NPS-AM-26-284

ACQUISITION RESEARCH PROGRAM SPONSORED REPORT SERIES

Improving Gear Provisioning Efficiency at the Individual Issue Facility (IIF) Through Size-Specific Inventory Simulation

December 2025

Capt Leah S. Weingarten, USMC

Thesis Advisors: Dr. Geraldo Ferrer, Professor
Thomas D. Kline, Lecturer

Department of Acquisition, Finance and Manpower

Naval Postgraduate School

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

The research presented in this report was supported by the Acquisition Research Program of the Department of Acquisition, Finance and Manpower Naval Postgraduate School.

ABSTRACT

As the U.S. expeditionary force in readiness, the Marine Corps depends on properly sized, serviceable individual combat equipment to maintain readiness. The Individual Issue Facility (IIF) currently stocks gear using sizing tariffs based on outdated anthropometric data, and replenishment occurs irregularly and without a consistent reordering policy. These practices contribute to shortages of high-demand sizes and excess inventory of others. This thesis uses helmets as a representative case and analyzes three years of Enterprise Logistics Management System (ELMS) issue and return data, developing a discrete-event simulation (DES) using SimPy to model the flow of helmets through the IIF. The data reveals size-demand patterns, return behavior, and loss rates, which are used to derive realistic model inputs. Baseline test results show that the model closely matches observed system behavior, and sensitivity tests demonstrate how longer retention periods and seasonal arrival patterns affect inventory levels. This research shows that a DES approach provides a practical tool for evaluating reorder strategies and improving size-specific inventory management that can be broadly applied to similar logistics systems facing long lead times, irregular returns, and size-specific demand. The model allows decision-makers to assess alternative policies and 'what-if' scenarios in advance, giving them a data-driven basis for setting reorder points and determining starting inventories.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First, I would like to acknowledge Professor Geraldo Ferrer, who has been with me from the very beginning of this project. Your guidance and patience kept me levelheaded through the countless visits to your office to stare at or re-draw the IIF process model on your whiteboard. Maybe one day I will find those elusive "Weingarten Numbers." Until then, thank you for your mentorship and support that made this thesis possible.

I would also like to acknowledge Thomas Kline. Sir, I've come a long way from staring blankly at Python error messages, wondering what it is I broke this time. Your help made the learning curve far less painful and made the direction of this project possible. As you said, I am genuinely excited to take these tools and prove that Marines can learn to code without contracting it out.

Finally, I am especially grateful to Commander Madeleine Fuentes for patiently enduring my many visits to her office and email inbox. Ma'am, I know I was in there a lot—but my experience at NPS, both academically and in terms of personal growth, would not have been possible without your support and encouragement.

THIS PAGE INTENTIONALLY LEFT BLANK

NPS-AM-26-284

ACQUISITION RESEARCH PROGRAM
SPONSORED REPORT SERIES

Improving Gear Provisioning Efficiency at the Individual Issue Facility (IIF) Through Size-Specific Inventory Simulation

December 2025

Capt Leah S. Weingarten, USMC

Thesis Advisors: Dr. Geraldo Ferrer, Professor
Thomas D. Kline, Lecturer

Department of Acquisition, Finance and Manpower

Naval Postgraduate School

Approved for public release; distribution is unlimited.

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

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AI	artificial intelligence
CV	coefficient of variation
CSP	Consolidated Storage Program
DES	discrete-event simulation
DLA	Defense Logistics Agency
DoD	Department of Defense
DPAS	Defense Property Accountability System
EDIPI	Electronic Data Interchange Personal Identifier
ELMS	Enterprise Logistics Management System
IIF	Individual Issue Facility
LOGCOM	Marine Corps Logistics Command
MC-ANSUR	Marine Corps anthropometric survey
MEF	Marine Expeditionary Force
MFSC	Marine Force Storage Command
NAVSUP	Naval Supply Systems Command
NSN	national stock number
PCA	permanent change of assignment
PCS	permanent change of station
PGI	personal gear issue
SAPI	small arms protective insert
SIOM	Site Demand-Based Inventory Optimization Model
SIOMsQ	Site Demand-Based Inventory Optimization Model with (*s,* Q) Pre-processing
sQUID	s and Q Unit Identification Model
USMC	United States Marine Corps
VSCode	Virtual Studio Code

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PROBLEM CONTEXT

As the Nation's expeditionary force in readiness, the United States Marine Corps (USMC) must be prepared to deploy rapidly and operate effectively in any environment. A key enabler of that readiness is the timely issue of properly sized, serviceable individual combat equipment to every Marine. The Individual Issue Facility (IIF), managed under the Consolidated Storage Program (CSP), serves as the central system responsible for providing Marines with essential gear such as helmets, body armor, and load-bearing equipment (USMC, 2016, Chapter 6, Paragraph 0603). The accuracy, efficiency, and adaptability of this system directly influence individual readiness, unit preparedness, and overall combat effectiveness of the force.

Maintaining readiness across a globally distributed force depends on the availability of the right equipment and the efficiency of the inventory systems that manage it. When Marines arrive in the fleet, delays or mismatched sizes in the issue of gear can have a real impact on the operational readiness of their units. Appropriately fitted equipment is essential to individual performance, safety, and combat effectiveness in the field. Thus, the reliability of the IIF system serves as a direct reflection of the Marine Corps' logistical readiness. However, despite its importance, the IIF's current inventory management approach remains largely static and reactive.

B. CURRENT INVENTORY PRACTICES

At each individual IIF across the Marine Corps, each Marine is issued personal equipment—such as helmets, plate carriers, and protective inserts—upon joining a unit and is required to return this gear upon reassignment or separation (USMC, 2016). The IIF is designed to provide accountability, maintain serviceability, and ensure that properly sized equipment is available to support training and deployment.

One tool that the Marine Corps currently uses to set inventory levels at IIFs is usage tariffs (Defense Logistics Agency, 2024). These tariffs specify the quantity of each sized item required to support a given Marine population and are intended to serve as standardized benchmarks for ordering personal equipment and readiness reporting. A "usage tariff" in the context of inventory management is a table that shows the proportion of each size (for example, helmet sizes S, M, L, XL) as a percentage of the total population, calculated using anthropometric data (Robinette et al., 2024). The tariff method assumes that static ratios accurately predict future restocking requirements. In reality, even small differences in unit size composition—as well as variation in how quickly certain units wear out or lose equipment—could change the rate at which specific sizes move. The inability to adapt to these dynamics leads to both overstocking of lower demand sizes and shortages of higher-demand sizes, creating inefficiencies that ultimately degrade readiness.

Tariffs are the only formalized guidance that units can follow, but there is still no standardized ordering policy across the Marine Expeditionary Forces (MEFs) for restocking size-specific inventory. Based on observed ordering patterns and inventory management practices, ordering cycles do not consistently follow tariff guidance. Orders seem to be placed on an annual or as-needed basis and are more often determined by when funding is available rather than by a fixed schedule or trigger. Because procurement often occurs when funds become available, usually at the end of the fiscal year, there is no consistent or synchronized ordering policy among the three MEFs.

This irregularity creates disparities in how effectively each MEF, as well as each individual IIF, can maintain required inventory. The absence of a formal inventory policy creates a risk of overcompensating during funding windows by ordering excessive quantities based on heuristic judgment rather than data-driven needs. It also increases the likelihood of deferring replenishment due to fiscal constraints. Together, the reliance on fixed tariffs and inconsistent ordering practices leaves the system both static—anchored to outdated size ratios—and reactive, responding to funding cycles and short-term shortages rather than proactively managing inventory based on real demand trends.

C. OPERATIONAL AND BEHAVIORAL FACTORS

Operational realities further complicate IIF inventory management. Marines may transfer between units serviced by the same IIF facility without formally de-issuing their equipment, making it difficult for the facility to account for those items when those Marines change duty stations. Over time, this creates artificial shortages and inaccurate inventory counts. Moreover, unit-level check-out and turn-in procedures vary significantly across MEFs, depending on command emphasis and local practices. These inconsistencies erode accountability and reduce the IIF's ability to maintain accurate asset visibility.

Differences in unit operational tempo may also influence equipment loss and degradation rates. Units that train in the field are more likely to experience higher rates of gear damage, loss, or inadvertent swaps among Marines during field exercises. Conversely, units that spend less time in the field intuitively would retain their equipment longer and in better condition. These differences contribute to uneven wear, unpredictable replacement rates, and data inconsistencies across the system. This also prevents the Marine Corps from applying the same lessons learned from one unit to other units with different missions or operating conditions.

D. AVAILABLE DATA

There is a large existing dataset collected through the IIFs and maintained in the Enterprise Logistics Management System (ELMS), formerly known as the Defense Property Accountability System (DPAS) before its rebranding to ELMS in 2025 (Defense Logistics Agency [DLA], 2025a). This data contains historical records of all equipment check-outs, returns, sizes, and serviceability data tracked for each Marine (USMC, 2023). The purpose of this study is to use this data as a foundation for developing more adaptive forecasting and ordering methods. Specifically, the ELMS historical records provide empirical input parameters—such as arrival rates, demand patterns, and return intervals—that can drive a discrete-event simulation (DES) model. Using the DES framework built in SimPy, an open-source Python library for modeling event-driven systems, this study will simulate the flow of gear through the IIF—from the arrival of Marines to the issue, return, or non-return of size-specific equipment. The research aims to test data-driven strategies for improving IIF ordering policies and to identify additional metrics needed to develop a reliable forecasting tool capable of supporting a standardized ordering policy across all MEFs.

The ELMS dataset does not record instances when Marines are issued the wrong size because the correct size was unavailable, which is a recognized deficiency in current data. This means there is a significant, undocumented process in action: when inventory for a required size is unavailable, Marines are issued the next closest size. For example, if a Marine requires a small helmet and none are in stock, the IIF issues a medium instead. Although corrective measures are being implemented by IIF facilities for future data collection purposes, this information is not present in the dataset used for this study. Consequently, size mismatches must be inferred indirectly (an unreliable approach), rather than directly observed.

E. DYNAMIC MODELING APPROACH AS A PATH FORWARD

Given these limitations, a simulation-based methodology may provide a useful approach. Various modeling approaches exist for analyzing inventory systems and identifying ways to improve performance. Traditional methods, such as statistical forecasting or optimization models, are useful for detecting trends in historical data and estimating future demand. However, these methods often struggle to capture the complex, event-driven nature of real operations where human behavior, timing, and uncertainty play a role.

A more flexible approach is DES, where a model is created in which the system is viewed as a state that changes over time in response to the occurrence of events. DES allows analysts to replicate real-world processes and test how policy or procedural changes might affect outcomes over time. Events of interest are those that change the state of one or more system resources. In the context of IIF operations, these events include the arrival of Marines, the issuance of size-specific equipment, the return of that equipment, and instances of non-returns or items reported as damaged.

SimPy allows users to observe entities (such as Marines, items, and facilities) interacting within a simulated system, which provides a framework for modeling the dynamic behavior of inventory operations. Using existing IIF issue and return data to derive parameters in a DES framework allows the model to approximate real-world operations as well as validate its accuracy by matching simulated outputs to historical

ELMS data. Once the model reliably mirrors observed system behavior, variables such as ordering policies, replenishment triggers, and tariff structures can be adjusted to observe their effects on inventory balance, availability, and overall readiness.

Thus, this thesis seeks to answer: Can a DES, informed by real IIF data, be used to build an accurate representation of IIF inventory behavior and then test alternative policies aimed at improving gear availability? To explore this question, the project will develop a SimPy-based DES model that captures key IIF processes and behaviors, integrating historical use data. The project will analyze inventory trends and evaluate alternative ordering and replenishment policies, with the goal of providing decision-makers with a data-driven tool to improve the availability and management of size-specific equipment within the IIF system. While size-specific equipment includes helmets, plate carriers, and Small Arms Protective Insert (SAPI) plates, the scope of this thesis is limited to helmet inventory as a representative case for modeling and analysis. The same methodology could be replicated for other size-specific equipment as needed.

F. ROADMAP

This introduction has outlined the challenges facing the Marine Corps' IIF system in managing size-specific equipment inventory, explained its current practices, highlighted operational and data limitations, and introduced simulation as a potential method for improvement. The next chapter reviews existing research relevant to this study, including the development of Marine Corps sizing tariffs, prior Department of the Navy efforts to apply demand data to inventory policy, and examples of simulation-based inventory modeling using SimPy. Chapter III analyzes the available IIF dataset for helmets, highlighting observable trends and how they will inform model assumptions. Chapter IV explains the structure of the system the model represents and shows how both the data and assumptions inform each part of the model. Chapter V presents the simulation results, beginning with a baseline validation of system behavior and followed by sensitivity tests that examine how changes in key parameters—namely, tour length and seasonality—can affect inventory performance and size availability. Chapter VI concludes the thesis with the model's key insights, implications for improving IIF inventory management, and recommendations for future enhancements.

II. LITERATURE REVIEW

This chapter reviews prior research and foundational concepts that inform the development of a data-driven inventory modeling approach for the Marine Corps IIF system. It begins by examining the origins and limitations of the current tariff-based sizing method used to determine equipment stock levels. It then reviews existing work within the Department of the Navy that has applied mathematical optimization and demand modeling to improve ordering policies, demonstrating the standard of data-informed decision-making. Following this, the chapter introduces DES and the SimPy framework as tools capable of modeling complex logistics systems. Finally, it highlights prior applications of Python-based simulation to validate and refine inventory management policies within the business sector.

A. THE ORIGINS AND LIMITATIONS OF TARIFF SIZING IN THE MARINE CORPS

As discussed in the introduction, in the context of Marine Corps supply, a usage tariff is a set of sizing distributions that dictate percentages of items for each size based on human measurement information (anthropometric data). The Marine Corps then uses these ratios to determine how many of each size are ordered and maintained at the IIFs across the Marine Corps for distribution to the fleet (Defense Logistics Agency, 2024).

The tariffs that are currently in use are based on the 2010 Anthropometric Survey of U.S. Marine Corps Personnel (MC-ANSUR) (Gordon et al., 2013). The purpose of this survey was to update anthropometric data for Marines and to replace reliance on legacy datasets, such as the 1966 Marine Corps survey and the 1988 U.S. Army ANSUR database, which had been statistically weighted to approximate Marine demographics. Researchers measured 1,921 Marines, including 1,301 men and 620 women, at Quantico, Camp Lejeune, and Camp Pendleton from September 2009 to August 2010. They recorded 94 body dimensions and 41 derived dimensions, as well as 3-D scans of Marines' heads, feet, and full bodies. The primary objective of the survey was to provide a statistically representative dataset for Marine uniform and equipment designers to reference when creating gender-inclusive and properly sized equipment for Marines.

In anthropometric engineering, a tariff is a quantitative output of the sizing loop that converts fit-test results and population data into practical stocking ratios—how many of each size should be produced, purchased, or maintained to achieve maximum accommodation at minimum cost (Robinette et al., 2024). The process relies on statistical mapping between body-measurement data and population coverage, ensuring that each size corresponds to a defined percentile range within the target population. Marine Corps Systems Command used the percentile distributions from the 2010 MC-ANSUR to develop usage tariffs for specific sets of equipment that set the proportion of gear to stock in each size. These tariffs became the basis for IIF and other gear-issuing facility inventory planning, and are found in the Marine Corps Stocklist, SL-8-09991A, Special List for Tariff Sizes (USMC, 2014).

This planning and the sizing tariffs assume that the distribution of body sizes across the force remains relatively constant. They do not consider dynamic factors like unit composition, changes in body size trends over time, or varying usage trends. As Robinette et al. (2024) explain in *Product Fit and Sizing: Sustainable Product Evaluation, Engineering, and Design*, sizing tariffs are only accurate as long as the target population and usage conditions remain stable. When either changes—through demographic shifts, new equipment designs, or evolving missions—the tariff must be recalculated and validated to remain effective. The 2010 MC-ANSUR study provided the most complete dataset available for the service at that time and was ideal for design and engineering applications, not for ongoing logistics forecasting. Tariffs based on the study are therefore a static picture of Marine anthropometry in 2010, rather than a continuously updated model.

A new anthropometric study is currently underway to replace the 2010 data. The USMC Anthropometry Survey 2025, conducted by Marine Corps Systems Command at Quantico, is collecting updated measurements from approximately 900 Marines between 2024 and 2026 (Marine Corps Systems Command, 2024). While the new study aims to provide an up-to-date anthropometric database for use in equipment design and sizing, this is still a static collection of data. No matter how current the measurements are, they still do not provide data on unit composition changes, changes in body size over time, or current usage levels throughout the force. There is still a need for a modeling tool that can use individual issue and return data to adjust inventory levels to the appropriate amount of gear in the right places as the force continues to change.

B. THE NEED FOR DATA-DRIVEN AND ADAPTIVE INVENTORY POLICIES

While tariff-based sizing provides a standardized baseline for procurement, it assumes that the size distribution and composition of the force remain constant. In practice, the Marine Corps experiences demographic changes, evolving body-size trends, and unit-level variation prompted by force reorganization—all of which affect usage patterns. To maintain readiness, inventory management must move toward data-driven models capable of adapting to real-time issue and return behavior.

Research within the Department of the Navy has already shown the value of using data and modeling to guide inventory decisions. Ersoz (2016) refined an analytical tool known as the *Site Demand-Based Level Inventory Optimization Model* (SIOM), which helps the Naval Supply Systems Command (NAVSUP) decide when and how much to reorder for thousands of stocked items. The SIOM uses mixed-integer linear programming to balance competing objectives—meeting unpredictable (stochastic) demand while minimizing total inventory cost, order frequency, and stockouts. Ersoz (2016) introduced a faster and more practical version called *SIOMsQ*, which reduced computational time by limiting the number of candidate (*s, Q*) pairs that the model tested. This refinement allowed NAVSUP planners to process large volumes of historical demand data more efficiently and generate near-optimal solutions more quickly than the original SIOM. In short, both the original SIOM and the *SIOMsQ* demonstrate how demand history can be translated into real, actionable reorder policies through optimization modeling. In both cases, the overall objective is to convert complex, stochastic demand behavior into useful logistics policy.

Building on this foundation, Deibler (2018) applied the *SIOMsQ* framework to smaller-scale, unit-level operations. His research developed a (*s*) and (*Q*) Unit Identification (sQUID) model to support the Navy's Personal Gear Issue (PGI) Divisions at Navy Expeditionary Support Units. The goal of this research was to use mathematical modeling to determine when and how much gear each PGI Division should reorder to keep costs as low as possible while still meeting demand. The model was built and tested using five years of real demand and lead time data—the same type of historical issue and return data analyzed in this research. This establishes a precedent for using individual-level demand data to optimize inventory policies for facilities that manage personally issued equipment, which is exactly what the Marine Corps IIFs manage.

C. DISCRETE-EVENT SIMULATIONS

DES is a modeling approach for representing complex, stochastic systems where events occur over various periods of time (Kelton & Law, 1991). It is widely applied to logistics planning, manufacturing, and defense operations to model and test systems under different conditions. In their book *Simulation Modeling and Analysis*, Kelton and Law (1991) define an *event* in the context of DES as "an instantaneous occurrence that may change the state of the system" (p. 7). In DES, time moves forward in steps from one *event* to the next, rather than continuously. The system is represented as a set of states that change when internal or external events occur, such as Marines arriving or leaving a unit, or them being issued and returning equipment. This approach is particularly useful for modeling queues, resource constraints, and equipment use in real-world operations.

Kelton et al. (2010) describe the key components of a DES: entities, attributes, variables, resources, queues, and events. They are the core building blocks of any model, as each represents an essential part of how real-world systems operate. Entities are the items moving through the process, attributes define an entity's characteristics, resources represent limited capacities like personnel or equipment, and queues capture waiting or processing delays (Kelton et al., 2010). This framework closely aligns with the structure of an IIF, where Marines (entities) interact with inventory items and staff (resources), each with defining attributes such as size or availability, while events include issue, return, and restock activities that occur over time.

Kelton et al. (2010) also emphasize that randomness is a fundamental component of DES, as both the input and output of real-world systems vary over time. They explain that random input—such as interarrival or service times—creates random output, which better represents the uncertainty and day-to-day fluctuations seen in operational systems. In this study, the same concept applies to the IIF, where the timing of equipment issues, returns, and losses, as well as the demand of size distributions, are naturally stochastic.

DES simulations also rely on probability distributions based on observed system data to represent stochastic processes such as arrivals, service durations, and equipment usage (Kelton & Law, 1991). Three distributions that are relevant to inventory and logistics modeling in this project are the Poisson, normal, and log-normal distributions. The Poisson distribution is a discrete probability distribution that gives the probability of an event occurring a specific number of times in a fixed time interval when those events occur independently and randomly (Mariappan, 2019, Chapter 10). In inventory and logistics applications, the Poisson distribution is widely used to model random arrival processes, demand occurrences, or event counts that unfold unpredictably over time. The normal distribution is a continuous, bell-shaped probability distribution defined by its mean and standard deviation and is widely used to represent naturally occurring variability in real-world systems (Mariappan, 2019, Chapter 11). In inventory contexts, the normal distribution is often used to represent processes where variation arises from many small contributing factors, such as repair times, processing durations, or size measurement variation of a population. The lognormal distribution describes a positive random variable whose natural logarithm follows a normal distribution (Krishnamoorthy, 2016, Chapter 23). In practical terms, it is used when most values are fairly small, but much larger values sometimes occur. In inventory systems, it commonly represents things like how long an item is used before it is returned or replaced, since most items follow a typical timeframe, but some remain in use much longer.

D. PYTHON-BASED SIMPY SIMULATION IN INVENTORY POLICY RESEARCH

Zinoviev (2024) introduces SimPy, a DES tool that uses the Python programming language. His paper provides a step-by-step exploration of system modeling using various examples. In particular, SimPy allows users to clearly define entities, processes, and resources being simulated. SimPy also allows for clear representation of state transitions, contention for shared resources, and process synchronization. All these features are essential for modeling inventory-based logistics systems.

SimPy implements inventory and resource flows through a structure called a “Store,” which functions as a queue-like container capable of holding any Python object (Team SimPy, 2023). Processes interact with a Store using operations such as “put()” and “get(),” which allow the model to represent item availability, stock movements, and waiting behavior (Team SimPy, 2023). Because a Store holds objects, it supports modeling where each item maintains its own attributes and state throughout the simulation life cycle.

Zinoviev (2024) points out that one of the key strengths of SimPy is the ease of integration with the larger Python ecosystem of analytical tools and libraries, including NumPy, Pandas, and matplotlib. This integration allows users to perform statistical analysis and data visualization directly within the model, combining simulation and analysis into one environment. This is particularly relevant to this study as it enables evaluation of how the model performs over time—tracking inventory levels, identifying shortfalls, and examining equipment flow throughout the simulated period. Zinoviev also explains how SimPy can capture considerations like resource limits, consumable supplies, and the chance of process failures—all of which reflect real-world conditions in military logistics. His examples of adding containers to represent limited resources and using probabilities to model failures are similar to how this project will model the issue, repair, and disposal of size-specific equipment at the IIF.

Python and its DES library, SimPy, have been increasingly used in logistics research to test and refine inventory policies using real-world data. Innuphat and Toahchoodee (2022) developed a simulation framework that combined SimPy with advanced forecasting techniques to validate spare-parts inventory policies in the petrochemical industry. Their study modeled a single-echelon inventory system—where each facility manages its own stock and reorder decisions independently. It used transaction and purchasing data of maintenance parts to replicate the behavior of plant maintenance and supply ordering operations. For example, to capture demand uncertainty, the authors used eleven years of historical data to create forecasts, which were then fed into a SimPy simulation to model realistic variation in spare-parts use and reordering cycles.

Innuphat and Toahchoodee's results demonstrated that the simulated fill rates closely matched observed real-world performance, confirming that Python-based simulation can effectively evaluate inventory policies before they are implemented. By combining the simulation with demand-forecasting methods, the study made its results more realistic and accurate than older, purely statistical approaches. The authors concluded that simulation not only provides an accurate testing ground for policy validation, but also generates actionable insights for improving stocking levels, reorder triggers, and service reliability.

Guerrero et al. (2022) also utilized SimPy to model a DES for an automotive wire harness production line. Specifically, the authors used SimPy to represent machines, workstations, and production flows to simulate one week of output under varying stochastic time distributions. This allowed them to assess system performance and apply SimPy to improve inventory and process policy within a manufacturing system by identifying bottlenecks and evaluating different ways of allocating resources. Although focused on the automotive sector, their study shows that SimPy can effectively model complex, multi-stage systems with stochastic variability—similar to the inventory processes at Marine Corps IIFs.

Both studies demonstrate that Python-based simulation using SimPy is a practical and proven method for modeling inventory systems. These precedents support applying a similar SimPy-based approach in this study to model the issue, return, and replenishment of size-specific combat equipment, evaluate alternative ordering policies, assess the effectiveness of current stocking methods, and identify data-driven opportunities for improvement.

THIS PAGE INTENTIONALLY LEFT BLANK

III. DATA ANALYSIS

This chapter presents a general analysis of the dataset using basic statistical methods to identify key patterns and relationships. The analysis establishes a foundation for drawing conclusions that will guide the integration of this data into the simulation model in later chapters. The chapter begins by describing the origin of the dataset and explaining the rationale for focusing on helmet issue data as a representative case for individual size-specific equipment. It then examines trends in helmet issues over time to identify demand patterns across sizes and months, followed by trends in helmet returns to explore factors influencing return rates and discrepancies. Together, these analyses provide insight into user behavior and inform how the model represents inventory flow, turnover, and size distribution within the IIF system.

A. DATA SOURCE

The data used for this analysis was sourced from the ELMS, which is defined on their official website as "a Department of Defense (DoD) property management system" and "the Accountable Property System of Record (APSR) for over 50 DoD Agencies and Military Services" (DLA, 2025b). Access to the dataset was provided through Marine Force Storage Command (MFSC), a subordinate organization of the Marine Corps Logistics Command (LOGCOM). The specific data needed for this thesis was requested through MFSC, which pulled and provided the relevant records from ELMS on behalf of this research. The dataset encompasses all IIF facilities operating under I MEF and includes three years of issue and return records, covering July 2022 through July 2025. This timeframe was selected to align with the average length of a Marine's tour of duty. Each record is linked to an individual Marine through a unique identification number and includes detailed information such as equipment type, size, and serviceability condition.

The focus of this analysis is on helmet issue and return data, selected because it provides a clean, size-specific dataset with minimal classification overlap or ambiguous sizing categories. While this chapter centers on helmet data, the analytical approach and resulting insights are designed to be scalable to other size-dependent items such as plate carriers and SAPI plates. The findings derived here will therefore serve as a model for evaluating and improving size-based inventory management practices across the broader IIF system.

B. HELMET ISSUE DATA

Analysis of the helmet issue dataset provides insight into how equipment demand fluctuates across time and size categories within the IIF system. From 2022 to 2025, monthly issue quantities for small, medium, large, and extra-large helmets displayed recurring surges that likely correspond to training cycles, unit turnovers, and pre-deployment preparation periods. As shown in Figure 1, these cycles generally align across all sizes, indicating shared external causes of demand. However, there are several instances—notably in August–September 2024—where one size spiked independently while others did not. Such deviations likely suggest that supply limitations and substitution may have influenced issue behavior in addition to true demand.

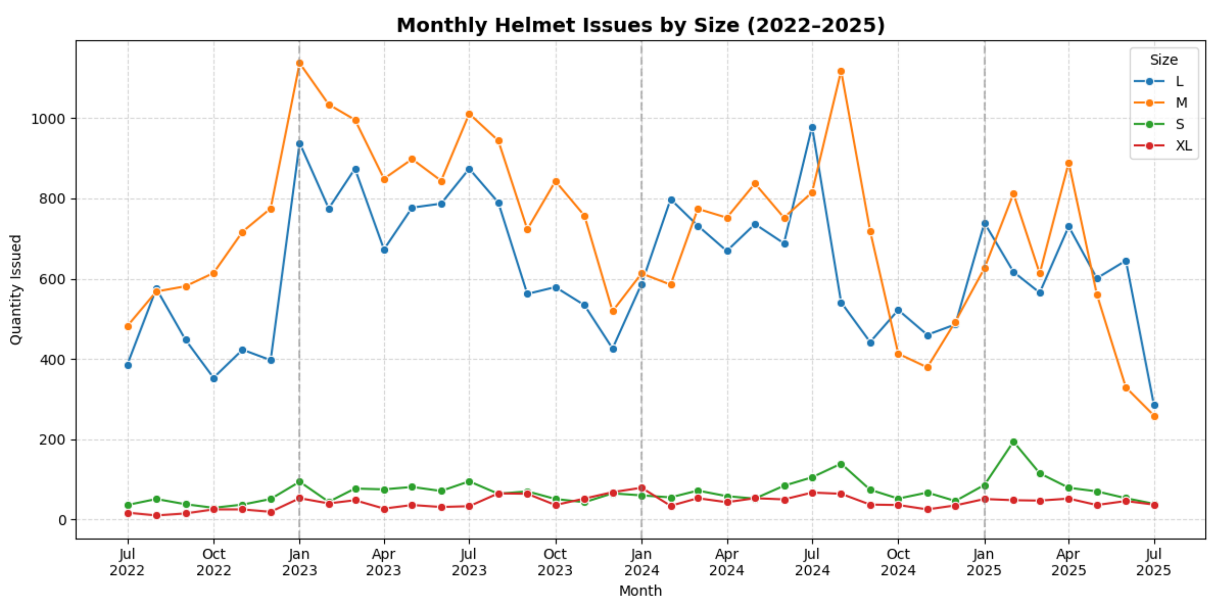

Figure 1. Monthly Helmet Issues by Size (2022-2025). Adapted from C.J. Grimes, personal communication (2025, July 10).

To better understand how each size contributes to the total issue mix, Figure 2 shows the proportional distribution of each size as a share of total helmets issued each month. While the relative proportions remain mostly stable over time, periodic distortions are evident, such as in mid-2024, when medium helmets temporarily spiked in proportion. These deviations suggest that when one size experiences a shortage or replenishment delay, other sizes may compensate through substitution. This pattern supports the idea that fluctuations in issue data may reflect both operational demand and supply constraints.

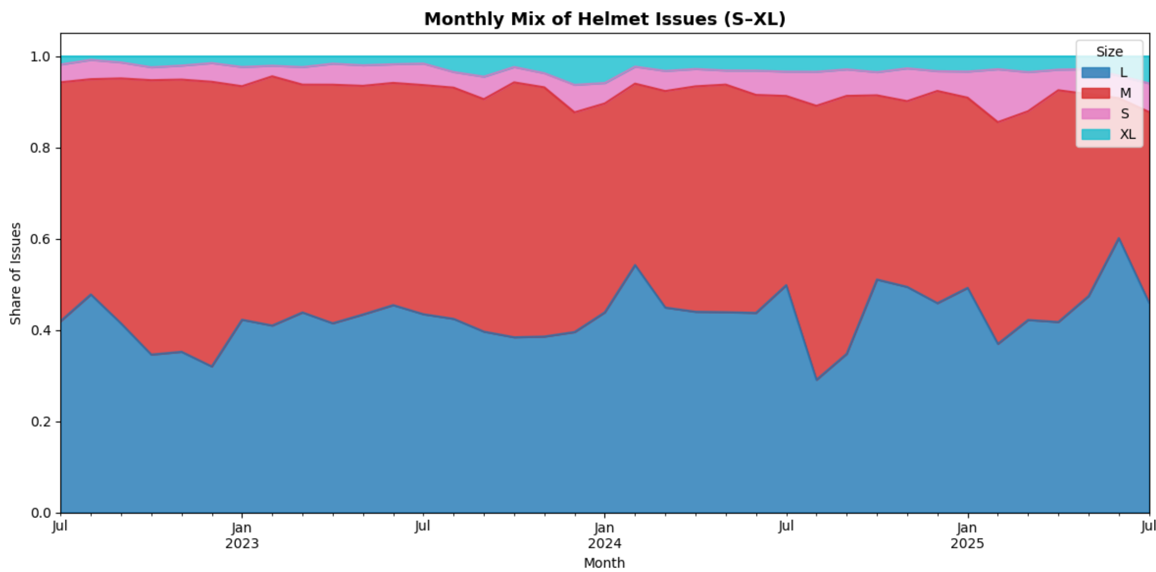

Figure 2. Monthly Mix of Helmet Issues (S-XL). Adapted from C.J. Grimes, personal communication (2025, July 10).

The relative volatility of demand for each size, measured using the coefficient of variation (CV), highlights differences in predictability. As shown in Figure 3, medium and large helmets exhibit the lowest variability (CV = 0.30 and 0.28, respectively). This shows stable and consistent demand over time for those sizes. However, small and extra-large helmets show higher volatility (CV = 0.45 and 0.39). This elevated variability means that issue quantities for these edge sizes fluctuate more sharply month to month. Some volatility is expected, but this may indicate substitution effects. It is possible that small and extra-large sizes, being the end sizes and representing a smaller proportion of total demand, are disproportionately affected by substitutions. For example, substituting 10 medium helmets with small ones represents a much larger share of total small demand than it does for medium.

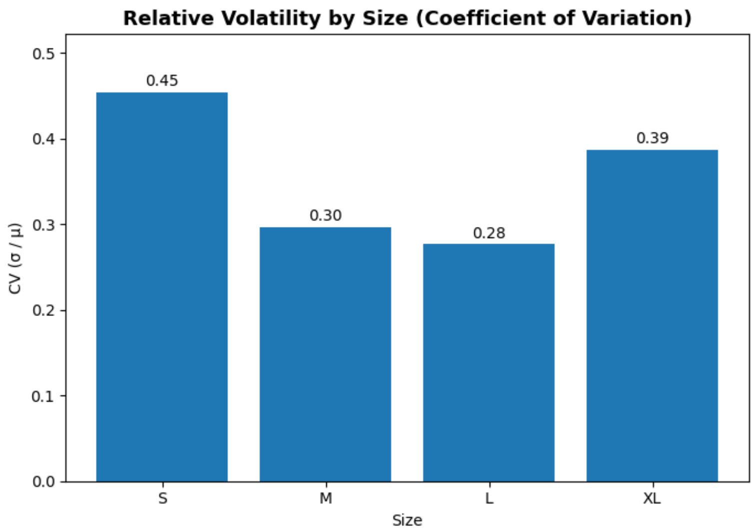

Figure 3. Relative Volatility by Size. Adapted from C.J. Grimes, personal communication (2025, July 10).

C. HELMET RETURN DATA

When evaluating return data, the most meaningful insight comes from analyzing the equipment that is not returned. Helmets recorded as unreturned—cases where a Marine completes the checkout process but fails to return the item—highlight potential friction points in the issue and recovery process, such as substitution, loss, or administrative error. These instances provide a clearer picture of system inefficiencies than the items that follow normal return patterns.

Figure 4 visualizes monthly counts of helmets that were issued but not returned, segmented by size. The trend shows that unreturned helmets generally rise and fall proportionally across all sizes. Larger spikes occur in 2023, but the relative distribution among sizes remains consistent. This proportionality suggests that the rate of non-return is consistent with the overall issue size distribution. In other words, there is no evidence from the data that a particular size is disproportionately unreturned. The 2023 surge in overall unreturned helmets likely reflects a broader operational or administrative event (e.g., deployment turnover, inventory reconciliation, or procedural change), but no explicit policy or contextual explanation is available within the scope of this study.

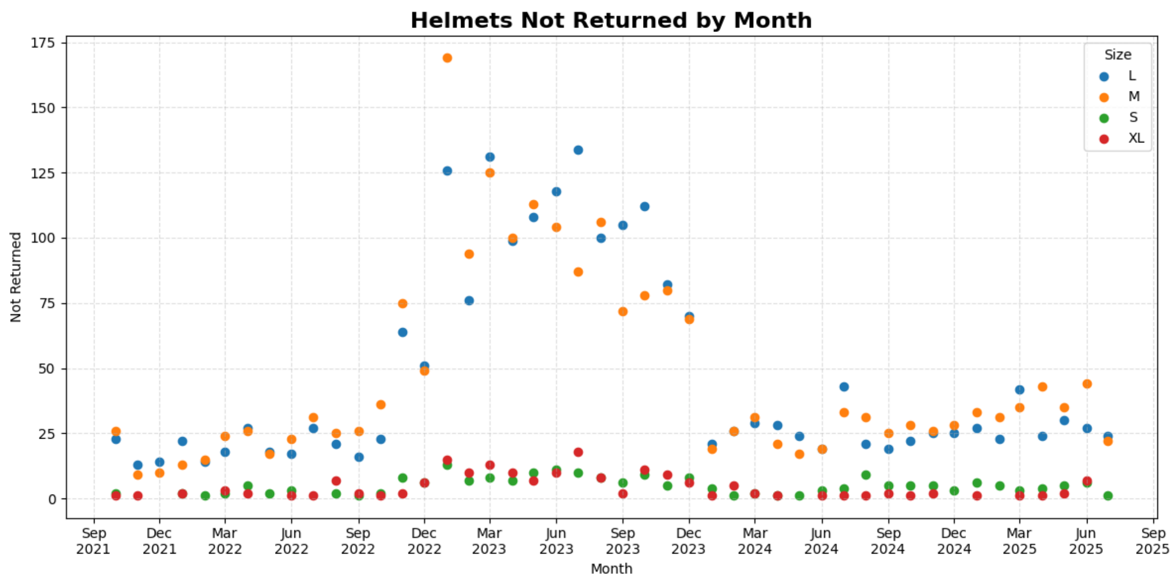

Figure 4. Helmets Not Returned by Month. Adapted from C.J. Grimes, personal communication (2025, July 10).

To gain insight into these inefficiencies, this analysis examines the reasons behind unreturned helmets to determine whether they result from true losses or administrative substitutions. Table 1 shows that among all unreturned records, the vast majority (75%) are due to returning a different size, directly indicating an undocumented size swap.

Table 1. Count of Reason Codes for Non-Returned Helmets. Adapted from C.J. Grimes, personal communication (2025, July 10).

Reason Code for Non-Return	Count
RD – Returned Different Size/Serial Nbr	3492
GA – FLIPL DD-FORM 200/Govt Authorized	541
GR – FLIPL DD-FORM 200/Govt Reimbursed	498
RL – Return Later	93
KI – KIA/MIA/WIA LOSS	23
RE – Returned Outside ELMS	10
CN – Consumed	5
LT – Left in Theatre	2
MD – Missing/Damaged Gear Statement	2

To confirm whether these unreturned helmets represent actual losses or simple size swaps, Figure 5 compares helmets that were issued but never returned with those recorded as returned without a corresponding issue. As helmets are not serialized, this comparison helps determine whether the discrepancies reflect administrative swaps between sizes rather than missing inventory.

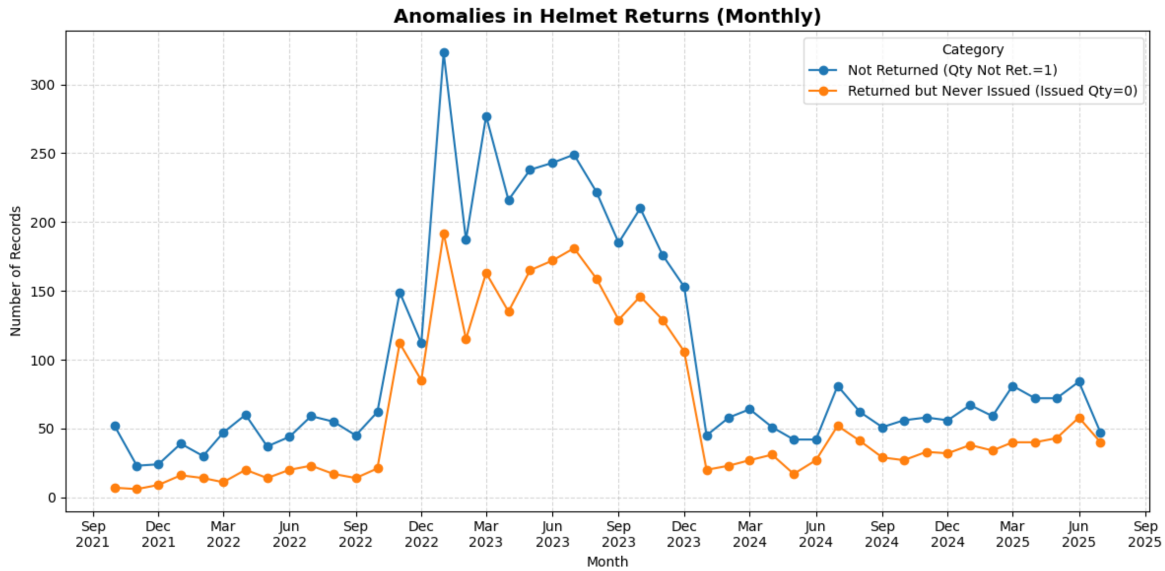

Figure 5. Anomalies in Helmet Returns. Adapted from C.J. Grimes, personal communication (2025, July 10).

The close alignment between the two trends in Figure 5 suggests that many of the "missing" helmets are not actual losses, but swaps or cross-returns between sizes. The data supports this interpretation, as the majority of unreturned helmets have corresponding transactions indicative of size swaps or reconciliation errors, rather than outright losses. Overall, the return data analysis indicates the following key conclusions:

- Unreturned helmet quantities align with expected size distributions.
- The correlation between "not returned" and "returned-but-never-issued" helmets suggests transactional overlap (swaps).
- Nearly all non-returned helmets have documented reason codes, the majority being size swaps.

D. SUMMARY

The data analysis reveals consistent demand patterns across helmet sizes and highlights how supply constraints and size substitutions influence issue and return behavior within the IIF system. Medium and large helmets show the most predictable demand, while small and extra-large sizes fluctuate more sharply due to lower overall volume. Return data confirms that most “missing” helmets result from size swaps rather than losses. These insights form the empirical foundation for the model introduced in Chapter 4, which uses these patterns to represent equipment flow, demand variability, and inventory responsiveness.

IV. SYSTEM MODEL AND SIMULATION DESIGN

This chapter describes how the model represents the IIF system. The goal is to translate observed patterns—such as size-specific demand, substitution, and return rates—into a dynamic simulation that reproduces real inventory behavior. The chapter first outlines the logical flow of the system, then explains how the simulation replicates that flow using DES methods, and how the data analysis from Chapter III informs its structure and behavior

A. SYSTEM OVERVIEW

The model represents how helmets move through the IIF, focusing on size-specific availability, substitution, and turnaround. It simplifies the real process to emphasize key interactions between Marines, inventory, and time. Figure 6 shows the conceptual flow of helmets through the IIF system.

The diagram reads from left to right: new helmets enter the system and are sorted into size-specific inventories, while incoming Marines enter the process and are assigned a required size. During the issue process, if the preferred size is unavailable, Marines are routed one size down according to the substitution rule. Once issued, both Marines and their helmets move into the fleet, represented by the central triangle labeled "Fleet." This element symbolizes the period during which Marines retain their helmets while serving their tours, during which it is possible for "lost" helmets to leave the system. After tour completion, Marines proceed through the return process, where helmets are inspected and routed for repair, disposal, or return to serviceable inventory based on their condition. Orange lines depict the flow of equipment, and green lines trace Marine movement and decision logic throughout the system.

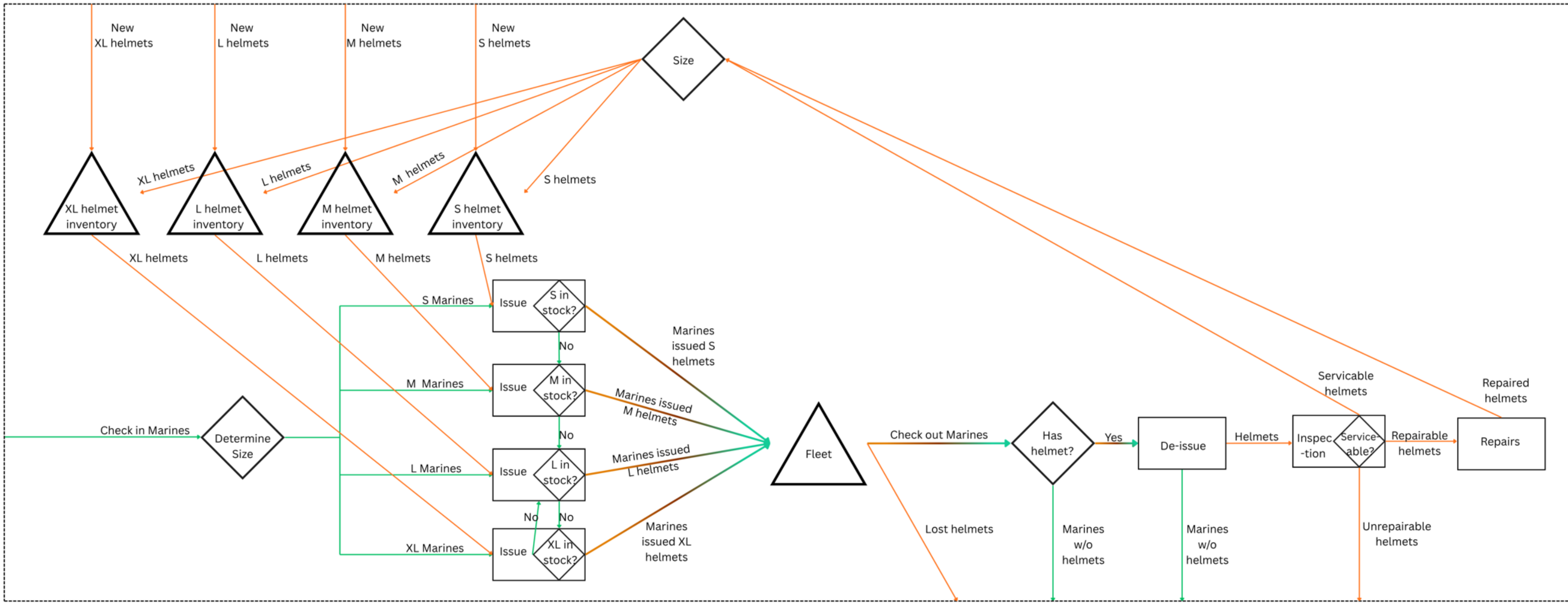

Figure 6. Individual Issue Facility Process Diagram for Helmets

B. THE MODEL

The simulation uses a DES framework in SimPy to replicate the processes illustrated in Figure 6, with the complete simulation code found in Annex A. Each Marine enters the system as a unique instance with a personal identification number and an assigned size requirement. The simulation environment continuously generates new Marines, each arriving randomly to mirror the unpredictable nature of real issue activity.

When a Marine checks in, the system attempts to issue a helmet of the requested size from the available inventory. If that size is unavailable, the Marine follows a substitution rule, receiving the next available size in the sequence. Every helmet also exists as an individual instance, labeled by size and serial number, and can only be checked out to one Marine at a time. Once issued, the Marine and helmet remain paired during a simulated tour of duty, after which the Marine returns—or sometimes fails to return—the helmet.

Inventory is modeled with per-size SimPy Store containers (helmet_store [‘S’, ‘M’, ‘L’, ‘XL’]). Each helmet is an object in a registry (all_helmets) and moves between the IIF store and Marines by the code “Store.put()/Store.get().” Daily snapshots record store items for on-hand counts and inspect the all_helmets registry to calculate checked-out counts while excluding disposed or lost items.

The model operates within a simulated timeline that allows days, months, and years to pass in compressed form. Early activity is treated as a warm-up period, giving the system time to reach a steady operating rhythm before recording results. Throughout the simulation, each transaction—issue, substitution, return, repair, or loss—is recorded in an activity log. This provides a detailed record of how helmets circulate through the system, how shortages and substitutions occur, and how inventory evolves over time.

The first version of the simulation was built using artificial intelligence (AI) coding assistant tools. I started by typing a short prompt into the built-in AI chat within Visual Studio (VS) Code Version 1.103.1, August 2025, and GitHub Copilot created a basic

Python script using the SimPy library. The prompt asked for a simple model where Marines arrive over time and receive different sized helmets from a limited supply. This generated a foundation that I later expanded by adding real data, substitution rules, and inventory reordering policies. With "agent mode" activated in VSCode, I used the integrated chat feature (powered by the GPT-4o model) to make edits, get explanations, and fix errors directly in the code as the model developed.

C. DERIVING MODEL INPUTS FROM ISSUE AND RETURN DATA TRENDS

This section describes how the data from Chapter III is incorporated into the simulation to ground its behavior in observed trends, specifically for size demand, arrival rate, tour length, and condition upon return.

1. Size Demand

To establish the baseline distribution of helmet demand by size, monthly issue was analyzed by issue date, size, and quantity issued. Issue dates were converted to monthly periods, and total quantities were aggregated by size for each month. The monthly totals were then used to calculate each size as a share of total helmets issued that month, giving a series of monthly size proportions over time. From this, the mean and standard deviation were calculated for each size to represent both the average tendency and the month-to-month variability of demand, shown in Table 2.

Table 2. Distribution of Helmet Size Demand by Mean and Standard Deviation. Adapted from C.J. Grimes, personal communication (2025, July 10).

Size	Mean_share	StdDev_share
S	0.0482	0.0175
M	0.4924	0.0657
L	0.4296	0.0604
XL	0.0298	0.0123

The official tariff factors from the Stocklist currently in use are: S = 0.030, M = 0.344, L = 0.594, XL = 0.032 (USMC, 2014). When comparing these values with those from Table 2, both distributions show that most Marines fall into the medium and large size ranges, while small and extra-large sizes represent a minor share. Table 2 shows a slightly higher share of small helmets and a lower share of large ones than the ones derived from the 2010 MC-ANSUR study, suggesting a modest demographic shift toward smaller sizes or sampling variation among the units surveyed. Nonetheless, both datasets reinforce that medium and large helmets dominate inventory demand.

The parameters from Table 2 inform the model's demand inputs: the mean defines the expected value of the size distribution, and the standard deviation determines the variability. For realistic variation in size demand, each month's size proportions are allowed to fluctuate within one standard deviation ($\pm 1\sigma$) of the mean for that size. This ensures that simulated demand reflects the variability observed in the data. A minimum floor of 1% is also applied to prevent any size exclusion. The code block from the simulation that represents these parameters is shown in Figure 7.

In the code from Figure 7, the size category is represented by the variable s. Each specific size is represented by S, M, L, and XL, which are sampled from a normal distribution defined by its mean (mu) and standard deviation (sd) using NumPy (np), a Python library for performing complex mathematical operations on many values at once. The function "get_monthly_size_demand()" uses these parameters to generate a random demand share for each size within these limits, then normalizes the results so that all four size proportions sum to one. In simple terms, this function creates a realistic month-to-month variation in helmet size demand while maintaining the distribution observed in the Table 2 data.

```python
# ------------------------------
# Monthly size-demand model (empirical mean ±1σ, 1% floor, normalized)
# ------------------------------
SIZE_STATS = {
    "S":  (0.0482, 0.0175),
    "M":  (0.4924, 0.0657),
    "L":  (0.4296, 0.0604),
    "XL": (0.0298, 0.0123)
}

def get_monthly_size_demand():
    """Generate monthly size proportions within ±1σ (1% minimum), then
normalize to sum to 1."""
    shares = {
        s: np.clip(np.random.normal(mu, sd), max(0.01, mu - sd), mu + sd)
        for s, (mu, sd) in SIZE_STATS.items()
    }
    total = sum(shares.values())
```

Lines 17–34 from Annex. Code defining the baseline helmet size demand distribution. Monthly proportions fluctuate within one standard deviation of each size's mean, with a 1% floor to prevent zero-demand sizes and normalized to maintain a total of 100%.

Figure 7. Monthly Size-Demand Function and Parameters

2. Arrival Rate

The arrival rate represents the tempo of new issues entering the system. Using the issue dataset, all transactions were grouped by month and converted into a daily rate. This shows how many Marines, on average, were issued helmets each day across the system. From July 2022 through July 2025, the average arrival rate was 47.8 issues per day, with a standard deviation of 12.3 and a coefficient of variation of 0.26. This means that daily issue activity was fairly consistent over time, with only moderate month-to-month variation. The rate ranged from about 30 to 60 issues per day, reflecting normal operational changes such as pre-deployment surges or rotation cycles.

In the model, new Marines enter the system following a Poisson arrival process based on the observed rate of $\lambda = 47.8$. The Poisson distribution is used because it represents random arrivals that happen independently over time, which fits the way

Marines check in and receive gear—steady on average, but unpredictable from one day to the next. This approach keeps the model realistic while reflecting the natural variation seen in the data, and it is shown in the code in Figure 8.

```
# Arrival rate (Poisson process) in arrivals per day
ARRIVAL_RATE_PER_DAY = 47.8

def continuous_marine_arrivals(env, iif, arrival_rate=ARRIVAL_RATE_PER_DAY):
    """
    Poisson arrivals: interarrival times ~ Exponential(rate=arrival_rate per
day).
    Each arriving Marine requests a size drawn from the current monthly size
distribution.
    """
    marine_count = 0
    while True:
        # exponential interarrival with rate lambda (arrivals per day)
        next_interarrival = random.expovariate(arrival_rate)
        yield env.timeout(next_interarrival)

        marine_count += 1
        marine_id = f"M-{marine_count:05d}"

        # draw size from current monthly demand mix maintained by IIF
        sizes = list(iif.size_demands.keys())
        weights = list(iif.size_demands.values())
        requested_size = random.choices(population=sizes, weights=weights,
k=1)[0]

        marine = Marine(env, marine_id, env.now, requested_size)
```

Lines 337–360 from Annex. This defines the arrival process for how Marines enter the simulation using a Poisson process with an average rate of $\lambda = 47.8$ arrivals per day. In a Poisson process, the time between arrivals (next_interarrival) follows an exponential distribution with rate λ. The code draws the next_interarrival time, advances the simulation clock by that amount, assigns a marine_id, selects a size, and then starts the Marine's life cycle in the model.

Figure 8. Poisson Arrival Process for New Marines

Although common knowledge indicates that typical rotation patterns have more arrivals occur in the summer months, with smaller off-cycle spikes in winter, the data did not show a strong seasonal trend. Because of that, using a single, constant Poisson is a realistic and defensible simplification of the model.

3. Tour Length

Tour length represents the amount of time a Marine retains an individually issued helmet before returning it to the IIF. To estimate this, issue and return datasets were merged by Electronic Data Interchange Personal Identifier (EDIPI) and helmet size, allowing the duration between the issue and return dates to be calculated for each Marine. Invalid or incomplete records were excluded to ensure accuracy. A box plot of this data is shown in Figure 9. Across all valid records (1,882), the average holding period was 399 days with a standard deviation of 253 days. The minimum observed duration was one day, and the maximum was just under three years (1,084 days).

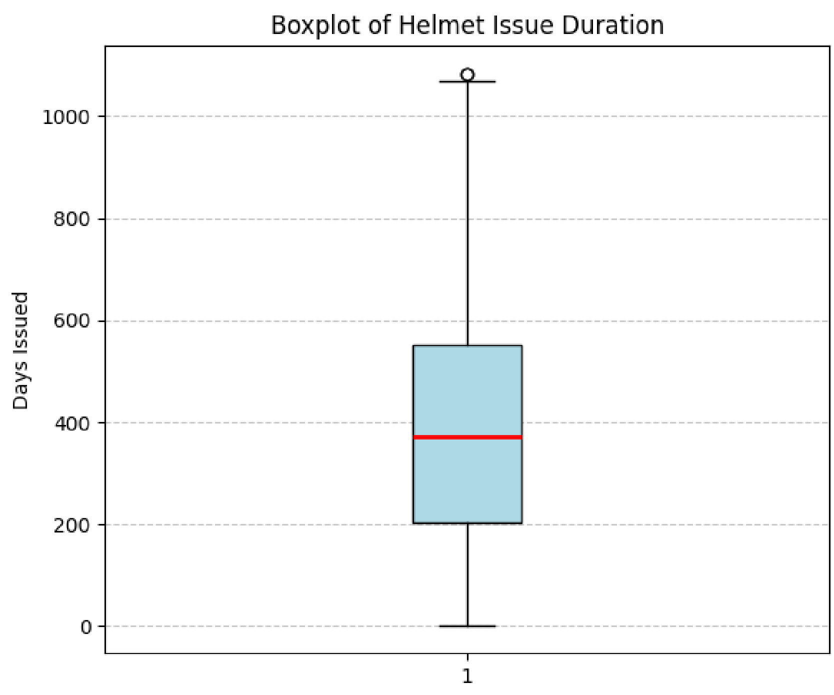

Figure 9. Boxplot of Helmet Issue Duration. Adapted from C.J. Grimes, personal communication (2025, July 10).

It is somewhat surprising that the average duration is just over one year—significantly below the nominal three-year tour length for Marines. However, this could make sense when considering how gear is managed throughout a Marine's assignment. Based on observation and experience, Marines will arrive and check in before they actually report to the IIF to receive their gear. Likewise, most will turn in their equipment weeks before departing. There are also Marines who participate in temporary assignments or inter-unit transfers—for example, when a junior Marine is sent to another unit for deployment purposes. In addition, Marines in the dataset have multiple issue and return events, which can occur for several reasons: lost or damaged gear, size swaps, or temporary duty assignments. Finally, there are those Marines who have early unplanned departures from their assignment. These factors together explain the shorter average while also contributing to the wide variation in the data ($\sigma = 253$ days).

In the simulation, tour length is modeled as a lognormal distribution with a mean of 399 days and a standard deviation of 253 days. Lognormal distribution is used because it supports a random variable that is always positive without an upper limit. It is a common choice for inventory management to represent demand with a high coefficient of variance always generating positive values. These parameters define the distribution of how long each Marine remains in the system and are depicted within the simulation code in Figure 10.

```python
class Marine(object):
    """Marine with ID, arrival day, required helmet size, and tour length.
    Tour length ~ LogNormal with target mean=399 and sd=253.
    """
    def __init__(self, env, marine_id, arrival_day, helmet_size):
        self.env = env
        self.marine_id = marine_id
        self.arrival_day = arrival_day
        self.helmet = None
        self.helmet_size = helmet_size
        # Draw from a log-normal parameterized to have approx mean=399 and
sd=253.
        target_mean = 399.0
        target_sd = 253.0
        var_ratio = (target_sd ** 2) / (target_mean ** 2)
        sigma_ln = math.sqrt(max(1e-12, math.log(1.0 + var_ratio)))
        mu_ln = math.log(max(1e-12, target_mean ** 2 / math.sqrt(target_sd **
2 + target_mean ** 2)))
        tour = random.lognormvariate(mu_ln, sigma_ln)
```

Lines 49–66 from Annex. Creates a blueprint (class) for Marines. Each Marine will be an object with properties like an ID number, helmet size, and tour length. The tour length parameter is how long a Marine retains an issued helmet before returning it to the IIF. It is sampled from a lognormal distribution parameterized to approximate a mean of 399 days and a standard deviation of 253 days.

Figure 10. Marine Class Definition and Tour Length Assignment Logic

4. Return and Condition Probabilities

The first option that the simulation tests is whether the helmet is returned upon completion of the tour. To estimate the probability that a helmet issued to a Marine is never returned, the full return dataset was analyzed. As each record represents an accountability event—either a successful return or a non-return—it inherently captures all helmets that reached the end of their expected issue period. Across all transactions, 8.3% of helmets were marked as not returned, representing instances where the Marine did not bring the item back to the IIF for standard turn-in. This rate therefore reflects the observed average probability that a helmet fails to re-enter the system after being issued, regardless of the specific reason.

However, most of these were the result of size-for-size swaps recorded under the code “RD – Returned Different Size/Serial Number.” These cases represent normal swap behavior rather than true losses since the Marine swapped one helmet for another and accountability was maintained. Excluding those transactions, the adjusted non-return rate is approximately 2.1% of all returnable helmets. In the simulation model, this 2.1% value represents the probability that a helmet is permanently lost or otherwise removed from the system.

The second option the simulation tests for is the condition of the helmet, assuming the Marine did return it. Helmet condition upon return was analyzed using the Condition Cd field from the MFSC return dataset. Each record was categorized as serviceable (codes A/B/D), repairable (codes E/F/G), or marked for disposal (code H) using the return data. Across all helmet returns from 2022–2025, 99.7% were serviceable, 0.2% required repair, and 0.1% were condemned for disposal. These proportions replace the notional weights previously assumed in the simulation, grounding the model in observed performance data.

As shown in Figure 11, these proportions are treated as fixed probabilities rather than stochastic variables for the simulation. The data indicates that the majority of helmets re-enter inventory immediately after return, with only a very small fraction delayed for repair or removed. In the model, helmets returned in serviceable condition are immediately restored to inventory, while those classified for disposal are permanently removed from the system. Although this study does not include data on repair turnaround times, repairable helmets are assumed to remain out of circulation for approximately one month before reentering inventory.

```python
# Helmet is returned — sample condition conditional on return (empirical)
        condition = random.choices(
            ["good", "repair", "dispose"],
            weights=[0.997, 0.002, 0.001],
            k=1
        )[0]
        helmet.condition = condition
```

Lines 294–300 from Annex A. This code determines what happens when a Marine finishes their tour. Each helmet has a small chance (2.1%) of never being returned, representing permanent loss. If it is returned, the code assigns a condition—good, repair, or dispose—based on observed data.

Figure 11. Helmet Condition Outcomes on Return

D. REMAINING ASSUMPTIONS AND PARAMETER RATIONALE

This section explains the remaining parameters and modeling decisions that were not directly derived from empirical data. These assumptions were informed by observed trends in the ELMS dataset, operational experience with the IIF process, and realistic fiscal or administrative constraints.

1. Repair Time Assumption

Direct repair-time data was unavailable. However, the ELMS dataset indicated that a very small percentage (0.2%) of helmets were returned for repair. Based on that observation, a mean repair duration of 30 days was assigned. This value represents the average time a helmet would remain unavailable for issue after being flagged as unserviceable.

2. Starting Stock Calculation

Table 3 shows the initial stock levels of helmets by size used at the start of the simulation. These levels were derived from the simulation design parameters. The model was structured to represent an IIF supporting a MEF-sized population, with the total number of helmets determined by an arrival rate of 47.8 Marines per day and an average helmet retention time of 399 days, as calculated earlier in this chapter in the section on tour length. These parameters correspond to roughly 19,072 Marines with helmets checked out at any given time. To maintain a stable system and account for items in repair, transit, or awaiting reissue, an additional 10 percent was added, bringing the total starting inventory to 20,993 helmets.

Table 3. Initial Helmet Inventory by Size

Size	Quantity	Percentage
Small	1,012	4.82%
Medium	10,337	49.24%
Large	9,019	42.96%
Extra-Large	625	2.98%
Total	20,993	100%

3. Reorder Logic and Frequency

Because reorder activity was not captured in the available dataset nor available at the time of this thesis for the time period observed, ordering behavior was modeled according to typical Marine Corps supply practices and annual fiscal constraints as discussed in the introduction. The model initiates one reorder cycle per year. A reorder is triggered whenever the on-hand inventory falls to 5 percent or less of its starting level, and each reorder replenishes 10 percent of the original starting stock. This reflects a conservative, budget-constrained approach that includes periodic replenishment.

4. Lead Time Source and Value

Lead time was fixed at 174 days, corresponding to the contracting data provided in FedMall for the item listed as "'HELMET, GROUND TROOP' (NSN 8470–01-592-6215)" (DLA, 2025c). FedMall serves as the DoD's centralized e-commerce platform for acquiring materiel and supplies, integrating catalog, procurement, and pricing data from across DLA's supply chain systems (DLA, 2021). The 174-day lead time represents the average total acquisition cycle, including administrative lead time and production lead time, derived from past contracting performance (R.L. Greene, personal communication, October 24, 2025). This captures the time required for both vendor manufacturing and delivery. While individual orders may vary depending on stock availability and vendor performance, the 174-day figure provides a realistic and conservative estimate that reflects actual procurement conditions.

5. Simulation Duration

The simulation was run for 10 years with a 4-year warm-up period to ensure the results reflected stable, steady-state behavior rather than effects caused by the initial starting conditions. Helmet retention times average roughly one year, and resupply operates on long procurement cycles, with a 174-day lead time and only one reorder opportunity each year. Because of these slow system dynamics, several full issue–return–reorder cycles are required before the effects of the initial starting inventory dissipate. A four-year warm-up allows multiple Marine cohorts to complete tours and ensures that at least one or two full resupply cycles have occurred, bringing the system into equilibrium. The remaining six years provide a sufficiently large sample—over 100,000 Marine arrivals—for estimating stockouts, substitutions, reorder behavior, and size-specific performance.

V. SIMULATION EXECUTION

This chapter presents the execution of the simulation, beginning with a baseline test to confirm that the model reaches a steady state and behaves realistically. It then conducts two sensitivity tests—one focused on longer tour durations and one introducing seasonal arrival patterns—to examine how changes in key assumptions affect inventory levels, reorder behavior, and size availability. Together, these tests verify that the model is functioning as intended and provide a foundation for evaluating inventory policies under varied operational conditions.

A. BASELINE TEST: MODEL VALIDATION AND REALISM

Before testing any policy changes, the model must show that it behaves in a way that matches how an IIF actually operates. This section runs a baseline test of the simulation and examines the results to verify whether the model reaches a stable steady state, whether size-specific issues and inventory levels look realistic, whether reorders occur at the right times, and whether swap and loss patterns make sense. These checks show that the model is behaving as expected and is reliable enough to use for the policy testing that follows.

1. Steady-State Behavior of the System

After the four-year warm-up period, the simulation converges to a stable steady state. As shown in Figure 12, the number of helmets checked out rises sharply during the early portion of the run and levels off around day 1500 to just under 20,000 Marines with helmets actively issued. This matches the expected steady-state population derived from the model's arrival and retention parameters.

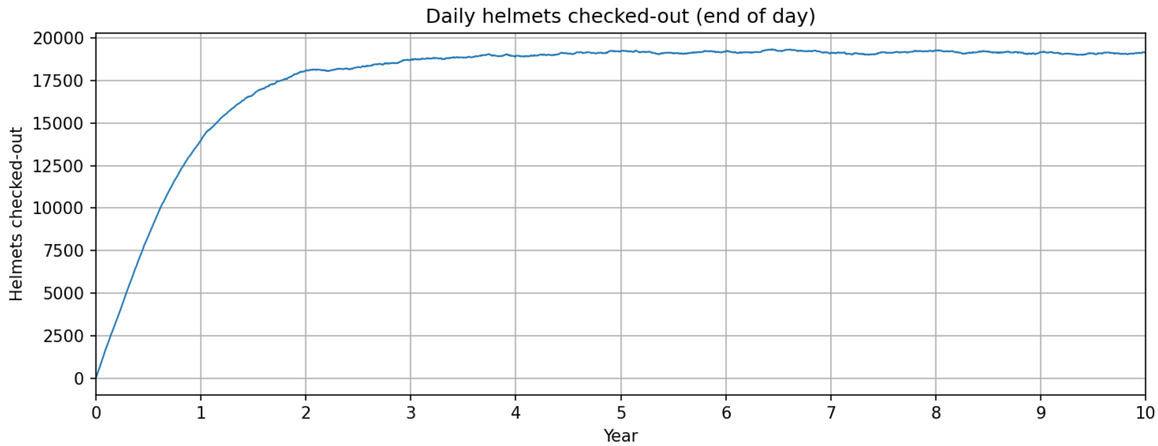

Figure 12. Total Daily Helmets Checked-Out for Baseline Simulation

2. Size-Specific Inventory Dynamics

Figures 13 and 14 display the on-hand and checked-out helmet levels by size. These graphs show that medium and large helmets stabilize at higher issued levels, reflecting their higher demand shares, while small and extra-large helmets exhibit lower but consistent steady-state rates. The on-hand inventory curves in Figure 13 also show periodic upward spikes, corresponding to reorder deliveries, while Figure 14 shows that size-based issue levels remain stable over time.

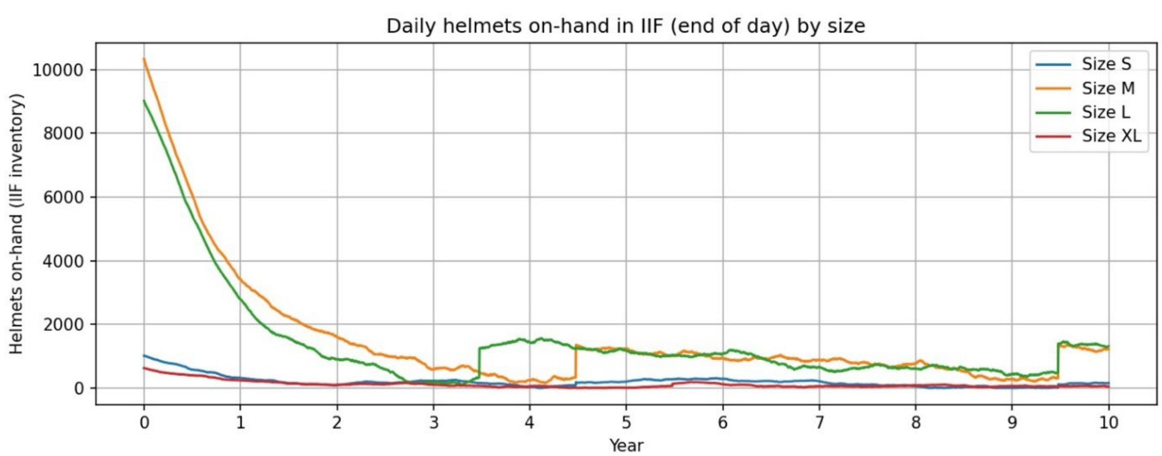

Figure 13. Daily Helmets by Size on Hand in IIF for Baseline Simulation

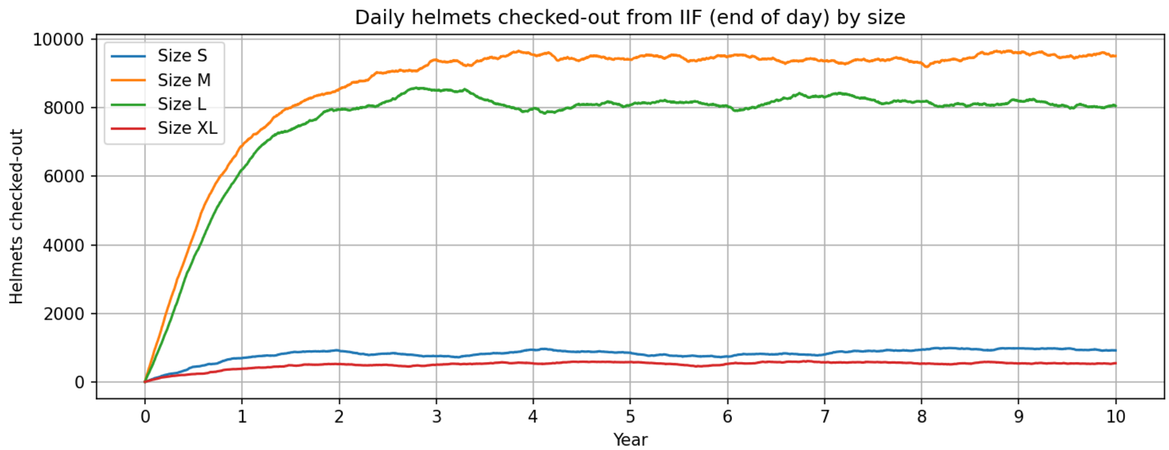

Figure 14. Daily Helmets by Size Checked-Out from IIF for Baseline Simulation

3. Reorder Behavior and Lead Time Verification

Reorders under the baseline policy occur naturally as inventories fall to 5 percent of the initial stock. These events are summarized in Table 4, which lists the day, year, size, and quantity of each replenishment.

Table 4. Reorders During Baseline Simulation

sim day	year	size	Qty
1095	3	L	901
1460	4	S	101
1460	4	M	1033
1825	5	XL	62
3285	9	S	101
3285	9	M	1033
3285	9	L	901

The timing of the spikes in Figure 13 aligns with these reorder events, appearing approximately 174 days after each end-of-year order modeled based on the FedMall-derived lead time. This confirms that the reorder logic and procurement delays are functioning properly within the system.

4. Swap and Loss Behavior

Swap events occurred whenever a Marine's requested size was unavailable, and the rate observed in the model remained low: 223 occurrences (0.21% of all issues processed). This matches expectations given the rarity of size-substitution events in current IIF operations. The model also produced a small but persistent number of permanently lost helmets: 3,238 (3.01% of all helmets in the system), reflecting the documented non-return rate and confirming that helmet attrition is being captured appropriately.

5. Comparison to Real-World Conditions

Because daily on-hand and checked-out numbers are not tracked consistently in the real system, a complete comparison over time was not feasible. However, data from an Enterprise Data Report produced by MFSC on 5 June 2025 shows 3,343 helmets on hand and 24,216 issued across I MEF (C.J. Grimes, personal communication, June 5, 2025). These values fall in the same range as the simulation's steady-state levels, though differences exist due to non-standardized reorder practices, local variations, and the lack of comprehensive daily snapshot data. Despite these differences, the alignment of overall totals suggests the model is capturing the essential behavior of a MEF-scale IIF inventory.

6. Overall Assessment and Transition to Policy Testing

Taken together, the steady-state behavior, size-specific stability, realistic reorder timing, and plausible swap and loss patterns all indicate that the simulation is operating as expected and providing a credible approximation of real-world IIF dynamics. Because the model reliably reflects current system behavior, it can now be used to evaluate alternative inventory strategies.

B. SENSITIVITY TEST 1: INCREASED TOUR DURATION

The first sensitivity test is designed to evaluate how sensitive the system is to changes in equipment retention behavior. This change reflects a realistic possibility: if a longer historical data window were available, average tour lengths might be higher than what the current three-year dataset shows. The dataset only captures the arrivals and departures of Marines who entered at the start of the window, and some Marines either extend at their unit or execute permanent changes of assignment (PCA) within the same MEF, both of which would lengthen true retention times.

1. Test 1 Scenario

Figure 15 shows how the simulation was adjusted to increase the average Marine tour length by 50% (from 399 days to 598.5 days). Because tour length in the model is drawn from a log-normal distribution, the standard deviation was also scaled proportionally. This is a reasonable assumption for a sensitivity test: if the underlying processes causing longer retention—such as within-MEF PCAs or unit extensions—increase the average time a Marine keeps a helmet, it is likely that the variability around that average would increase in a similar proportion. After adjusting the distribution, the initial inventory was also increased to match the new steady-state requirement, while all other model logic (reorder rules, tracking, and data collection) remained unchanged.

```
class Marine(object):
    """Marine with ID, arrival day, required helmet size, and tour length.
    Tour length ~ LogNormal with target mean=399 and sd=253.
    """
    def __init__(self, env, marine_id, arrival_day, helmet_size):
        self.env = env
        self.marine_id = marine_id
        self.arrival_day = arrival_day
        self.helmet = None
        self.helmet_size = helmet_size
        # Draw from a log-normal parameterized to have approx mean=399 and
sd=253.
        # Increase average tour length by 50% and scale sd proportionally to
preserve relative spread
        target_mean = 399.0 * 1.5         # new mean ≈ 598.5 days
        target_sd   = 253.0 * 1.5         # new sd ≈ 379.5 days
        var_ratio = (target_sd ** 2) / (target_mean ** 2)
        sigma_ln = math.sqrt(max(1e-12, math.log(1.0 + var_ratio)))
        mu_ln = math.log(max(1e-12, target_mean ** 2 / math.sqrt(target_sd **
2 + target_mean ** 2)))
        tour = random.lognormvariate(mu_ln, sigma_ln)
```

This code implements the sensitivity test for increased tour duration. To replicate this test, use the Annex code shown in Figure 10, replacing the original tour-length target_mean and target_sd with the adjusted versions from this figure.

Figure 15. Simulation Adjustment for Increased Tour Length

2. Test 1 Results

For completed cycles within this test, tour durations averaged 512.7 days (median 452.1; 90th percentile 886.2). Increasing the mean retention period slowed the rate at which helmets cycle through the system; thus, longer runs would allow the final mean to converge on the new distribution. Nonetheless, the observed distribution aligns with the adjusted log-normal inputs and confirms that the DES correctly generates longer retention periods.

As expected, extending tour length raises the number of helmets simultaneously checked out. The curves maintain the same overall shape as the baseline because Marines arrive at the same rate, but the system reaches equilibrium later—around year 4, as shown in Figure 16, instead of year 3 in Figure 14. Once equilibrium is reached, inventories by size remain within stable ranges, but at higher steady-state levels than in the baseline simulation.

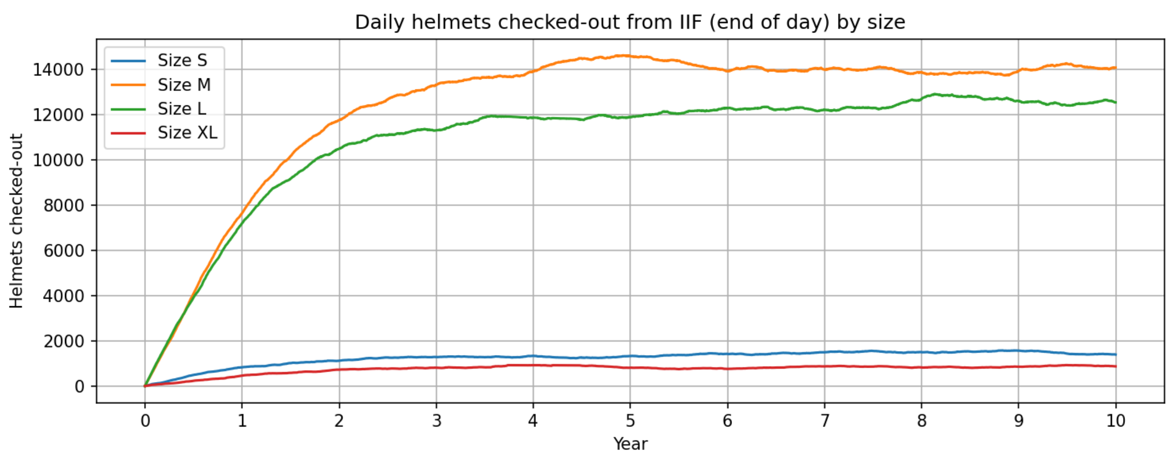

Figure 16. Daily Helmets by Size Checked-Out from IIF for Extended Tour Length Scenario

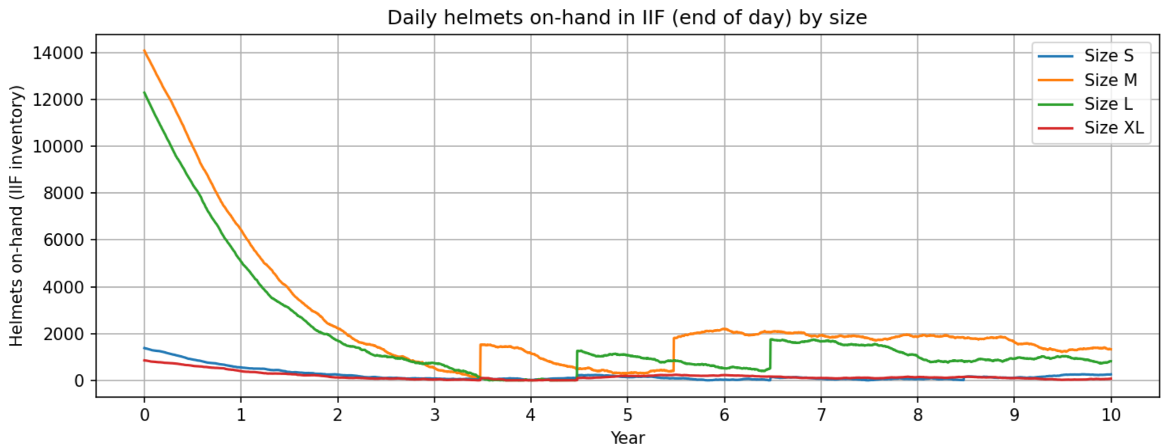

Figure 17. Daily Helmets by Size on Hand in IIF for Extended Tour Length Scenario

Reorder events increased only slightly, and the delivery of those orders appears as the inventory spikes in Figure 17. Under the existing reorder policy, which purchases 10% of the current stock whenever a size falls below its threshold, the model generated the orders shown in Table 5.

Table 5. Reorders During Extended Tour Length Scenario

sim day	year	size	qty
1095	3	M	1408
1095	3	XL	85
1460	4	S	137
1460	4	L	1228
1460	4	XL	85
1825	5	M	1408
2190	6	S	137
2190	6	L	1228
2920	8	S	137

Compared to the baseline, which had 7 total reorders, the change is moderate. The system only triggers orders slightly more often, mainly because higher usage brings inventory levels closer to the reorder point. Overall, this shows that longer tours do not significantly strain the current reorder policy as long as starting inventories are raised to match the new expected steady-state level.

The rate of size substitutions increased but remained operationally low. A total of 600 Marines were issued a non-requested size (0.57% of processed issues), compared to 223 events (0.21%) in the baseline. This increase is consistent with tighter availability resulting from longer retention but does not indicate systemic strain. Loss behavior also remained proportional. The model produced 2,973 permanently unreturned helmets (2.75% of all helmets processed), compared to 3,238 losses (3.01%) in the baseline. Similarly, 129 helmets were disposed after return, consistent with prior runs.

3. Test 1 Assessment

Increasing the average tour length did not destabilize the system but did raise the steady-state number of helmets checked out and slightly increased reorder frequency and substitution events. Overall, the results indicate that the system is robust, as it preserves its fundamental behavior even when key parameters—such as tour length—are substantially different from those observed in the MEF data. These results support the hypothesis that if Marine retention periods are longer than currently recorded—whether due to PCAs within a MEF or unit extensions—the Marine Corps may require either higher authorized inventory levels or a more adaptive reorder mechanism to maintain target availability. Additional longer-term data on actual retention behaviors would further improve calibration of this effect.

C. SENSITIVITY TEST 2: SEASONALITY

Although the available dataset does not show clear seasonality in Marine arrivals to the IIF, it is plausible that seasonal patterns would emerge in a larger or longer dataset. Fleet activity is not uniform across the year: certain months consistently have higher permanent change of station (PCS) movements, while others are comparatively quieter. Because these operational rhythms could translate into higher or lower equipment arrival rates, introducing seasonality into the simulation is a reasonable and informative sensitivity test.

1. Test 2 Scenario

To see how seasonal swings might affect inventory stability, this scenario adjusts the baseline arrival rate using month-specific multipliers—higher during typical PCS surge months and lower during quieter periods—while keeping the total number of yearly arrivals the same. All other aspects of the model remain unchanged.

The code block in Figure 18 introduces seasonal variation into the arrival rate for the sensitivity test. First, each month is assigned a relative weight that reflects how busy that month is expected to be. May through September are given the highest weights to represent the heaviest arrival periods, while January and December receive the next-highest weights. The remaining months are assigned lower weights to represent slower periods. The values for each month are not arrival rates—they simply indicate which months should be higher or lower relative to others.

```
def continuous_marine_arrivals(env, iif, arrival_rate=ARRIVAL_RATE_PER_DAY):

    # relative (unnormalized) weights for months 1..12
    rel_weights = {
        1: 1.3,  # Jan (next-heaviest)
        2: 0.8,
        3: 0.8,
        4: 0.9,
        5: 1.6,  # May-Sep (heaviest)
        6: 1.6,
        7: 1.6,
        8: 1.6,
        9: 1.6,
        10: 0.9,
        11: 0.8,
        12: 1.3   # Dec (next-heaviest)
    }
    # normalize so the average multiplier across 12 months = 1.0
    avg_weight = sum(rel_weights.values()) / 12.0
```

This code implements the sensitivity test for time-varying Poisson arrivals with monthly seasonality. To replicate this sensitivity test, replace the arrival-rate calculation function shown in Figure 8 with this seasonally adjusted version in this figure.

Figure 18. Simulation Insert for Seasonal Arrival-Rate Logic

The specific monthly patterns were chosen based on general observations of typical summer and winter PCS cycles in Marine Corps operations, but they function mainly as illustrative placeholders to show how the model reacts to seasonal variation. Next, these monthly weights are normalized so that their average equals 1.0. This ensures that the total number of arrivals over the year stays the same as in the baseline simulation. Only the timing of arrivals changes, not the overall volume.

2. Test 2 Results

Figures 19 and 20 show the system response when monthly arrival rates are adjusted by seasonal weights while total annual arrivals remain constant. Unlike Test 1, which modified retention behavior, this test adjusts only the timing of demand. As expected, the most visible effect of seasonality is the appearance of cyclical fluctuations in both helmets checked out and on-hand inventory levels, even once the system reaches steady state. There is a clear periodic pattern in the number of helmets checked out. Inventory in the IIF shows a corresponding pattern, with dips occurring during months of elevated demand. After the initial warm-up period ends (the end of year 3), inventory oscillates annually rather than settling into a relatively stable band as in the baseline test.

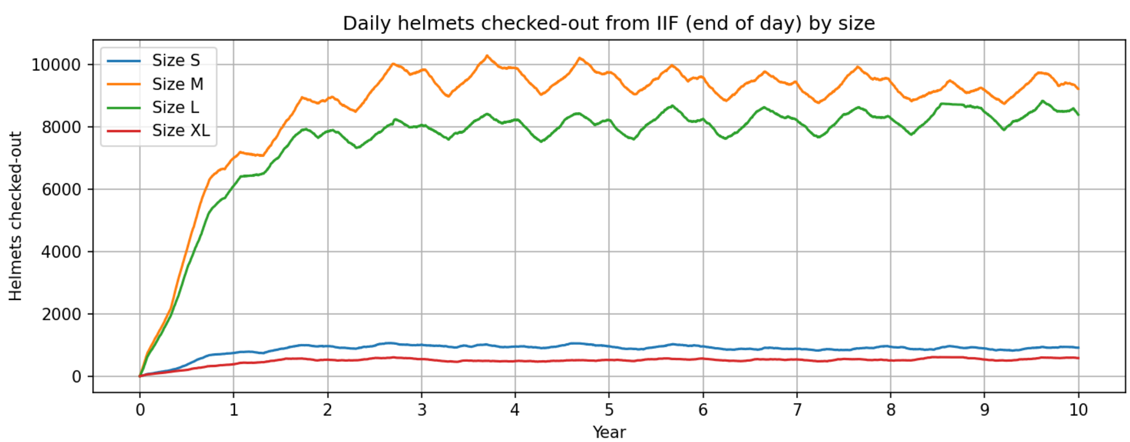

Figure 19. Daily Helmets by Size Checked-Out from IIF for Seasonality Scenario

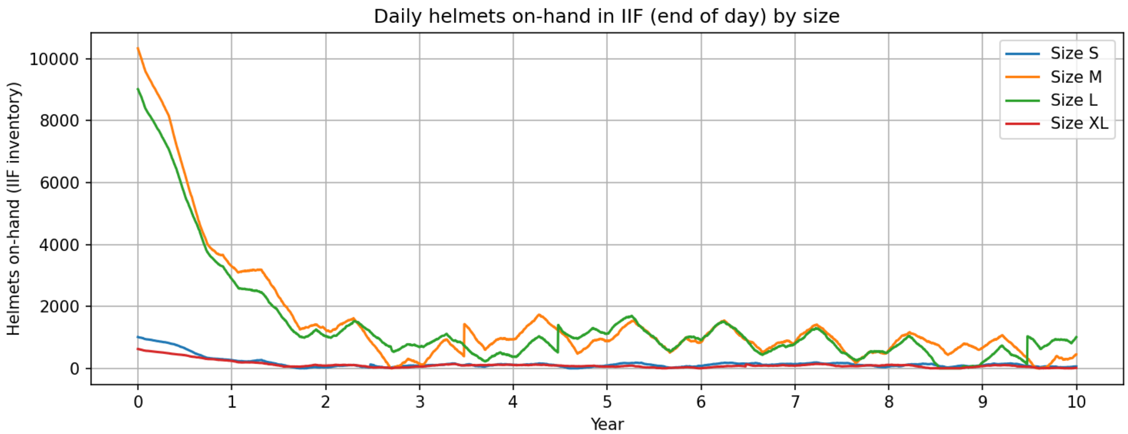

Figure 20. Daily Helmets by Size on Hand in IIF for Seasonality Scenario

Seasonal arrivals temporarily push inventories closer to reorder points even though total annual demand does not change. However, despite December's elevated arrival weight, overall reorder activity actually decreased relative to the baseline (baseline: 7 orders; test 2: 5 orders). The specific reorder events are shown in Table 6.

Table 6. Reorders During Seasonality Scenario

sim day	year	size	qty
730	2	S	101
1095	3	M	1033
1460	4	L	901
2190	6	XL	62
3285	9	L	901

More notably, the test shows an increase in substitution behavior. The model generated 632 wrong-size issues (0.61% of all processed Marines), compared to 223 events (0.21%) in the baseline. While still operationally low, this increase may suggest that seasonal fluctuations concentrate demand in ways that periodically narrow size availability even when annual totals remain stable. Loss behavior, by contrast, remained effectively unchanged. The simulation produced 3,199 permanently unreturned helmets (2.99%) and

163 disposals, nearly identical to baseline proportions. As expected, seasonal peaks affect availability more than they affect long-term loss or disposal rates.

3. Test 2 Assessment

Introducing seasonal arrival patterns did not destabilize the system but did create predictable swings in inventory levels, with some sizes pushed closer to their reorder points during high-demand months. Overall reorder activity decreased compared to the baseline, indicating that seasonality changes the timing of inventory pressure rather than the total amount. However, substitution events increased from 0.21% to 0.61%, suggesting that concentrated arrival surges temporarily tighten size availability even when annual demand is unchanged. If future data confirm stronger seasonal effects—or if operational patterns suggest that seasonality should be expected—then maintaining higher inventory levels or adjusting reorder thresholds may be necessary to preserve target availability during peak months.

VI. CONCLUSION

This thesis set out to determine whether a discrete-event simulation (DES) built on real Individual Issue Facility (IIF) data could accurately represent size-specific helmet inventory dynamics and provide a practical tool for evaluating alternative policies. The results across the baseline and sensitivity tests in Chapter V show that the model successfully captures the core behaviors of the system—arrival patterns, retention, size distribution, substitutions, losses, and reorder timing—and responds logically when key assumptions change. The following sections conclude this project by summarizing the main insights gained from the model, explaining how the simulation could be used to support inventory planning and decision-making, and outlining opportunities for future improvement.

A. MODEL INSIGHTS

The simulation results, together with the underlying issue data used to derive the model inputs, provide several key insights into how the IIF system behaves. First, the data shows that although there are modest differences between observed size demand and the current tariff factors, both reflect the same overall pattern dominated by medium and large sizes. This suggests that systemic availability issues for certain sizes are unlikely to stem from major errors in tariff sizing guidance. Even so, maintaining accurate and regularly updated tariffs is critical for long-term planning, as well as for supporting any future shifts in Marine demographics.

Second, the model shows the value of using DES to incorporate operational nuances that are difficult to implement in traditional spreadsheet approaches, such as substitutions, non-returns, disposal after return, and delayed reorder delivery. Representing these mechanisms explicitly allows the model to reflect the real conditions Marines encounter and makes the framework highly useful for testing a wide range of system behaviors.

Third, the baseline test results demonstrate that DES is an effective tool for representing IIF operations. The model reaches a realistic steady state, produces size-specific issues within expected ranges, and replicates reorder timing, substitution behavior, and loss rates that align with both available data and practical expectations. This confirms that DES is a suitable method for analyzing size-specific equipment supply systems and can reliably capture the dynamics of a MEF-scale IIF.

Finally, the sensitivity tests illustrate how adjustments to core assumptions—such as longer tour durations or seasonal arrival patterns—affect inventory stability. These tests show that the model responds logically, though not drastically, to changes in demand and retention behavior, making it useful for evaluating policy changes under different conditions. Together, these insights indicate that the model is both credible and operationally relevant for analyzing and improving IIF inventory policies.

B. HOW THE MODEL SHOULD BE USED

A key consideration in applying this model is recognizing how severely long DLA/FedMall procurement lead times constrain responsiveness. Because helmets and other size-specific equipment compete with broader DoD supply priorities, IIFs cannot rely on frequent reorders to maintain availability. In practice, units are more likely to receive replenishment only once per year, or at best semi-annually. The model highlights that reorder timing—not tariff accuracy—is the primary factor shaping size stability within these constraints. This makes it essential for IIFs to enter each year with enough inventory to sustain long periods without resupply. Within this context, the model is most useful in the following ways:

1. Developing and evaluating reorder policies

2. Decision-support tool for planners

The model provides a structured way to test alternative reorder thresholds, order quantities, and timing strategies that account for long procurement delays. Rather than relying on experience-based assumptions or ad hoc practices, the DES gives commands a data-driven basis for setting reorder points that prevent shortages across extended lead time windows. The model also enables decision-makers to explore "what-if" scenarios before adjusting inventory policies. By manipulating arrival patterns, retention times, loss rates, or size distributions, planners can preview how the system reacts under conditions where resupply may not arrive for an entire year. This capability is especially important for sizes with narrow buffers or historically tight availability.

C. RECOMMENDATIONS FOR FUTURE IMPROVEMENT

While the current model provides a realistic and flexible representation of IIF helmet inventory dynamics, several enhancements would improve its accuracy, usefulness, and applicability to broader Marine Corps logistics planning for other size-specific equipment.

1. Incorporate Multiple IIF Facilities

The current model treats the MEF as a single consolidated IIF. However, in reality, equipment is distributed across multiple facilities that support different units. Future versions could simulate multiple IIFs operating parallel to each other within the same system. Each IIF would use its own demand patterns, stock levels, and reorder histories. This would allow planners to examine imbalances between sites and whether redistribution strategies could improve availability—especially given long lead times for ordering new equipment.

2. Expand Usage Data

The model's accuracy depends heavily on the quality of IIF arrival, issue, and return data. Better visibility into daily issue patterns and demographics would allow for more accurate calibration and more precise forecasting. The dataset used for this study covered only a three-year window, which limits visibility into longer-term trends such as multi-year retention patterns, demographic shifts, or changes in operational tempo. Any improvement in reporting would immediately enhance the model by reducing the number of assumptions required and replacing them with directly observed parameters.

Additionally, the current model does not incorporate data from daily on-hand inventory counts, which would allow more precise validation of size availability. The model would also benefit from information on historical ordering patterns, which would provide insight into how often IIFs actually reorder and how long replenishment typically takes, instead of only relying on the published lead time in FedMall. These improvements would collectively strengthen the model's ability to represent real IIF operations and support future policy analysis.

3. Broader Applications

The framework for the IIF helmet inventory model can be adapted to other size-specific equipment such as plate carriers, SAPI inserts, cold-weather gear, or even boots. The logic behind size demand distributions, retention patterns, substitutions, losses, and procurement lead times applies across these systems, only requiring adjustments to item-specific data. The approach can also be scaled to represent the needs of different sized units, such as a Marine Expeditionary Unit, a Marine Littoral Regiment, or any other Marine Air-Ground Task Force.

This scalability makes the model useful not only for routine IIF management but also for planning in contingency, surge, or distributed operations where size-specific constraints become more significant. During surge periods—such as rapid mobilization, large-scale exercises, or unplanned contingency deployments—the normal demand rate can multiply within days, and units may have to rely on whatever inventory is on hand for an extended time. The model would allow planners to stress-test the system under elevated arrival rates and evaluate how long available stock can sustain demand if replenishment is delayed or unavailable. This helps decision makers anticipate when shortages will emerge, determine the minimum starting inventory required to cover long gaps in resupply, and understand the trade-offs when storage space or lift capacity limits how much inventory can be deployed. By adjusting arrival patterns, operational tempos, or loss rates, planners can use the model to determine equipment requirements and maintain size availability across a wide range of operational conditions.

APPENDIX. IIF SIMPY SIMULATION PYTHON CODE

```
# -*- coding: utf-8 -*-
""""
Created on August 15, 2025
Last updated: October 29, 2025
@author: Leah S. Weingarten, utilized GitHub Copilot

This simulation models Marines checking out helmets from a Individual Issue Facility (IIF).
""""

import simply
import random
import math
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# ------------------------------
# Monthly size-demand model (empirical mean ±1σ, 1% floor, normalized)
# ------------------------------
SIZE_STATS = {
"S": (0.0482, 0.0175),
"M": (0.4924, 0.0657),
"L": (0.4296, 0.0604),
"XL": (0.0298, 0.0123)
}

def get_monthly_size_demand():
"""Generate monthly size proportions within ±1σ (1% minimum), then normalize to sum to 1."""
shares = {
s: np.clip(np.random.normal(mu, sd), max(0.01, mu – sd), mu + sd)
for s, (mu, sd) in SIZE_STATS.items()
}
total = sum(shares.values())
return {s: v / total for s, v in shares.items()}


# ----------------------------
# Entities
```

```
# ----------------------------
class Helmet(object):
“““IIF helmet with ID, size, checkout status, and condition.”““
def __init__(self, helmet_id, size):
self.helmet_id = helmet_id
self.size = size
self.checked_out = False
self.current_marine = None
self.condition = “good”

class Marine(object):
“““Marine with ID, arrival day, required helmet size, and tour length.
Tour length ~ LogNormal with target mean=399 and sd=253.
“““
def __init__(self, env, marine_id, arrival_day, helmet_size):
self.env = env
self.marine_id = marine_id
self.arrival_day = arrival_day
self.helmet = None
self.helmet_size = helmet_size
# Draw from a log-normal parameterized to have approx mean=399 and sd= 253.
target_mean = 399.0
target_sd = 253.0
var_ratio = (target_sd ** 2) / (target_mean ** 2)
sigma_ln = math.sqrt(max(1e-12, math.log(1.0 + var_ratio)))
mu_ln = math.log(max(1e-12, target_mean ** 2 / math.sqrt(target_sd ** 2 + target_mean ** 2)))
tour = random.lognormvariate(mu_ln, sigma_ln)
self.tour_length = float(tour)

class IIF(object):
“““
Individual Issue Facility (IIF): per-size stores, activity log, size-demand mix (monthly),
and counters.
“““
def __init__(self, env, num_helmets_by_size):
self.env = env
self.processing_time = 30 # minutes (checkout/checkin processing)

# Registry of all helmets ever created (by size) for accurate tracking
```

```
self.all_helmets = {s: [] for s in num_helmets_by_size} # <-- MOVE THIS UP HERE

# Per-size inventory stores
self.helmet_store = {
"S": simpy.Store(env),
"M": simpy.Store(env),
"L": simpy.Store(env),
"XL": simpy.Store(env)
}

# Activity log (post-warmup only)
self.activity_log = pd.DataFrame(columns=[
'marine_id', 'arrival_date', 'helmet_id', 'helmet_size',
'helmet_checkout_date', 'tour_end_date', 'helmet_checkin_date'
])

self.wrong_size_issued = 0
self.lost_helmets = 0

# Initialize inventory
for size, num in num_helmets_by_size.items():
for i in range(num):
helmet = Helmet(f"{size}-H-{i+1:03d}," size)
self.helmet_store [size].put(helmet)
self.all_helmets [size].append(helmet) # <-- Now this works!

# simple sequence counters to generate new helmet IDs when reordering
self.helmet_seq_counters = {s: num for s, num in num_helmets_by_size.items()}

# Reorder policy defaults (tweak as needed)
# By default: reorder when inventory <= 5% of initial, order 10% of initial, 174-day lead time
self.reorder_point = {s: max(1, int(0.05 * n)) for s, n in num_helmets_by_size.items()}
self.reorder_qty = {s: max(1, int(0.10 * n)) for s, n in num_helmets_by_size.items()}
self.lead_time_days = 174 # increased lead time to 174 days
self._pending_orders = set() # sizes with outstanding orders

# Reorder tracking / reporting
self.reorder_count_by_size = {s: 0 for s in num_helmets_by_size}
```

```
self.total_reorders = 0
self.reorder_history = [] # list of (day, size, qty)

# Initialize current monthly size demand (will be refreshed by a process)
self.size_demands = get_monthly_size_demand()
# Checked-out tracking and daily snapshots
self.checked_out_count = 0
self.daily_checked_out = []
self.daily_on_hand = []
self.daily_days = []
self.daily_on_hand_by_size = {s: [] for s in num_helmets_by_size}
self.daily_checked_out_by_size = {s: [] for s in num_helmets_by_size}

def add_marine_record(self, marine_id, arrival_date,
helmet_id=None, helmet_checkout_date=None,
tour_end_date=None, helmet_checkin_date=None):
""""
Create or update a Marine's record. Only records events after warm-up.
""""
if self.env.now < WARMUP_TIME:
Return

existing = self.activity_log [self.activity_log ['marine_id'] == marine_id]
if len(existing) > 0:
idx = existing.index [0]
if helmet_id is not None:
self.activity_log.at [idx, 'helmet_id'] = helmet_id
if helmet_checkout_date is not None:
self.activity_log.at [idx, 'helmet_checkout_date'] = helmet_checkout_date
if tour_end_date is not None:
self.activity_log.at [idx, 'tour_end_date'] = tour_end_date
if helmet_checkin_date is not None:
self.activity_log.at [idx, 'helmet_checkin_date'] = helmet_checkin_date
else:
new_record = pd.DataFrame({
'marine_id': [marine_id],
'arrival_date': [arrival_date],
'helmet_id': [helmet_id],
'helmet_size': [None], # filled on checkout
'helmet_checkout_date': [helmet_checkout_date],
'tour_end_date': [tour_end_date],
'helmet_checkin_date': [helmet_checkin_date]
})
```

```
self.activity_log = pd.concat([self.activity_log, new_record], ignore_index=True)

def place_reorder(self, size):
“““Place a reorder for the given size (schedules receipt after lead time).”““
if size in self._pending_orders:
Return
qty = self.reorder_qty.get(size, 0)
if qty <= 0:
Return
self._pending_orders.add(size)
# track reorder event
self.reorder_count_by_size [size] += 1
self.total_reorders += 1
self.reorder_history.append((self.env.now, size, qty))
print(f”Day {self.env.now:.0f}: Placing reorder for {qty} helmets size {size} (lead {self.lead_time_days} days)”)
# schedule receipt
self.env.process(self._receive_reorder(size, qty))

def _receive_reorder(self, size, qty):
“““SimPy process: wait lead time then add helmets to inventory.”““
yield self.env.timeout(self.lead_time_days)
for _ in range(qty):
self.helmet_seq_counters [size] += 1
hid = f”{size}-H-{self.helmet_seq_counters [size]:06d}”
helmet = Helmet(hid, size)
self.helmet_store [size].put(helmet)
self.all_helmets [size].append(helmet) # <-- Add this line!
self._pending_orders.discard(size)
print(f”Day {self.env.now:.0f}: Received reorder {qty} helmets size {size}; new count = {len(self.helmet_store [size].items)}”)

def inventory_reorder_manager(self, check_interval=1):
“““Process that checks inventory daily and places reorders when counts drop to reorder_point.”““
while True:
for size, store in self.helmet_store.items():
current = len(store.items)
if current <= self.reorder_point.get(size, 0) and size not in self._pending_orders:
self.place_reorder(size)
yield self.env.timeout(check_interval)

def daily_snapshot_manager(self, check_interval=1):
```

```
“““Record daily snapshots of checked-out and on-hand helmets (at day boundaries).”““
while True:
self.daily_days.append(self.env.now)
self.daily_checked_out.append(self.checked_out_count)
self.daily_on_hand.append(sum(len(s.items) for s in self.helmet_store.values()))
# Record by-size on hand and checked out
for size in self.helmet_store:
# On hand: helmets in store for that size
on_hand = len(self.helmet_store [size].items)
self.daily_on_hand_by_size [size].append(on_hand)
# Checked out: helmets that are checked out and not lost/disposed
checked_out = sum(
1 for helmet in self.all_helmets [size]
if helmet.checked_out and helmet.condition not in (“dispose”,) and helmet.current_marine is not None
)
self.daily_checked_out_by_size [size].append(checked_out)
yield self.env.timeout(check_interval)
# ------------------------------
# Processes
# ------------------------------
def marine_life cycle(env, marine, iif):
“““Marine arrives, checks out a helmet (prefers requested size), serves tour, returns helmet.”““
# Process is started by the arrival generator at the correct time,
# so do not wait again here. Record arrival immediately.
print(f”Day {env.now:.0f}: Marine {marine.marine_id} arrives at command”)
iif.add_marine_record(marine.marine_id, env.now)
print(f”Day {env.now:.0f}: Marine {marine.marine_id} goes to IIF for helmet (size {marine.helmet_size})”)
sizes_order = [“S,” “M,” “L,” “XL”]
wanted_index = sizes_order.index(marine.helmet_size)
# Preference order: requested size then larger sizes (XL tries downward)
try_sizes = sizes_order [::-1] if marine.helmet_size == “XL” else sizes_order [wanted_index:]
# Attempt immediate issue by preference
```

```
helmet = None
issued_size = None
for size in try_sizes:
if len(iif.helmet_store [size].items) > 0:
helmet = yield iif.helmet_store [size].get()
issued_size = size
print(f"Day {env.now:.0f}: Marine {marine.marine_id} got helmet size {size}")
Break

# If none available, wait until any preferred size returns
if helmet is None:
print(f"Day {env.now:.0f}: Marine {marine.marine_id} waiting for any helmet in preference order")
while helmet is None:
for size in try_sizes:
if len(iif.helmet_store [size].items) > 0:
helmet = yield iif.helmet_store [size].get()
issued_size = size
print(f"Day {env.now:.0f}: Marine {marine.marine_id} finally got helmet size {size}")
break
if helmet is None:
yield env.timeout(1) # check daily

# Bind helmet to Marine
marine.helmet = helmet
helmet.checked_out = True
helmet.current_marine = marine.marine_id
# increment global checked-out counter
iif.checked_out_count += 1

# Count wrong-size issue
if issued_size and issued_size != marine.helmet_size:
iif.wrong_size_issued += 1

# Processing time (30 min)
yield env.timeout(iif.processing_time / (24 * 60))

print(f"Day {env.now:.0f}: Marine {marine.marine_id} checked out helmet {helmet.helmet_id}")
# Record checkout and the size (add size value at checkout time)
iif.add_marine_record(marine.marine_id, None, helmet.helmet_id, env.now)
# Also store size into the record's 'helmet_size' column if present
```

```
if iif.env.now >= WARMUP_TIME:
idx = iif.activity_log [iif.activity_log ['marine_id'] == marine.marine_id].index
if len(idx) > 0:
iif.activity_log.at [idx [0], 'helmet_size'] = marine.helmet_size

# Serve tour
print(f"Day {env.now:.0f}: Marine {marine.marine_id} begins tour
({marine.tour_length:.0f} days)")
yield env.timeout(marine.tour_length)

# Tour complete → return helmet
print(f"Day {env.now:.0f}: Marine {marine.marine_id} tour complete, returning
helmet {helmet.helmet_id}")

# Record tour end date
iif.add_marine_record(marine.marine_id, None, None, None, env.now)

# Processing time to return
helmet.checked_out = False
helmet.current_marine = None
marine.helmet = None
yield env.timeout(iif.processing_time / (24 * 60))
# helmet no longer with Marine (returned to IIF processing / will be returned or
lost) -> decrement counter
iif.checked_out_count = max(0, iif.checked_out_count – 1)

# First: chance the Marine never returns the helmet at all (permanent loss)
p_not_return = 0.021 # 2.1% chance the helmet is never returned
if random.random() < p_not_return:
# Helmet permanently lost / removed from the system
iif.lost_helmets += 1
print(f"Day {env.now:.0f}: Helmet {helmet.helmet_id} PERMANENTLY LOST /
NOT RETURNED by Marine {marine.marine_id}")
# Record: tour end already recorded; mark helmet as permanently lost and
record a "checkin" timestamp
# so the Marine is not counted as still-serving (helmet is gone).
iif.add_marine_record(marine.marine_id, None, None, None, None, env.now)
# Do NOT put helmet back into inventory.
else:
# Helmet is returned — sample condition conditional on return (empirical)
condition = random.choices(
["good," "repair," "dispose"],
weights=[0.997, 0.002, 0.001],
```

```
k=1
)[0]
helmet.condition = condition

if helmet.condition == “good”:
iif.helmet_store [helmet.size].put(helmet)
iif.add_marine_record(marine.marine_id, None, None, None, None, env.now)
print(f”Day {env.now:.0f}: Helmet {helmet.helmet_id} returned in GOOD condition”)
elif helmet.condition == “repair”:
print(f”Day {env.now:.0f}: Helmet {helmet.helmet_id} needs REPAIR (≈30 days)”)
yield env.timeout(30)
helmet.condition = “good”
iif.helmet_store [helmet.size].put(helmet)
iif.add_marine_record(marine.marine_id, None, None, None, None, env.now)
print(f”Day {env.now:.0f}: Helmet {helmet.helmet_id} repaired and back in inventory”)
else: # dispose (returned but condemned / removed)
print(f”Day {env.now:.0f}: Helmet {helmet.helmet_id} RETURNED BUT CONDEMNED/DISPOSED (not returned to inventory)”)
iif.add_marine_record(marine.marine_id, None, None, None, None, env.now)

print(f”Day {env.now:.0f}: Marine {marine.marine_id} helmet processed, departing command”)


# Arrival rate (Poisson process) in arrivals per day
ARRIVAL_RATE_PER_DAY = 47.8

def continuous_marine_arrivals(env, iif, arrival_rate= ARRIVAL_RATE_PER_DAY):
“““
Poisson arrivals: interarrival times ~ Exponential(rate=arrival_rate per day).
Each arriving Marine requests a size drawn from the current monthly size distribution.
“““
marine_count = 0
while True:
# exponential interarrival with rate lambda (arrivals per day)
next_interarrival = random.expovariate(arrival_rate)
yield env.timeout(next_interarrival)
```

```
marine_count += 1
marine_id = f"M-{marine_count:05d}"

# draw size from current monthly demand mix maintained by IIF
sizes = list(iif.size_demands.keys())
weights = list(iif.size_demands.values())
requested_size = random.choices(population=sizes, weights=weights, k=1)[0]

marine = Marine(env, marine_id, env.now, requested_size)
env.process(marine_life cycle(env, marine, iif))


def monitor_iif_status(env, iif, check_interval=30):
"""Periodic inventory snapshot by size (≈ monthly)."""
while True:
available = {size: len(store.items) for size, store in iif.helmet_store.items()}
print("Day {:0.0f}: IIF Status – {}".format(
env.now, ," ".join([f"{s}: {c}" for s, c in available.items()])
))
yield env.timeout(check_interval)


def monthly_size_demand_updater(env, iif, month_len_days=30):
"""Refresh the size mix once per simulated month."""
while True:
yield env.timeout(month_len_days)
iif.size_demands = get_monthly_size_demand()
print(f"Day {env.now:.0f}: Updated monthly size mix -> {iif.size_demands}")


# ---------------------------
# Simulation Parameters
# ---------------------------
NUM_MARINES = 300 # Informational; arrivals are time-based and unbounded
NUM_HELMETS_BY_SIZE = {
"S": 1012,
"M": 10337,
"L": 9019,
"XL": 625
}
SIMULATION_TIME = 3650 # Run for ~10 years (10 * 365 days)
WARMUP_TIME = 1460 # Ignore first ~4 years for statistics (in days)
```

```
# ----------------------------
# Run
# ----------------------------
print(“=== Marine IIF Helmet Checkout Simulation ===“)
print(f”Simulating up to {SIMULATION_TIME} days (~10 years)”)
print(“Initial IIF inventory: “ + ,” “.join([f”{v} {k}” for k, v in NUM_HELMETS_BY_SIZE.items()]))
print(“=“ * 50)

env = simpy.Environment()
iif = IIF(env, NUM_HELMETS_BY_SIZE)

# Start processes
env.process(continuous_marine_arrivals(env, iif))
env.process(monitor_iif_status(env, iif))
env.process(monthly_size_demand_updater(env, iif))
env.process(iif.inventory_reorder_manager(check_interval=365)) # <-- start reorder manager (once-yearly checks ≈ 365 days)
env.process(iif.daily_snapshot_manager()) # <-- start daily snapshot recorder

# Go!
env.run(until=SIMULATION_TIME)

# ----------------------------
# Final Report
# ----------------------------
print(“=“ * 50)
print(“=== Simulation Complete ===“)
final_helmets = {size: len(store.items) for size, store in iif.helmet_store.items()}
print(“Final IIF Status: “ + ,” “.join([f”{size}: {count}” for size, count in final_helmets.items()]))

# Save activity log (post-warmup only)
activity_log_after_warmup = iif.activity_log [iif.activity_log [‘arrival_date’] >= WARMUP_TIME]
output_filename = “marine_iif_activity_log.csv”
activity_log_after_warmup.to_csv(output_filename, index=False)
print(f”Activity log saved to: {output_filename}”)

# ==============================
# SUMMARY STATISTICS
# ==============================
print(“\n” + “=“*50)
```

```
print("=== SUMMARY STATISTICS ===")

# Basic counts
total_processed = len(activity_log_after_warmup)
completed_tours =
activity_log_after_warmup.dropna(subset=['helmet_checkin_date'])
still_serving = activity_log_after_warmup [activity_log_after_warmup
['helmet_checkin_date'].isna()]
returned_count = len(completed_tours)
still_count = len(still_serving)

print(f"Total Marines processed: {total_processed}")
print(f"Marines who completed tours: {returned_count} ({(returned_count/
total_processed*100 if total_processed else 0):.2f}%)")
print(f"Marines still serving: {still_count} ({(still_count/total_processed*100 if
total_processed else 0):.2f}%)")

# Size distribution at checkout
if total_processed > 0:
size_counts = activity_log_after_warmup ['helmet_size'].value_counts(dropna=
True)
size_perc = (size_counts / total_processed * 100).round(2)
print("\nSize distribution (checked out) [count (percent)]:")
for size, cnt in size_counts.items():
print(f" {size}: {cnt} ({size_perc [size]:.2f}%)")
unknown_size = activity_log_after_warmup ['helmet_size'].isna().sum()
if unknown_size:
print(f" Unknown size at checkout: {unknown_size} ({unknown_size/
total_processed*100:.2f}%)")
else:
print("\nNo post-warmup activity to summarize sizes.")

# Wrong-size issuance
pct_wrong_size = (iif.wrong_size_issued / total_processed * 100) if
total_processed else 0.0
print(f"\nMarines issued the wrong size helmet: {iif.wrong_size_issued}
({pct_wrong_size:.2f}% of processed)")

# Helmets lost and disposed
lost_abs = iif.lost_helmets
den = total_processed + lost_abs
pct_lost = (lost_abs / den * 100) if den else 0.0
```

```
print(f”Helmets permanently lost / not returned: {lost_abs} ({pct_lost:.2f}% of processed + lost)”)

disposed_count = sum(
1 for size in iif.all_helmets
for helmet in iif.all_helmets [size]
if helmet.condition == “dispose”
)
print(f”Helmets disposed after return: {disposed_count}”)

# Empirical tour durations (where checkout and tour end exist)
durations = (activity_log_after_warmup [‘tour_end_date’] – activity_log_after_warmup [‘helmet_checkout_date’]).dropna()
if len(durations) > 0:
mean_dur = durations.mean()
median_dur = durations.median()
p90 = durations.quantile(0.9)
print(“\nEmpirical tour durations (days) for completed tours:”)
print(f” Mean: {mean_dur:.1f}, Median: {median_dur:.1f}, 90th pct: {p90:.1f}”)
else:
print(“\nNo completed-tour duration data available (checkout or tour_end missing).”)

# Final inventory snapshot
print(“\nFinal IIF inventory (by size): “ + ,” “.join([f”{size}: {count}” for size, count in final_helmets.items()]))

# Reorder statistics
print(“\n=== Reorder Statistics ===“)
print(f”Total reorders placed: {iif.total_reorders}”)
for size, cnt in iif.reorder_count_by_size.items():
print(f” {size}: {cnt} orders”)
if iif.reorder_history:
first = iif.reorder_history [0]
last = iif.reorder_history [-1]
print(f”First reorder: Day {first [0]:.0f}, size {first [1]}, qty {first [2]}”)
print(f”Most recent reorder: Day {last [0]:.0f}, size {last [1]}, qty {last [2]}”)
else:
print(“No reorders placed during simulation.”)

# Reorder events table (all reorders placed during the simulation)
if iif.reorder_history:
```

```
reorder_df = pd.DataFrame(iif.reorder_history, columns=['day', 'size', 'qty']).sort_values('day').reset_index(drop=True)
# Save CSV
reorder_csv = "iif_reorder_history.csv"
reorder_df.to_csv(reorder_csv, index=False)
# Try to save Excel (openpyxl may be required)
reorder_xlsx = "iif_reorder_history.xlsx"
try:
with pd.ExcelWriter(reorder_xlsx, engine="openpyxl") as writer:
reorder_df.to_excel(writer, index=False)
print(f"Saved reorder history to: {reorder_csv} and {reorder_xlsx}")
except Exception:
print(f"Saved reorder history to: {reorder_csv} (Excel not written: openpyxl may be missing)")

# Print a compact table to the console
print("\nReorder events (day, size, qty):")
print(reorder_df.to_string(index=False))
else:
print("No reorder events to export.")

# ------------------------------
# Plots and Time Series CSVs (by size)
# ------------------------------
try:
days = list(iif.daily_days)
size_labels = ["S," "M," "L," "XL"]
if len(days) > 0:
# Checked-out plot: one line per size
plt.figure(figsize=(10,4))
for size in size_labels:
plt.plot(days, iif.daily_checked_out_by_size [size], label=f"Size {size}")
plt.xlabel("Year")
plt.ylabel("Helmets checked-out")
plt.title("Daily helmets checked-out from IIF (end of day) by size")
plt.grid(True)
years = int(days [-1] // 365) + 1
plt.xticks([y*365 for y in range(years+1)], [str(y) for y in range(years+1)])
plt.legend()
plt.tight_layout()
plt.savefig("checked_out_time_series_by_size.png," dpi=150)
print("Saved: checked_out_time_series_by_size.png")
```

```
# On-hand plot: one line per size
plt.figure(figsize=(10,4))
for size in size_labels:
plt.plot(days, iif.daily_on_hand_by_size [size], label=f"Size {size}")
plt.xlabel("Year")
plt.ylabel("Helmets on-hand (IIF inventory)")
plt.title("Daily helmets on-hand in IIF (end of day) by size")
plt.grid(True)
plt.xticks([y*365 for y in range(years+1)], [str(y) for y in range(years+1)])
plt.legend()
plt.tight_layout()
plt.savefig("on_hand_time_series_by_size.png," dpi=150)
print("Saved: on_hand_time_series_by_size.png")

# Save on-hand by size CSV
on_hand_df = pd.DataFrame({"day": days})
for size in size_labels:
on_hand_df [f"on_hand_{size}"] = iif.daily_on_hand_by_size [size]
on_hand_df.to_csv("on_hand_by_size_timeseries.csv," index=False)
print("Saved: on_hand_by_size_timeseries.csv")

# Save checked-out by size CSV
checked_out_df = pd.DataFrame({"day": days})
for size in size_labels:
checked_out_df [f"checked_out_{size}"] = iif.daily_checked_out_by_size [size]
checked_out_df.to_csv("checked_out_by_size_timeseries.csv," index=False)
print("Saved: checked_out_by_size_timeseries.csv")
else:
print("No daily snapshot data collected; no plots or time series CSVs
produced.")
except Exception as e:
print(f"Could not produce plots or time series CSVs: {e}")

print("\nSimulation finished!")

size_labels = ["S," "M," "L," "XL"]
checked_out_end = sum(iif.daily_checked_out_by_size [size][-1] for size in
size_labels)
print(f"\nSanity check: Marines still serving = {still_count}, Helmets checked out
at end = {checked_out_end}")
if still_count == checked_out_end:
print("PASS: Marines still serving matches helmets checked out at end.")
else:
```

```
print("WARNING: Mismatch! Investigate logic for lost helmets or tour completion.")
```

LIST OF REFERENCES

Defense Logistics Agency. (2021, December 16). *FedMall program information*. https://www.dla.mil/Working-With-DLA/Applications/Details/Article/2877007/fedmall/

Defense Logistics Agency. (2024, April 24). *Introduction of new clothing and textile (C&T) items into the Department of Defense (DoD) supply system (*Joint Service Regulation 4140.34).

Defense Logistics Agency. (2025a, October 10). *ELMS release 2025.4.1.* https://elmssupport.golearnportal.org/index.php/resource-center/release-center/3827-elms-release-2025-4-1

Defense Logistics Agency. (2025b). *The DoD APSR for materiel management.* https://elmssupport.golearnportal.org/

Defense Logistics Agency. (2025c). *FedMall Commerce*. https://www.restricted.fedmall.mil/commerce

Deibler, C. M. (2018). *Expeditionary logistics warehouse inventory optimization* [Master's thesis, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/60392

Ersoz, T. (2016). *A new approach to site demand-based level inventory optimization* [Master's thesis, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/49452

Gordon, C. C., Blackwell, C. L., Bradtmiller, B., Parham, J. L., Hotzman, J., Paquette, S. P., Corner, B. D., & Hodge, B. M. (2013). *2010 anthropometric survey of U.S. Marine Corps personnel: Methods and summary statistics* (Technical Report NATICK/TR-13/018). U.S. Army National Soldier Research, Development and Engineering Center. https://apps.dtic.mil/sti/tr/pdf/ADA581918.pdf

Guerrero, R., Serrano-Hernandez, A., Pascual, J., & Faulin, J. (2022). Simulation model for wire harness design in the car production line optimization using the SimPy library. *Sustainability, 14*(12*)*, 7212. https://doi.org/10.3390/su14127212

Innuphat, P., & Toahchoodee, M. (2022). The implementation of discrete-event simulation and demand forecasting using temporal fusion transformers to validate spare parts inventory policy for petrochemicals industry. ECTI Transactions on Computer and Information Technology, 16(3), 247–258. https://doi.org/10.37936/ecti-cit.2022163.246900

Kelton, W. D., & Law, A. M. (1991). *Simulation modeling and analysis* (2nd ed.). McGraw-Hill.

Kelton, W. D., Sadowski, R. P., & Swets, N. B. (2010). *Simulation with arena* (5th ed.). McGraw-Hill Higher Education.

Krishnamoorthy, K. (2016). *Handbook of statistical distributions with applications*. https://doi.org/10.1201/b19191

Mariappan, P. (2019). *Statistics for business.* https://doi.org/10.1201/9780429443244

Marine Corps Systems Command. (2024). *USMC anthropometry survey 2025 participant recruitment letter*. Quantico, VA: United States Marine Corps.

Robinette, K. M., Veitch, D., Alemany, S., & Bredenkamp, K. (2024). *Product fit and sizing: Sustainable product evaluation, engineering, and design.* CRC Press.

Team SimPy. (2024). *Shared resources*. Documentation for SimPy. https://simpy.readthedocs.io/en/latest/topical_guides/resources.html#store

United States Marine Corps. (2014, May). *Marine Corps stocklist: Special list for tariff sizes* (SL-8-09991A).

United States Marine Corps. (2016). *Management of property in the possession of the Marine Corps.* (MCO 4400.201-V13). https://www.marines.mil/Portals/1/Publications/MCO%204400.201%20Volume%2013%20Final.pdf

United States Marine Corps. (2023, April). *DPAS user manual: Volume I—Garrison property, non-tactical vehicles, other property & equipment.* https://www.marines.mil/Portals/1/Publications/MARINE%20CORPS%20DPAS%20USER%20MANUAL%20VOL%201%20APRIL%202023.pdf?ver=yrbZ-N3U4hxp2ruvgGh2Xw%3D%3D

Zinoviev, D. (2024). Discrete event simulation: It’s easy with SimPy. https://doi.org/10.48550/arxiv.2405.01562

Acquisition Research Program
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net