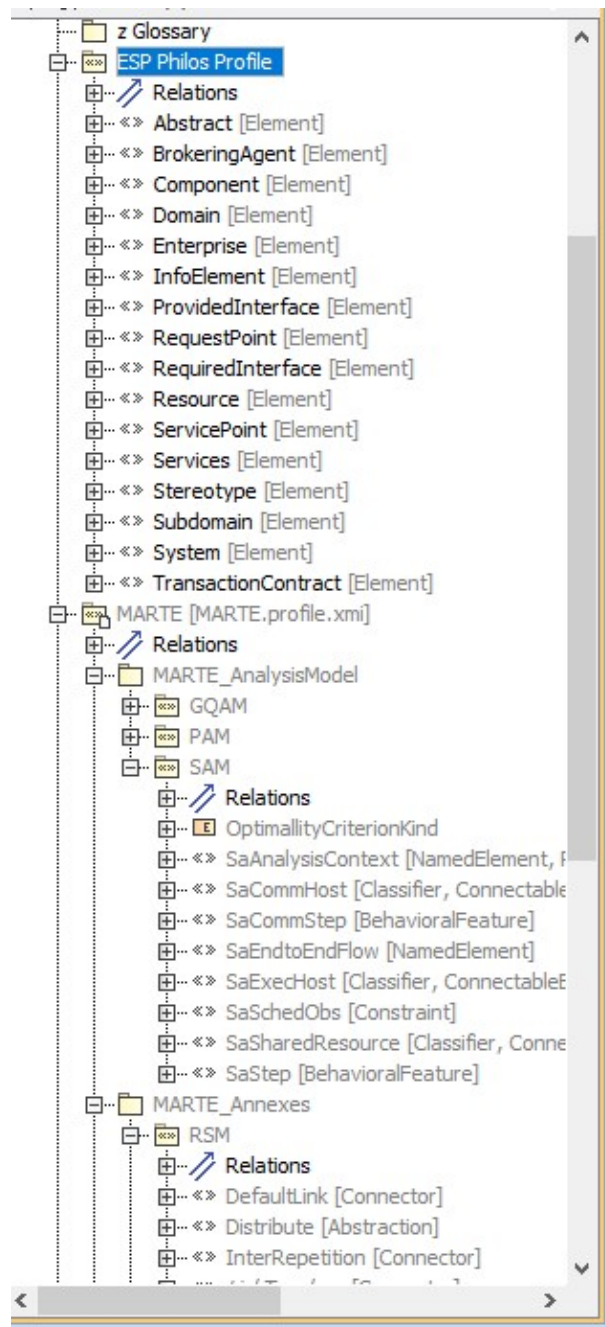


Figure C.24. ESP™/Philos™ Profile with new Stereotypes, including some MARTE examples.



Figure C.25. Example of an ESP™ Block using added User-defined Stereotypes

Constraint Blocks define mathematical relationships among Values

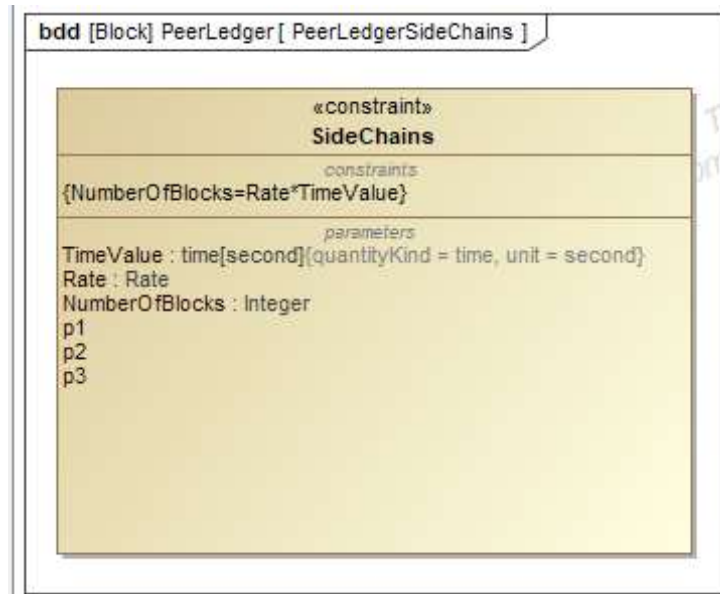


Figure C.26. Non-Functional Constraint Block BDD

A Parametric Diagram created using Constraints

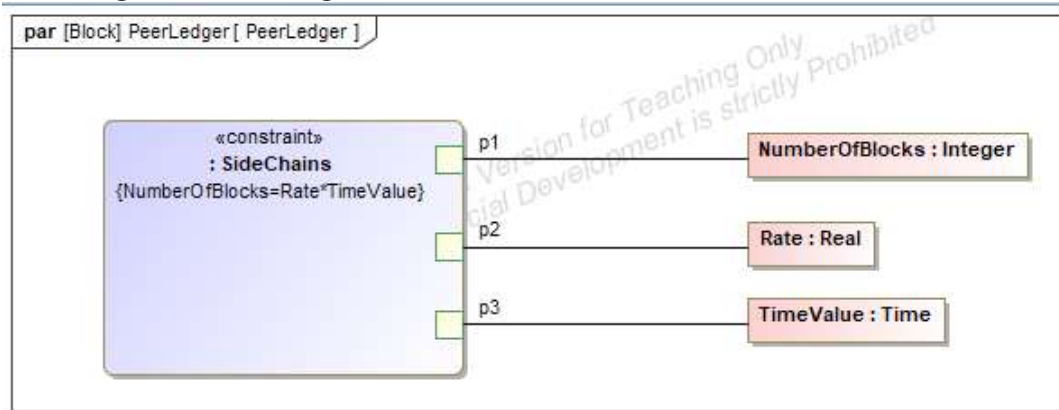


Figure C.27. Non-Functional Constraint Properties and Parameters

Block Instance Specifications in the Physical Viewpoint.

Figure C.28 represents three Block Instances within the :PhilosBrokeringAgent. It is likely that this entire structure would be realized in software code, so there may be nothing “physically tangible” in this section of the PV. The <<proxy>> Port would not connect directly to any external components but might abstractly be “connected” with the SystemContainer SecurityLayer to pass all input-output data to other components within the SystemContainer, or through the SystemContainer to any external Nodes, as shown in Figures C.29 and C.30. Since it is likely that much of this architecture could be instantiated strictly within software, the “ports” could be instantiated to various virtual entities, with “virtual firewalls,” that protect from cyber threats, but might not be “ports” in the functional sense.

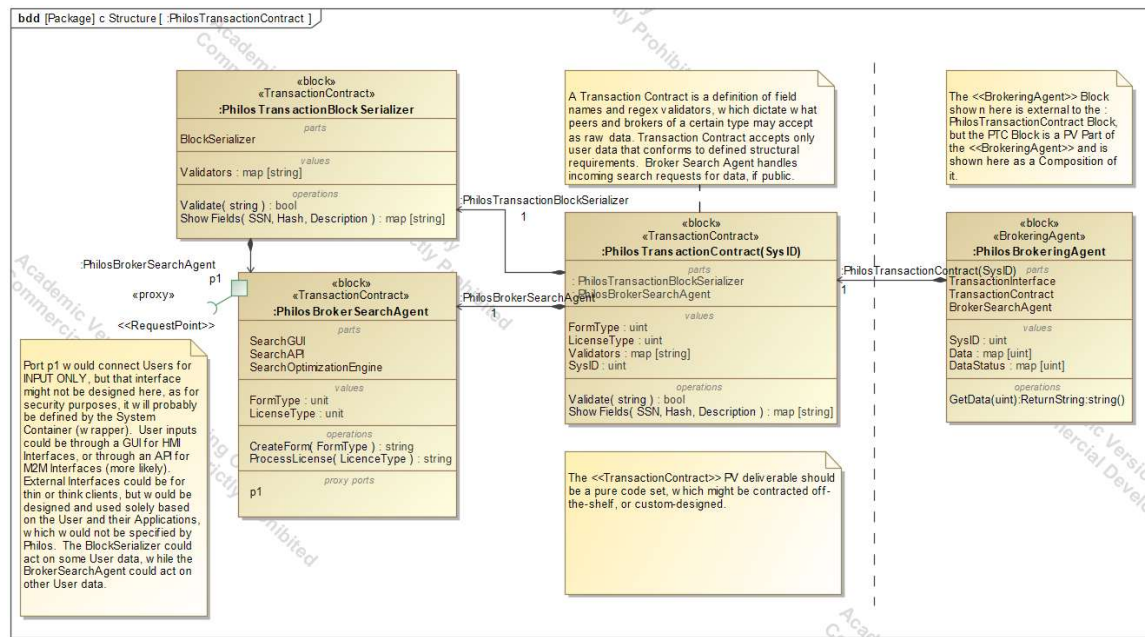


Figure C.28. Examples of Philos™ Block Instances within the :PhilosBrokeringAgent
Interface Blocks used to type Proxy Ports

To create an Interface Block and to type Proxy Ports, the best and perhaps the *only* true <<full>> Interfaces in either the ESP™ or the Philos™ Enterprises all connect through functionally-identical SystemContainers – of which there could be millions of instantiations. The vision of the Systems Engineering model is intended to eventually be a digital twin of the actual Enterprises. As such, I identified the SystemContainer as a <<stereotype>> in the diagrams since in the future, what is described in the SystemContainer and its contents would potentially *be* the dynamic configuration of common elements of the entire Enterprise. Figures C.27 and C.28 show a BDD and an IBD of the SystemContainer. Note that the multiplicities to the BrokeringAgent, the PeerLedger, and the ConsortiumLedger (0..1) should actually be accurate. Although in reality the SystemContainer would contain only one of the three, the “stereotypical” diagram has depicted all three because the SC would have to be compatible with any of the three, and modular enough to be used in any of the three roles.

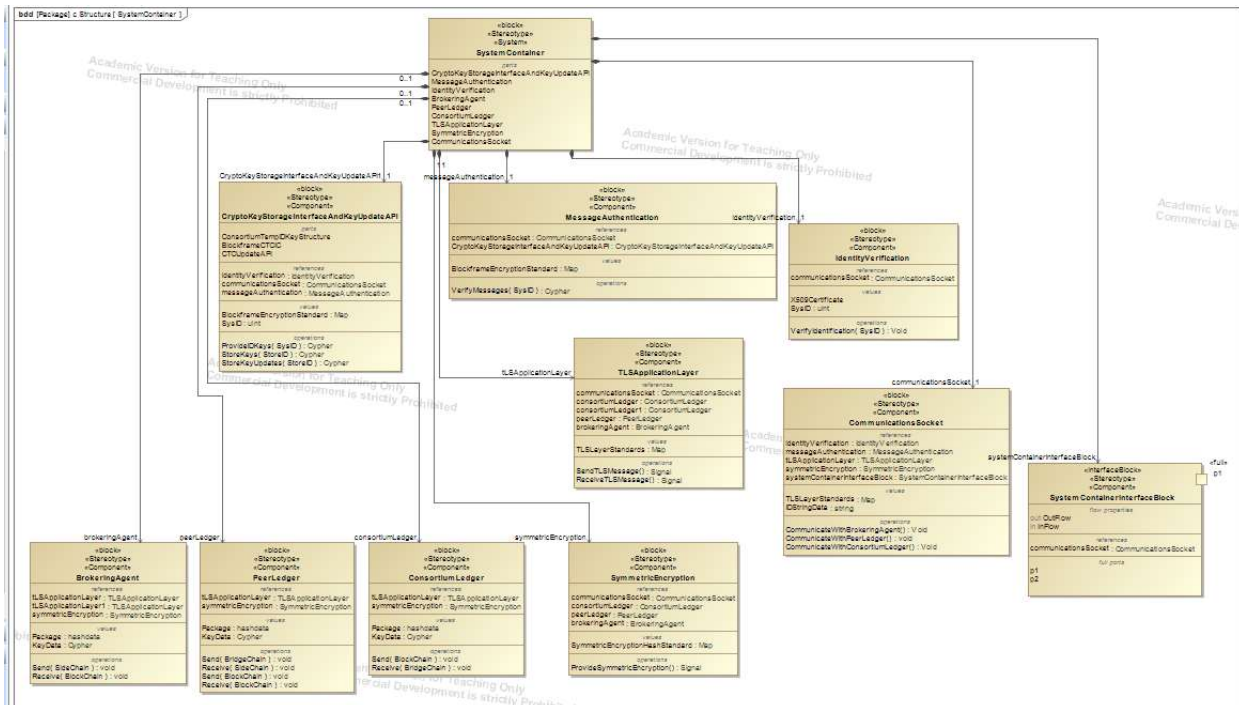


Figure C.29. BDD of SystemContainer

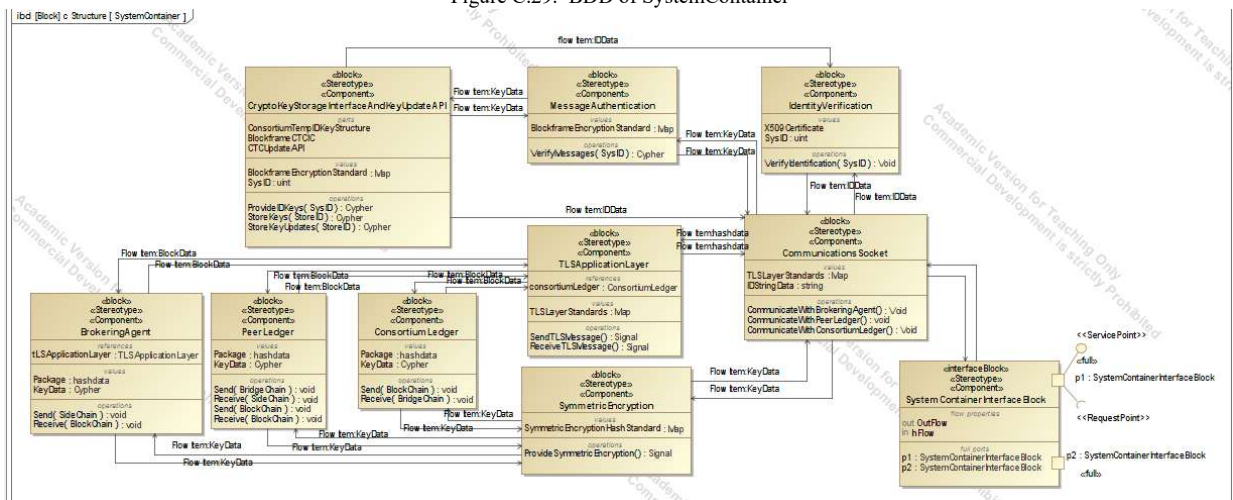


Figure C.30. IBD of SystemContainer

A <<proxy>> Port is shown in Figure C.28 to the :PhilosBrokerSearchAgent of the TransactionContract Block. As explained in the Note, this will probably not be a tangible physical Port, as it will probably be by a software code set. Physical connections will only be made through a SystemsContainer. In this instantiation, the Transaction Contract would be physically inside the SystemsContainer (if it is actually physical), or within code (if it is totally virtual), and would receive requests for Blockchain operations from Users only through a SystemsContainer SecurityLayer, TLS, firewalls, etc.

Despite the uninteresting number (probably two) of actual Port types in Systems Containers throughout the Enterprise architectures, other than in a common stereotypical Systems Container, Figures C.13, C.14, C.15, and C.16

in the “Services Architectures have perhaps more important definitions for Service Interfaces. Each <<ServicePoint>> and <<RequestPoint>> are represented as Interfaces on the SystemsContainer, which would then redirect all service requests to the appropriate Service-providing Component within each SystemContainer. Although this would probably not be a physical “Port,” it should help conceptualize the fact that all Services must transit a robust SecurityLayer that Users have to carefully consider and comply with.

Satisfying Defined Requirements

Chapter 5 examined Emergence Theory and assumed that Systems Engineering and SysML could be used to define a set of requirements and help prove whether those requirements can be mapped to a set of capabilities that satisfy those requirements. This section of Appendix C demonstrates in detail how that can be done using the BlockFrame® Satisfaction Matrix

Due to the large size and layout of the SysML requirements matrix, we will examine three tables. Table C.6 shows a list of previously defined Requirements. Table C.7 shows a summary list of all of the CMMC Requirements that were added. Table C.8 is a fragment of an example strip of screenshots of the ESP™/Philos™ Satisfy Requirement Matrix, showing ESP™/Philos™ Structural elements along the top, and Requirements along the left axis.

Within a Satisfy Requirement Matrix, it appears that every Requirement, including CMMC compliance Requirements, can be mapped to Enterprise capabilities, and to the ESP™/Philos™ capabilities that should satisfy each requirement. The Matrix is also an immediately useful design tool to match each engineering design team to the CMMC and other Requirements they would be expected to satisfy and provide objective goals to prove those levels of satisfaction have been realized.

Satisfaction of ESPTTM/PhilosTM Requirements

Table C.6. Summary List of Previously Defined ESPTTM/PhilosTM Requirements

RM.5.155	Analyze the effectiveness of security solutions at least annually to address anticipated risk to the system and the organization based on current and accumulated threat intelligence.
R1.1	The system shall provide an Eco-Secure Provisioning Framework for IoT Devices (Project Domain Nickname ESP)
R1.1.1	The system shall provide uniformity features for supply and trust chains
R1.1.2	The system shall be scalable to many use cases in large marketplaces across any industry
R1.1.3	The system shall be able connect to individual IoT devices through embedded logistics devices
R1.1.4	The system shall program mass distributed IoT devices in network
R1.1.5	The system shall provide cryptologically-secure protection for IoT devices
R1.1.6	The system shall provide cryptologically-secure protection for system components
R1.1.7	The system shall provide security management and control services at the organization/business level
R1.1.8	The system shall provide global services across jurisdictional boundaries
R1.1.9	The system shall provide verification and validation of user/actor/device identities
R1.1.10	The system shall operationally increase measurable trust levels among users (and reduce risk) over time
R1.1.11	The system shall use open architectures of modular components
R1.1.12	The system shall design components to accept expandable technologies
R1.1.13	The system shall maintain and provide standardized libraries for embedded logistics devices
R1.1.14	The system shall have the capability to prove the origin of data -- Proof-of-Origin POC
R1.1.15	The system shall coordinate, manage, and synchronize all operational ecosystem comms for Zero Trust
R1.1.16	The system shall have the ability to wrap legacy equipment by tools/APIs to support cyber security methods
R1.1.17	The system shall manage the use and delivery of cryptographic tools and secret codes
R1.1.18	The system shall assure the security of the handling and delivery of products
R1.1.19	The system shall conduct capabilities and security testing and evaluation
R1.1.20	The system shall (after initial stability) be economically self-sustaining by user fees
R1.1.21	The system shall provide education and training to developers and users
R1.1.22	The system shall manage cryptographic key management throughout the IoT life cycle
R1.1.23	The system shall detect, authenticate, and update any IoT device upon linking to the trust chain
R1.1.24	The system shall manage the secure distribution and installation of softw are to users
R1.1.25	The system shall facilitate media control or digital rights management (DRM)
R1.1.26	The system shall manage "New Supply Chain Industry Creation" initiatives and modifications
R1.2	The system shall provide Records, Documents, and Transactions management (Project Domain Nickname Philos)
R1.2.1	The system shall provide immutable records of data transactions
R1.2.2	The system shall provide privacy of individual records
R1.2.3	The system shall minimize the use of energy for management
R1.2.4	The system shall transact at speeds to support large-scale operations
R1.2.5	The system shall incentivize cooperative distributed sharing of managed data
R1.2.6	The system shall limit the growth of managed data to a controllable size
R1.2.7	The system shall support both continuous and non-continuous operations
R1.2.8	The system shall manage data phase-out and end-of-life
R1.2.9	The system shall support managed data blocks of different sizes
R1.2.10	The system shall balance transparency and visibility with privacy
R1.2.11	The system shall manage transactions to allow indefinite number of large-scale smart contracts
R1.2.12	The system shall use management subcomponents that allow indefinite expandability and operational consensus of trust
R1.2.13	The system shall have an open architecture to allow developers and general public to make their own contracts
R1.2.14	The system shall have the capability to add and remove users and organizations
R1.2.15	The system shall have usable controls -- like web interfaces, mobile apps, Zero-Trust
R1.2.16	The system shall manage management subcomponents to provide service to multiple organizations and industry types
R1.2.17	The system shall provide end-to-end transaction security for participating transaction users on supply chains
R1.3	The system shall provide an Industry Utility Registrar
R1.3.1	The system shall use expandable industry identification registers of users/devices on multiple networks
R1.4	The system shall Manage Neutral Governance
R1.4.1	The system shall use neutral governance infrastructure to manage and optimize the (DNS-like) transaction engine
R1.4.2	The system shall use a consortium of common components for all indefinitely scalable governance

Satisfaction of CMMC Requirements¹⁸⁷

Table C.7. Summary list of all of CMMC Requirements
List of Practices

This subsection itemizes the practices for each domain and at each level.

* CUI means Controlled Unclassified Information

Each practice is specified using the convention of [DOMAIN].[LEVEL].[PRACTICE NUMBER] where:

- DOMAIN is the two-letter domain abbreviation;
- LEVEL is the level number Level 1 Level 2 Level 3 Level 4 Level 5; and
- PRACTICE NUMBER is the identifier assigned to that practice.

DOMAINS:

ACCESS CONTROL (AC)
ASSET MANAGEMENT (AM)
AWARENESS AND TRAINING (AT)
AUDIT AND ACCOUNTABILITY (AU)
SECURITY ASSESSMENT (CA)
CONFIGURATION MANAGEMENT (CM)
IDENTIFICATION AND AUTHENTICATION (IA)
INCIDENT RESPONSE (IR)
MAINTENANCE (MA)
MEDIA PROTECTION (MP)
PERSONNEL SECURITY (PS)
PHYSICAL PROTECTION (PE)
RECOVERY (RE)
RISK MANAGEMENT (RM)

The System shall...

- AC.1.001 Limit information system access to authorized users, processes acting on behalf of authorized users, or devices (including other information systems).
- AC.1.002 Limit information system access to the types of transactions and functions that authorized users are permitted to execute.
- AC.1.003 Verify and control/limit connections to and use of external information systems.
- AC.1.004 Control information posted or processed on publicly accessible information systems.
- AC.2.005 Provide privacy and security notices consistent with applicable CUI rules.
- AC.2.006 Limit use of portable storage devices on external systems.
- AC.2.007 Employ the principle of least privilege, including for specific security functions and privileged accounts.
- AC.2.008 Use non-privileged accounts or roles when accessing nonsecurity functions.
- AC.2.009 Limit unsuccessful logon attempts.
- AC.2.010 Use session lock with pattern-hiding displays to prevent access and viewing of data after a period of inactivity.
- AC.2.011 Authorize wireless access prior to allowing such connections.
- AC.2.013 Monitor and control remote access sessions.
- AC.2.015 Route remote access via managed access control points.
- AC.2.016 Control the flow of CUI in accordance with approved authorizations.
- AC.3.012 Protect wireless access using authentication and encryption.
- AC.3.014 Employ cryptographic mechanisms to protect the confidentiality of remote access sessions.

¹⁸⁷ Mitre. "Mitre ATT&CK for Industrial Control Systems." Accessed July 21, 2022. https://collaborate.mitre.org/attackics/index.php/Main_Page.

AC.3.017 Separate the duties of individuals to reduce the risk of malevolent activity without collusion.

AC.3.018 Prevent non-privileged users from executing privileged functions and capture the execution of such functions in audit logs.

AC.3.019 Terminate (automatically) user sessions after a defined condition.

AC.3.020 Control connection of mobile devices.

AC.3.021 Authorize remote execution of privileged commands and remote access to security-relevant information.

AC.3.022 Encrypt CUI on mobile devices and mobile computing platforms.

AC.4.023 Control information flows between security domains on connected systems.

AC.4.025 Periodically review and update CUI program access permissions.

AC.4.032 Restrict remote network access based on organizationally defined risk factors such as time of day, location of access, physical location, network connection state, and measured properties of the current user and role.

AC.5.024 Identify and mitigate risk associated with unidentified wireless access points connected to the network.

AM.3.036 Define procedures for the handling of CUI data.

AM.4.226 Employ a capability to discover and identify systems with specific component attributes (e.g., firmware level, OS type) within your inventory.

AU.2.041 Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.

AU.2.042 Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.

AU.2.043 Provide a system capability that compares and synchronizes internal system clocks with an authoritative source to generate time stamps for audit records.

AU.2.044 Review audit logs.

AU.3.045 Review and update logged events.

AU.3.046 Alert in the event of an audit logging process failure.

AU.3.048 Collect audit information (e.g., logs) into one or more central repositories.

AU.3.049 Protect audit information and audit logging tools from unauthorized access, modification, and deletion.

AU.3.050 Limit management of audit logging functionality to a subset of privileged users.

AU.3.051 Correlate audit record review, analysis, and reporting processes for investigation and response to indications of unlawful, unauthorized, suspicious, or unusual activity.

AU.3.052 Provide audit record reduction and report generation to support on-demand analysis and reporting.

AU.4.053 Automate analysis of audit logs to identify and act on critical indicators (TTPs) and/or organizationally defined suspicious activity.

AU.4.054 Review audit information for broad activity in addition to per-machine activity.

AU.5.055 Identify assets not reporting audit logs and assure appropriate organizationally defined systems are logging.

AT.2.056 Ensure that managers, system administrators, and users of organizational systems are made aware of the security risks associated with their activities and of the applicable policies, standards, and procedures related to the security of those systems.

AT.2.057 Ensure that personnel are trained to carry out their assigned information security-related duties and responsibilities.

AT.3.058 Provide security awareness training on recognizing and reporting potential indicators of insider threat.

AT.4.059 Provide awareness training focused on recognizing and responding to threats from social engineering, advanced persistent threat actors, breaches, and suspicious behaviors; update the training at least annually or when there are significant changes to the threat.

AT.4.060 Include practical exercises in awareness training that are aligned with current threat scenarios and provide feedback to individuals involved in the training.

CM.2.061 Establish and maintain baseline configurations and inventories of organizational systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles.

CM.2.062 Employ the principle of least functionality by configuring organizational systems to provide only essential capabilities.

CM.2.063 Control and monitor user-installed software.

CM.2.064 Establish and enforce security configuration settings for information technology products employed in organizational systems.

CM.2.065 Track, review, approve, or disapprove, and log changes to organizational systems.

CM.2.066 Analyze the security impact of changes prior to implementation.

CM.3.067 Define, document, approve, and enforce physical and logical access restrictions associated with changes to organizational systems.

CM.3.068 Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.

CM.3.069 Apply deny-by-exception (blacklisting) policy to prevent the use of unauthorized software or deny-all, permit-by-exception (whitelisting) policy to allow the execution of authorized software.

CM.4.073 Employ application whitelisting and an application vetting process for systems identified by the organization.

CM.5.074 Verify the integrity and correctness of security critical or essential software as defined by the organization (e.g., roots of trust, formal verification, or cryptographic signatures).

IA.1.076 Identify information system users, processes acting on behalf of users, or devices.

IA.1.077 Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.

IA.2.078 Enforce a minimum password complexity and change of characters when new passwords are created.

IA.2.079 Prohibit password reuse for a specified number of generations.

IA.2.080 Allow temporary password use for system logons with an immediate change to a permanent password.

IA.2.081 Store and transmit only cryptographically-protected passwords.

IA.2.082 Obscure feedback of authentication information.

IA.3.083 Use multifactor authentication for local and network access to privileged accounts and for network access to non-privileged accounts.

IA.3.084 Employ replay-resistant authentication mechanisms for network access to privileged and non-privileged accounts.

IA.3.085 Prevent the reuse of identifiers for a defined period.

IA.3.086 Disable identifiers after a defined period of inactivity.

IR.2.092 Establish an operational incident-handling capability for organizational systems that includes preparation, detection, analysis, containment, recovery, and user response activities.

IR.2.093 Detect and report events.

IR.2.094 Analyze and triage events to support event resolution and incident declaration.

IR.2.096 Develop and implement responses to declared incidents according to pre-defined procedures.

IR.2.097 Perform root cause analysis on incidents to determine underlying causes.

IR.3.098 Track, document, and report incidents to designated officials and/or authorities both internal and external to the organization.

IR.3.099 Test the organizational incident response capability.

IR.4.100 Use knowledge of attacker tactics, techniques, and procedures in incident response planning and execution.

IR.4.101 Establish and maintain a security operations center capability that facilitates a 24/7 response capability.

IR.5.106 In response to cyber incidents, utilize forensic data gathering across impacted systems, ensuring the secure transfer and protection of forensic data.

IR.5.102 Use a combination of manual and automated, real-time responses to anomalous activities that match incident patterns.

IR.5.108 Establish and maintain a cyber incident response team that can investigate an issue physically or virtually at any location within 24 hours.

IR.5.110 Perform unannounced operational exercises to demonstrate technical and procedural responses.

MA.2.111 Perform maintenance on organizational systems.

MA.2.112 Provide controls on the tools, techniques, mechanisms, and personnel used to conduct system maintenance.

MA.2.113 Require multifactor authentication to establish nonlocal maintenance sessions via external network connections and terminate such connections when nonlocal maintenance is complete.

MA.2.114 Supervise the maintenance activities of personnel without required access authorization.

MA.3.115 Ensure equipment removed for off-site maintenance is sanitized of any CUI.

MA.3.116 Check media containing diagnostic and test programs for malicious code before the media are used in organizational systems.

MP.1.118 Sanitize or destroy information system media containing Federal Contract Information before disposal or release for reuse.

MP.2.119 Protect (i.e., physically control and securely store) system media containing CUI, both paper and digital.

MP.2.120 Limit access to CUI on system media to authorized users.

MP.2.121 Control the use of removable media on system components.

MP.3.122 Mark media with necessary CUI markings and distribution limitations.

MP.3.123 Prohibit the use of portable storage devices when such devices have no identifiable owner.

MP.3.124 Control access to media containing CUI and maintain accountability for media during transport outside of controlled areas.

MP.3.125 Implement cryptographic mechanisms to protect the confidentiality of CUI stored on digital media during transport unless otherwise protected by alternative physical safeguards.

PS.2.127 Screen individuals prior to authorizing access to organizational systems containing CUI.

PS.2.128 Ensure that organizational systems containing CUI are protected during and after personnel actions such as terminations and transfers.

PE.1.131 Limit physical access to organizational information systems, equipment, and the respective operating environments to authorized individuals.

PE.1.132 Escort visitors and monitor visitor activity.

PE.1.133 Maintain audit logs of physical access.

PE.1.134 Control and manage physical access devices.

PE.2.135 Protect and monitor the physical facility and support infrastructure for organizational systems.

PE.3.136 Enforce safeguarding measures for CUI at alternate work sites.

RE.2.137 Regularly perform and test data back-ups.

RE.2.138 Protect the confidentiality of backup CUI at storage locations.

RE.3.139 Regularly perform complete, comprehensive, and resilient data back-ups as organizationally defined.

RE.5.140 Ensure information processing facilities meet organizationally defined information security continuity, redundancy, and availability requirements.

RM.2.141 Periodically assess the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals, resulting from the operation of organizational systems and the associated processing, storage, or transmission of CUI.

RM.2.142 Scan for vulnerabilities in organizational systems and applications periodically and when new vulnerabilities affecting those systems and applications are identified.

RM.2.143 Remediate vulnerabilities in accordance with risk assessments.

RM.3.144 Periodically perform risk assessments to identify and prioritize risks according to the defined risk categories, risk sources, and risk measurement criteria.

RM.3.146 Develop and implement risk mitigation plans.

RM.3.147 Manage non-vendor-supported products (e.g., end of life) separately and restrict as necessary to reduce risk.

RM.4.149 Catalog and periodically update threat profiles and adversary TTPs.

- RM.4.150 Employ threat intelligence to inform the development of the system and security architectures, selection of security solutions, monitoring, threat hunting, and response and recovery activities.
- RM.4.151 Perform scans for unauthorized ports available across perimeter network boundaries over the organization's Internet network boundaries and other organizationally defined boundaries.
- RM.4.148 Develop and update as required, a plan for managing supply chain risks associated with the IT supply chain.
- RM.5.152 Utilize an exception process for non-whitelisted software that includes mitigation techniques.
- RM.5.155 Analyze the effectiveness of security solutions at least annually to address anticipated risk to the system and the organization based on current and accumulated threat intelligence.
- CA.2.157 Develop, document, and periodically update system security plans that describe system boundaries, system environments of operation, how security requirements are implemented, and the relationships with or connections to other systems.
- CA.2.158 Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.
- CA.2.159 Develop and implement plans of action designed to correct deficiencies and reduce or eliminate vulnerabilities in organizational systems.

Table C.8. An exemplary strip of screenshots of ESPTM/PhilosTM Satisfy Requirement Matrix using the Cameo tool. ESPTM/PhilosTM Structural elements are shown along the top, and Requirements along the left axis. (If reading left-to-right, start at the bottom panel.)

	Satisfy	Satisfy (Implied)
BlockchainDevelopmentCommunity	✓	✓
BlockchainDevelopmentCommunityServices	✓	✓
BlockchainDistributedLedger	✓	✓
BlockchainDistributedLedgerServices	✓	✓
BrokeringAgent	✓	✓
BrokeringAgentServices	✓	✓
ConsortiumWebSvc	✓	✓
ConsortiumWebSvc5Services	✓	✓
eSEntrprise : ESFEnterprise [1]	✓	✓
PeerLedgerServices	✓	✓
SystemContainer	✓	✓
PhilosTransactionBlockSerializer	✓	✓
SymmetricEncryption	✓	✓
BrokeringAgent : BrokeringAgent	✓	✓
communicationsSocket : CommunicationsSocket	✓	✓
consortiumLedger : ConsortiumLedger	✓	✓
peerLedger : PeerLedger	✓	✓
BrokeringAgent : BrokeringAgent	✓	✓
communicationsSocket : CommunicationsSocket	✓	✓
ConsortiumLedger : ConsortiumLedger	✓	✓
CryptoKeyStorageInterfaceAndKeyUpdateAPI	✓	✓
IdentityVerification	✓	✓
MessageAuthentication : MessageAuthentication	✓	✓
PeerLedger : PeerLedger	✓	✓
SymmetricEncryption	✓	✓
systemContainerInterfaceBlock : SystemContainer	✓	✓
TLSApplicationLayer	✓	✓
TLSApplicationLayer : TLSApplicationLayer	✓	✓
brokeringAgent : BrokeringAgent	✓	✓
communicationsSocket : CommunicationsSocket	✓	✓
consortiumLedger : ConsortiumLedger	✓	✓
peerLedger : PeerLedger	✓	✓

Academic Version for Teaching Only

Commercial Development

Version for Teaching Only

Commercial Development is strictly Prohibited

Legend		15.39 AU.3.046	15.40 AU.3.048	15.41 AU.3.049	15.42 AU.3.050	15.43 AU.3.051	15.44 AU.3.052	15.45 AU.4.053	15.46 AU.4.054	15.47 AU.5.055	15.48 CA.2.157
	Satisfy										
	Satisfy (Implied)										
MessageAuthentication											
communicationsSocket : CommunicationsSocket											
CryptoKeyStorageInterfaceAndKeyUpdateAPI											
NeutralGovernance											
ConsortiumCommonComponents											
BDLContainer											
ConsortiumCommonComponentsServices											
ConsortiumCommonElements											
ConsortiumLedger											
LedgerConsortiumGovernanceSvc											
BDLContainer											
LedgerConsortiumGovernanceSvc											
ConsortiumCommonComponents : Consorti-											
ConsensusAlgorithm											
LedgerConsortiumGovernanceSvc											
NeutralGovernance											
ConsortiumCommonComponents : Consorti-											
ConsortiumCommonElements											
LedgerConsortiumGovernanceSvc : LedgerCo											
LedgerConsortiumGovernanceSvc											
LedgerConsortiumGovernanceSvc											
PolicyGovernance : PolicyGovernance [1]											
PolicyGovernanceServices											
PhilosEnterprise : PhilosEnterprise [1]											
PolicyGovernance											
BlockFrameInc											
ColoradoIT											
LedgerConsortiumGovernanceSvc : LedgerCo											
UCCS											
PolicyGovernanceServices											
UCCS											
PeerLedger											
SideChains											
SymmetricEncryption : SymmetricEncryption											
TLSApplicationLayer											
PhilosBrokerSearchAgent											

Legend	15.39 AU.3.046	15.40 AU.3.048	15.41 AU.3.049	15.42 AU.3.050	15.43 AU.3.051	15.44 AU.3.052	15.45 AU.4.053	15.46 AU.4.054	15.47 AU.5.055	15.48 CA.2.157
Teaching Only is strictly Prohibited Satisfy (Implied)										
BlockSerializer										
PhilosTransactionContract(SysID)										
PhilosBrokerSearchAgent										
PhilosTransactionBlockSerializer										
PhilosBrokerSearchAgent : PhilosBrokerSearch										
PhilosTransactionBlockSerializer : PhilosTrans										
BrokeringAgent										
symmetricEncryption : SymmetricEncryption										
tlSApplicationLayer : TLSApplicationLayer										
tlSApplicationLayer : TLSApplicationLayer										
tlSApplicationLayer : TLSApplicationLayer										
IdentityVerification : IdentityVerification										
messageAuthentication : MessageAuthentication										
symmetricEncryption : SymmetricEncryption										
systemContainerInterfaceBlock : SystemContair										
tlSApplicationLayer : TLSApplicationLayer										
ConsortiumLedger										
symmetricEncryption : SymmetricEncryption										
tlSApplicationLayer : TLSApplicationLayer										
CryptoKeyStorageInterfaceAndKeyUpdateAPI										
BlockFrameCTCIC										
communicationsSocket : CommunicationsSocket										
ConsortiumIDKeyStructure										
CTCUpdateAPI										
IdentityVerification : IdentityVerification										
messageAuthentication : MessageAuthentication										
ASU										
BlockFrameInc										
BlockFrameInService										
UCLibrary										
UCLibraryService										
UCLibraryService										
UCLibraryService										
LogisticC										
LogisticCService										
NetworkProgrammingService										
NetworkProgrammingWebSvc										
ESPPhilosEnterprises										
ESPEnterprise : ESPEnterprise [1]										
ESPEnterprise : ESPEnterprise [1]										
IdentityVerification										
communicationsSocket : CommunicationsSocket										

Carneo Systems Modeler 19.0 - BlockFrameModeling [C:\Users\Kent.Lambeth\Documents\System Engineering\Eng 607\607 MODELS\Carneo Files\BlockFrame1]

File Edit View Layout Diagrams Options Tools Analyze Collaborate Window Help

ESPPhilosFunctionalRequ... X

Criteria

Row Element Type: Requirement

Row Scope: 3 Requirements

Dependency Criteria: Satisfy Satisfy (Implied)

Column Scope: c Structure

Direction: Both

Show Elements: All

Property/Reference Property: Subsystem Synte

Legend

Satisfy

Satisfy (Implied)

PhilosBrokerAgent

PhilosTransactionContract(SYSID) : PhilosTran

BrokerSearchAgent

TransactionContract

TransactionInterface

PhilosBrokerSearchAgent

SearchAPI

SearchGUI

SearchOptimizationEngine

PhilosTransactionBlockSerializer

PhilosBrokerSearchAgent

BlockSerializer

13 37 50 17 17 17 41 14 22 77

15.39 AU.3.046

15.40 AU.3.048

15.41 AU.3.049

15.42 AU.3.050

15.44 AU.3.052

15.45 AU.4.053

15.46 AU.4.054

15.47 AU.5.055

15.48 CA.2.157

Ready

Type here to search

Windows Taskbar

Mathematical Analysis Using Formal Methods

This System includes: the Philos™ Blockchain/Distributed Ledger (B/DL), and its associated systems: the Eco-Secure Provisioning System™ (ESPTM), the Neutral Governance Organization (NGO), the Industry Utility Registrar™ (IUR™), and their associated development organizations, BlockFrame, Inc., and the BlockChain Development Consortium (BCDC).

I have not yet observed a need for the highest level of rigor within the System, although that might be useful later, in this discussion. Formal Methods might be an appropriate approach to organize that research. “In [computer science](#), formal methods are [mathematically](#) rigorous techniques for the [specification](#), development, [analysis](#), and [verification](#) of [software](#) and [hardware](#) systems.”¹⁸⁸ “The use of formal methods for software and hardware design is motivated by the expectation that, as in other engineering disciplines, performing appropriate mathematical analysis can contribute to the reliability and robustness of a design.”¹⁸⁹ Some highly critical specifications might be necessary for certain algorithms, particularly for the Philos™ consensus algorithm. Although the exact algorithm has not yet been finalized, it is critical for adding new blocks to the Philos™ B/DL, and for adjudicating several trust factors across the Philos™ Enterprise to “score” Users on their relative levels of trust.¹⁹⁰ That is a core function of Philos™, and the System could fail if trust factors are not fairly and consistently applied. Due to the Philos™ goal to be a highly efficient engine to process an indefinite number of blockchain requests rapidly, the consensus algorithm must also be highly efficient – another reason that a very efficient and validated code set with very rapid processing times could be critical. That suggests that once the algorithm is determined, or as a process to compare different feasible approaches. A trade-off study might be appropriate to see if a lower-level language, an assembly language, or a “first-order logic model,” along with formal mathematical analysis, might be appropriate to perform and validate those calculations. By extension, that might also include careful analyses of the input/output elements, networks, hardware, and code sets that work in close functional proximity to the consensus algorithm. The network topology is also not yet clear as to

¹⁸⁸ Butler, R. W. (2001-08-06). *“What is Formal Methods?”*. Retrieved 2006-11-16. Cited in Wikipedia, “Formal Methods,” https://en.wikipedia.org/wiki/Formal_methods.

¹⁸⁹ Holloway, C. Michael. *“Why Engineers Should Consider Formal Methods” (PDF)*. 16th Digital Avionics Systems Conference (27–30 October 1997). Archived from [the original](#) (PDF) on 16 November 2006. Retrieved 2006-11-16. Cited in Wikipedia, “Formal Methods.” https://en.wikipedia.org/wiki/Formal_methods.

¹⁹⁰ Gorog, Christopher, and Terrance E. Boulton. “Solving Global Cybersecurity Problems by Connecting Trust Using Blockchain.” In *2018 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 1425–32, 2018. https://doi.org/10.1109/Cybermatics_2018.2018.00243.

whether the consensus algorithm will typically run on a single central processor, or if several processors might need to run in parallel to process data prior to a final consensus decision process. Several discussions about the consensus algorithm have centered on which statistical distribution functions might be most useful to score the trust calculations. Initially, a prevailing opinion might have been that an exponential function, or a set of exponential functions, might be the best consideration, but more recently after some initial testing that might be reconsidered. A more linear representation might be “good enough,” and might bring the benefits of mathematical simplicity, might be more efficient to explain to stakeholders and Users, as well as possibly decrease processing time. Another reason why mathematical formal methods might be considered is that Philos™ is designed to process “smart transactions” among separate and possibly competing business entities where the trust factors adjudicated by Philos™ could be the basis for trust for very high value business, security, or military operations. A failure of trust could have drastic consequences, which could include catastrophic damage to User systems, and/or criminal or civil legal liabilities, if they do not work well. Just on that practical point, conducting a rigorous formal methods analysis might contribute to a good “legal defense” if Philos™ performance, accuracy, or trust chaining was ever to be challenged. Formal methods could also be useful to review the appropriateness of current programming languages, code peer-review or code automated-review practices. Until the time of this writing, several programmers have sometimes used languages that they are familiar with and are expedient. Those will probably to be re-translated to more robust languages – perhaps Go-Lang or others – based on a more formal review process. As described above, some assembly language-level coding might also be considered.

Applying a “Lightweight Formal Method”

A “lightweight formal method” can help establish important characteristics of the system in terms of meeting requirements while minimizing cost, schedule, and risk.

It is not clear whether the application of a complete SysML/MBSAP is best described as a “Level 0, lightweight formal method”, or as a “Level 1, high integrity specification” formal method, or both. Ideally, SysML/MBSAP should lead to satisfying the “high integrity” goals because of the need for high levels of operational security, cryptological soundness, and other factors have the requirements of satisfying some of the highest (and most elusive) and current national requirements for Cybersecurity Maturity Model Compliance (CMMC), and Statement of Insolvency Practice 13 (SIP-13). So, it is difficult to determine whether that is best defined as a Level 0 or a Level 1.

Whichever definition, I would describe the benefits of SysML/MBSAP considering the following list of characteristics:

- Requirements. Used as structured design requirements/capabilities, a priori, the CMMC and SIP-13 standards, if satisfied, tested, and evaluated, form a highly visible set of pre-defined security and federal government mandated requirements, especially for the energy industry, that are generally considered highly problematic to attain. It might be considered that every major energy producer/stakeholder would seek compliance with these requirements, and if SysML/MBSAP can help prove their satisfaction and verification, the project should be highly accepted.
- Cost. Initial modeling (primarily from a framework designed in Module 10), and now being refined, suggest that costs will tend to be “optimized” if reference architecture topologies can be represented for typical Use Cases, and then modularized and repeated for other Use Cases. For the Philos™ Blockchain, that should generally involve the reduction of infrastructure, and therefore personnel costs, while encouraging more Users, customers of User Services, and more smart transactions using the blockchain services – assuming that a core business structure involves charging fees for each blockchain transaction. A logical starting point is with the first contracted use of Philos™, which involves support for Randa Solutions to secure records for the Colorado Department of Education. As a relatively simple Use Case, the System and Network definitions should be able to be transformable to a modular reference architecture and with loose coupling which can be used as the basis for similar Use Cases (or Scenarios). That also has good potential for containing costs. The basic Philos™ architecture (of using System Containers as the common component of an extensive Enterprise), by its nature makes modularity and security integral to the design. Follow initial fielding, more complex additions, like integration with ESP™ and other components will serve more complex Use Cases, but due to their modular design, should also allow modular reuse to avoid having to customize each instance for new Use Cases/Scenarios. One current challenge for modeling cost flows might be a limitation for the Cameo tool: how transactions, tokens, or “stamps” representing the costs of specific products will be represented within the System. Petri Nets models might be a better way to represent or simulate those flows of Phylis™ through the System in order to simulate financial flows, velocities, and accounting of payments. Once better represented, those models might be better subject to “optimization,” and/or used as tools to validate fee payment structures. Whether by a centralized pricing structure, or a more freely competitive

floating price structure, those decisions will probably be better informed by modeling, analysis, and/or simulation.

- Schedule. As with physical and digital components, typical or reference architectures can now allow projections for workflows, milestones, delivery schedules, etc., as modular parts of typical Use Cases. Tables C.12 and C.13 represent such “typicals” for a supply chain Use Case and a smart energy Use Case. Over time, these examples can be refined as better knowledge and experience from deliveries can be validated, and “optimized”.

Table C.12. Typical Work Breakdown Structure and Milestones Supply Chain Project

Work Break Down			
Num ber	Activity	Description	Labor Category
1	Segment 1	Customer and Use Case Scoping Efforts	
1.1	Customer Coordination	Identification and Prioritization of Air Force Use Cases	SME
1.2	Industry SME Analysis	Engagement and Integration Operations with Air Force Software and IoT Vendors	PM
1.3	Technical Architecture	Legacy Integration Support ESPTM CTC Implemented for Air Force	SME
1.4	Software/ Hardware Engineer	Cyber Physical ESPTM CTC Implemented for Air Force Vendor	Developer
1.5	Customer Coordination	Define Air Force Supply Chain Governance Needs and Implementation Plan	SME
2	Segment 2	Implement Customer Supporting Logistics-as-a-Service	
2.1	Military Liaison	Identification and Prioritization of Broker Agent and Field Service Needs	PM
2.2	Customer Coordination	Identification and Prioritization of Air Force Governance Workflow	SME
2.3	Software Engineer	Customization of value-added Air Force Logistics-as-a-Service	Developer
2.4	Software/ Hardware Engineer	Supply Chain-as-a-Service provided by BlockFrame IUR	Developer
2.5	Software Engineer	Air Force Distributed Ledger vendor supply chain Broker placed in Operation	Developer
2.6	Military Liaison	Scaling and Prototype Field Operations Testing	SME
3	Segment 3	Implement Air Force Wide Supply Chain Eco-System	
3.1	Business Analysis	Identification and Prioritization of Custom Visibility	BA-PM
3.2	Military Liaison	Establish Custom Air Force Governance Workflow	SME
3.3	CTC Design	Implemented Air Force supply chain maintenance and visibility access	Developer
3.4	IUR Design	End-to-End Operational Testing	Developer
3.5	Business Analysis	Development and Integration Training	BA-SME

4	Segment 4	Conversion to Operational Secure Supply Chain	
4.1	Project Management	Implement Operation of Governance Workflow	BA-PM
4.2	Military Liaison	Air Force Supply Chain Governance Capability Established	SME
4.3	CTC Design	Additional Use Case Rollout and Scalability Verification	Developer
4.4	IUR Design	Operational Support for Scalability	Developer
4.5	IUR Design	Operational and User Training	BA

Milestone Schedule

RAPID Phase II				
MILESTONE - TASK	EXPECTED DELIVERY DAYS AFTER CONTRACT AWARD	DELIVERABLE	ACCEPTANCE CRITERIA	PAYMENT AMOUNT
01- Scope and Prioritize Customer Supply Chain Use cases	Award +15 Days	Stakeholders approval Deliverable: 1	AF customers and AF stakeholder POCs approve the project plan	\$219,264 - Evidence: Signoff, AF Excepted Work Plan
02 - ESP™ IOT capability delivered	Award +90 days	Deliverable: 2, 3, 4. Report: 1 & 2	AF customers and AF IoT Vendors gain integration access	\$75,000 - Evidence: Deliverables received and accepted by AF
03 - ESP™ legacy software capability delivered	Award 150 days	Deliverable: 5, 6. Report: 1, 2, 3	AF customers evaluate the legacy application integration on schedule	\$75,000 - Evidence: Air Force Acceptance of Deliverables
04 – Design Produced Plan for Philos™ Broker	Award +6 Months	Stakeholders approve design plan. Deliverable: 7 Report: 1 & 3	AF customer and vendor POCs approve the project plan	\$75,000 - Evidence: Email confirmation that AF excepted work plan
05 – Philos™ Broker capability delivered	Award +9 Months	Deliverable: 8&9 Report: 1, 2, & 3	AF customers and AF end-users POCs approve the demonstration	\$ 73,336 - Email confirmation of successful

				demonstration by AF
06 – Philos™ Governing Operations and Procedures In place	Award +12 Months	Deliverable: 10&11 Report 1 & 3	AF customer and vendor POCs approve the operation and governing procedures.	\$ 249,516 - Email confirmation of demonstration and deliverables accepted by AF
07 – Philos™ Broker submitting BDL transactions	Award +15 Months	Stakeholders Demonstration of successful Deliverable: 12&13 Report: 1	AF customers and AF end-users verify operation of demonstration. Deliverables and	\$ 50,000 - Email confirmation of Deliverables accepted by AF
08 – ESP™ Provisioning-as-a-Service Operational	Award +18 Months	Deliverable: 14,15,16 Report: 1	AF customers access Administrative and Dashboard web-based tools.	\$ 50,000 - Email confirmation of Deliverables accepted by AF
09 – ESP™ customer legacy software patched to operational with initial vendor	Award +21 Months	Deliverable: 17 Report: 1	AF customers and legacy application vendor test end to end Provisioning operation.	\$ 50,000 - Email confirmation of Deliverables accepted by AF
10 – ESP™ IOT capability implemented with initial vendor	Award +24 Months	Deliverable: 18 Report: 1	AF customers and IoT product vendor test end to end Provisioning operation.	\$ 63,244 - Email confirmation of Deliverables accepted by AF

Table C.13. Typical Work Breakdown Structure and Milestones Smart Energy Project

Work Break Down			
Number	Activity	Description	Labor Category
1	Segment I	CUS Vendor and Use Case Scoping Efforts	
1.1	Vendor Coordination	Identification and Prioritization of CSU Vendor needs	SME
1.2	Industry SME and Hardware Design	Engagement and Integration Operations with CSU DER Vendors	PM
1.3	Software/Hardware Engineer	Vendor Integration Support ESP CTC libraries	Developer
1.4	Industry SME and Hardware Design	Manufacturing Cyber Physical ESP CTC product line support	Developer

1.5	Customer Liaison and Software Des.	Define CSU Sensor Data Collection Needs and IUR integration plan	SME
-----	------------------------------------	--	-----

2	Segment II	Implement Customer Supporting Logistics as a Service	
2.1	Customer Liaison	Identification and Prioritization of Broker Agent support for Vendors	PM
2.2	Business Analyst	Identification and Prioritization of CSU Provisioning and sensor data workflow	BA
2.3	Business Analyst /Software Engineer	Customization of CSU on premises IUR Logistics As A Service	BA-Dev
2.4	Software/Hardware Engineer	ESP Vendor provisioning as a Service provided by BlockFrame IUR	Developer
2.5	Software Engineer	ESP Network provisioning as a Service provided by CSU IUR	Developer
2.6	Customer Liaison	Scaling and IUR(s) and provisioning prototype for customer operations testing	SME

3	Segment III	Implement CSU ESP Framework for Vendor Eco-System:	
3.1	Business Analysis	Identification and Prioritization of administration and visualization needs	BA-PM
3.2	Customer Liaison	Establish CSU Administrative workflow capability for on premises CSU IUR	SME
3.3	Software Engineer	Implemented CSU administration and maintenance and visibility access	Developer
3.4	Business Analysis	End to End ESP Framework Operations Test	BA
3.5	Business Analysis	Development and Integration Training	BA

4	Segment IV	Conversion to Operational Secure Supply Chain:	
4.1	Business Analyst/ Software Engineer	Implement operation of administration and sensor data collection	BA-PM
4.2	Customer Liaison	Operational ESP Capability implemented on CUS IUR	SME
4.3	Business Analyst/ Customer Liaison	Sensor data collection services scalability verification	BA-SME
4.4	Business Analyst	Operational Support for Scalability	BA
4.5	Business Analyst	Operational and User Training	BA

Milestone Schedule

RAPID Phase II	EXPECTED DELIVERY			
MILESTONE - TASK	DAYS AFTER CONTRACT AWARD	DELIVERABLE	ACCEPTANCE CRITERIA	PAYMENT AMOUNT
01- Scope and Prioritize Customer Use cases	Award +15 Days	Deliverable:1	AF customers and Utility approve the project plan	\$219,264 - Evidence: AF Signoff
02 - ESP™ IOT capability delivered	Award +3 Months	Deliverable:2,3,4 Report: 1&2	Quarterly Status Report	\$75,000 - Evidence: Deliverables accepted
03 - Design Produced Plan for Philos™ Broker Integration	Award +6 Months	Deliverable: 5 Report: 1	AF approve plan and Workflow	\$75,000 - Evidence: AF accept deliverables
04 – BlockFrame IUR Capability Available Remote	Award +9 Months	Deliverable:6 Report: 1	Verify end to end Provisioning demonstration	\$ 73,336 - Email confirmation of successful demonstration
05 – On Premises Installation work plan	Award +12 Months	Deliverable:7 Report: 1	AF approve the project plan	\$ 249,516 - Email Deliverables accepted
06 – Migrate Network Provisioning to On-Premises IUR	Award +15 Months	Deliverable:8,9 Report: 1&2	AF verify demonstration. Deliverables	\$ 50,000 - Email accepted by AF
07 – Philos™ Broker for On-Prem Network Provisioning available.	Award +18 Months	Deliverable: 10,11,12 Report: 1	Verify OnPrem functionality and Access to Administrative and Dashboards	\$ 50,000 - Email successful vendor test, and deliverables
08 – ESP™ IUR service Installation Complete	Award +21 Months	Deliverable: 13&14 Report:1	Verify end to end Provisioning operation.	\$ 50,000 – Email: Test Complete and Deliverables
9 – ESP™ IUR Operational Status and migration to Phase III.	Award +24 Months	Deliverable: 15 Report: 1&2	Approve migration to operational status.	\$ 63,244 - Email confirmation

Risk Reduction. Table C.14 displays some identified risks in Philos™ development, and some SysML/MBSAP-based tasks that can be taken to help mitigate those risks. All of these have been considerations have been identified to a new Philos™ and BlockFrame systems engineering teams for discussions.

Table C.14. Risks and Mitigation Considerations

Risks	Mitigation Steps
Intellectual property loss could risk corporate market share	Trademarks used; initial patents applied for; need to be finalized (non-functional rqmt)
Corporate startup cash flow until full production	R&D grants received; more being submitted; initial subcontract with State prime contractor; new military Cooperative R&D Agreements (CRADA) being sought, and MOU with industries (non-functional rqmt)
Schedule risks: to assure initial delivery timing	Create and refine WBS and Milestones to plan with better certainty of research and product delivery
Systems engineering personnel staffing	Started new Philos™/BlockFrame SE team
Deficiency in Phylia flow model	Acquired Petri Net flow model for Matlab to study various options
Incomplete requirements/capabilities analyses	Expand requirements analyses: continue to validate CMMC compliance; add SIP 13 requirements; plan for T&E traceability to validate requirements; add to Cameo tool
Consensus model might lack robust processes	Consider mathematical formal methods (described in answer above)
Network topology risks	More specifically model and analyze topologies, especially near the consensus model to help validate structure, processes, and performance
Schedule risks: modularity and reusability	Modify small Use Case models into Reference Architectures (RAs) for reuse; compare existing WBS and milestones into more generic re-useable RA.
Organizational risks	Expand Architecture Overview and Summary to better institutionalize inter-corporate planning; expand draft Service-Level Agreements to specify Philos™ B/DL Services and other User Services
Phylia™ pricing risk	Setting an initial value for Phylia™ is a time-urgent problem, but the level of risk is difficult to determine. If Phylia™ are allowed to float in value, they could be dynamically instable, or they could quickly converge to stability. Attempting to model those dynamics could be useful.

Verification and Validation (V&V) Plan

Following previous work to map requirements to Enterprise components that satisfy those requirements, a reasonable V&V plan would be to:

- 1) Coordinate the large requirements cross-map with development engineers are organized into several working groups. The team approach with both system engineers and developers is a good way to document whether the requirements are indeed satisfied, and how they are satisfied. For this research, those requirement sets

currently include all CMMC requirements, and a self-generated set of functional Philos™ performance requirements;

- 2) Expand the existing requirements sets to include SIP 13 requirements for energy production-related Use Cases that already should be satisfied, but are currently not documented nor included in the SysML model;
- 3) Create a similar traceability matrix for test and evaluation actions to trace <<verification>> dependencies back to the tested components. Some of this work has already started but has not yet been added to the SysML model. For example, interactive dashboards have already been used to record and control Philos™ blockchain interactions within test platforms, as have tests with BlockFrame ESP™ components that have already successfully verified their performance but have not yet been documented. All of these actions should be added to the SysML model to trace and document those <<verification>> relationships, which should expand over time as more component and system testing is performed.
5. Consensus Model/Trust Behavior. Figure C.34 shows actual test data from Philos™ B/DL servers, performance parameters, and trust parameters that increase and converge over time. Again, some of these may represent functional performance parameters, but also could be considered non-functional quality attributes (QAts) that are also appropriate to track as quality metrics and behaviors over time. Examples are given in Figure C.2 for metrics like, “Determine Merchant Performance,” “Ownership Track Record,” and “Identify High Risk Items/Organizations,” that can monitor trends in QAts, and rapidly identify quality problems that might compromise trust within business transactions, Users “gaming the system,” or other trends that could compromise the goals and performance of the trust chains. This also demonstrates the importance of the Consensus algorithms, and the incumbent risks that might occur if those algorithms were compromised, inefficient, or possibly fail economic stress tests.

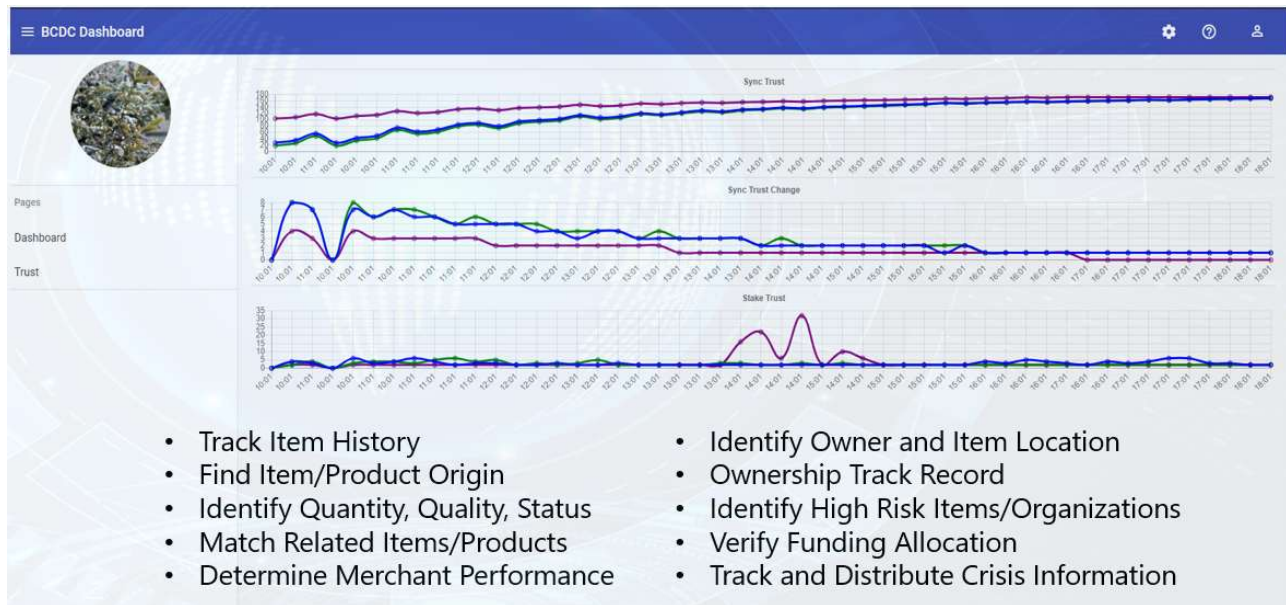


Figure C.34. Philos™ Dashboard

6. Economic Stability of Phylai™. Stakeholder concerns about the initial pricing of the in-system utility tokens (Phylai™), are due to the unknowns concerning their economic behaviors. The initial pricing mechanism is important because it must behave nearly as expected for initial demonstrations, or there could be risks of perceived “failures” of the whole B/DL. Although Phylai™ should not be considered cryptocurrencies that could be used in open markets and could be subject to changes in value due to speculative investor behaviors – possibly either risk averse or risk prone – Phylai™ theoretically should not since they are primarily used for administrative or accounting purposes to accurately assure payments of fees-for-services. However, safeguards must be created and constantly monitored throughout the system lifecycle to ensure that dynamic instabilities, or even too much dynamic stability, could adversely affect fee payment structures. Since the initial startup of Philos™ and the active use of Phylai™ could begin with no transaction history, that could make that phase unpredictable and risky. Like a parent teaching their child to ride a bicycle, the neutral governance system might have to artificially control the initial conditions until initial stability appears to be sustainable – like the point where the parents can safely release their grip on the bicycle and the child can confidently start pedaling on their own. A simulation model could be very useful to test the behaviors of that initial phase, but in the long run, the stability levels could be considered as QATs that should be self-sustaining but also require constant or at least periodic monitoring for inserted events that could break the predicted cycles.

Real-time Considerations

This section identifies and demonstrates the ability of SysML to represent time-sensitive behaviors with deadlines and to create an initial timing analysis using Sequence and other behavioral diagrams. One Sequence Diagram will use MARTE constructs to define and demonstrate real-time behavior.

Figures C.35 and C.36 show two time-sensitive behaviors based on time-sensitive criteria. Peer Ledgers must receive and process their subordinate Brokering Agent transactions, process them to assure that transactions comply with pre-determined security requirements, format all of the transactions into candidate block requests to pass to Consortium Ledgers. Figure C.35 shows 30-minute deadlines for each Peer Ledger to process all of their respective Brokering Agent transactions, and all of the other activities listed above. If those deadlines are exceeded, the request fails and must be delayed to the next 30-minute cycle. The Figure also only shows two Peer Ledgers for this example. An actual system could have far more than that. Figure C.35 also shows a representative time deadline for Consensus Ledgers to collect a set of Peer Ledger requests, synchronize them into a new blockchain block, and transmit the new blocks back to the Peer Ledgers so they will have that information for validation that their previous requests have been processed, and to inform future requests. For this example, a period of 72 minutes is used as an arbitrary example of that deadline.

Figure C.36 shows a more complete and unlimited Peer Ledger multiplicity performing similar processes, but with the emphasis on another deadline, in this case a global requirement to have a fixed percentage of all PhilosTM come into consensus every 24 hours. In both cases, the Peer Ledger deadlines and the 24-hour deadline should be considered “soft” deadlines in that the B/DL will not suffer catastrophic failure if those deadline criteria are not met. However, if the system cannot meet those deadlines on a sustainable basis, the business model would probably not be as effective and could be compromised over time.

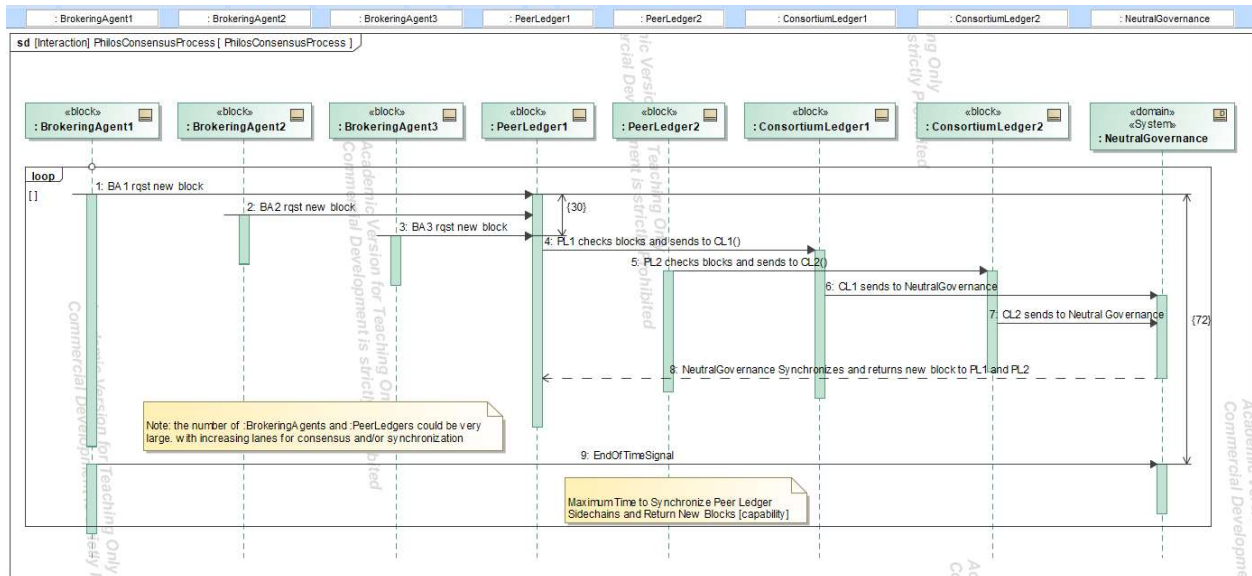


Figure C.35. Philos™ Consensus Process with Emphasis on Brokering Agent to Peer Ledger to Consortium Ledger Transactions

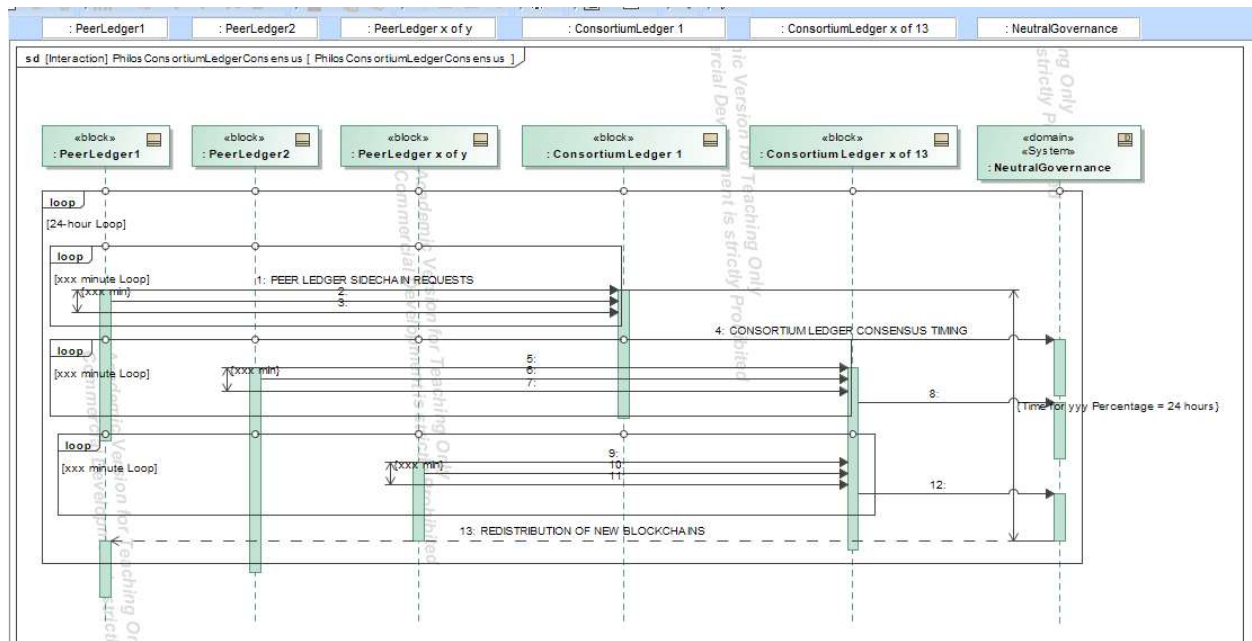


Figure C.36. Philos™ Consensus Process with Emphasis on Peer Ledger to Consortium Ledger Transactions

Figure C.37. shows a State Machine Diagram that monitors the status (states) of ESP™ LogisticsIntegratedCircuit (LIC) devices within Internet of Things (IoT) devices. This monitoring process may not be a time-sensitive process with deadlines but is shown here to describe the starting point of another sequence. The Figure shows how a state machine might monitor the status of a “Good Cypher” on the LIC. If the `{LICDeviceTestFlag=false}`, a failure report is sent, the LIC will be shut down and will report a failure. Depending on the criticality of the IoT device, a failed LIC could be caused by a lot of events, probably all bad: attempted or

actual hostile penetration of cyphers, battle damage, etc. Whatever the cause, the next step leads to an attempt to ProvisionLICCypher in an attempt to re-provision the LIC and return the device back to normal operations. That process leads to the Sequence Diagram in Figure C.38, which may be time-sensitive with strict deadlines.

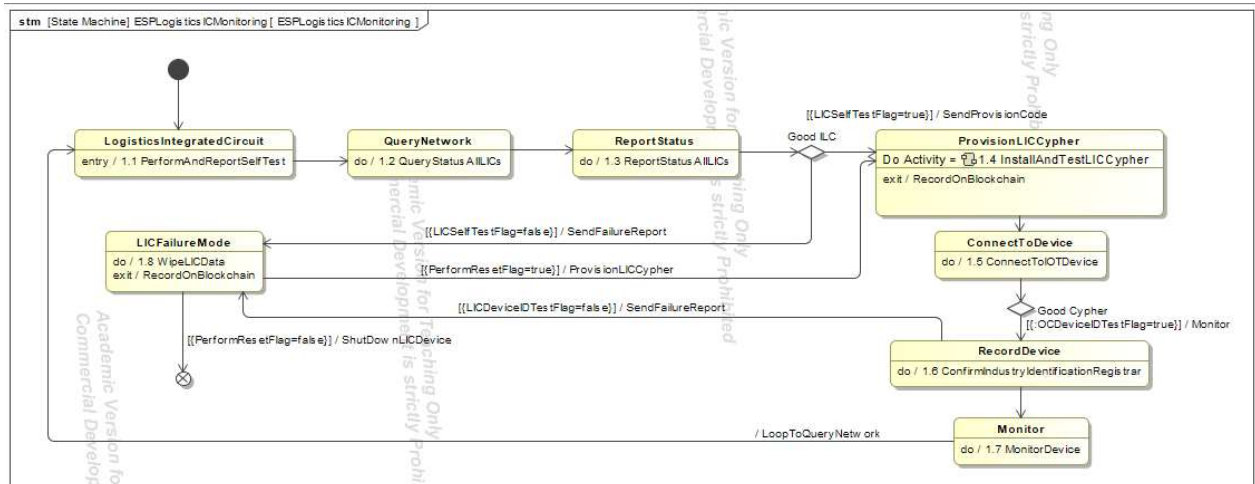


Figure C.37. State Machine showing ESP™ LogisticsIntegratedCircuit Monitoring

Figure C.38 uses a Sequence Diagram in the SysML model with MARTE constructs to define a real-time behavior.

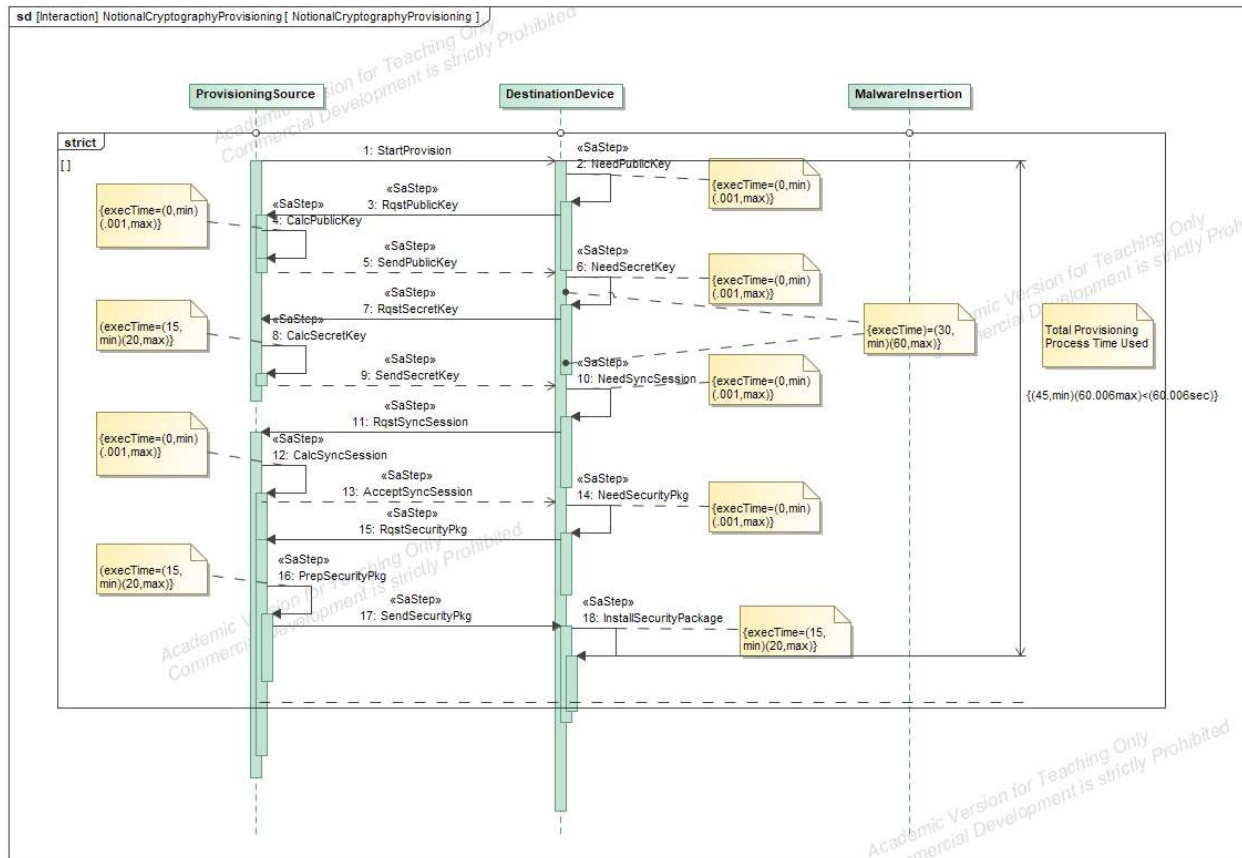


Figure C.38. Sequence Diagram showing ESP™ LogisticsIntegratedCircuit Cypher Code Provisioning

The intent of Figure C.38 is to demonstrate very rapid time constraints to provision secret cyphers onto LogisticsIntegratedCircuits. If time constraints are not met, that could indicate that unexpected malware packages could have been inserted, and/or are “backpacking” into the system along with the code. Therefore, very precise time criteria should be enforced, or the provisioning should be rejected.

Open System Characteristics

The SysML model can be used to analyze the existing architecture in terms of measures of openness and make appropriate changes to enhance modularity, loose coupling, and other open system characteristics. As an Enterprise-of-Systems that is designed to connect to and provide blockchain services for a wide variety of dissimilar Users, Use Cases, devices, and applications, almost all of the nodes within the Enterprise are actually Philos™ common components that are either connected to or embedded within User system devices. Modularity, loose

coupling, and most other open system characteristics are inherent within the Philos™ design, with some specific exceptions that are mentioned in the OSA Questionnaire below. Some components inside trusted security layers might not be considered to be totally “open,” unless it would be on a “need to know” basis for designers, but that would be limited to the corporate secrets of either the designers or the Users, intellectual property, protected cyphers or keys, or protected security processes, and only within those trusted security layers.

Open standards for interfaces, services, components, and other architecture elements

The standards for interfaces and services are currently being driven under an effort that will lead to operational tests and evaluation as a component for a Randa Solutions contract with the State of Colorado where Randa is the Prime, and the BlockChain Development Community (BCDC) in the Sub-contractor to provide blockchain services. Randa’s previous design was to use a common blockchain with another standards. Randa seems to have found less-than-ideal performance from that solution, which probably involved the same rationale that BlockFrame, Inc., had previously rejected the same solution and other existing solutions. That previous research by BlockFrame led to the decision to start up the BCDC, and to develop the Philos™ Blockchain/Distributed Ledger. So, for the immediate need, the basic Philos™ design might need to be modified to replace the current “legacy” blockchain product and the existing interface for the Randa test. It is not known at this time whether following that test whether that interface will be adopted for other future projects or not, or if it will become a “standard” at all. It is also not known if the test interface “standard” will be uniquely designed into the BCDC side of the interface, the Randa side of the interface, or both, so that makes it difficult to specify right now what the interface solution will be, and whether it will be temporary just for the test, or whether it will be considered “permanent” for all other future configurations. If successful, the Randa/BCDC combination should be used for the Colorado Department of Education but also could be back-fitted into other Randa legacy systems currently operating in three other states.

Philos™ attempts to include a wide variety of security standards, like Advanced Encryption Standard AES-256, and Transport Layer Security (TLS). All external interfaces are designed to use synchronous encryption, initially facilitated by asynchronous public key infrastructure (PKI) encryption standards. Although actually more of a compliance regime than a standard, Philos™ should meet or exceed the mandates of Cybersecurity Maturity Model Compliance (C2M2), and of the North American Electric Reliability Corporation (NERC) Cyber Security Supply Chain Risk Management Standard CIP-013-1 for Bulk Energy Systems. The rationale is to meet a wide range of security requirements for cybersecurity resilience, monitoring, operational feedback, and compliance with government

mandates for critical infrastructure protection of energy and supply chain systems, and to provide open architectures to the greatest extent possible while protecting customers using advanced cryptology methods, and unique blockchain products. We know of no existing standards that are available or would apply to blockchains in general, or Philos™ in particular, or to some of the security services that it would provide, simply because it is currently unique, and no standards have yet been accepted.

Advanced Model-Based Systems Engineering Open System Architecture (OSA) Questionnaire

The following OSA Questionnaire applied to the BlockFrame® system architecture is a useful checklist to determine whether open architecture principles are being applied.

A. Modular designs based on standards, with loose coupling and high cohesion, that allow for independent acquisition of system components:

- Question: Does the system design decouple hardware, operating system and middleware from applications?
 - Answer: Yes. All internal functions within a Systems Container are partitioned, firewalled, and disconnected from the User apps outside the trusted security layer of the System Container interfaces.
- Question: Are the system's applications functionally segregated to function as independent entities (emphasis on single responsibilities and separation of concerns)?
 - Answer: Yes. All internal components are functionally separated. All interactions among separate System Containers are also separated. Since some of the functional components inside of System Containers are all contained within the trusted security layer within the same System Container, there might not be separate partitions within the same units.
- Question: Can the computing hardware be upgraded without the necessity to change operating system, middleware or applications?
 - Answer: That might depend on what needs to be upgraded. Obviously, some interface software might need to be changed with the new hardware, to assure the compatibility of interfaces with new User products, but generally not. In some cases, if some hardware components crash or are compromised, it might be possible to remotely re-provision their software components remotely back to the state they had prior to the crash.

- Question: Are the functional components of the system well defined with clearly specified behaviors and interfaces?
- Answer: Yes. Virtually all components exist within System Containers, which might have multiple internal components, but each internal component should execute its own particular function.
- Question: Are unique or proprietary solutions used only when they have published interfaces and use widely accepted standards?
- Answer: All interfaces with System Containers should be standardized with open architecture principles and standards. Some internal components are based on intellectual property, but that should not have much to do with the interfaces, which should all be open architectures. However, since the entire Enterprise (with components within System Containers) are based on cryptological secrets for cybersecurity for corporate and/or User secret data, for those reasons some components, software, or processes could be held secret without prior permission. (Obviously, there could be future configurations where some or all of an instantiation could be a closed architecture if required by, for instance, a classified military application or system.)
- Question: Are the software development environment tools for each application "industry standard," and openly available as a set of products?
- Answer: All software development packages that I am aware of use standard programming languages, although some different languages could be used for things like database management, time-sensitive routines, or other specialized functions. Some code or blockchain creation routines might not comply with certain common standards, like some of the NIST standards, because many of the functions are probably not yet defined by NIST.

B. Enterprise investment strategies that maximize reuse of proven hardware system designs (these questions apply when a system is being developed as an element of a larger enterprise):

- Question: Has the system development program investigated potential reusable/ modifiable components from other programs?
- Answer: The Philos™ Enterprise and the Eco-Secure Provisioning™ (ESPTM) Enterprise are designed for compatibility with each other, and for very large-scale compatibility with a wide

variety of other Users. In that sense, the design could be considered “reusable” for a very-large multiplicity of Users. Both systems, but the ESPTTM in particular, are designed to be able to deliver coordinated updates of code versions, updates, and patches remotely through Continuous Integration/Continuous Delivery (CI/CD), so those might also be considered as “modifiable” components. If the intent of the question is whether PhilosTM or ESPTTM will use reusable or modifiable components from other systems, the answer is generally no because of their unique nature of their security functions. However, the use of some COTS electronic components, operating systems, etc., are planned within the design, but might be ordered under specific security design specifications.

- Question: Are development contractors/subcontractors incentivized to identify potential reuse candidates from a broad spectrum of providers?
 - Answer: As in the previous answer, some electronic components would be used, but most components are internal and must be custom-designed. In some cases, specialized sub-components or data storage components will be considered and used, but generally not for the customized components. In one case, a “front end” component was considered and evaluated as to whether it could operate as a “plug-and-play” internal component, but it turned out not to be as desirable as current custom design.
- Question: Does the development program use, modify, or extend data models, service taxonomies, and other potentially reusable elements?
 - Answer: Generally customized modules and code are unique and do not lend themselves to reuse of non-customized elements. Services and service level agreements should accompany User contracts and should generally be standardized and reusable but could also be specialized for individual User agreements. Reuse of components, data models, service taxonomies of the PhilosTM Blockchain across several Use Cases is definitely likely, but since there are no known near-peer products, it is currently unlikely that they would probably not be exported outside of the PhilosTM Enterprise.
- Question: Has the system development been planned to include separate contracts/ product procurements for the various components of the system in a “best of breed” strategy?

- Answer: Generally, custom-developed components are unique and cannot be separately acquired. If Philos™ or ESP™ interfaces with external systems or components, there might be a possibility of “best of breed” external interface components on the User side. For internal components, like an operating system as an example, different alternatives might definitely be considered but would be selected on the basis of the best value and performance.
 - Question: Have incentive structures such as tiered fees and future business opportunities been built into the program plan and contracts to reward cooperation and collaboration among the architect, integrator, and component providers?
 - Answer: In a sense, yes. Since the Philos™ and ESP™ networks are designed to involve “cooperative” processes – such as Users providing storage space, selling or transferring energy into the system, or other “exchange” agreements for mutual services within the network – those considerations could result in fees-for-service, discounts, and/or premiums to account for those cooperative processes.
 - Question: Does the system development program exploit and contribute to repositories of reusable architecture and design artifacts, including reference architectures?
 - Answer: RAs are not actively used yet but are being discussed because of the probability of a highly repetitious number of Users and Use Cases. Generally, the repetitious of the use of Philos™ System Containers as common components should be considered to be RAs, as well as the repetitious use of ESP™ components. Many Use Cases are so similar it is also likely that new Use Cases for different Users will follow very similar, if not almost identical interfaces and service level agreements. As both Enterprises become operational, it is very likely that RAs will be very useful, both for repeating similar patterns, but also through making minor changes to adapt to new business opportunities.
- C. Enhanced life cycle sustainment for software intensive systems through proven technology insertion and software product upgrade techniques:
- Question: Does the system development program include provisions for long term technology refreshment, additional/enhanced software capabilities, emerging technologies and products, and other evolutionary measures to maintain operational effectiveness and affordability?

- Answer: The Philos™ Blockchain is specifically designed with end-of-life considerations in mind, largely as a reaction to the lack of that consideration in other blockchains where their unsustainable growth and size make them unsustainable. Updates or revisions should be able to be re-provisioned through Continuous Integration/Continuous Delivery™ (CI/CD) processes for most software components.
- Question: Does the system design promote lower life cycle cost by leveraging modularity to reduce the effort and cycle time of system modernization?
 - Answer: As mentioned in the last answer, this is one of the primary motivations for the Philos™ Blockchain. There have been major concerns expressed about the long-term viability of other common blockchains due to over-growth of the size of their chains, and therefore their expensive storage requirements and costs imposed on Users. The costs of extensive “data mining” is also extremely wasteful of valuable energy resources, and as those costs exceed those blockchains’ profitability, they will likely be simply too expensive to maintain and will not be able to maintain their modernization cycle times. Philos™ does not use those techniques, thereby avoiding those particular life cycle cost problems.
- Question: Does the system design appropriately exploit commodity COTS computing and networking hardware to reduce procurement and maintenance costs?
 - Answer: Yes, in a sense, but probably not as an external service. Since computing of transactions, and/or encryption calculations, will generally either be done by Users, or by Philos™ as a service to Users, the computational, hardware, and User Use Case development should often be a voluntary, shared responsibility, possibly with discounts or premiums applied to either service-level agreements, or perhaps to each cooperative blockchain transaction, depending on the terms of the agreements.
- Question: Are decisions to use specific COTS products supported by test results, architectural suitability, “best value” assessments, etc.?
 - Answer: The broadest use of COTS products will probably be in custom-ordered COTS electronics components. For both the Philos™ and ESP™ Enterprises, some initial testing and proofs-of-concept have already been completed, but generally with prototypes, breadboards,

and/or not-to-scale components. As opportunities become available to scale up for large-scale production, some agreements are already in place to move to more efficient, less-expensive electronic components.

D. Reduce development risk by maintaining transparency of system designs, continuous design disclosure, and peer reviews:

- Question: Are the system/subsystem/component/application specifications and design data available to a broad cross section of potential providers?
 - Answer: All system designs have followed Agile development practices to the greatest extent practical and are vetted in group settings for review. Due to the unique security designs, some of which involve sensitive intellectual property with national security implications, disclosure is generally limited to “need-to-know” partners, and not across broad public disclosure.
- Question: Are the end users and other system stakeholders included in the system design and upgrade process as well as the definition of training requirements?
 - Answer: That is currently practiced and planned following partner agreements. Partnering Users have effectively become co-developers and share information on things like interfaces and training. End Users (i.e., customers of the User that interfaces with the Philos™ blockchain), might not have a need to know about the internal working of Philos™. (Presumably, they would also not even care about how it works.)
- Question: Are independent reviews of system designs, component selections, risk assessments and mitigation plans, integration and test plans, etc. conducted on a periodic basis?
 - Answer: The status of Agile development “Stories,” are generally reviewed on a weekly basis, or as necessary as development progress is made. Due to the amount of work required with a limited number of development personnel, prioritization is also pretty “agile,” and often driven by the availability of resources. As resources become available, workload and growth are typically accelerated.

E. Effective use of data rights to promote initial and subsequent competitive procurements and access to alternative solutions and sources across the system life cycle:

- Question: Have the appropriate data rights been obtained with each system component, especially software applications?
- Answer: Generally, internal data for the operation of Philos™ or ESP™ are the internal property of BCDC or BlockFrame® and would not be exchanged as data. Customer data may be held by Philos™ or ESP™ temporarily, or encrypted and decrypted, but generally for “permanent” storage (by agreement) on User storage equipment, and/or on the public blockchain. Philos™ or ESP™ would generally not have “data rights” to User or End User data. Any data held on the blockchain, or the governing rules, would be governed by the Neutral Governance Organization (NGO).
- Question: If a product contains proprietary elements, are the license requirements for use clearly documented, and those proprietary elements segregated with well-defined interfaces such that modification of another component will not require modification of the proprietary product?
- Answer: For Philos™ or ESP™, proprietary elements are internal corporate property – except perhaps for the use of common commercial COTS operating systems or languages, like Linux, JSON, GoLang, etc., that would be used under licensing agreements. Since all BlockFrame or BCDC-proprietary information, business information, or intellectual property that is “instantiated” into the system would be contained within secure System Containers, they should all be highly secure behind corporate-designed and operated interfaces.
- Question: Have the asset packages (i.e., the deliverables) been reviewed prior to Government acceptance to ensure that they reflect the agreed upon licenses and data rights markings?
- Answer: No commercial productization has been accomplished to date¹⁹¹. The first use of Philos™ is likely to be as a sub-contract for the State of Colorado. Upon delivery, licenses and data rights markings would probably be part of the compliance with both the State contract, as well as the service-level agreements.

¹⁹¹ Since the date of this writing, BlockFrame® a public commercialization process, but that will not be described in detail here.

Appendix D: Risk Analyses of Selected Functions of the Philos™ Blockchain/ Distributed Ledger and Ecosystem¹⁹²

Introduction

This appendix contains a compilation of several examples of risks and risk analysis techniques that could be applied to the BlockFrame® system-of-systems. These were all researched as part of CSU research on risk analysis but are not specifically connected to the scope of the dissertation research objectives included in Section 7.2. They are documented and retained here because they could be used to support future business decisions and use cases in the Philos™ Ecosystem.

Risk Information and Models^{193 194}

Risk Factors in Philos™ Blockchain Funding, Development, and Operationalization

In this example, three risks are selected, and a standard risk model (diagram) was used to depict them. For each diagram, the risk event, impact, drivers, probabilities, and total loss estimate are included. A typical standard risk model (diagram) is used as shown in Figure D.1:

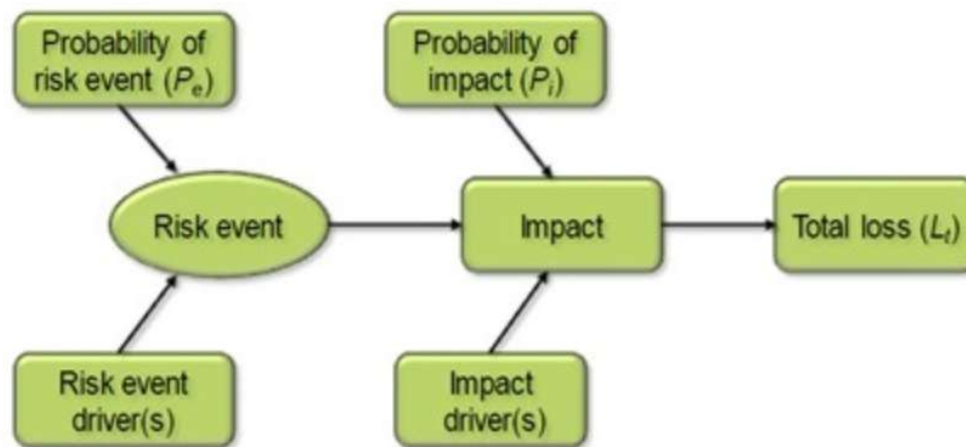


Figure D.1. Standard Risk Model Diagram^{195 196}

¹⁹² Lambert, Kent. Engr 531, Engineering Risk Analysis. Colorado State University. August 4, 2019.

¹⁹³ Lambert, Kent. Engr 531, Engineering Risk Analysis. Colorado State University. *Exercise 1, CSU Engr 531*. May 26, 2019.

¹⁹⁴ Lambert, Kent. Engr 531, Engineering Risk Analysis. Colorado State University. *Exercise 2, CSU Engr 531*. June 2, 2019.

¹⁹⁵ <http://www.ricksturdivant.com/2016/06/09/standardriskmodel/>

¹⁹⁶ <http://www.ricksturdivant.com/2016/06/13/benefits-of-the-standard-risk-model/>

For this example, Philos™ Risk 1, “Funding/Cost Risk”; Philos™ Risk 4, “Threat/Security Risk”; and Philos™ Risk 10, “Deprivation Risk”; are presented as shown below.

1. Philos™ Risk 1. Funding/cost risk: Assuming a startup with zero cash on hand, research and development funding commitments from users need to be identified almost immediately. What is the likelihood that revenue can be secured for all for initial system, subsystem, and component research and development (R&D), production, and operational cost goals to meet all research, development and integration to support an estimated 3 system users by 2021, 10 users by 2023, and 100 users by 2026, and then sustain a positive cash balance of 5% of the revenue? Without adequate up-front funding, the project will not be able to attain a point of long-term sustainability from other funding sources.

Cost estimates were as assumed in Table D-1.

Table D-1¹⁹⁷

Year	2019	2020	2021	2022	2023	2024	2025	2026	Total
Annual Cost \$M	0.7	1.3	1.9	2.4	2.8	4.9	7.0	9.2	30.2

From Philos™ Risk 1 assumptions above, we can assume that 3 users (0 small/3 medium/0 large) would be expected by 2021, increasing to 10 (2 small/7 medium/1 large) by 2023, and 100 (20 small/70 medium/10 large) by 2026. Total cumulative costs from 2019 through 2026 are therefore estimated to be \$30.2 million. It is required that adequate up-front revenue is needed to cover R&D and initial production costs until service fees can be charged for each end user. In this case it is assumed that it would include costs from 2019 to 2021, or \$3.9 million in total. The required revenues would be costs plus 5% per year. Estimates of the cost breakdowns per year are available, but for this purpose only the total annual costs are relevant. To continue to grow the cash balance at 5% per year through 2026, it is assumed that it would be held in reserve and would grow each year. It is also assumed that fees for small, medium and large businesses would scale based on the number of “brokering agent” (BA) and “peer ledger” (PL) system container modules required for each sized organization, based on a ratio of (BA/PL) of (.25/0) for small organizations, (1/0) for medium organizations, and (9/1) for large

¹⁹⁷ Lambert. Kent. 2019. *Exercise 1*. Data from: Lambert K and Bishop V. 2019. *Engineering the Philos™ Blockchain System*. Final report, Engr 530. Colorado State University.

organizations. Assuming each BA and PL subsystem will cost users \$150,000 and \$300,000 in annual fees respectively, the total fee revenue per year can be estimated. Not knowing those exact costs, those fee revenues are calculated parametrically to estimate what the fees would have to be to cover the system costs through 2026.

Using Excel to visualize the funding and cost relationships, and using the assumptions above, Figure D.2 shows the required revenues, both from required “up-front” investment and sustainable revenue from user fees. Following the assumptions above, securing up-front investment revenue of \$4.095 million to cover initial R&D costs for three years should cover costs until 2021. However, if annual fee revenue is only available after 2021, annual costs may exceed revenue each year until 2025 due to the slow assumed rate of integration with end users during that period. That might be mitigated if the integration schedule is accelerated or if revenue is increased. After 2025, it appears that annual revenue can sustain annual costs.

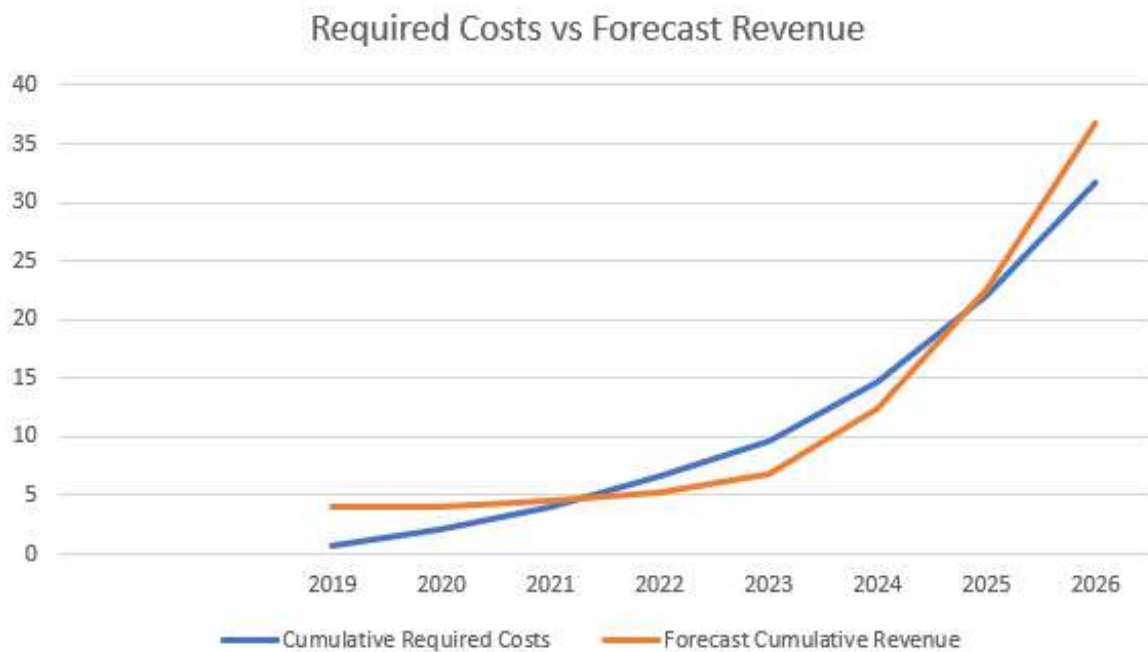


Figure D.2. Required Costs vs Forecast Revenue (without assumed risk)

Funding/Cost Risk Event	Risk Event Driver	Risk Impact Driver
Revenue from investments	Expected investors refusing to invest	Insufficient start-up revenue
Revenue from annual user fees	Expected end users refusing to purchase	Unsustainable growth
User fee level	Variability could price users out of market	Unsustainable growth
Integration schedules with end users	Faster or slower schedules possible	Positive or negative effect
Inflation risks	Could increase current expected costs	Less profit if sales prices were fixed

Figure D.3. Risk events, event drivers, and impact drivers

Figure D.3 shows that there could be several possible risks that could affect the problem such as risk event drivers and impact event drivers. Because these probabilities are currently unknown, and could be mutually dependent, it is assumed for this example that expected revenue from investors is already assured and will not change. User fees could be changed to attract more revenue, but that could also drive some users out of the market. For this report, it is assumed that the user fees are fixed and reasonable because they are about as low as possible and still meet sustainable goals in 2026. Long-term inflation could be a factor, but it is assumed that inflation might not be highly significant over the seven-year period. If all those assumptions are made, the remaining variable for discussion would be the end users' decisions not to buy the Philos™ service at the assumed prices. Even though a complex multi-variate analysis could be considered, using the end users' decisions as a surrogate for all the other variabilities should be sufficient for an initial estimate. Based on subjective estimates of the customer base, it is assumed that government organizations and private organizations would purchase on a 50/50 ratio, however, for a variety of reasons government purchasers might be less likely to commit than private organizations. On an annual basis with the other risk event drivers constant, it is forecast that expected end users refusing to purchase, might be 10% for government organizations, and 4% for private organizations. If those expected end users drop out, it can be expected that new users will be found to accept new contracts, but it is assumed that one year of revenue would be lost before new contracts could be put in place with a 1-for-1 replacement. Therefore, the total estimated annual loss from risk could be calculated as: $Le = Pe \times Pi \times Lt$.¹⁹⁸ $Pe = (0.5 \times 0.90) + (0.5 \times 0.96) = 0.93$; $Pi = 1.0$. Lt = Previously forecast annual fee revenues. Therefore, $Le = 0.93 \times 1.0 \times Lt(t)$. Using Excel with those calculated risks, Figure D.4 shows the revised Required Costs vs Forecast Revenue estimates. The total reduction, Lt , in cumulative revenue due to the assumed risk is about \$36.76 million - \$34.47 million = \$2.29 million over the specified time period. That also exacerbates the magnitude of the expected revenue shortfalls

¹⁹⁸ Miller, E. 2019. *Lecture Notes Module I, Engr 531, Engineering Risk Analysis*. Slides 122-123. Colorado State University. May 22, 2019.

between 2021 and 2025. In summary, the standard risk model diagram for funding/cost risk is shown at Figure D.5.

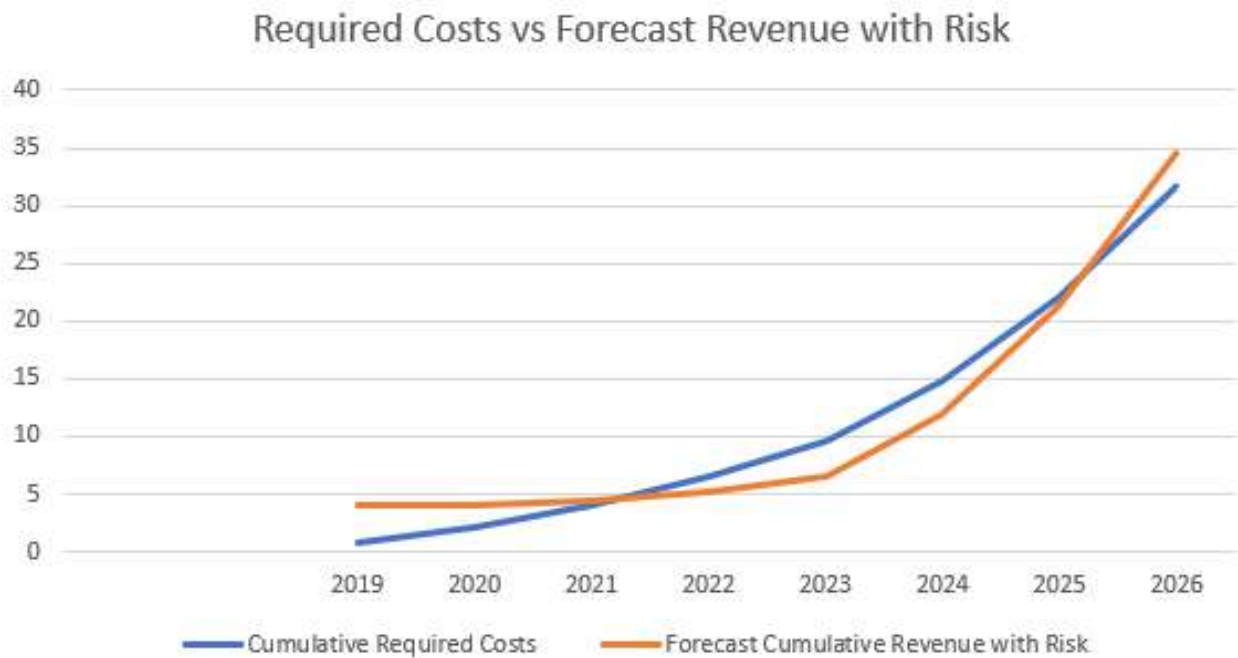
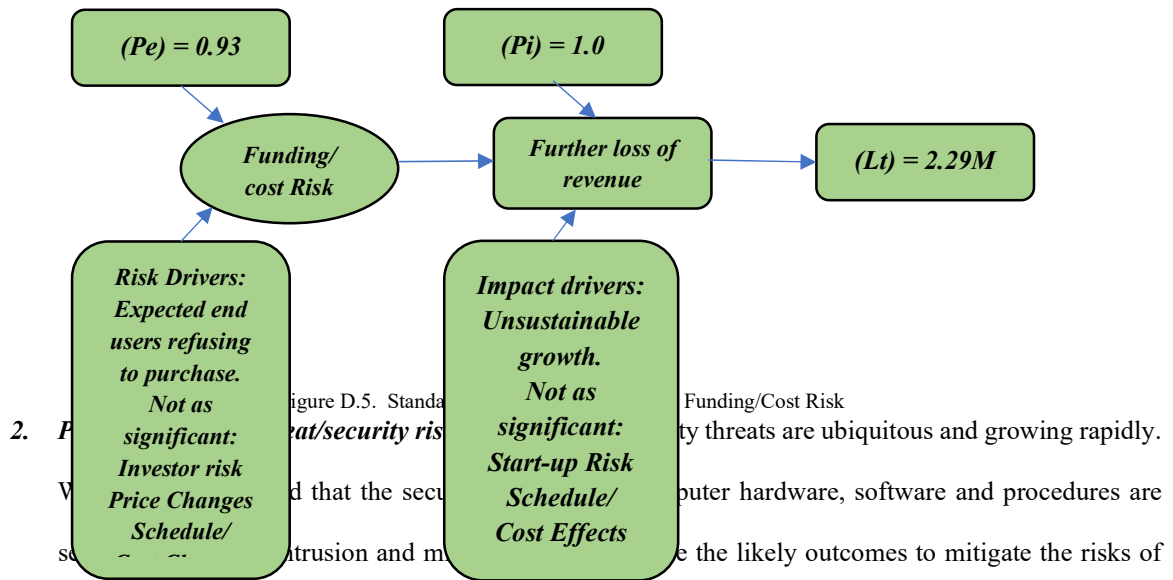


Figure D.4. Required Costs vs Forecast Revenue (with assumed risk)



end user systems once they are integrated with Philos™?

Cybersecurity risk is a major consideration of the fundamental design of Philos™. While it is too early in test and evaluation to complete comprehensive cybersecurity penetration testing, the deliverable Philos™ product must be very robust from hostile intrusion. Since its engineered design strategy is also based on defense-in-depth, a zero-trust cyber-security philosophy, and robust encryption standards, the Philos™ system should be assumed to meet a requirement of greater than a 0.999 resistance to hostile penetration over time. That, in turn, should help deter hostile actors that might be looking for an easy target.

Philos™ is also designed to help increase trust (lower risk) using five specific cryptographic cybersecurity methods that were discussed in detail in Section 7.2. Those methods should help host end-user systems to mitigate those specific cybersecurity risks. Philos™ system containers, that contain brokering agents (BAs) and peer ledgers (PLs), from the systems engineering viewpoint will act as common security components within a broad system-of-systems. Three main threat/security problems still might remain:

1. Since Philos™ units will typically reside either within or nearby host servers, the interfaces between the host servers and the Philos™ software (or physical units) could make the interfaces themselves attractive targets for attackers to attempt to penetrate the host systems.
2. Interactions between Philos™ software, operating systems, and customized user applications should always be required to be tested for compatibility prior to operational use. However,

there is a non-zero probability that some unknown incompatibilities could still exist, even after testing, that might not have been detected during testing. The “deprivation trust” methodology for Philos™ Risk 10 will be discussed in the next section and is intended to identify those unexpected abnormalities. It therefore might catch some unknown anomalies that got past the test so they can be examined and corrected even if they have been previously missed.

3. The end users’ hardware, software, and procedures might have other internal vulnerabilities that cannot be solved by Philos™. While the use of the Philos™ blockchain should increase trust over time, it is not intended as a comprehensive cybersecurity system against all threats for all users. End users will ultimately have the responsibility for the security of their own systems, and Philos™ operators will likely not be able to direct the configuration of government or corporate security systems without mutual agreement. In summary, interfaces, untested interactions, and end-user-specific security problems should be decreased within the Philos™ framework, but some risks might remain.

According to the United Kingdom National Cyber Security Centre (UKNCSC), 81% of large businesses reported a breach, with the average cost of each breach of £600k - £1.15m.¹⁹⁹ That document also lists “10 Steps to Cyber Security” that contain several good general recommendations for improvement. Although Philos™ cannot apply every solution to every threat, it can be characterized as helping to identify, target, and mitigate many key threats over time.

To populate a standard risk model diagram, the following assumptions are made:

- The Philos™ system itself is significantly hardened and will be a minimal risk to the system.
- 81% of user systems may be breached over time. Assuming a ten-year span, and assuming a uniform threat over time, that could roughly translate into approximately an 8.1% expected probability of a breach per year. This would aggregate all attacks, but since the threats might dynamically change over time, it does not specify or anticipate every possible attack vector.

¹⁹⁹ United Kingdom National Cyber Security Centre. <https://www.ncsc.gov.uk/staticassets/documents/NCSC%20Cyber%20Attacks.pdf>

- For this exercise, all large, medium, and small organizations will be assumed to have the same level of vulnerability. Although large companies might be more attractive targets, small companies might typically be more vulnerable, so a simple uniform probability is assumed.
- From the UKNCSC source, it is assumed that damage from breaches for large corporations can be expected to lose a value of £875 loss per breach, or about \$1.1 million USD.
- For sake of scaling, it is assumed that the ratio of Philos™-served businesses is 20% small, 70% medium, 10% large.²⁰⁰
- Expected damages to medium and small businesses per breach would be one and two orders-of-magnitude lower than for large organizations respectively: \$110k for mediums, \$11k for small.
- Other losses could typically include loss of personally identifying information (PII), or other non-monetary losses, but for the purposes of comparison, this exercise will only consider and compare monetary losses.
- If an attack against an end user without Philos™ is not mitigated, it is assumed the probability of impact (damage) is 0.50. For a Philos™-installed system, it is assumed that probability of damage is reduced to 0.10 initially and is then reduced thereafter by 80% per year over time. The reason for that impact reduction is that the Philos™ Blockchain should initially reject untrustworthy transactions, and then “poll” transactions and identify and mitigate other discrepancies and threats over time.

To calculate an unmitigated threat, $Le = Pe \times Pi \times Lt$.²⁰¹ Where $Pe = 0.081$ per year; $Pi = 0.5$ (unmitigated); $Lt = \$1.1M \times 0.1 + \$110k \times 0.7 + \$11k \times 0.2 = \$189k/\text{unmitigated event}$. Therefore, $Le = 0.081 \times 0.5 \times \$189k = \$7654.50$ per year.

To calculate a mitigated threat, $Pe = 0.081$ per year; $Pi(t) = 0.1 \times 0.8^t$; $Lt = \$189k$. An Excel comparison of unmitigated and mitigated threats is shown in Figure D.6.

²⁰⁰ Lambert Kent, and Bishop V. 2019.

²⁰¹ Miller, E. 2019. *Lecture Notes Module I, CSU Engr 531*. Slides 122-123. May 22, 2019.



Figure D.6. Expected Loss (Le) per Year (in \$1000)

If left unmitigated, the expected loss (Le) is \$7,655 per year. Those relationships are shown in Figure D.7a. If the probability of incidents is mitigated from an expected 0.5 to 0.1 initially due to cybersecurity actions by initiating Philos™, and if the Pi is further mitigated each year by 0.8, the expected loss (Le) should be reduced initially to \$1,531 per year in the first year, and to \$321 per year by 2026. Those relationships are shown in Figure D.7b.

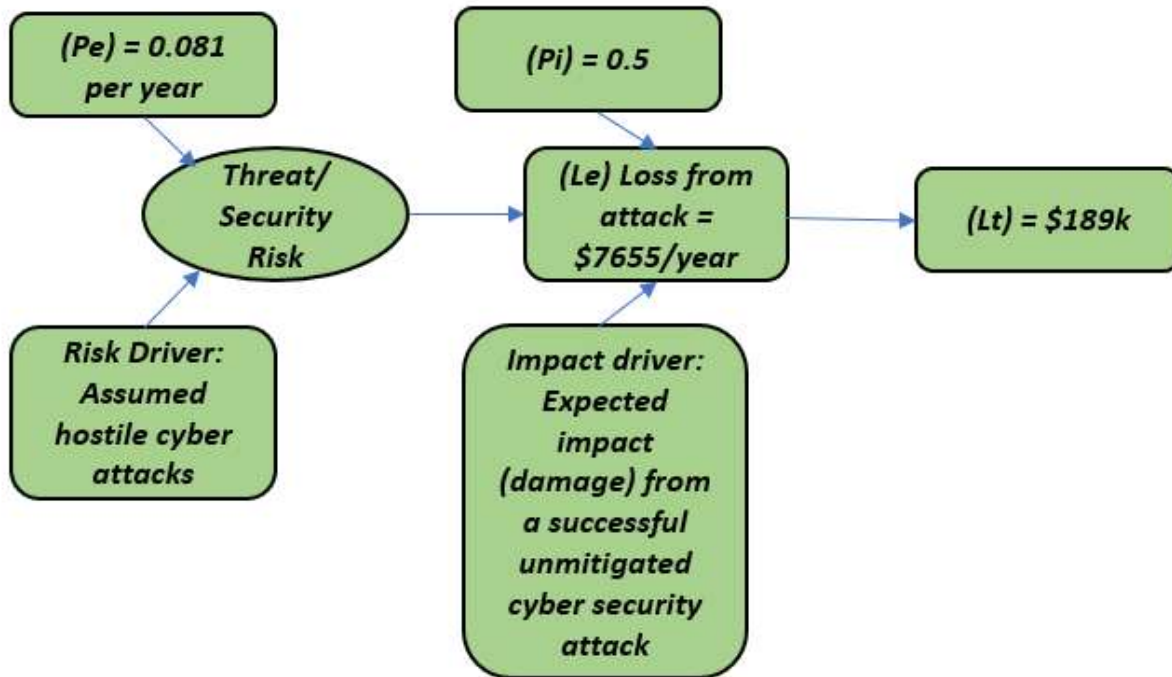


Figure D.7a. Standard Risk Model Diagram for Threat/Security Risk (Unmitigated)

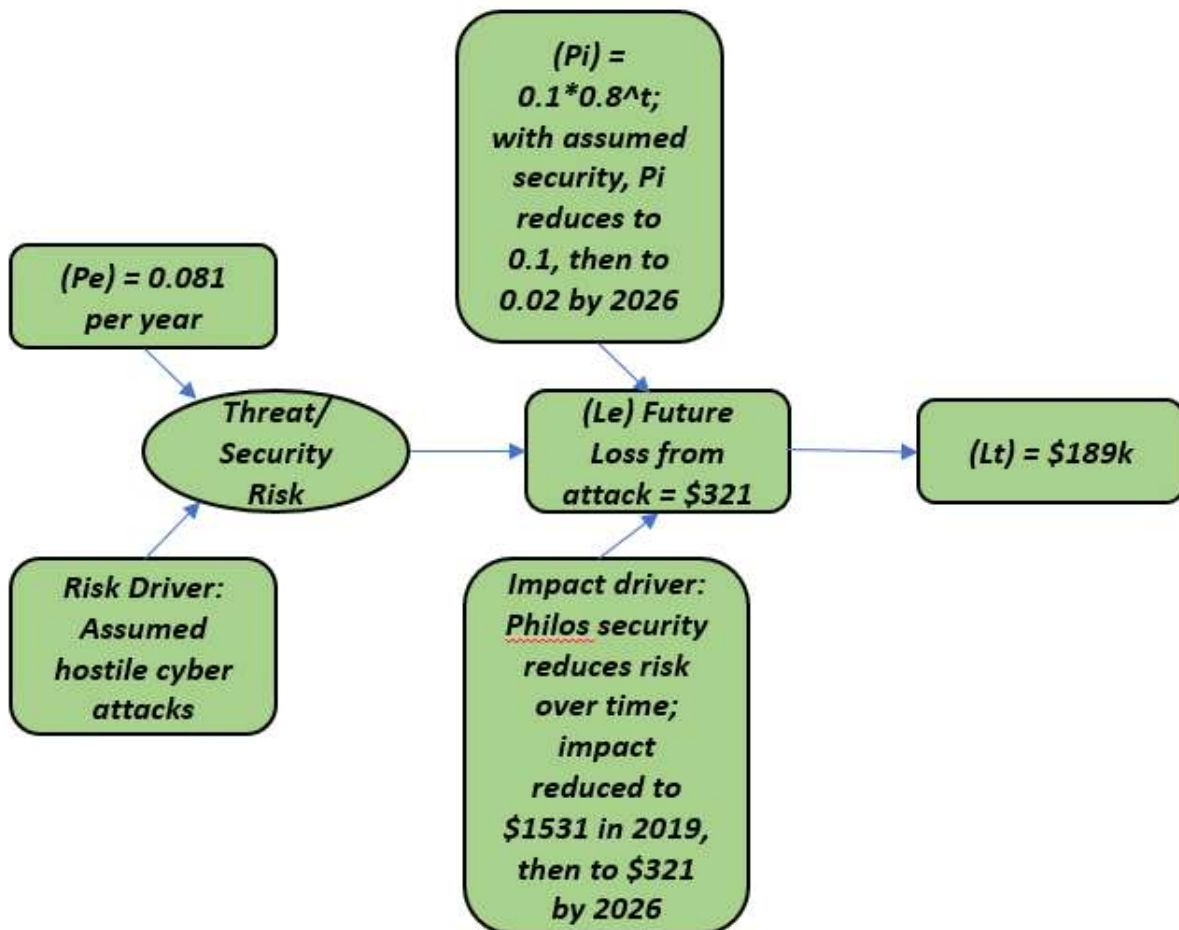


Figure D.7b. Standard Risk Model Diagram for Threat/Security Risk (Mitigated)

3. **Philos™ Risk 10. Deprivation Trust (TDP) Risk:** “A blockchain transaction entry incorporating the resultant degradation information in relation to a reliability trust transaction.”²⁰². This is a quantifiable measure of trust related to cryptographic operations to prove data availability and examinable degradation over time to reduce the risk of abnormally losing information that could otherwise be examined and corrected.

Of the five trust-increasing/risk lowering “categories of trust,” deprivation trust (TDP) is unique²⁰³. The first four Philos™ risk factors have been designed to cryptographically prove that conditions have been met to prove that a trust relationship exists. Once those factors have been proved, then theoretically the fifth factor should always be 1.0, implying that there should be no unexplained discrepancies in data availability. However, if the fifth factor, if the probability of deprivation trust, is anything less than 1.0, that should be an alarm that something else is abnormal in the data, and that further examination is necessary. (I.e., The probability of risk ($Risk_{TDP}$) would therefore be greater than 0.0.) The TDP factor will probably not identify what is wrong but should dictate that further examinations should attempt to isolate the fault and take action to correct it. Over time, if those faults can be associated with a particular user data set or other data or transaction sources, that could be evidence that a data breach might have occurred, that data might have been exfiltrated by hackers, that infiltrated bots might have become active, that denial-of-service attacks might have occurred, that there is a catastrophic failure in the cryptological system, or that significant hardware or software faults might have occurred (possibly in end-user applications). If things are operating normally, that should never happen, but that’s just the point, this event would not be normal. That’s essentially the purpose of whether we can trust data to be available. If hackers *deprive* us of any data, especially over time, deprivation trust is supposed to detect that. Again, TDP isn’t supposed to prove how it happened, just that the normal trust chain, or something outside of the normal trust chain, however slight, went wrong over time. Some systems engineers might initially call that detection of an “unknown unknown.” Figure D.8 shows a standard risk model diagram for Philos™ deprivation trust. It should normally be impossible that $Le > 0$, but if Le is in fact greater than zero, that’s a warning that Pe , Pi , or Lt might also

²⁰² Gorog C. 2018.

²⁰³ Lambert K. 2019. *Exercise 1, CSU Engr 531*.

be non-zero, and that trust has been violated somewhere, however slightly. In that case, the magnitudes of a positive Le might be irrelevant because any such event could be major or catastrophic. Loss might also appear negligible, but it could be discoverable. Any positive value of Le could initially make this a logical Boolean problem that a problem exists (Y/N), and trust alarms should go off to audit and fix that problem.

A “zero-trust” cyber-security strategy would assume that the system-of-systems should always be looking for faults over time, “just in case,” even if they are initially believed to be impossible. Whatever the cause, such a violation of trust could potentially indicate a catastrophic systemic failure.

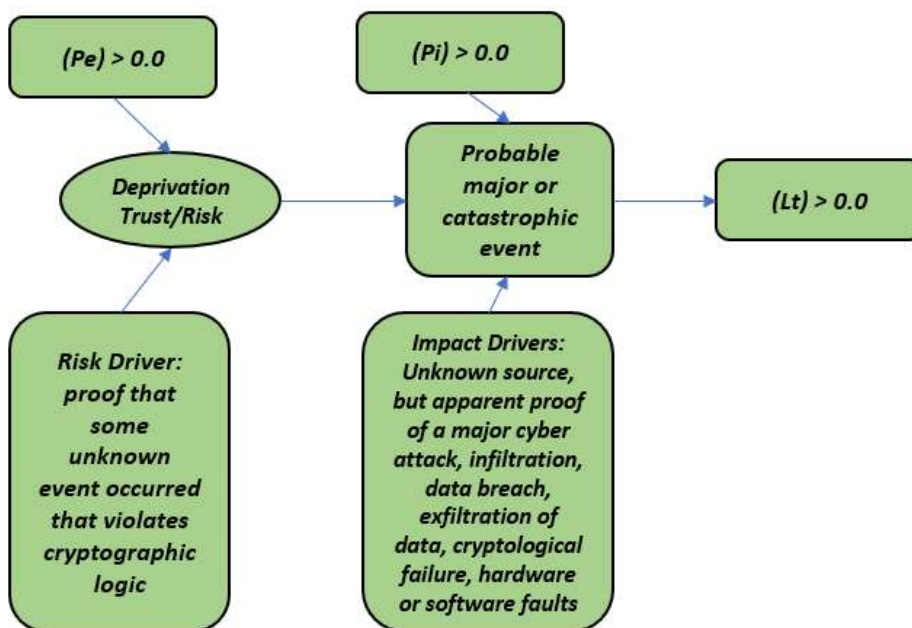


Figure D.8. Standard Risk Model Diagram for Deprivation Trust (TDP) Risk

Preliminary Hazards Analysis (PHA)²⁰⁴

This section applies a PHA to Philos™ to identify potential hazards to humans using a table format with columns for: Subsystem, Mode, Hazardous Element, Event Causing Hazardous Condition, Hazardous Condition, Event Causing Potential Accident, Potential Accident, Effect, Hazard Class, and Accident Preventative Measure²⁰⁵. Five example hazards are considered.

Since Philos™ is designed to provide cybersecurity for an actual system-of-systems, and would typically involve highly virtual environments, hazards to humans may typically be considered to affect end users by financial loss, loss of confidentiality (privacy or authentication), loss of integrity (information integrity or non-repudiation), loss of availability of data, or damage to the system itself. For this report, this Preliminary Hazard Analysis (PHA) is applied to the system, not to procedural or product design. This PHA is limited to only five rows representing different hazard examples at a high level. The actual system-of-systems could be many orders of magnitude more complex than that. The actual system involves the Philos™ blockchain system, specialized networks, and interfaces to end-user systems. For simplicity, we will just refer to the entire system-of-systems as “Philos™.”

Since typical relative hazard risk charts do not adequately define or characterize the cybersecurity hazards of systems, some representative risks as shown in Table PHA.1. The Philos™ Preliminary Hazards Analysis is shown in Table PHA.2.

Table PHA.1. Philos™ Revised Hazard Categories

- Class I Hazards: Negligible. Worst-case effects will cause less than minor financial loss, loss of confidentiality (privacy or authentication), loss of integrity (information integrity or non-repudiation), loss of availability, or less than minor system damage
- Class II Hazards: Marginal Effects. Worst-case effects will cause minor financial loss, loss of confidentiality (privacy or authentication), loss of integrity (information integrity or non-repudiation), loss of availability, or minor system damage

²⁰⁴ Lambert, K. June 16, 2019. *Exercise 4, CSU Engr 531*.

²⁰⁵ Ostrom & Wilhelmsen, *Critical Functions for Continuity of Operations*, in Ostrom & Wilhelmsen, which originated from *FEMA Continuity of Operations Plans (COOP) and Continuity of Government (COG) Plans*. Table 6.2, p. 71.

- Class III Hazards: Critical. Worst-case effects will cause severe financial loss, loss of confidentiality (privacy or authentication), loss of integrity (in information integrity or non-repudiation), loss of availability, or severe system damage
- Class IV Hazards: Catastrophic. Worst-case effects will cause death, catastrophic financial loss, loss of confidentiality (privacy or authentication), loss of integrity (in information integrity or non-repudiation), loss of availability, or catastrophic system damage

Table PHA.2. Philos™ Preliminary Hazards Analysis

Subsystem	Mode	Hazardous element	Event causing hazardous condition	Hazardous condition	Event causing potential incident	Potential incident	Effect	Hazard class	Effect prevention measure
Financial business functions of planning schedule and/or revenue	Primary concern is during pre-operational system R&D phase, but could occur at other times	Financial funding/ cost system to support system R&D, test & evaluation, production, integration, and/or operations	Revenue reductions due to Investment, management performance, personnel or legal failures; or adverse competitor actions	Loss or reduction of system R&D, T&E and operating revenue before sustainable funding	Triggered by failure to secure revenue, loss of funds due to loss of investor confidence, or legal disruptions	Reductions of funding significantly lower than planned for necessary startup or sustainment	Loss or delay of system, loss of supporting finances, WBS delays, financially unsustainable operations	Class III. Since the system is currently a startup, loss of expected revenue could cause severe long-term damage to system sustainability	Allow adequate time, focus, and management resources to assure initial funding for Philos™ system meets sustainable goals and timelines
End user-designed apps, data and human interfaces, data storage	Routine ops. Probability of attack could increase during code or patch changes	Vulnerable end-user security elements including firewalls, un-coded data, user code, and data and human interfaces	Successful hostile cyber-attack to penetrate, probe or exfiltrate end user systems or the Philos™ system	Successful cyber-attack against end users for successful theft of data; and lack of adequate effective user cyber defense	Failure to effectively and fully secure end-user elements including firewalls, un-coded data, user code, data and human interfaces	Cyber penetration allowing further exploits, system breach, loss and/or theft of data	Primary: Loss of confidentiality, integrity, and/or availability. Secondary: Loss of privacy, authentication, info security, non-repudiation, and system degradation for end-users	Class III. Successful attacks against end users can cause severe financial or system damage. Possible data recovery or containment of effects to one end-user	Segmentation and defense in depth. Apply Philos™ system to contain hazards through confidentiality, integrity and trust controls and to assist re-constitution after damage
End user apps, networks, possibly parts of the whole Philos™ system	Routine ops. Probability of attack could increase during code or patch changes	Vulnerable end-user security elements including firewalls, un-coded data, user code, and data and human interfaces	Hostile cyberattacks to shut down user systems and/or to demand ransom	Successful denial of service attack and/or ransomware against end users; lack of adequate user cyber defense	Cyber penetration followed by insert of smart “robotic” malware code to destroy system, commit extortion or spread to other sub-systems or systems	Potential total loss of user systems. Smart bots could spread to other networks and/or systems. Attacker could financially extort users	Total loss of user systems or networks including confidentiality, integrity, and/or availability of data. Loss of revenue and/or data through extortion that may or may not be recoverable	Class IV. Successful attacks can cause catastrophic operational or financial damage against end-users. Local subsystem damage might not be recovered or contained	Conduct regular secure encrypted backups through Philos™ to use the cryptology-based Philos™ blockchain to prevent penetrations, prove security quality and trust prior to transactions, and to re-constitute data following attacks
End-user and possibly multiple end-user apps, multiple networks, possible system-wide damage to financial functions	Routine ops, ops following initial attacks, or ops while under attack	Supply chain system, including multiple end users, comms, computers, apps, electronic and physical sub-systems	Hostile cyberattacks to disrupt end-user or Philos™ networks, operations, divert or steal logistics, supplies, or information	Successful cyberattack against user networks or supply chains; lack of user segmentation, internal defense-in-depth, and adequate proof or trust	Cyber-attacks destroy or disrupt supply chains, possibly with extended effects over time. Could include but not require, a ransom or denial of	Shutdown of for one or more supply chains. Could include energy, military, intelligence, food, or private commerce	Time, financial, power, information, military, food, or commercial losses	Class IV. Successful attack against supply chains can cause catastrophic distributed financial or operational damage, possibly affecting entire	Conduct regular secure encrypted backups through Philos™ to use the cryptology-based Philos™ blockchain to prevent penetrations, prove security quality and trust prior to transactions,

					service attack			system operations	and to re- constitute data following attacks
Entire system-of-systems and all sub-systems including end users and networks, power grids, and societal structure	Emergency modes prior to or following major systemic damage to electrical power grids: emergency shutdowns modes or planned automatic re-constitution modes	Abnormal stress on power grids and power supplies to all end-users, and all cyber-security systems within affected regions	Major natural solar eruption event or hostile EMP attack affecting world-wide or major regional electrical power grids; inadequate preparation for resilience against solar max or HEMP attack	Loss of electrical power grid over wide areas for extended duration; external natural or wartime threat; inadequate preparation for threat and/or denial of threat	Solar flares greater than 1859 Carrington Event ¹ directed towards Earth. Successful hostile high-altitude nuclear detonation causing high altitude EMP to target North American power grid	Solar: massive deposition of charged particles into magnetosphere and magnetic poles, disrupting or destroying power and electronic devices. HEMP: targeted high-altitude nuclear event creating radiation resulting in regional electronic destruction or degrades of power and electronics	Broad long-term damage to energy sources and distribution. Secondary effects could include almost total loss of energy, communication, transportation, food and product distribution, industrial base, healthcare, governments, possibly for years until re-constitution ²	Class IV. Damage from a major natural solar flare or human attack using EMP could cause catastrophic distributed financial or operational damage, possibly causing permanent damage to power supplies, very broad secondary effects and mass casualties ²	Use event sensors to trigger pre-programmed shutdowns, rapid data backups, electronic and physical asset protection. Pre-position extra critical parts since damaged power sources and grids might be replaced over very long periods. Use the highly distributed Philos™ blockchain for automatic self-re-constitution of critical infrastructure data

Ref 1: <https://www.history.com/news/a-perfect-solar-superstorm-the-1859-carrington-event>

Ref 2: White, Dr. Richard. June 12, 2019. *Podcast: NCF-126 Cyber Security & Department of Homeland Security*. University of Colorado, Colorado Springs. Broadcast on New Cyber Frontier Podcast.

Failure Mode and Effects Analysis (FMEA)²⁰⁶

This section applies a limited FMEA to Philos™, with columns for: Process Step/Input, Potential Failure Mode, Potential Failure Effects, Severity Rating, Potential Causes, Occurrence Rating, Current Controls, Detection Rating, RPN, Action Recommended, Resp., Actions Taken *1, SEVERITY (1 - 10), OCCURRENCE (1 - 10), DETECTION (1 - 10), RPN.

Table FMEA.1. Philos™ Failure Mode and Effects Analysis

FMEA Form															
Process/Product Name: <u>Project Philos Block Chain</u>						Prepared By: <u>Kent D. Lambert</u>									
Responsible: <u>Kent D. Lambert</u>						FMEA Date (Orig.): <u>30-Jun</u>				(Rev.): <u>0.0</u>					
Process Step/Input	Potential Failure Mode	Potential Failure Effects	SEVERITY (1 - 10)	Potential Causes	OCCURRENCE (1 - 10)	Current Controls	DETECTION (1 - 10)	RPN	Action Recommended	Resp.	Actions Taken *1	SEVERITY (1 - 10)	OCCURRENCE (1 - 10)	DETECTION (1 - 10)	RPN
What is the process step, change or feature under investigation?	In what ways could the step, change or feature go wrong?	What is the impact on the customer if this failure is not prevented or corrected?		What causes the step, change or feature to go wrong? (how could it occur?)		What controls exist that either prevent or detect the failure?			What are the recommended actions for reducing the occurrence of the cause or improving detection?	Who is responsible for making sure the actions are completed?	What actions were completed (and when) with respect to the RPN?				
FINANCIAL Pre-operations business functions (or possibly later phases) to support plans & schedule of R&D, IOT&E, production, and integration phases	Loss of contracts, investments, grants, revenue, or budget problems could cause financially unsustainable operations unless corporate ops and business are well monitored and managed	Loss or delay of system schedule (WBS), delays of R&D, IOT&E, production, integration, operations; loss of service; system or corporate failure	6	Revenue reductions due to lower than expected investment, contracts, grants, management, performance, personnel, or legal failures; adverse competitor actions	3	Senior leaders' relationships & customer service with investors & end users; high visibility PR capabilities to publicize threats, contextualize concerns to customers, and positively support system reputation	4	72	Assure adequate time, focus, and management resources, and funding for Philos to meet sustainable goals and timelines; assure WBS and budget meet milestone targets; gain new contracts & grant funding	Senior leadership; PR team; WBS manager; technical developers; CFO	Some baseline funding is secured; DHS grant application submitted 6/19; other grant applications anticipated in 2019 and 2020; angel investor meetings and demos scheduled; defense contractor requirements being analyzed	4	1	2	8
END-USER SECURE OPERATIONS. Integrating and operating end-user applications with Philos blockchain to improve trust and security in confidentiality, integrity, and availability to user systems	End-user security firewalls, un-encrypted data, user code, human interfaces may be vulnerable. Hostile attacks could occur during routine operations, but likelihood increases during code or patch changes, especially through interfaces, data in transit, or in data storage	Failure to effectively secure end-user elements or the Philos system, along with a successful hostile cyberattack could penetrate, probe or exfiltrate user data, resulting in loss or theft, further exploits, system breaches, degradation, or destruction and severe financial loss	7	Hostile cyberattack penetrating security firewalls, combined with failure of internal security systems or personnel; insufficient personnel training	6	Existing cyber defenses vary with each system, but traditional firewall defenses, internal monitoring and logging have often proved ineffective for timely detection or system defense against attacks	6	252	Apply segmentation, defense in depth, zero trust and Philos system to contain hazards through confidentiality, integrity, availability, and trust controls; attempt to contain effects to one segment or end user; assist system reconstitution after damage	End users are generally responsible for security of their own apps; Philos is responsible for their system design and development; both are jointly responsible for integration, interfaces, operational governance; the Philos distributed blockchain itself should autonomously govern security of system-of-systems transactions	Consensus coding and testing is being done now on prototypes; due to distributed governance, the use of Philos should improve confidentiality, integrity, availability, and trust values for all users over time.	5	2	3	30
SECURE NETWORK OPERATIONS. End user apps, networks and data management. Routine network ops among end users, and end user use of internet of things (IoT) devices. Probabilities of attack could increase during code or patch deployments	Vulnerable end-user and network security firewalls, un-coded data, user code, data or human interfaces, transmission circuits, IoT devices could be most likely points of hostile cyber attack. Philos is intended to increase security, but is always assumed to have a non-zero probability of failure	Hostile cyberattack may degrade or catastrophically destroy user networks, apps, and/or devices; revenue loss, unrecoverable data loss or extortion; disruption or failure of user systems could delay operations and disrupt major system operations	8	Successful hostile denial of service or ransomware attacks against networks or the Philos system; inadequate user defenses; cyber penetration followed by insert of smart "robotic" malware to destroy, degrade, or delay systems, commit extortion, and spread malware to other systems or subsystems	4	Existing cyber defenses vary with each system, but traditional firewall defenses, internal monitoring and logging have often proved ineffective for timely detection or system defense; existing network defenses might include encryption, access and authorization controls, but is often not used	6	192	Deploy the Philos system to help detect and block hostile network penetrations using strict authentication, cryptology, distributed governance and security, and help restore systems if hostile attacks succeed	End users are generally responsible for security of their own apps; Philos is responsible for their system design and development; both are jointly responsible for integration, interfaces, operational governance; the Philos distributed blockchain itself should autonomously govern security of system-of-systems transactions	Initial prototype deployments in network test environments being planned for 7/19. Secure encrypted transmissions, synchronizations and backups through Philos blockchain designed to help prevent penetrations, prove security quality and trust prior to transactions, and help reconstitute data following network-wide attacks	3	3	2	18

²⁰⁶ Lambert, Kent. June 29, 2019. *Exercise 6, CSU Engr 531*.

SUPPLY CHAINS. Protection of complex multiple end-user apps, and systems-of-systems and IoT objects during network processing of information, financial, or physical logistics supply chains	Hostile cyber attacks could target whole systems-of-systems or single components. Attacks could disrupt end-user or Philos networks, operations, divert or steal money, logistics, supplies, or information	Supply chains with multiple end users, comm systems, computers, aocs, electronic and physical subsystems could be disrupted, or catastrophically affect energy, military, intelligence, financial, food, or private commerce deliveries. This may include ransom or denial-of-service attacks	8	Multiple simultaneous hostile cyber attacks, or malware attacks that spread through internet or other communications lines could deliver malware "smart bots" throughout systems-of-systems; not all supply chain components might need to be destroyed if "critical paths" or "critical components" are disrupted	7	Traditional cyber defenses, encryption, end-user defenses and defense-in-depth are already common, but successful attacks still occur; good logging procedures can identify some attacks, but the first recognition of cyber breaches may be several months after successful hostile penetrations, at which point damage may already be severe	7	392	Segment and secure chain nodes; deploy Philos to detect abnormalities that indicate attacks across supply chain systems-of-systems and IoT; locate and identify threat to construct counter-measures; deliver secure patches through the Philos system to secure end-user transactions. Philos should also help reconstitute damaged elements	Philos and end-user security operators should work jointly to contain and resolve chain damage; Philos should autonomously help prevent supply chain penetrations and damage, but would be managed by 24/7 security operators who could reset and repopulate data across the system-of-systems	Capabilities mentioned are currently being designed into the Philos system. Test and evaluation has not been completed, but testing on system requirements is planned for prototypes. Continuous integration/continuous delivery of upgrades and security patches and counter-measures; proof of code versions and origins is in the system design	5	3	2	30
RADIATION THREATS. System emergency mode resilience prior to or following major continental or global damage to electrical power grids, networks, and electronic and digital infrastructures from either natural or human sources and events.	Solar: massive solar deposits of charged particles into the magnetosphere could destroy or disrupt power supplies and electronic devices. HEMP: targeted high-altitude nuclear weapon could result in continental destruction or degradation of power and electronics. Damage effects could persist for years *3	Damage from solar flares greater than 1859 Carrington Event *2 or attack using HEMP weapon could cause catastrophic loss of continental energy grids, networks, logistics, transportation, food, health, financial systems, industry, commerce, and societal structure with broad secondary effects over very long durations, and mass casualties of up to 90% of the population *3	10	A combination of a major natural solar eruption or hostile HEMP attack, combined with inadequate defense preparation and electronic resilience; political or psychological denial of the threats; inadequate operational and technical planning	1	Good capability to detect solar flares. Effects not instantaneous, but time to react is uneven and problematic. Some space, military, utility assets might respond, but might be ineffective. Nuclear HEMP weapons might be detected, but automated rapid response currently inadequate to prevent major damage or facilitate reconstitution. HEMP effects also extremely unpredictable, making technologies and attack planning very costly and problematic	9	90	Contain HEMP weapon development; preposition critical replacement parts; increase electronic resilience to threats; mirror data bases; develop event sensors to trigger pre-programmed emergency backup and shutdown modes for electronic and physical assets; plan automatic reconstitution modes in the highly-distributed Philos system for automatic reconstitution of critical infrastructure data	National, military, state governments and critical infrastructure providers and budget offices should establish requirements for pre-planning, actual events, and reconstitution; plan for secure, resilient, survivable backups or mirrored data systems and procedures; most effective responses will require designed-in technologies that can autonomously respond at near-relativistic speeds	The Philos system has some automated capabilities to restore or reconstitute critical data, especially over widely distributed geographical areas that could mitigate local effects of both solar flares and HEMP weapons. Author is studying HEMP design alternatives through UCCS, the US Air Force, and presidential executive orders	7	1	5	35
* Ref 1: Actual project actions are in planning stage; for this exercise, RPN mitigation value will only be estimated subjectively to represent potential future capabilities															
* Ref 2: https://www.history.com/news/a-perfect-solar-superstorm-the-1859-carrington-event															
* Ref 3: White, Dr. Richard. June 12, 2019. Podcast: NCF-126 Cyber Security & Department of Homeland Security. University of Colorado, Colorado Springs. New Cyber Frontier Podcast															

Human Reliability Analysis (HRA)²⁰⁷

This section applies a Human Reliability Analysis (HRA) to the Philos™ blockchain system using an example use case for hypothetical Colorado drug prescriptions.

Step 1. Bound the System. This example HRA scenario process would start with the decision by a doctor to proceed to write a prescription, and end with the prescription being transmitted to a pharmacy and documented into the Colorado Prescription Drug Monitoring Program (PDMP) database. The Philos™ system-of-systems could literally be applied to millions of different use cases for different needs, as well as hundreds of state-government-specific uses. For example, this analysis is limited to demonstrate one simple use case of transactions and transfers of prescription information of an opioid (like Oxycodone) from a hypothetical doctor's office to a pharmacy and to the state PDMP.

Step 2. Perform Task Analysis.

1. A certified doctor determines that a legitimate patient would benefit from Oxycodone.
 - a. Background information²⁰⁸: Often encouraged by “Big Pharma” companies and specifically encouraged by the FDA, doctors were directed for years to treat pain first, often using opioid medications, and to worry less about long-term effects, or to seek or fund other non-opioid remedies. Subjective pain levels (1 to 10) were instituted as a “vital sign” that became a standardized practice to collect from every patient for any condition, but which could not be confirmed objectively. Without adequate follow-on, opioid pain medications typically exceeded their maximum beneficial effects after about a week but continued to cause long-term dependencies and addictions. The legal system sometimes punished responsible doctors for causing “pain and suffering” for not over-prescribing opioids. According to the DEA, legal prescription drugs became a major cause of illegal drug use. Many doctors lost their licenses over illegal drug prescriptions or “recommendations” for the use of medical marijuana. It is therefore important to monitor and prove that doctors’ credentials are valid and current. Patients also practiced “doctor shopping” to find multiple doctors to receive multiple prescriptions, or simply bought cheaper opioids, like illegal heroin, to cope with their addictions. Many also use

²⁰⁷ Lambert, Kent. Exercise 7, CSU Engr 531. July 14, 2019.

²⁰⁸ Colorado Legislative Opioid Task Force. 2017.

false identities or false credentials to obtain drugs fraudulently. (Assume that the terms “doctor” and “doctor’s office” are used synonymously.)

- b. Failure modes: Doctor misidentifies themselves (1a), or misidentifies patients (1b), or misdiagnoses (intentionally or unintentionally) (1c).
 - c. Recovery action: Philos™ can immutably and non-repudiably document the digital signature authority of the doctor, prove who the doctor is, and prove whether that doctor is currently licensed (non-repudiation and authentication) (1a), and prove the patient identification (non-repudiation and authentication) (1b). If doctors later deny submitting prescriptions, the non-repudiable evidence will already be in place to prove that they did. Because “pain levels” are subjective, Philos™ would not correct that problem, but could help permanently and non-repudiable document that patient and that “vital sign” for further consideration (non-repudiation) (1c). (Note: Other legislative and administrative actions are being considered to remove or modify subjective “pain levels” as mandated “vital signs.”)
2. Doctor checks the PDMP to determine whether the prescription meets the legal requirements for duration since the last prescription (usually seven days, with some specific exceptions).
- a. Background information: Over-prescription of prescription drugs – especially opioids – was a prime consideration for state statutes creating the PDMP and placing statutory time limits on opioid prescriptions. If used correctly, the PDMP can monitor legal limits on opioid prescriptions.
 - b. Failure mode: Doctor does not check or intentionally or unintentionally avoids or violates PDMP criteria.
 - c. Recovery action: Philos™ is designed to document and prove the origin of all data. PDMP checks could be also documented with an electronic token of proof that the doctor correctly checked the PDMP for the patient’s drug history (validation, integrity, non-repudiation). If incorrectly documented, or if the PDMP was not checked, the pharmacy could also use that information to refuse to fill in the prescription until that action is remedied.
3. Doctor writes and enters the prescription to the pharmacy and into the PDMP.
- a. Background information: Secure data accuracy is essential to safe and effective patient outcomes. Patient privacy and compliance is required by law. Although the loss or theft of patient data would

probably be committed intentionally by hackers and not doctors, the increase in tort lawsuits (including class-action suits) and corresponding increased medical insurance fees demonstrate that doctors would most likely be punished for the consequences of hacking – and could have to pay for damages and probably transfer those costs back to their patients – whether they are the cause or not.

- b. Failure mode: Doctor does not correctly enter the prescription (3a), or prescription data are stolen or modified in transit or in storage (3b).
 - c. Recovery action: Philos™ can authenticate and assure non-repudiability of the doctor’s prescription (3a) (authentication, non-repudiation) and encrypt the contents of the doctor and patient identities while data are in transit or in storage for patient privacy and data integrity (3b) (privacy, integrity). It could also ensure that the prescription to the pharmacy and the record sent to the PDMP are identical.
4. PDMP determines whether the prescription meets legal standards.
- a. Background information: Data mining techniques have been applied to audit and enforce prescription drug compliance. Over time, compliance with the PDMP is expected to increase medical safety through better oversight of prescription drugs.
 - b. Failure mode: The PDMP does not accurately challenge an illegal prescription or does not adequately secure the information. (Probably errors of omission.)
 - c. Recovery action: PDMP analysis can help complete the “feedback loop” of potential system failures. Philos™ services would typically precede those actions but would assure that corrective legal actions would be based on data privacy, authentication, secure data integrity, and non-repudiable data, and thus assuring a clean chain of evidence back to the originator of the data.

Step 3. Model HRA. The Philos™ Human Reliability Analysis (HRA) Model is depicted in Figure HRA.1.

Step 4. Quantify Human Error Probability (HEPs). Table HRA.1 lists the assumed probabilities of each failure mode and each recovery action for each sequence step. Using the assumed probabilities, the compounded probability of system failure modes is 0.19 per year. By applying the assumed recovery actions, the compounded probabilities of successful recovery actions by using Philos™, could reduce the compounded probability of system failure to 0.019 per year. For reasons and assumptions shown in the documentation, the assumed probabilities of failure modes were selected conservatively high, and the assumed probabilities of success of recovery actions were

selected liberally low until more data are available and validated. Following testing, it is possible that the cryptographic strength of Philos™ could theoretically approach a probability of success closer to unity (1.0), but that should not be assumed until it is proved by operational testing.

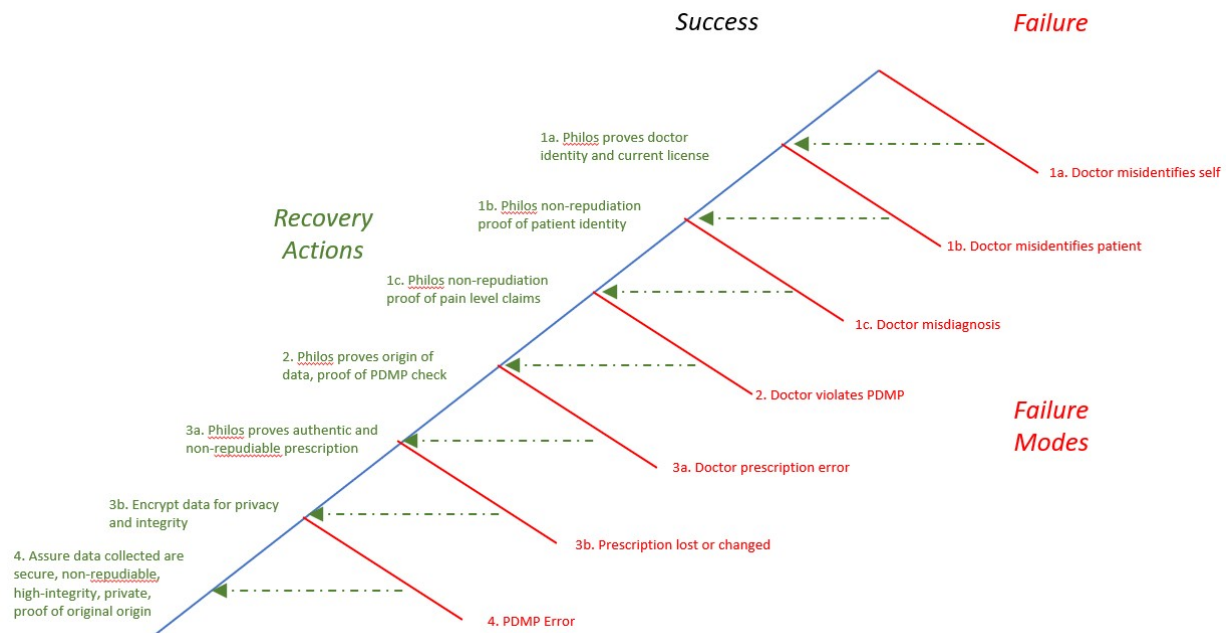


Figure HRA.1. Philos™ HRA Model

Table HRA.1 Human Error Probabilities in Failure Modes, and Recovery Actions

			P (Success) Philos recovery actions	P (Failure) Philos recovery actions	Compound Π P(Fail) after recovery	Philos™ methods used			
						Suppression (Privacy)	Validation (Authentication)	Reliability (Integrity)	Refutation (Non-Repudiation)
Sequence step failure mode	P (Fail)	Philos™ recovery actions							
1a. Doctor misidentifies self	0.02	1a. Proof of Dr ID and license	0.90	0.10	0.002		X		X
1b. Doctor misidentifies patient	0.02	1b. Proof of patient ID	0.90	0.10	0.002		X		X
1c. Doctor misdiagnoses	0.02	1c. Proof/documentation of pain claims	0.90	0.10	0.002				X
2. Doctor violated PDMP	0.03	2. Proof of PDMP check	0.90	0.10	0.003		X	X	X
3a. Doctor prescription error	0.04	3a. Authentic prescription	0.90	0.10	0.004		X		X
3b. Prescription lost or changed	0.04	3b. Encrypt data for privacy/integrity	0.90	0.10	0.004	X		X	
4. PDMP error	0.02	4. Assure data confidence & integrity	0.90	0.10	0.002	X	X	X	X
Total Σ P (Fail)	0.19				0.019				

Step 5. Documentation.

Background. The Philos™ blockchain system is associated with ten risk factors.²⁰⁹ Of those, five were generally managerial, and five were related to forms of cryptological trust.²¹⁰ Four of the five forms of cryptological trust are: suppression trust, validation trust, reliability trust, and refutation trust. Those forms of trust are designed to address the largely human-error-based problems of data privacy, data authentication, data integrity, and data non-repudiation, respectively. Examples could be from hackers using “social engineering” to co-opt passwords from human users or other forms of cyber-attacks to compromise digital computer systems, or to cause human errors on the part of network users. For the purposes of this HRA, those problems may be considered as failure modes, while the forms of cryptological trust may be considered as recovery paths. Philos™ should normally act as an almost fully autonomous, self-governing service for a system-of-systems, thereby helping to avoid common human-induced failures. (That’s not to say that human failures in software programming, administration, and management would not occur, just that the blockchain itself should be more isolated from negative human factors, and therefore more immune from human errors once validated.) Therefore, is more appropriate to apply an HRA to the human failures that Philos™ is intended to solve, and to view the “non-human” Philos™ solutions as the recovery path steps. It should be very useful to demonstrate the capabilities and value of Philos™ as a system to quantitatively mitigate human failure errors which exist within the larger system-of-systems, or as they are applied to individual use cases.

Assumptions:

1. The given scenario is used to demonstrate the methodology with notional data for a typical doctor’s office that dispenses opioids, and that might experience common human factor problems that are known to occur. Other scenarios of “recommending” medicinal marijuana (which is never prescribed), prescribing psychotropic drugs to children, or other medications that are known to be abused could be of great interest, but are not considered in this scenario.
2. Approximate probabilities of failures and successes were derived subjectively for this report, assuming a typical doctor’s office, or similar unit, over a one-year period, not on a transaction-by-

²⁰⁹ Lambert, Kent. Example 1, CSU Engr 531. May 26, 2019.

²¹⁰ Gorog C. and Boulton T. *Solving Global Cybersecurity Problems by Connecting Trust using Blockchain*. 2018.

transaction basis. Data are representative approximations based on public testimonies²¹¹ and frequency of occurrence considerations.²¹²

3. Sequence step failure modes are assumed to be independent and mutually exclusive. However, they might not be collectively exhaustive. Compounded probabilities are assumed to be additive. I.e., Cumulative Total P (failure of all failure modes) = Σ P (failure of each failure mode). Note: One exception to that might be the last sequence step failure mode, #4 PDMP Error, because it could be argued that its success or failure could be dependent on the success of previous failure modes, in which case the failures might be multiplicative. On the other hand, the PDMP is a separate organization and whether the interrelationships between doctors and the PDMP are effective (and dependent) remains to be proved. The use of Philos™ might help validate that effectiveness, but it is not clear at this point whether that probability should be additive or multiplicative. For this report, I will assume for now that #4 PDMP Error is independent and additive to avoid the possibility of undercounting the estimated effects.
4. Considering recovery actions, compounded probabilities of failure for each step failure mode are dependent and non-mutually exclusive, because the recovery actions are dependent on the failure modes. They are therefore assumed to be multiplicative. I.e., Compound P (Failure after recovery) = P (failure of step failure mode) X P (failure of each recovery action). [P (failure of each recovery action) = 1 – P (success of each recovery action.)]
5. Compounded probabilities after recovery are assumed to be additive. I.e., Cumulative Total P (failure of all failure modes after recovery actions) = Σ P (failure of each failure mode after recovery actions).
6. Each P (failure of each failure mode) is assumed to be representative for the given scenario, but given the lack of verified data, they are intentionally skewed to the “high side” (i.e., intentionally not understated) because of very significant concerns about opioid abuse and historical systemic failures to control it to date. Until the actual direct causes of opioid abuse can be proved, it should

²¹¹ Colorado Legislative Opioid Task Force. 2017.

²¹² Ostrom, L., and Wilhelmsen, C. “*Risk Assessment*.” Wiley. 2012. p. 67.

be acknowledged that there are significant “unknowns” that are not yet accounted for. If those concerns can be determined to be over-stated, the P (failure) could be decreased later.

7. Each P (failure of each failure mode after recovery actions) is assumed to be representative for the given scenario but given the lack of verified Philos™ testing to date, estimates of probabilities of success of Philos™ are intentionally on the “low side” of expected system performance until they can be proved in operational settings. That may help ensure that the capabilities are not overstated. After testing, if system performance can be confirmed, their probabilities of success could be increased. If engineering expectations are met, the Philos™ probabilities of success (in this case for recovery actions) could be closer to 1.0.

Critical Function Analysis (CFA)²¹³

Philos™ is the application of the BlockFrame® distributed ledger blockchain, as common elements within a very large-scale system-of-systems. This analysis presents a failure mode, effects, and criticality analysis (FMECA) on five representative applications of Philos™.²¹⁴ The CFA in this section is limited to a continuity-of-operations mission scenario. The “mission,” as defined here, is to plan for, respond to, and recover from threats to the North American Power Grid from cyber-attack, maximum solar winds, or a nuclear attack producing a high-altitude electromagnetic pulse (HEMP), by using Philos™. The critical functions could be considered a subset of primary critical infrastructure functions listed in Table 13.5 in *Critical Functions for Continuity of Operations*, in Ostrom & Wilhelmsen, pp. 198-199, which originated from *FEMA Continuity of Operations Plans (COOP) and Continuity of Government (COG) Plans*.²¹⁵ I would further limit these functions to visualize the solar wind threat only. The critical functions that apply to the other threats would be structurally similar. In this case, there are no controls over the initiating event, and it might not be possible to protect all infrastructure without losses. It is assumed that the focus is on successfully “riding out” the attack, minimizing damage, and reconstituting sufficient critical infrastructure to provide essential utilities and government services. By necessity, this is a very high level of abstraction, but there are several points of real-world interest in these functions. This CFA would only involve the protection of the information within the infrastructure. This CFA is a useful decomposition to visualize how Philos™ components could support the stated mission. Designs to protect physical or electronic vulnerabilities such as generators, transmission lines, communications lines, electro-physical hardening (Faraday) cages, or hardening of electronic components would have to be considered separately. Figure CFA.1 shows an example CFA for Philos™ components to help protect critical information within power grids from severe Solar Max threats.

²¹³ Lambert, Kent. *Exercise 8, CSU Engr 531*. July 21, 2019.

²¹⁴ Lambert, Kent. *Exercise 6, CSU Engr 531*. June 29, 2019.

²¹⁵ Ostrom & Wilhelmsen, *Critical Functions for Continuity of Operations*, in Ostrom & Wilhelmsen, which originated from *FEMA Continuity of Operations Plans (COOP) and Continuity of Government (COG) Plans*. pp. 198-199

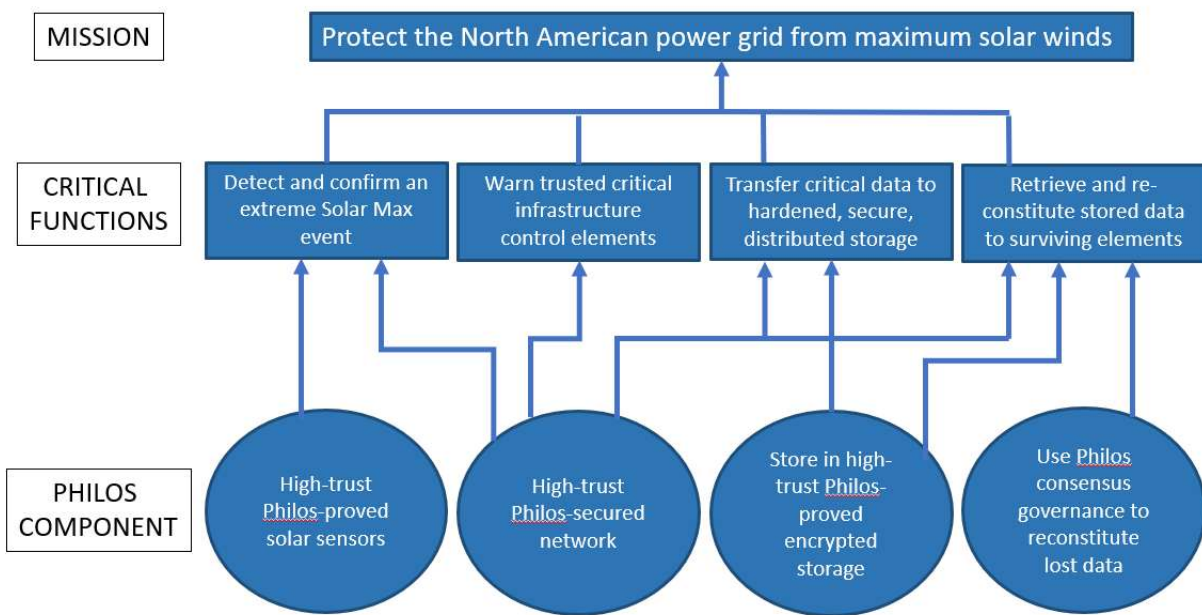


Figure CFA.1. Critical Functional Analysis Example for Philos™

Probabilistic Risk Analysis (PRA)²¹⁶

This section performs a simplified probabilistic risk assessment (PRA) procedure for Philos™ on a simple specific event which could cause failure, using an event fault tree diagram, with quantitative probabilities included. For this example, one typical internal component, a Philos™ brokering agent, is shown in Figure PRA.1 in its context as a common component within an indefinitely expandable system-of-systems ecosystem. For this example, it is assumed that brokering agents are designed as a hybrid of a virtual machine and a physical machine with different failure paths. Each element should have very high reliability and should be replaceable or updateable if/when necessary.

²¹⁶ Lambert, Kent. *Exercise 9, CSU Engr 531*. July 25, 2019.

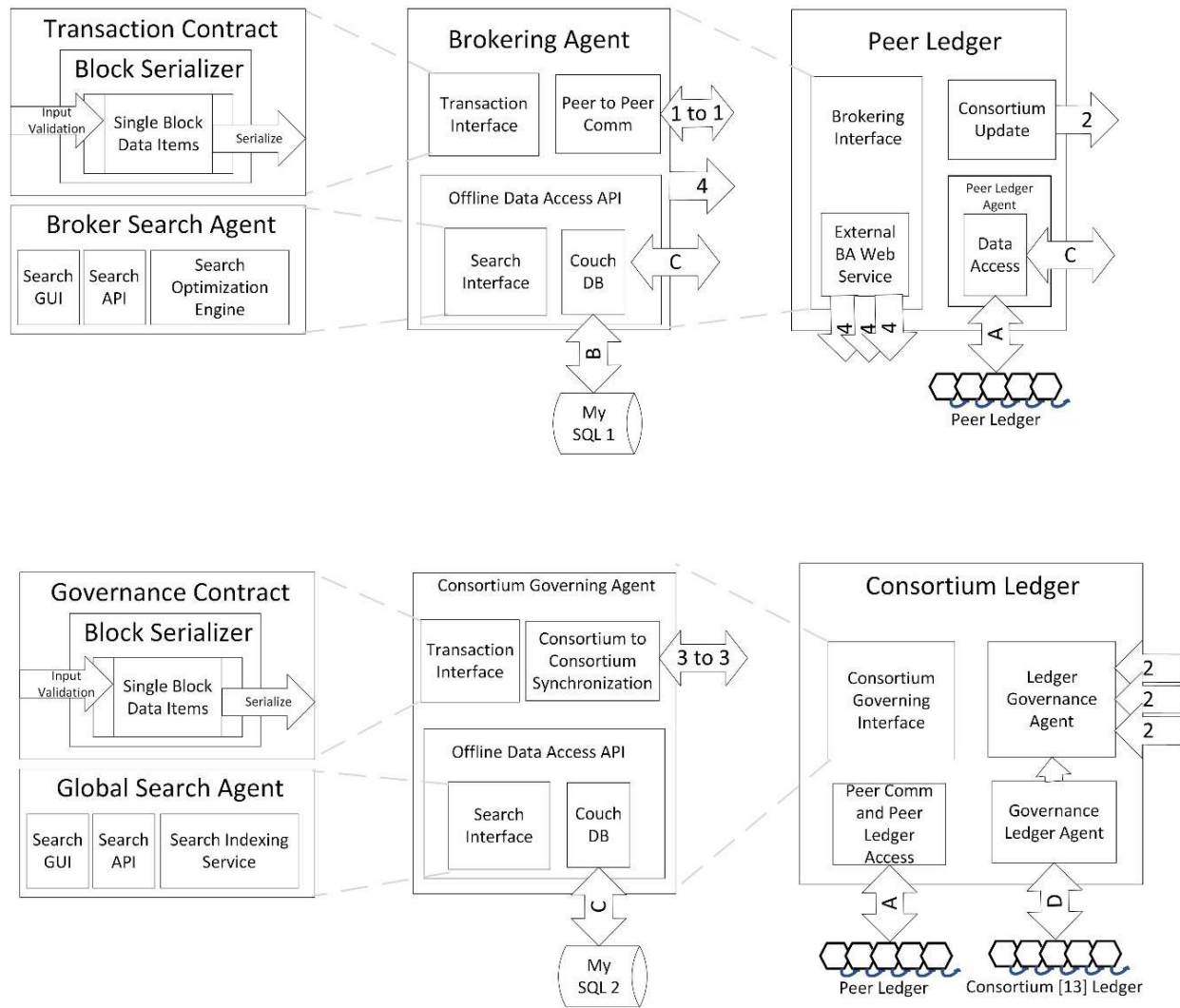


Figure PRA.1. Philos™ Subsystem Architecture (Typical)²¹⁷

Figure PRA.2 shows how brokering agents should connect to the larger systems-of-systems through a blockchain/distributed-ledger system-of-systems to serve highly diverse and dissimilar applications.

²¹⁷ Gorog C, et, al. "Software Design Document Blockchain Synchronization Consensus V3." Source Document Proprietary. 2019.

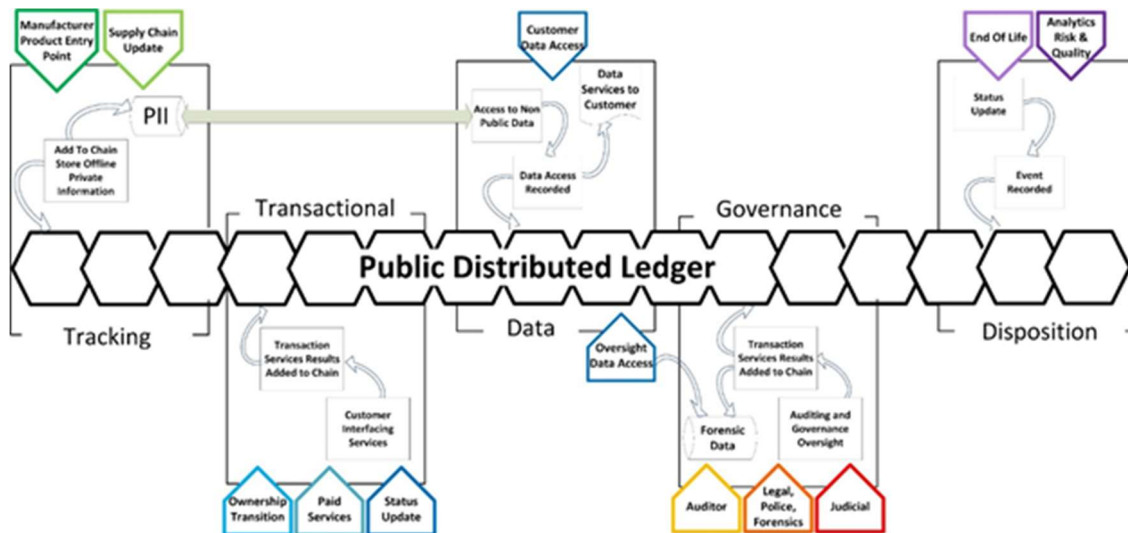


Figure PRA.2. Philos™ Customer Interfaces (Typical)²¹⁸

Because each virtual brokering agent element could be remotely replaceable, and each physical element could be physically replaceable, a probabilistic risk assessment could be a useful tool to anticipate times to failure, costs of replacement and labor, trade-off studies between functions of virtual and physical elements, and several other factors.

For this example, which by necessity must be simplified due to the complexity of the actual components, it is further assumed that only the following typical functions would be examined, and that the scope of the examination will be limited to reliability failures, not failures of cryptography or damage from intentional cyber-attack.

Functions for virtual elements:

1. Brokering agent-to-brokering agent connections and communications
2. Brokering agent-to-peer ledger (PL) connections and communications
3. Brokering agent-to-external web service connections and communications
4. Brokering agent-to-data storage connections and communications
5. Critical internal functions within virtual elements

Functions for physical elements:

6. Brokering agent-to-brokering agent connections and communications
7. Brokering agent-to-peer ledger (PL) connections and communications
8. Brokering agent-to-external web service connections and communications
9. Brokering agent-to-data storage connections and communications

²¹⁸ Ibid.

10. Critical internal functions within physical elements

Although any of these functions could fail simultaneously, we can assume that the failure of any element could be considered failure of the entire brokering agent element. In some cases, system, code, or cryptological updates could be updated or reset remotely. When they can, those failures or faults will not be considered in this example.

Although the virtual and physical elements in this example would be physically separated, we will assume that their respective failure trees should be separated functionally, not physically, because of the parallelism of their functions. Figure PRA.3 represents the failure/fault tree for a typical Philos™ brokering agent.

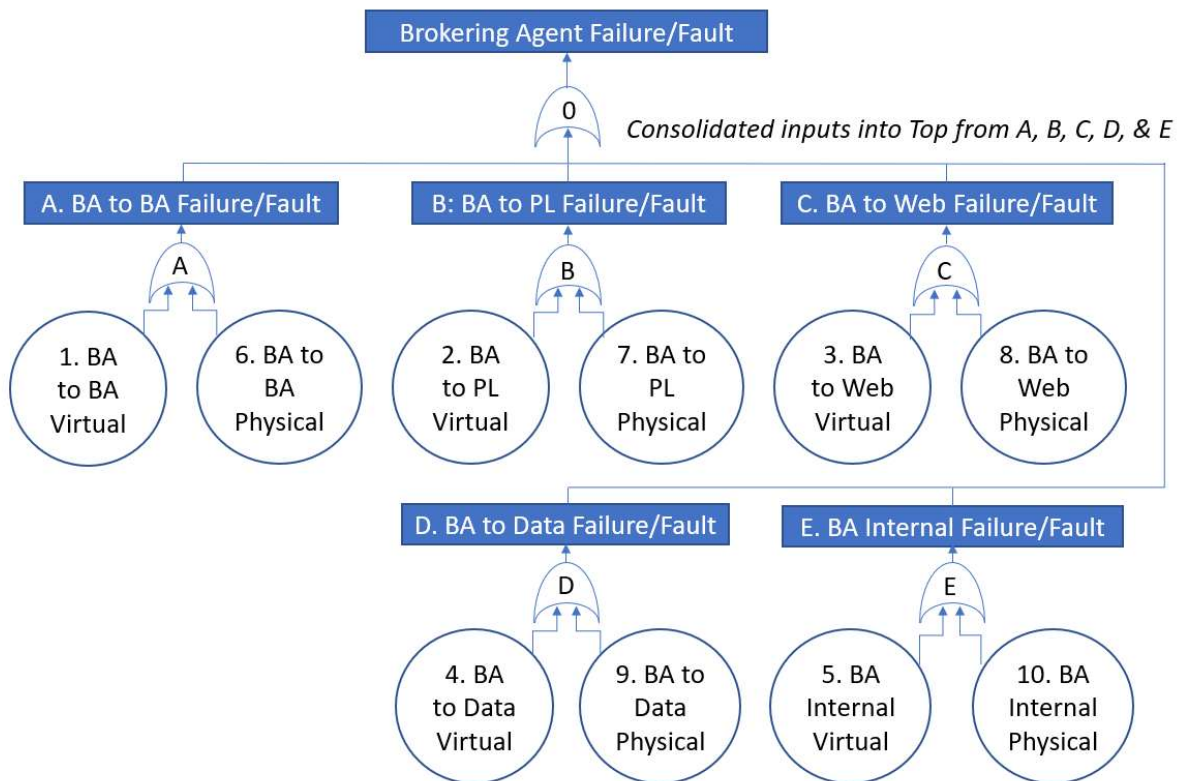


Figure PRA.3. Failure/Fault Diagram of Philos™ Brokering Agent

To calculate the risk probabilities, Table PRA.1 displays the relative probabilities of failure of each element, assuming that they are independent, that virtual reliability is relatively higher than physical reliability, and that “internal” element reliability is approximately equivalent to the reliability of communications components. Actual virtual and physical Philos™ brokering agents have not yet been tested, but based on similar devices, it is assumed the expected life duration of a Philos™ brokering agent could be similar to a typical microprocessor relay, which is

about 50 years.²¹⁹ For comparison of reliabilities, it is assumed we would want to compare the probabilities of failure in one year or less. Although the probabilities of these events should be very rare, since no testing has been done, we will arbitrarily assume that the probability of failure of a typical Philos™ physical element at 0.0001 per year for purposes of comparison.²²⁰ Since the reliability of virtual elements should typically be higher than physical components, we will also arbitrarily assume that the probability of failure of a typical Philos™ virtual element will be 0.00001 per year. Table PRA.1 displays these assumed reliabilities for each brokering agent element.

Table PRA.1. Representative Philos™ elements	Type	Reliability
1. Brokering agent-to-brokering agent connections and communications	Virtual	0.00001
2. Brokering agent-to-peer ledger (PL) connections and communications	Virtual	0.00001
3. Brokering agent-to-external web service connections and communications	Virtual	0.00001
4. Brokering agent-to-data storage connections and communications	Virtual	0.00001
5. Critical internal functions within virtual elements	Virtual	0.00001
6. Brokering agent-to-brokering agent connections and communications	Physical	0.0001
7. Brokering agent-to-peer ledger (PL) connections and communications	Physical	0.0001
8. Brokering agent-to-external web service connections and communications	Physical	0.0001
9. Brokering agent-to-data storage connections and communications	Physical	0.0001
10. Critical internal functions within physical elements	Physical	0.0001

Since the failure/fault of either the virtual OR physical elements of each combined function is additive, for this example, $P_f(A) = P_f(B) = P_f(C) = P_f(D) = P_f(E) = 0.0001 + 0.00001 = 0.00011$ per year.

Since the failure/fault of any combined function is additive, for this example, $P_f(\text{Brokering Agent Failure/Fault})$ is $5 \times 0.00011 = 0.00055$ per year. Therefore, the reliability of the entire system could be approximately 0.99945 per year.

Figure PRA.4 represents the Philos™ failure/fault risk assessment diagram with probabilities from Table PRA.1.

²¹⁹ Kleman, Michael C., and Todd Moyer. *Life Cycle Considerations for Microprocessor Relays*. Accessed July 25, 2019.

²²⁰ Ostrom, L., and Wilhelmsen, C. *Risk Assessment*. P. 67.

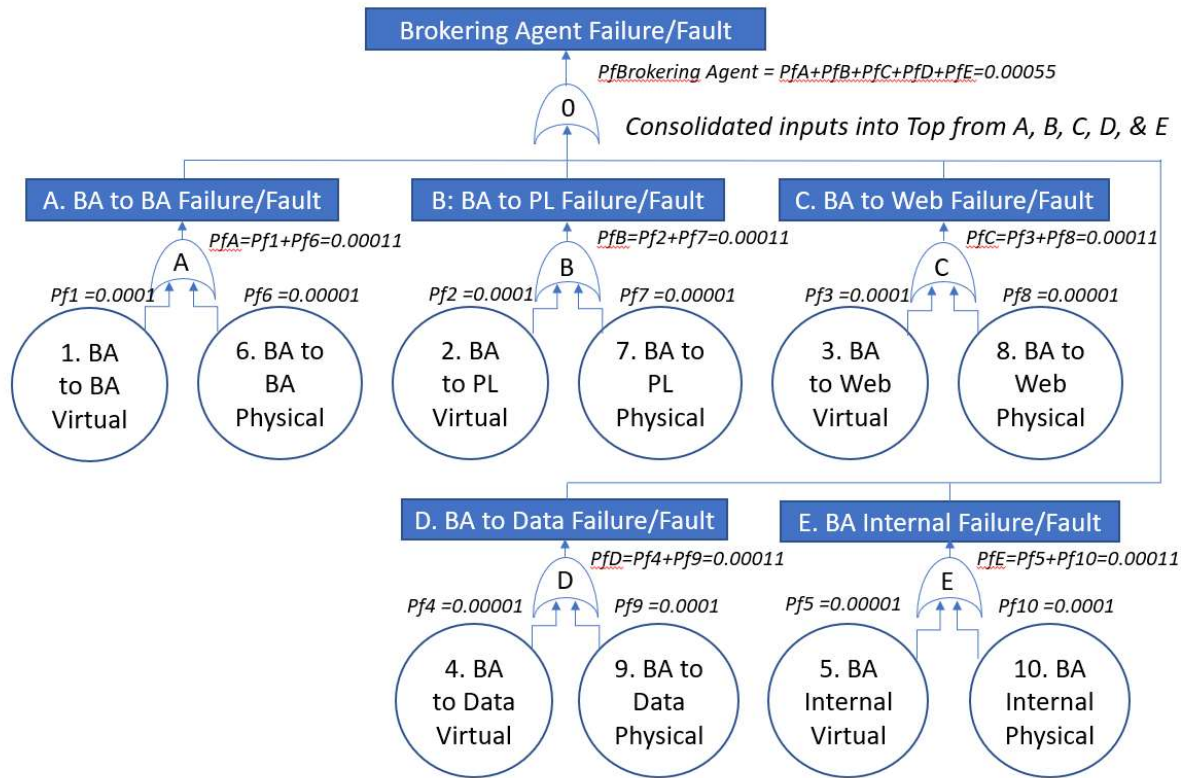


Figure PRA.4. Failure/Fault Diagram of Philos™ Brokering Agent with Probabilities

Conclusions

This Appendix is a compilation of previous research. Each section presents examples of how various risk assessment techniques could be applied to the Philos™ blockchain/distributed ledger (BDL). Philos™ is designed in part to fulfill requirements specified in Colorado Senate Bill SB18-086²²¹, and other real-world applications. Since the actual blockchain and the system-of-systems ecosystem of potential applications is very complex, this does not intend to provide a comprehensive risk analysis of all aspects of Philos™.

There were a few relevant observations that are worth noting:

1. Although each example of risk assessment analytic techniques applied herein to the actual Philos™ blockchain are highly limited in scope, they are still useful demonstrations, even as “snapshots,” of how structured assessments might be scaled up to the larger Philos™ system-of-systems and ecosystem, especially as the system becomes operational.
2. Because Philos™ is designed to be indefinitely expandable in scope, each future use case will require education and training for engineers to conceptualize how blockchain applications can be applied to a wide variety of cybersecurity, logistics, and other operational requirements. Several of these risk assessment techniques could support those educational processes.
3. There seem to be very few published works that systematically apply risk assessment or other systems engineering techniques specifically to cyber-security problems and systems. Two exceptions are Geer, D and McClure, S. 2016,²²² and “What you can do to combat cyber-attacks” by the United Kingdom National Cyber Security Centre²²³. It was necessary and convenient to modify several common definitions²²⁴ because they were more oriented to physical systems and hazards, rather than virtual and cybersecurity systems and hazards. That said, with some modifications, this research should help demonstrate that many normal risk assessment techniques can be applied effectively for supply chain or cybersecurity blockchain systems like Philos™.
4. Although some statistical techniques were not included in this section, a practical application of some statistical distributions could be very useful for other Philos™ research. Deriving statistical

²²¹ Lambert, K, A Williams, J Ginal, and B Rankin. “Colorado Senate Bill SB-086, ‘Cyber Coding Cryptology for State Records.’” Colorado General Assembly, 2018. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.

²²² Geer, D and McClure, S. 2016. *How to Measure Anything in Cybersecurity Risk*. Wiley, Hoboken, New Jersey. pp 9, 11.

²²³ United Kingdom National Cyber Security Centre, “What you can do to combat cyber-attacks”, <https://www.ncsc.gov.uk/static-assets/documents/NCSC%20Cyber%20Attacks.pdf>.

²²⁴ Ostrom, L., and Wilhelmsen, C. “*Risk Assessment*.” Wiley. 2012. p. 67.

distributions from theoretical or empirical data could be directly applied to values that are needed to reduce complex blockchain governance algorithms to simpler algorithms with a smaller and more efficient number of parameters. Some distributions have already been advocated for that purpose, but more research will be necessary before governance algorithms are optimal. The scope of this research is too narrow to solve a problem with that complexity, but some useful ideas were considered for future research.

5. Several risk analysis techniques have already become relevant to of Philos™ applications, and possible future contract proposals could support several industries including, hospital network device security, supply chain management, protection or mitigation of HEMP, Solar Max, cyber-attacks against power and utility critical infrastructure, and life cycle costs. There also seem to be useful structures for quantitatively measuring component reliability, planning for operational test and evaluation, and identifying critical components within the larger system-of-systems ecosystems. Although specific applications of qualitative analyses have not been discussed, several non-quantitative techniques could be useful for initial assessments of legal and/or ethical requirements and issues. For example, Colorado House Bill HB18-1128, *Protections for Consumer Data Privacy*,²²⁵ was passed to specify victim notification requirements following data breaches, which is largely a qualitative issue. Social media privacy is a similar qualitative concern due to loss of individual privacy and the high potential for identity theft. Structured qualitative analyses of the legal, ethical, and social consequences of poor data handling and security could also help discover quantitative impacts on companies, individuals, or other organizations. Those factors could in turn motivate the need for mitigating policies, legislation, systemic quantification of financial damage and liability claims, or other processes to control the negative effects of data breaches.

²²⁵ Wist, Bridges, Lambert and Court, Colorado House Bill HB18-1128, *Protections for Consumer Data Privacy* 2018. Accessed at https://leg.colorado.gov/sites/default/files/2018a_1128_signed.pdf.

Glossary

Blockchain. The blockchain is a distributed ledger with growing lists of records (*blocks*) that are securely linked together via cryptographic hashes. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a Merkle tree, where data nodes are represented by leaves). Since each block contains information about the previous block, they effectively form a *chain* (compare linked list data structure), with each additional block linking to the ones before it. Consequently, blockchain transactions are resistant to alteration because, once recorded, the data in any given block cannot be changed retroactively without altering all subsequent blocks and obtaining network consensus to accept these changes. Blockchains are typically managed by a peer-to-peer (P2P) computer network for use as a public distributed ledger, where nodes collectively adhere to a consensus algorithm protocol to add and validate new transaction blocks. Although blockchain records are not unalterable, since blockchain forks are possible, blockchains may be considered secure by design and exemplify a distributed computing system with high Byzantine fault tolerance.²²⁶

Complexity. Describing something that cannot be adequately dealt with using only simple, intuitive tools and methods. Complex systems routinely exhibit behaviors that are difficult to predict or even to understand.²²⁷ Dr Larry Rainey defines complexity as, “a characteristic of all modern systems which means the number of multi-level interactions among the components or elements of [the] system.”²²⁸

Defense-in-Depth. Deployment of diverse and mutually supportive security controls at various points in architecture to create concentric shells of protection that an attacker must defeat to reach protected assets²²⁹.

Distributed Energy Resource (DER): Distributed energy resources (DER) are small, modular, decentralized power-generating systems located near the point of energy use, which can include energy storage and delivery technologies. DER systems are designed to meet increased energy demand, reduce peak operating costs, and enhance the reliability of electric power generation.²³⁰

Distributed Ledger Technology (DLT): Distributed Ledger Technology (DLT) refers to a technology that allows organizations to share data and code without the need for a single centralized authority. It enables the development of decentralized applications (DAPPs) where multiple organizations collaborate to address the challenges of decentralization. However, the lack of centralized governance poses difficulties in enforcing regulations, and it can be challenging to coordinate trust between distrusting organizations. This has led some projects to resort to creating new centralized entities to oversee the governance of their DAPPs, limiting the potential benefits of distributed ledger technology.²³¹

Edge Computing. Edge computing is a distributed computing paradigm that brings computation and data storage closer to the sources of data. This is expected to improve response times and save bandwidth. It is an architecture rather than a specific technology. It is a topology- and location-sensitive form of distributed computing²³².

²²⁶ “Blockchain.” *Wikipedia*. Accessed: Aug 29, 2025. [Blockchain - Wikipedia](#).

²²⁷ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>. P.1.

²²⁸ Mittal, Saurabh, and Rainey, “Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering. SCSC’15, Summer Simulation Multi-Conference, Chicago, IL.”

²²⁹ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>. P.353.

²³⁰ “Distributed Energy Resource.” <https://www.sciencedirect.com/topics/engineering/distributed-energy-resource>.

²³¹ “Distributed Ledger Technology.” <https://www.sciencedirect.com/topics/computer-science/distributed-ledger-technology>.

²³² “Edge computing.” *Wikipedia*. Jul. 14, 2022. Accessed: Jul. 18, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Edge_computing&oldid=1098098050.

Emergence. The potential outcome/result (negative or positive) of the interaction of dissimilar and nonconnected systems²³³.

Law of Requisite Variety.

1. the amount of appropriate selection that can be performed is limited by the amount of information available.
2. for appropriate regulation the variety in the regulator must be equal to or greater than the variety in the system being regulated. Or the greater the variety within a system, the greater its ability to reduce variety in its environment through regulation. Only variety (in the regulator) can destroy variety (in the system being regulated). The law was formulated by Ross Ashby²³⁴²³⁵²³⁶.

Research and Development (R&D). The term R&D covers three types of activity: basic research, applied research and experimental development. Basic research is experimental or theoretical work undertaken primarily to acquire new knowledge of the underlying foundations of phenomena and observable facts, without any particular application or use in view. Applied research is original investigation undertaken in order to acquire new knowledge. It is, however, directed primarily towards a specific, practical aim or objective. Experimental development is systematic work, drawing on knowledge gained from research and practical experience and producing additional knowledge, which is directed to producing new products or processes or to improving existing products or processes²³⁷.

Smart Grid Architecture: Smart Grid Architecture refers to the design and structure of a more accurate and detailed system that ensures the security of the Smart Grid²³⁸.

System. A set of interrelated components functioning together towards some common objective(s) or purpose(s)²³⁹.

System-of-Systems

1. The integration of a set of independently developed and operated systems with the goal of creating different, usually greater, capabilities than the individual participants can achieve²⁴⁰.
2. A system of systems (SoS) is a composition of systems, where often component systems have legacy properties, e.g., autonomy, belonging, diversity, and emergence²⁴¹.

Token (in networking). In networking, the word token has a different meaning. It is a special frame that is passed from node to node around a ring network. When it gets to a node that needs to transmit data, the node changes the token into a data frame and transmits it to the recipient. Tokens are essential to the inner workings of a token ring network. A token can only be handled by a single node at a time. The bearer of the token is the only one allowed to send data around the network to a recipient node within the network. The bearer of the token writes the address of the recipient and the data to be sent and then sends it to the next node in the series. When the

²³³ L. B. Rainey and M. Jamshidi, Eds., *Engineering Emergence: A Modeling and Simulation Approach*. Boca Raton: CRC Press, 2018. doi: 10.1201/9781138046412.

²³⁴ “The Law of Requisite Variety.” Aug. 31, 2001. <http://pespmc1.vub.ac.be/REQVAR.html> (accessed Jul. 21, 2022).

²³⁵ “Requisite Variety - an overview | ScienceDirect Topics.” <https://www.sciencedirect.com/topics/computer-science/requisite-variety> (accessed Jul. 21, 2022).

²³⁶ K. Kijima, “Generalization of Law of Requisite Variety,” in *Systems Research I: Essays in Honor of Yasuhiko Takahara on Systems Theory and Modeling*, K. Kijima, J. Iijima, R. Sato, H. Deguchi, and B. Nakano, Eds. Singapore: Springer Nature, 2022, pp. 27–46. doi: 10.1007/978-981-16-9937-5_2.

²³⁷ “Research and Development.” National Science Foundation. <https://ncses.nsf.gov/pubs/ncses25215>.

²³⁸ “Smart Grid Architecture.” Science Direct. <https://www.sciencedirect.com/topics/computer-science/smart-grid-architecture>.

²³⁹ B. Blanchard and W. Fabrycky, *Systems Engineering and Analysis*, 5th ed., Upper Saddle River, NJ: Pearson Hall, 2011. ISBN: 9780132217354.

²⁴⁰ Borky, John Michael, and Bradley, Thomas, *Effective Model-Based Systems Engineering*. 2019. Accessed: Jun. 22, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-95669-5>.

²⁴¹ J. Boardman and B. Sauser, “Systems of Systems -- The Meaning of ‘of’.” Los Angeles, CA, 2006.

sender node sends the token to the next node, that node reads the address. If that node is not the intended recipient, it sends the data to the next node and so on. Finally, when the recipient node reads the data and knows that it is the recipient, it takes the data and sends the token back to the sender's address with a message indicating that the data was received. The token is then sent around the ring again until it reaches the sender/bearer of the token. After using the token, a node releases it back into the network so other nodes will be able to use it.²⁴²

Translational. Adjective.

1. of, or relating to [translating](#), or to a [translation](#)
2. of, or relating to a [body](#) in [motion](#)
3. of or relating to the transfer of scientific knowledge into practical applications²⁴³.

Trust/Trustworthiness.

1. The attribute of a person or organization that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities²⁴⁴.
2. The attribute of a person or enterprise that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities²⁴⁵.
3. Security decisions with respect to extended investigations to determine and confirm qualifications, and suitability to perform specific tasks and responsibilities²⁴⁶.

Trustworthy System. Computer hardware, software and procedures that:

1. are reasonably secure from intrusion and misuse;
2. provide a reasonable level of availability, reliability, and correct operation;
3. are reasonably suited to performing their intended functions; and
4. adhere to generally accepted security procedures²⁴⁷.

Uniformity 1. The quality or state of being the same: the quality or state of being uniform or identical.²⁴⁸

Uniformity 2. Uniformity as used by BlockFrame® does not yet have a rigorous academic definition but might be best described as an evolution of the concepts of defense-in-depth, zero trust, standardization, and the digital management and enforcement of system rules. In practice, uniformity across a system of systems can help apply consistent practices and standards in logistics, identification, traceability, trust, accountability, and balancing privacy with visibility²⁴⁹.

Zero Trust (ZT)/Zero Trust Architecture. Zero trust (ZT) is the term for an evolving set of cybersecurity paradigms that move defenses from static, network- based perimeters to focus on users, assets, and resources. A zero-trust architecture (ZTA) uses zero trust principles to plan industrial and enterprise infrastructure and workflows. Zero trust assumes there is no implicit trust granted to assets or user accounts based solely on their physical or network location (i.e., local area networks versus the internet) or based on asset ownership (enterprise or personally owned). Authentication and authorization (both subject and device) are discrete functions performed before a session to an enterprise resource is established. Zero trust is a response to enterprise network trends that include remote users, bring your own device (BYOD), and cloud-based assets that are not located within an enterprise-

²⁴² "Token." Techopedia. <https://www.techopedia.com/definition/4140/token>.

²⁴³ "Translational — definition, examples, related words and more at Wordnik," *Wordnik.com*. <https://www.wordnik.com/words/translational> (accessed Jul. 22, 2022).

²⁴⁴ "Trust/Trustworthiness." National Institute of Standards and Technology NIST Special Publication SP 800-79; Kissel, R. *Glossary of Key Information Security Terms*. NISTIR 7298 Rev 2. May 2013.

²⁴⁵ Committee on National Security Systems CNSSI-4009; NIST Special Publication SP 800-39.

²⁴⁶ NIST Federal Information Processing Standards FIPS 201.

²⁴⁷ NIST Special Publication SP 800-32.

²⁴⁸ "Uniformity Definition & Meaning | Britannica Dictionary." <https://www.britannica.com/dictionary/uniformity> (accessed Jul. 18, 2022).

²⁴⁹ C. Gorog and K. Lambert, "BlockFrame, Inc., Technology -- Use Cases for Blockchain Technologies," Jul. 2021.

owned network boundary. Zero trust focuses on protecting resources (assets, services, workflows, network accounts, etc.), not network segments, as the network location is no longer seen as the prime component to the security posture of the resource. This document contains an abstract definition of zero trust architecture (ZTA) and gives general deployment models and use cases where zero trust could improve an enterprise's overall information technology security posture.²⁵⁰

²⁵⁰ S. W. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," *NIST*, Aug. 2020, Accessed: Jul. 19, 2022. [Online]. Available: <https://www.nist.gov/publications/zero-trust-architecture>.

Bibliography

- Assembly. "How Ford's Willow Run Assembly Plant Helped Win World War II." Assembly. Jan. 3, 2019. <https://www.assemblymag.com/articles/94614-how-fords-willow-run-assembly-plant-helped-win-world-war-ii>. Accessed Jul. 22, 2022.
- Auguston, Mikhail. "Monterey Phoenix, or How to Make Software Architecture Executable." October 2009.
- Auguston, Mikhail. "Software Architecture Built from Behavior Models." *ACM SIGSOFT Software Engineering Notes*, 34, no. 5 (2009).
- Bayuk, Jennifer L., and Barry M. Horowitz. *An Architectural Systems Engineering Methodology for Addressing Cyber Security*. *Systems Engineering*, 14, no. 3 (2011). 16 February 2011. 294–304. <https://onlinelibrary.wiley.com/doi/abs/10.1002/sys.20182>.
- BH Sky Associates. *ALERT: GSA's SAM Falls Victim to Third Security Breach*. 2018. Accessed June 22, 2022. <https://www.bhskyassociates.com/blog-post/alert-gsa%E2%80%99s-sam-falls-victim-third-security-breach>.
- Blockchain Development Community (BCDC) Policy & Governance Team. "Meeting Notes: Policy and Governance for Philos Distributed Ledger," February 7, 2020.
- BlockFrame, Inc. Accessed July 20, 2022. <https://blockframetech.com/>.
- BlockFrame, Inc. "Patents Filed by BlockFrame, Inc. U.S. Patent Application No. 17/488,529; No. 17/488,562; No. 17/488,589; No. 17/488,655; Patent Cooperative Treaty (PCT) Application No. PCT/US21/52839; No. PCT/US21/52854; No. PCT/US21/52865; No. PCT/US21/52872." United States Patent and Trademark Office., Oct. 01, 2021. The most current information is available at <https://www.blockframetech.com/patents/>.
- Boardman, J., and B. Sauser. "Systems of Systems -- The Meaning of 'Of'." Presented at the IEEE/SMC International Conference on System of Systems Engineering, Los Angeles, CA, 2006.
- Borky, John Michael. *Cybersecurity for Systems Engineers*. Colorado State University. Revised January 16, 2018.
- Borky, Dr John M., and Dr Thomas H. Bradley. *Effective Model-Based Systems Engineering*. 1st ed. Colorado State University: Springer, 2019.2019. Accessed June 22, 2022. <https://link.springer.com/book/10.1007/978-3-319-95669-5>.
- Britannica. "Uniformity Definition & Meaning | Britannica Dictionary." Accessed July 18, 2022. <https://www.britannica.com/dictionary/uniformity>.
- Carnegie Mellon University and the Johns Hopkins University Applied Physics Laboratory, LLC. "'Cybersecurity Maturity Model Certification' Report for the Office of the Secretary of Defense." January 31, 2020. <https://www.acq.osd.mil/cmmc/index.html>.
- Columbus, Louis. *10 Charts That Will Challenge Your Perspective of IoT's Growth*. Forbes. Accessed June 22, 2022. <https://www.forbes.com/sites/louiscolumbus/2018/06/06/10-charts-that-will-challenge-your-perspective-of-iots-growth/>.
- Dassault Systèmes®. "Cameo Systems Modeler - CATIA - Dassault Systèmes®." Accessed July 21, 2022. <https://www.3ds.com/products-services/catia/products/no-magic/cameo-systems-modeler/>.
- Diogenes, Yuri, and Dr Erdal Ozkaya. *Cybersecurity – Attack and Defense Strategies - Second Edition*. Packt, December 2019. <https://www.packtpub.com/product/cybersecurity-attack-and-defense-strategies-second-edition/9781838827793>. Accessed Jul. 19, 2022.
- Giammarco, Kristin, and Mikhail Auguston. "Monterey Phoenix Tutorial." May 2019.
- Gorog, C. *Blockchain Consortium Governance Community Sourced Project. Software Design Document for Blockchain Synchronization Consensus Method*. Unpublished. 2019. All rights reserved.
- Gorog, Christopher. *BlockChain Cybersecurity Logistics Organization for Globally Scalable Security Management*. Unpublished, 2019.
- Gorog, C. *Blockchain Distributed Ledger for State Governance*. Prepared for discussions with the State of Colorado. 2019.
- Gorog, Christopher, et.al. *Blockchain Synchronous Trust Consensus Model*. University of Colorado -- Colorado Springs. Unpublished. 2020.
- Gorog, Christopher, and K. Lambert. *BlockFrame, Inc., Technology -- Use Cases for Blockchain Technologies*. PowerPoint. Unpublished. July 2021.
- Gorog, Christopher, Pam Russell, and Terrance E. Boulton. *Carbon Neutral Distributed Ledger*. In *IEEE Innovative Smart Grid Technologies (ISGT)*. New Orleans, LA: IEEE Pre-Release, 2022.
- Gorog, C. *DARPA Cybersecurity Logistics Management for Distributed IOT V2*. Unpublished. BlockFrame, Inc. Source Document Proprietary. 2019.

- Gorog, Christopher, and Terrance E. Boulton. *Solving Global Cybersecurity Problems by Connecting Trust Using Blockchain*. In *2018 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 1425–32, 2018. University of Colorado in Colorado Springs. https://doi.org/10.1109/Cybermatics_2018.2018.00243.
- Gorog, C, et.al. *Software Design Document Blockchain Synchronization Consensus V3*. Source Document Proprietary. 2019.
- Gorog C. "Personal discussions with Lambert K." Unpublished. 2019.
- GS1 US®. *Applying Standards for Supply Chain Visibility in Blockchain Applications*, Release 1.1. January 30, 2020.
- Herber, Dr Daniel R, and Dr John M. Borky. *Engr 667: Session 8: Mathematical Network Models 2 and Pattern in Architecture Decisions*. Colorado State University, March 23, 2020.
- IEEE. IEEE Digital Privacy Initiative. <https://digitalprivacy.ieee.org/>
- IEEE. "IEEE Digital Privacy Model." <https://digitalprivacy.ieee.org/about/digital-privacy-model>.
- IEEE Future Directions. Digital Privacy Workshop. "Policies and Frameworks around the World." Oct 7, 2021. Accessed July 22, 2022. <https://cmte.ieee.org/futuredirections/events/digital-privacy-workshop/>.
- Jackson, Daniel. *Software Abstractions: Logic, Language, and Analysis*. MIT Press. Cambridge, MA. 2012.
- Kijima, K. "Generalization of Law of Requisite Variety," in *Systems Research I: Essays in Honor of Yasuhiko Takahara on Systems Theory and Modeling*, K. Kijima, J. Iijima, R. Sato, H. Deguchi, and B. Nakano, Eds. Singapore: Springer Nature, 2022, pp. 27–46. doi: 10.1007/978-981-16-9937-5_2.
- Lambert, Kent. *Applications of Defense-in-Depth and Zero-Trust Cryptographic Products in Emergent Cybersecurity Environments*. In *Emergent Behavior in System of Systems Engineering*. Pre-Release. CRC Press, 2022.
- Lambert, Kent, and C Gerou. "Colorado Senate Bill SB14-190. 'Statewide Discovery Sharing System'." Colorado General Assembly, 2014. http://www.leg.state.co.us/clics/clics2014a/csl.nsf/fsbillcont3/622646C65800A5FF87257CA00080C333?Open&file=190_enr.pdf.
- Lambert, Kent D., Angela Williams, Joann Ginal, and Bob Rankin. "Colorado Senate Bill SB-086, 'Cyber Coding Cryptology for State Records.'" Colorado General Assembly. 2018. http://leg.colorado.gov/bill-search?search_api_views_fulltext=sb18-086.
- Lambert, Kent. *Bridging the Technologist Policymaker Divide in Digital Privacy*. Published in IEEE-USA InSight; (accessed Feb 26, 2024).
- Lambert, Kent. *Cybersecurity for the Blockchain Development Consortium*. Colorado State University. May 8, 2019.
- Lambert, Kent, and Vincil Bishop. *Engineering the Philos Blockchain System; Engr 569 Semester Project*. Colorado State University, 2019.
- Lambert, Kent, and Bishop, V. 2019. *Engr 569 Semester Project: Engineering the Philos™ Blockchain System*. Colorado State University.
- Lambert, Kent. *Engineering 567*. Colorado State University. 2019 And derivative sources.
- Lambert, Kent. *Final Project Report for Engr 567*. Colorado State University. Unpublished., 2019.
- Mitre. "Mitre ATT&CK for Industrial Control Systems." Accessed July 21, 2022. https://collaborate.mitre.org/attackics/index.php/Main_Page.
- Mittal, Saurabh, and Larry Rainey. *Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering*. SCSC'15, Summer Simulation Multi-Conference, Chicago, IL. © Society of Computer Simulation International., 2015.
- Moser, A. *Conversations and Correspondence with Kent Lambert*. BlockFrame, Inc. Unpublished. Sept 29, 2019.
- National Conference of State Legislatures. "Task Force on Cybersecurity." Accessed July 22, 2022. <https://www.ncsl.org/ncsl-in-dc/task-forces/task-force-on-cybersecurity.aspx>.
- PIACET. <https://piacet.com/home#features>. Accessed 2019.
- Principia Cybernetica Server. *Law of Requisite Variety Definition*. Accessed July 21, 2022. http://pespmc1.vub.ac.be/ASC/LAW_VARIE.html.
- Principia Cybernetica Server. *The Law of Requisite Variety*. Accessed August 31, 2021. <http://pespmc1.vub.ac.be/REQVAR.html>.
- Rainey, Larry B., and Mo Jamshidi, eds. *Engineering Emergence: A Modeling and Simulation Approach*. Boca Raton: CRC Press, 2018. <https://doi.org/10.1201/9781138046412>.

- Readkong. "Analysis of the Cyber Attack on the Ukrainian Power Grid - TLP: White March 18, 2016 - SANS ICS," March 18, 2016. <https://www.readkong.com/page/analysis-of-the-cyber-attack-on-the-ukrainian-power-grid-6826988>. Accessed Jul. 19, 2022.
- Ready, Frank. "Epiq's Ransomware Response Is By The Books, But Dangers Still Lurk." Law.Com, March 3, 2020. <https://www.law.com/legaltechnews/2020/03/03/epiqs-ransomware-response-is-by-the-books-but-dangers-still-lurk-2/?slreturn=20200205161632>.
- Rose, Scott W., Oliver Borchert, Stuart Mitchell, and Sean Connelly. *Zero Trust Architecture*. NIST, August 10, 2020. <https://www.nist.gov/publications/zero-trust-architecture>.
- Runyon, Jennifer. "Power Grid Faces Elevated Russian Cyber Threats during War in Ukraine." *POWERGRID International* (blog), March 31, 2022. <https://www.power-grid.com/news/power-grid-faces-elevated-russian-cyber-threats-during-war-in-ukraine/>.
- Science Direct. "Requisite Variety - an Overview | ScienceDirect Topics." Accessed July 21, 2022. <https://www.sciencedirect.com/topics/computer-science/requisite-variety>.
- Selva, D, B Cameron, and E Crawley. *Patterns in System Architecture Decisions. Systems Engineering*, Wiley, 19, no. 6 (2016).
- Solarium. "Cyberspace Solarium Commission." Accessed July 5, 2022. <https://www.solarium.gov/>.
- Szathmari, Gabor. "Ukraine Power Grid Cyberattacks." Rainbow and Unicorn, May 17, 2022. <https://blog.gaborszathmari.me/ukraine-power-grid-cyberattacks/>.
- The White House. *Executive Order on Securing the United States Bulk-Power System*. May 1, 2020. <https://www.whitehouse.gov/presidential-actions/executive-order-securing-united-states-bulk-power-system/>.
- The White House. "Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure – The White House." Accessed June 22, 2022. <https://trumpwhitehouse.archives.gov/presidential-actions/presidential-executive-order-strengthening-cybersecurity-federal-networks-critical-infrastructure/>.
- U.K. National Cyber Security Centre. *The cyber threat to UK Business 2017-2018 report*. United Kingdom. 2019. <https://www.ncsc.gov.uk/information/the-cyber-threat-to-uk-business-2017-2018-report>.
- U.S. Cybersecurity and Infrastructure Security Agency. *Russian Government Cyber Activity Targeting Energy and Other Critical Infrastructure Sectors | CISA*. Accessed June 22, 2022. <https://www.cisa.gov/uscert/ncas/alerts/TA18-074A>.
- U.S. Department of Energy. *Colonial Pipeline Cyber Incident*. Energy.gov. Accessed November 27, 2021. <https://www.energy.gov/ceser/colonial-pipeline-cyber-incident>.
- U.S. Department of Energy. *EAC FERC Order 2222 Recommendations. Approved Pdf*. Accessed July 21, 2022. https://www.energy.gov/sites/default/files/2021-04/EAC%20FERC%20Order%202222%20Recommendations_Approved.pdf.
- U.S. Federal Bureau of Investigation and National Security Agency. *Alert (TA18-074A) Russian Government Cyber Activity Targeting Energy and Other Critical Infrastructure Sectors*. FBI/NSA document. Last revised: March 16, 2018. <https://www.us-cert.gov/ncas/alerts/TA18-074A>.
- U.S. Federal Energy Regulatory Commission. "FERC Order No. 2222: Fact Sheet | Federal Energy Regulatory Commission." Accessed July 21, 2022. <https://ferc.gov/media/ferc-order-no-2222-fact-sheet>.
- U.S. General Services Administration. "FedRAMP." Accessed July 21, 2022. <https://www.gsa.gov/technology/government-it-initiatives/fedramp>.
- U.S. National Institute of Standards and Technology. *NIST Publishes SP 800-171 Revision 2: Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations*, NIST, Feb. 2020, Accessed: Jul. 21, 2022. [Online]. Available: <https://www.nist.gov/news-events/news/2020/02/nist-publishes-sp-800-171-revision-2-protecting-controlled-unclassified>.
- U.S. Office of Personnel Management Security Resource Center. *Security Incidents*. Undated. <https://www.opm.gov/cybersecurity/cybersecurity-incidents/>.
- U.S. Office of the Secretary of Defense. *OUSD A&S - Cybersecurity Maturity Model Certification (CMMC)*. <https://www.acq.osd.mil/cmmc/model.html> (accessed Jul. 21, 2022).
- Utermohlen, K. *Facebook Data Breach 2019: 540 Million Users' Records Exposed Facebook Data Breach 2019: 540 Million Users' Records Exposed*. InvestorPlace (blog), April 4, 2019. <https://investorplace.com/2019/04/facebook-data-breach-2019/>.
- Verizon. *Verizon 2019-Data-Breach-Investigations-Report.Pdf*. Accessed June 22, 2022. <https://www.verizon.com/business/resources/reports/2019-data-breach-investigations-report.pdf>.

- Wikipedia. "2022 Ukraine Cyberattacks." *Wikipedia*. Accessed June 9, 2022.
https://en.wikipedia.org/w/index.php?title=2022_Ukraine_cyberattacks&oldid=1092281378.
- Wikipedia. "Domain Name System." *Wikipedia*. Accessed June 12, 2022.
https://en.wikipedia.org/w/index.php?title=Domain_Name_System&oldid=1092781489.
- Wikipedia. "Edge Computing." *Wikipedia*. Accessed July 14, 2022.
https://en.wikipedia.org/w/index.php?title=Edge_computing&oldid=1098098050.
- Wikipedia. "Samuel Andrews (Chemist)." *Wikipedia*. Accessed July 19, 2021.
[https://en.wikipedia.org/w/index.php?title=Samuel_Andrews_\(chemist\)&oldid=1034354875](https://en.wikipedia.org/w/index.php?title=Samuel_Andrews_(chemist)&oldid=1034354875).
- Wikipedia. "United States Railroad Administration." *Wikipedia*. Accessed January 21, 2022.
https://en.wikipedia.org/w/index.php?title=United_States_Railroad_Administration&oldid=1067020171.
- Wordnik.com. "Translational — Definition, Examples, Related Words and More at Wordnik." Accessed July 22, 2022. <https://www.wordnik.com/words/translational>.
- Zeigler, Bernard P. *DEVS-Based Modeling and Simulation Framework for Emergence in Systems of Systems*. In *Engineering Emergence, A Modeling and Simulation Approach*, First., 548. CRC Press, 2019.