3-2022

Salvaging Value from the Measure of "Requirements Volatility" in the DoD's Software Resources Data Report (SRDR)

Robert T. Walker

SALVAGING VALUE FROM THE MEASURE OF "REQUIREMENTS VOLATILITY" IN THE DOD'S SOFTWARE RESOURCES DATA REPORT (SRDR)

THESIS

Robert T. Walker, 1st Lieutenant, USAF

AFIT-ENV-MS-22-M-270

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENV-MS-22-M-270

SALVAGING VALUE FROM THE MEASURE OF "REQUIREMENTS VOLATILITY" IN THE DOD'S SOFTWARE RESOURCES DATA REPORT (SRDR)

THESIS

Presented to the Faculty

Department of Systems Engineering and Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cost Analysis

Robert T. Walker, BS

1st Lieutenant, USAF

March 2022

AFIT-ENV-MS-22-M-270

SALVAGING VALUE FROM THE MEASURE OF "REQUIREMENTS VOLATILITY" IN THE DOD'S SOFTWARE RESOURCES DATA REPORT (SRDR)

Robert T. Walker, BS

1st Lieutenant, USAF

Committee Membership:

Lt Col Scott T. Drylie, PhD
Chair

Robert D. Fass, PhD
Member

Jonathan D. Ritschel, PhD

Member

AFIT-ENV-MS-22-M-270

Abstract

The DoD and the software development industry as a whole has long dwelt over the idea of requirements volatility (RV). The DoD has toiled with this concept so much so that its guidance was modified four times over a 13 year span. In these changes, its policy completely transformed the what, how, and when regarding RV information. As a result, the volatility data it has received is quite varied and seemingly useless for anything more than anecdotal analysis. This study takes several approaches to salvage value from this data. It begins with a survey of the uncertain concept of volatility, and provides an array of descriptive statistics to make clear what DoD currently has available for analysis. It then places volatility in its intended place as a mediator between problem characteristics and problem outcomes. It does this in two steps. First, it evaluates various volatility measurement schemes against an array of possible measures of growth where the impact of volatility may occur. This work is exploratory to determine which scheme may be most predictive and how so. Second, it identifies relationships between these various measures of volatility and the program attributes which program managers may be contemplating when they try to portray volatility. Both initial and final relationships are tested, capturing the ability of managers to assess volatility in any meaningful way at the start of the program and as a retrospective, or post-mortem. All tests are completed utilizing Contingency Analysis and the respective Odds Ratio to determine the strength of the relationships identified.

Acknowledgments

I would like to express my sincere appreciation to my faculty advisor, Lieutenant Colonel Scott Drylie, for his patience, experience, and guidance through this research process. The insight and keen since of intellectual rigor were both greatly appreciated.

Table of Contents

List of Figures

Page

List of Tables

SALVAGING VALUE FROM THE MEASURE OF "REQUIREMENTS VOLATILITY" IN THE DOD'S SOFTWARE RESOURCES DATA REPORT (SRDR)

I. Introduction

The Department of Defense (DoD) has long solicited measures of Requirements Volatility (RV) from its software development contractors. RV is understood as the likelihood that requirements will change, and it is perceived to be an important risk effecting cost and schedule, and for the DoD, operational readiness (Henry, 1993; Standish, 1995; Stark, 1998). A few studies have, indeed, found positive correlations between measures of RV and schedule overruns in the commercial sector (Nurmuliani, 2004; Singh,2012; Stark,1998; Zowghi, 2002). But we are aware of no published research on the DoD dataset, nor any internal DoD studies, which can comment on whether its own measures of RV can predict such effects and thus serve in a manner which could improve program management. The quality of the data has likely hampered efforts to date. The current study will employ exploratory techniques to mine the value of the current data, to identify the implicit concepts of volatility guiding contractor inputs, and to determine how the DoD might improve its guidance for the solicitation of RV measurement in the future.

The poor quality of DoD data in regard to volatility gives this paper its form and purpose. Despite the DoD requiring RV measures from its contractors, the extant data is limited. Moreover, across time and between contractors, the measurement schemes are inconsistent. These shortcomings are likely the result of both weak oversight and changing guidance, and they render a dataset that appears, on its surface, to have little value. What, if anything, can be learned from RV data? And should we care about RV at all? These are very rudimentary questions. But, in our case, the data neither allows nor provides clear answers. And so, we must start at the beginning. We examine the concept of RV; we attempt to decode our data with these concepts; and, in the exploration of empirical relationships, we try to clear up how contractors are conceiving of volatility. The work aims to establish clearer guidance and to contribute to the broader academic discussion of the variable of volatility.

The dynamic nature of the software environment means it is inevitable that changes will occur (Barry, 2002; Henry, 1993; Jones, 2004; Kulk, 2008; Nidumolu, 1996; Nurmuliani, 2004; Stark, 1998; Zowghi, 2002). Uncertainty is at the heart of a changing program. In software development, uncertainty is broadly defined in the commonsensical way as the absence of complete information (Nidumolu,1996). The inevitability of uncertainty makes the development and estimation of software difficult. Both DoD and commercial projects will experience great cost and schedule growth, sometimes to the point of program failure (Standish, 1995). Some of this cost growth is a result of poor estimating of uncertainty of task, cost or schedule, but much of it is due to changing requirements (Luketic, 2020). The term "requirements volatility" is frequently used in the software development community to describe changing requirements in the development cycle. Other terms seem to relate to it: changing requirements, requirements uncertainty, requirements creep, and requirements churn. The DoD, in its Software Resource Data Report (SRDR), tracks requirements volatility, while labeling it as merely "Volatility."

The 2008 Software Development Cost Estimating Handbook reveals the basic intuition of RVs mediating role between specific problems and general results. It describes RV, first, as “the cost penalty due to expected frequency and scope of requirements changes after the baseline Software Requirements Review;” and second, as something that “projects the impact of those changes on delivery cost and schedule” (p.10-6). Those are ambitious aims-- to capture the true impact that changing requirements will have on the overall performance of a project. But rigorous testing of its potential is minimal for the commercial sector, and non-existent within the DoD. How does RV relate to underlying changes? What kind of scale best represents these changes in such a way they may predict cost and schedule growth? Can a single global measure add value to other assessments of the program? The challenge for RV as a measure is manifold, and it should not be presumed that subject matter experts (SMEs) can capture the significance of change into a single measure, nor represent it in a scale that predicts program growth.

The first basic challenge of RV as a measure is that of capturing the significance of change. The challenge can be quite intuitive in the fact that not all changes are the same. Adding and subtracting requirements are likely to increase or decrease the amount of work respectively, but adjustments to requirements may have ambiguous impacts. Similarly, some changes are more interconnected to the whole system than others. And even the timing of changes may matter: Early changes may have smaller impacts than later ones (Zowghi, 2002). Any effort that weights all requirements equal will not be capable of accurately predicting growth through volatility. For the global assessment of RV to have any value for management–creating alarm, producing confidence, and overall advising on risk--management must be able to understand the relative importance of the various underlying changes (Jones, 2004; Kulk, 2008; Nidumolu, 1996; Zowghi, 2002).

The second basic challenge of RV as a measure is that of choosing a scale which is predictive of growth. Even if the ingredients/precursors and their weight for RV can be roughly formalized within an organization, a suitable scale must be selected. This scale must predict growth, but it must both be understandable among users if it is to be employed consistently, and intelligible to its recipients if it is to lead to action. Little work has been done to investigate the merits of various RV scales seen in industry.

The DoD has appeared to have dwelt on these issues and has made efforts to better understand and capture RV over time. Recently the DoD has begun requesting a calculated, numerical input instead of a categorical or qualitative rating (DoD 5000.4-M-2, 2004). The change was intended to reduce bias. A prescribed calculation allows the DoD an attempt to dictate what ingredients should drive RV inputs. They have, in other words, by asking for a more precise measure, conveyed a more technical standard.

Contractors clearly contemplate this issue as well. Regardless of guidance, contractors employ different schemes in their submissions. Three primary measurement scales have been employed throughout the years: an Adjective rating (ex. low, nominal, high), a numerical scale rating using whole numbers (1-5), and a percentage scheme. Table 1 shows the proportional usage of each in DoD's database. As can be seen, the data is messy. The schemes fluctuate in usage over time. Moreover, there is a large proportion of data with either no inputs or inputs of "zero."

Table 1: SRDR Volatility Inputs for All Data

RV Input style	Count	Percentage
No Input	2174	43%
Numerical	1090	22%
Adjective	813	16%
Zero's	548	11%
Percentage	444	9%

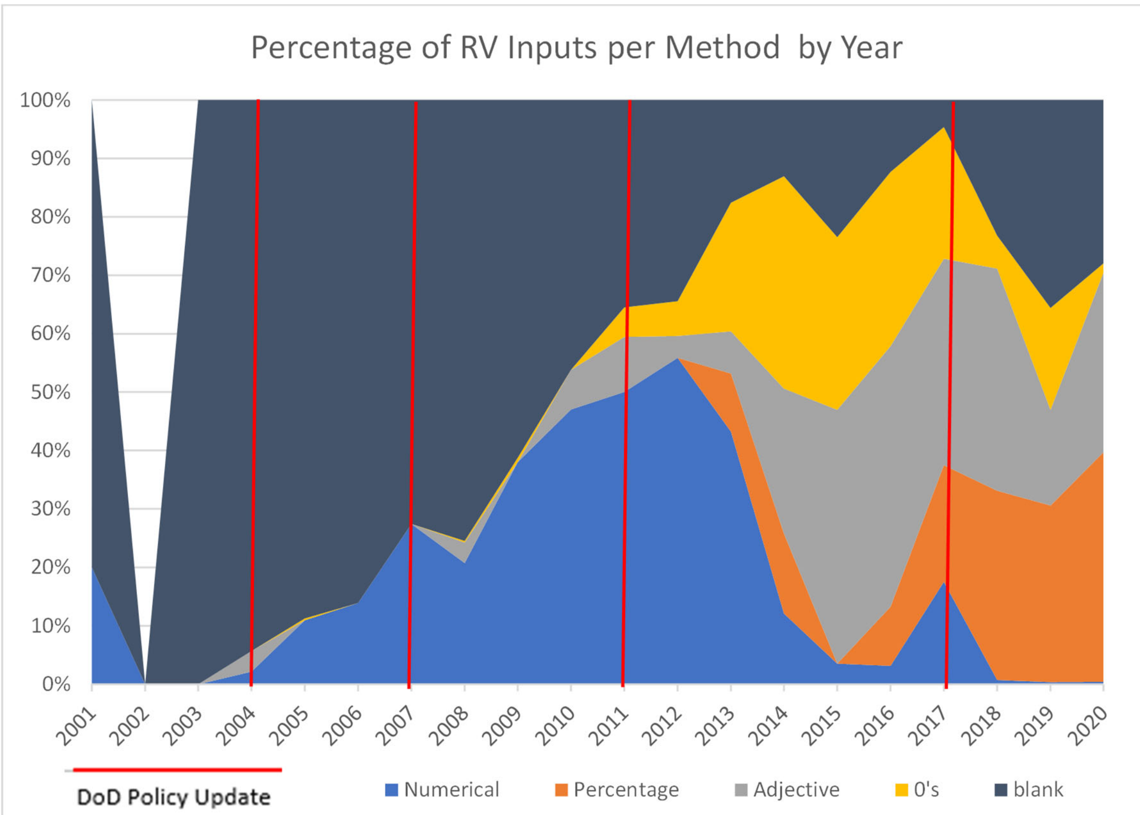


Figure 1: Time Series of All RV Data by Percentage of Method

The messy co-existence of competing scales provides a unique opportunity. Figure 1 shows the relationships which can be explored with the data set. The specific questions asked are:

1. To what degree have contractors adhered to DoD data requirements?

2. Do projects which submit RV data perform better or worse than those that do not?

3. Which RV schemes are best predictors of growth? Growth will be measured as change in ESLOC, schedule, peak head count, total hours, hours in development phases, internal requirements and staffing change. RV in this test is the independent variable. It is shown in the Figure 2, labeled with the letter A. *Initial* RV data will be used for this test.

4. Which characteristics of the program correlate to RV measurements? In this test, RV is the dependent variable. It is shown in Figure 2, labeled with the letter B. Both Initial and Final reports are tested. Initial will reflect the ability of program managers to use the RV measure early in the program. The final reflects the ideal state of a program manager, able to apply a retrospective, where learning of associations may take place.

4. Which characteristics of a program also relate to the intended program growth? This test is shown by the letter C. (*Initial* inputs for items which change)

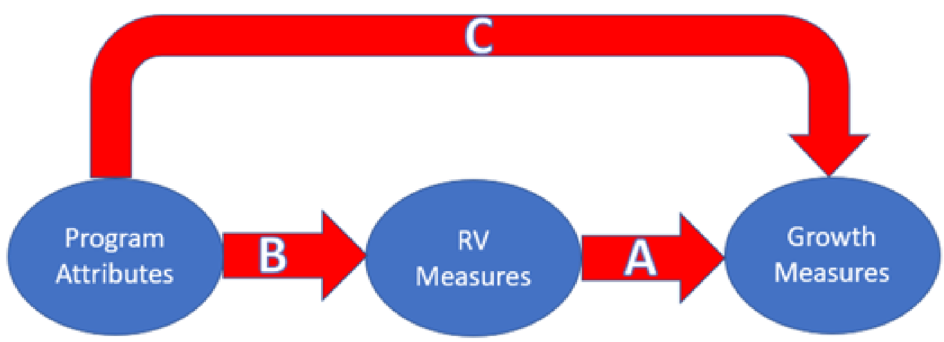


Figure 2: Mediation Models for Analysis for RV, Growth and Program Attributes

Because this research is exploratory, we test these relationships in a myriad of ways. The RV inputs have been primarily received in three ways, a Numerical 1-5 rating, an Adjective rating, or as a Percentage, each of these three methods will be tested independently. Additionally, the three measures will be conjoined to make a unified scheme that will be tested as well as testing submissions which supplied input against those which did not. For measures of growth, we will look items that are known to be drivers of costs, some schedule elements and some more general change indicators that may be indications of complexity. For program attributes we will primarily focus on the items that can be clearly defined at the beginning of the program as well as a few items like requirements, size and experience level of team that may be more telling of a program and how RV is considered. The testing of RV as a mediation variable is being done in less traditional sense, typically mediation is tested when a strong relation between the predictor and criterion variable is known (Baron 1986). For our study there are no known relationships, let alone known strong relationships, just theoretical ones will a willingness to explore a wide range of variables. This exploratory study into RV will utilize contingency tables, a Pearson test with an alpha of 0.10 to identify significant relationships, and an Odds Ratio to calculate the strength of relationships.

Answering these questions is fundamental to place RV in the important role it is supposed to play—that of a mediator between the sense of volatility a program may experience and the impacts those may have of cost and schedule. In a further exploration, this study will also seek to determine if the disparate schemes are somehow relatable or translatable to each other, revealing possible commonalities of the thinking among inputters when wrestling with the ill-defined but intuitively important concept of RV. That is, perhaps we can learn something about the psychology of measuring RV through the disparate attempts to do so within our dataset.

In its current state, the DoD data is of little use for practitioners within the DoD. The DoD must provide guidance that solidifies the concept of volatility, and which prescribes a scheme which can be consistently applied and be predictive and intelligible. The current thesis aims to contribute to those ends. It aims to assist in improving DoD program management. It also aims to contribute to the larger academic pursuit in clarifying and operationalizing the concept of volatility in the sphere of software development.

II. Literature Review

In its most basic definition, Requirements Volatility should be seen as the change in requirements. The purpose of RV is to capture the rate at which requirements are changing, how much the requirements have already changed, or predict how much they should be expected to change. The implications are commonly discussed. Volatility has proven to have implications for both a project's ability to finish on time and within budget (Standish,1995). The Standish group conducted a study that investigated the success and failure of software projects. Their 1995 study found that 31.1% of projects fail to ever be completed and of the project which do get completed, the average cost overrun was 189%. Capers Jones similarly identified that, with volatility a persistent problem for projects, only 10 percent of the 250 projects observed managed to finish on time and within budget (Jones, 2004). Despite the broad recognition that volatility can be problematic, the concept of volatility is quite varied, revealing the challenge of creating a singular, global measure.

Definitions vary both in terms of conceptual elements and form of measurement. A few examples capture the range of definitions. Nurmuliani (2004) defines RV as the "tendency of requirements to change over time." He is mathematically precise in his definition, stating that "the rate of RV is measured as a ratio of the total number of requirements changes (add, delete, and modify) to the total number of requirements in the system over a period of time" (p.4). The count of items governs his thinking. The SEER Software Estimation Model describes RV as the "anticipated frequency and scope of change in the requirements during the development" (p.30). Both count and magnitude of impact figure into this definition. And, in contrast to Nurmuliani (2004), the model applies broad measures, classifying change as: minor, moderate, or major. Didar Zowghi (2002) in her article, “Study of the Impact of Requirements Volatility on Software Project Performance,” adds sociological or institutional aspects to the definition. She describes RV as the “potential for change in business environment, fluctuation in users' requirements (instability), and disagreement among users/stakeholders on requirements (diversity)”(p.2). Her measure relied on an 8-question survey with a five-point Likert-style scale to assess the RV present in a project from the SME perspective. Finally, Constructive Cost Model (COCOMO) has evolved in its understanding of volatility. Initially, it was understood in reference to the hardware aspect of the project not the requirements. But over time, it became associated with requirements change. The COCOMO II Model Definition Manual 1995 edition now uses the acronym REVL to refer to the “requirements evolution and volatility.” REVL is used as a factor to help estimate a project's size. Since COCOMO's release, and perhaps due to its impact on the field, volatility is often meant to reference changes in requirements (Aaramaa, 2017; Henry, 1993; Nurmuiani, 2004, 2006; Stark, 1998; Zowghi, 2002).

Factors

As can be gleaned in these definitions, numerous factors may contribute to the RV of a software project. While the count of change figures into each of these, count alone is a misleading guide to try to capture the implications of those changes. Consider the simplest fact that not all change is additive. Some changes, of course, can reduce the scope of the project, and others are more horizontal in terms of scope. Increasing the number of requirements is often referred to as requirements creep or scope creep (Zowghi, 2002). As a manager, managing these types of requirements is crucial for success in software development (Jones, 1996; Standish, 1995; Stark, 1998; Zowghi, 2002). Programs that experience scope creep are much more likely to fail or finish well over budget when compared to projects that did not (Jones, 1996). The decreasing of requirements can happen when a function is no longer needed, or requirements have been consolidated reducing the total number of requirements. These types of changes are often referred to as scrap (Kulk, 2008). Horizontal change, or swapping of requirements, happens when a requirement is removed but a very similar requirement replaces it, swapping does not change the total number of requirements. This is referred to as requirements churn (Kulk, 2008). Churn typically is related to small changes that do not advance or inhibit the first intended function of the project. An example would be if colors needed to be changed, or the placement of an output needed to be adjusted (Kulk, 2008). Each of these can have different directional impact on cost and schedule.

There is also a time element involved with the implications of volatility. It has been recognized by multiple scholars that there is a distinct difference in impact depending on when RV occurs. The timing of RV is often referred to relative to the phases of development or more broadly the software lifecycle. The key one, is the early requirements identification phase. Those that happen before the plan is agreed upon are less likely to hinder performance than those that happen after (Zowghi, 2002; Singh, 2012; Nidumolu, 1996; Stark, 1998). The literature is predominantly concerned with the RV that occurs after requirements have been agreed upon. The RV which occurs before requirements have been reviewed is constructive, while the RV after is considered destructive (Singh, 2012; Zowghi, 2002).

The literature provides finer detail regarding the impact of RV in respect to when it occurs in the development cycle. RV may occur through all phases of software development, but that large degrees of RV towards the end of projects were indicative of failing or struggling projects (Zowghi, 2002). For a traditional waterfall development method, RV that occurs in later phases tend to be more costly (Singh, 2012). A poor quality or error in requirements discovered in operation can be 100 times costlier than had it been discovered when the requirements were still being defined (Singh, 2012).

It is widely accepted that early detection is a key to success in Software Development, especially when success is based on finishing on time and within budget (Zowghi, 2002). Capers Jones, perhaps one of the most published in the field of software management, believes requirements change in the design, coding, and testing phases must be near zero for success in managing requirements (Jones 1996). The further along a project is, the more likely that a requirement that is changed will have implications on other requirements. This scenario will lead to greater rework than simply fixing or updating one requirement.

The degree to which requirements are interconnected matters too. Interconnected requirements that are changed will increase the amount of rework. A recent publication on using machine learning to assess RV uses classes to categorize types of requirement behavior as result of a related requirement being changed (Hein, 2021). The four classes used are Multiplier, Absorber, Transmitter and Robust. These classifications are important to understanding the different impacts a changed requirement may have on a system. A Multiplier requirement will propagate further change in other related requirements. An Absorber requirement is changed by the initial related requirement but does not propagate further changes. A Transmitter is itself unaffected by the initial change but necessitates other changes down the line. A Robust change class is for requirements that remain unchanged by a related requirement and do not propagate further changes (Hein, 2021). Figure 3 is a visual representation of each class of RV. This classification system demonstrates that not all requirement change is equal, therefore a RV rating that relies more on count than nature of change may under-measure RV. The interconnectedness in software requirements complicates tracking and measuring changes.

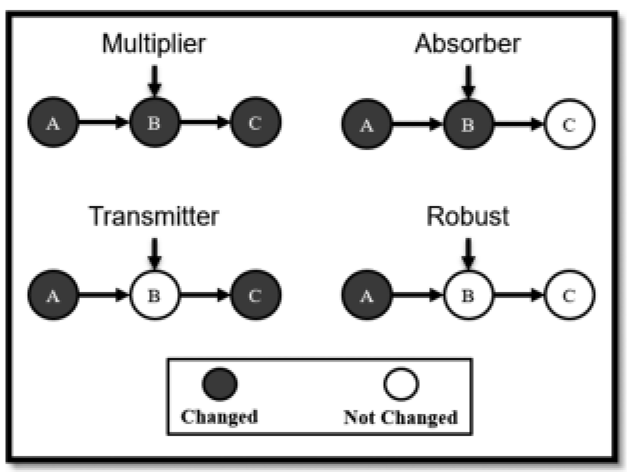


Figure 3:Visual Description of Requirement Volatility Classes (Hein 2021)

The forces that drive or necessitate a change in requirements are often referred to as factors. These factors can be split into two main categories, internal and external (Zowghi, 2002; Mundlamuri, 2005). External factors are things outside the control of the developers or managers or external to the system but will affect performance. Internal factors are things that lead to requirements change that are internal to the system, common examples are the quality of requirements and requirements management. Two common external factors are government regulation and market competition (Zowghi, 2002; Mundlamuri, 2005). Unforeseen changes in regulations or the market are likely to impact development by necessitating a change to stay compliant or competitive.

Internal factors are the factors which can be managed or affected by those involved in the project. Any requirement change that is not driven by an external factor will be a result of an internal factor. Of the internal factors, management of requirements is the most impactful to RV (Nidumolu, 1996; Stark, 1998). Requirements management will include the solicitation of requirements, as well as the tracking and vetting of any change request throughout a project. Poor initial requirements can often be attributed to poor communication and will likely result in rework and contribute to low productivity. Programs that suffer from low productivity are likely to go past schedule and over budget (Singh, 2012). Poor management of requirements is similarly impactful. The communication between the user and the development team is crucial to managing RV. Two experts in the field Capers Jones and Didar Zowghi draw a connection between the quality of requirements management and the quality of the software produced (Jones, 2004; Zowghi 2002). Jones (2004) finds that projects of higher quality are much more likely to be within budget and completed on schedule.

Looking into Requirements Volatility

It is widely accepted that the RV is important, but there is no standard system for assessing or measuring it. There are two lenses in which estimators and researchers typically view RV, a forward look, or a backwards look. A forward-looking assessment will be aimed at trying to estimate the future volatility of a project's requirements. This assessment can be used for trying to estimate the size and complexity of a project (COCOMO II, 1995). On the other hand, the backwards looking study will be looking at past performance of a project and assessing the RV that has already occurred. These two interpretations for RV mirror the RV data available in the SRDR Compilation dataset that is used for this study. RV input provided at the beginning of projects are to be predictive by nature and the RV analyzed after completion is assumed to be reflective.

COCOMO II uses a term they call REVL which stands for Requirements Evolution and Volatility. REVL is the percentage of code estimated to be discarded due to requirements evolution and volatility due to mission or user interface evolution, technology upgrades or commercial off the shelf software (COCOMO II, 1995). The size of the project is then adjusted by the percentage of REVL. (See Equation 1 for the COCOMO II Size Equation). In the case of a forward-looking estimate, the actual count is not going to be known so the question will be based on what the estimator decides the count will likely be. COCOMO generally assesses uncertain parameters using a system of 5 categories, (Very Low, Low, Nominal, High, Very High) and each category will have a numerical value that is associated with each rating. No such values or descriptions exists for REVL.

By way of comparison, SEER uses a 5-scale measurement as well, but it does provide descriptions to guide an estimator with assessing RV. The ratings and descriptions can be found in Table 2.

Equation 1:COCOMO II Size Equation

$$Size = \left(1 + \frac{REVL}{100}\right) \times Size_D$$

*100 represents the total instructions

Table 2: SEER for Software Ratings and Descriptions

Requirements Volatility (Change): Anticipated frequency and scope of requirements change once they are baselined (after preliminary design starts).	
Rating	**Description**
Very High	Very High Frequent moderate and occasional major changes.
High+	High+ Evolutionary development with significant user interface requirements.
High	High Occasional moderate redirections, typical for evolutionary developments.
Nominal	Nominal Small noncritical redirections.
Low	Low Essentially no requirements changes.

RV as a rate can be used as an indicator for when a project may be headed out of control. G.P. Kulk (2008) used the compound interest formula that is often seen in accounting to calculate RV as a rate. (The formula used can be found in Equation 2) Kulk decided on this formula based on the prior work of Jones who had used it in work calculating monthly RV rates and comparing them across industries. Kulk determines this method to be best for his study based on the idea that volatility growth is not linear as changes later in projects are known to have greater implications. Kulk goes on to show that with a known industry acceptable rate of growth his methods could be used to find a project spiraling out of control prior to it reaching its cancellation point.

Equation 2: RV as Rate Formulas (Kulk 2008)

$$SizeAtEnd = SizAtStart \cdot \left(1 + \frac{r}{100}\right)^t$$

$$r = \left(\sqrt[t]{\frac{SizeAtEnd}{SizeAtStart}} - 1 \right) \cdot 100$$

When conducting backward-looking research, the most common measurement for RV is as a percentage of change (Nurmuliani, 2006; Singh, 2012). Typically, it will be the percentage of change from when requirements were set to the end of the development cycle. There has also been research conducted using categorical ratings as measurement in backward looking studies. Zowghi used a Low, Medium, High system when gathering data from practitioners on their past projects and their assessment of the RV they experienced. For Zowghi's assessments she used a survey which asked a series of simple questions that could be scored, then used to determine where the project should be categorized. Knowledgeable software developers and managers completed the surveys.

The conclusion one should draw from this literature is that the attempt to measure change in a meaningful way—that is, as a single predictor of cost impact, or as a summary post-mortem—is no straight-forward ambition. The complexities of RV make it difficult to measure and hinders its correlation to cost and schedule growth. When RV is used as a forward-facing measure, it is aiming to capture risk and uncertainty. It is impossible to measure what is unknown. This idea adds to the complexities of capturing RV accurately. A simplistic measure like taking a percentage is likely to fall short of fully capturing either risk or uncertainty in requirements. The interconnectedness of requirements is likely a contributor to the shortcomings of percentages as RV measurement. Depending on what is counted when taking a measurement is also cause for concern. Requirements can experience churn, so counts that just count beginning and end numbers could potentially underestimate the impact that churning requirements can have on cost and schedule. The dynamic nature of software in general may make it understandable that requirements are going to change but it will always be difficult to know how much change will occur. The external factors that can never truly be fully known also contribute to the complexity of predicting RV levels. Things like government regulation or pressures of competition are going to be difficult to capture in predictive RV measure. With the inherent complexities of software perhaps a measured approach is not best for forward looking RV assessments.

How does DoD Understand RV?

The DoD has a long-understood RV to be important. Prior to the existence of the Cost Assessment Data Enterprise (CADE), there was a data collection agency known as Quality Software Measure (QSM). In the era when DoD software data collection was being accomplished by QSM, data was reported on a Software Description Annotated Outline or DD Form 2630 (R. Curry, personal communication, January 31, 2022). The very first data collection item on the 1992 form is System Requirement Volatility. The volatility information was collected in a series of three questions with boxes for the inputter to check the most relevant box. Figure 4 displays the initial portion of DD Form 2630, August 1992 version. As software development progressed and evolved so did data collection efforts, and the importance and understanding of RV. The Form 2630 eventually developed into three separate forms, the 2630-1 known as the *Initial Government Report*, 2630-2 known as the *Initial Developers Report*, and 2630-3 known as the *Final Developers Report* (DoD 5000.4M-2, 2004). The 2004 SRDR Manual outlines the expectations for RV reporting, the guidance states,

> As part of the final DD Form 2630-3 report, indicate the amount of requirements volatility using a qualitative scale (very low, low, nominal, high, very high) relative to similar systems of the same type. This should be a relative measure rather than an absolute one in order to understand how initial expectations were or were not met during the course of the software development (DoD 5000.4M-2, 2004 p.16).

This 2004 regulation is critical because the dataset this study analyzes starts around that time. In 2007, SRDR Data Item Descriptions (DIDs) (also known as DI-MGMT-81739 & 40) were released for both *Initial* and *Final* developer reporting. The key changes are in the requirement for developers to provide RV input for both the *Initial* and *Final* reports. The DID for *Initial* and *Final* reporting asks for a qualitative scale of expected RV for *Initial*, while the DID for *Final* ask for the RV encountered. Both DIDs like the DoD 5000.4M-2 ask the contracted developers to also provide definitions for each qualitative ranking, and the overall definition they used to assess RV.

SOFTWARE DESCRIPTION ANNOTATED OUTLINE
(See DoD 5000.4-M for additional guidance)

GENERAL INSTRUCTIONS

Describe the characteristics of the system software. Supply requested data for both the top level and each Computer Software Configuration Item (CSCI) (and CSC when available). Information presented at the top level should apply to all the levels below.

Other data that could affect system costs should be provided at the appropriate level of detail. This includes any information not requested below but which is necessary to prepare a cost estimate. Other input data that are used in a software cost model should be included as an appendix to the Cost Analysis Requirements Description (CARD) submission.

In each question, if a response pertains only to selected software items, identify those items in the "Additional Comments" block.

Section I - Top-Level Characteristics. Above the CSCI Level. Information provided in this section should apply across the system's software, including each CSCI (and each CSC when available) and each software build.

Section II - Lower Level Characteristics. Complete for each CSCI (or each CSC when available) and each build.

SECTION I - TOP-LEVEL CHARACTERISTICS (Above CSCI Level)

1. SYSTEM REQUIREMENT VOLATILITY

a. LEVEL OF DEFINITION AND UNDERSTANDING OF SYSTEM REQUIREMENTS *(X one)*
- [ ] (1) Very little
- [ ] (2) Questionable
- [ ] (3) Fairly complete
- [ ] (4) Very complete

(5) Additional Comments

b. HOW WILL OVERALL TECHNOLOGY ADVANCES DURING DEVELOPMENT AFFECT THE PROJECT? *(X one)*
- [ ] (1) Significant advances; more than one system upgrade
- [ ] (2) Between one and three significant system modifications
- [ ] (3) Minor modifications
- [ ] (4) No changes to system or requirements

(5) Additional Comments

c. REQUIREMENTS VOLATILITY DURING DEVELOPMENT *(X one)*
- [ ] (1) No changes
- [ ] (2) Small noncritical changes
- [ ] (3) Frequent noncritical changes
- [ ] (4) Occasional moderate changes
- [ ] (5) Frequent moderate changes
- [ ] (6) Many large changes

(7) Additional Comments

2. SYSTEM INTEGRATION DIFFICULTY

a. EXPECTED LEVEL OF DIFFICULTY OF INTEGRATING AND TESTING THE CSCI's TO THE ELEMENT LEVEL *(X one)*
- [ ] (1) Very little integration, no complex interfaces
- [ ] (2) Average degree of system integration/interface complexity
- [ ] (3) Several system interfaces, some complex
- [ ] (4) Complex, time-intensive integration process anticipated

(5) Additional Comments

Figure 4: DD Form 2630, Aug 1992, Section 1

Within the DoD acquisition realm, the hunt to improve software estimation and thus the importance of RV continued to evolve. The Naval Center for Cost Analysis and the Air Force Cost Analysis Agency teamed up to publish a handbook that would highlight the best practices and comply with DoD policies. The result of this work became the 2008 Software Development Cost Estimating (SDCE) Handbook. Among the general software estimation information, is discussion on RV and its role in an estimate. This SDCE Handbook describes an RV rating as evaluating, "the cost penalty due to expected frequency and scope of requirements changes after the baseline Software Requirements Review, and projects the impact of those changes on delivery cost and schedule" (2008, p.10-6). The SDCE Handbook recommends using a weighting system. Table 3 shows the values and descriptions associated with each value. Each rating has a description of the products that qualify for that rating. Although, they refer to the cost in the description the impact of the rating manifests in the sizing of the estimate (2008, p.10-7). The lowest category that describes, “no” requirement changes, is to be equated to software being produced on an assembly line for a well-defined product, these products have a less than 1 value which effectively shrinks the size of the project or suggest learning. The next category is described as a familiar product, this means that the developer has created the product before, and the customer is “familiar” with it, no growth or efficiency is expected for this type of product, thus its value is one. This differs from the next rating of a “known” product, which is taxed with a 15% growth, because the product is known, by the developer and customer, but it will be advance in some way. The final two categories are characterized by technology, the terms “exists” and “new” face the heaviest growth tax of 29% and 46% respectively. Existing technology is distinguished from new technology, by if the product has ever been produced before, even if the developer themselves did not produce it. New is characterized by a true research and development type of effort (2008, p.10-7). The 2008 SDCE Handbook also identifies that many of commercial software estimation techniques out there use similar but different measures and assessments, a modified version of this table can be found in Table 4. We believe the value down the left-hand side to primarily be for comparison, not for use in the estimates these models produce. The comparison is to distinguish the difference in how the prescribed model will weight that type of software comparatively to other models. The overall level of uncertainty in a model RV is expected to capture for the estimate will likely drive the weighting of the value assigned.

Table 3: RV Values and Description from 2008 SDCE Handbook

RVOL - Requirements Volatility	
Value	Description
0.93	Essentially **no** requirements changes
1.00	**Familiar** product – small non-critical redirections
1.15	**Known** product – occasional moderate redirections
1.29	Technology **exists** – unfamiliar to developer
1.46	Technology **new** – frequent major redirections

Table 4: Modified 2008 SDCE Handbook Commercial Estimation Models RV descriptions and Value

Value	CA Definition 9/10	Sage	SEER-SEM	REVIC	COCOMO	PRICE-S
0	No changes	Essentially no requirements changes	Essentially no requirements changes	No changes		No changes
1					Essentially no changes	
2					Small, non-critical redirections	
3	Very few changes expected	Familiar product, small noncritical redirections	Small noncritical redirections	Very few changes expected	Occasional moderate redirections	
4					Frequent Moderate or occasional redirections	
5	Minor changes to requirements caused by design reviews or changing mission requirements	Known product, occasional moderate redirections	Occasional moderate redirections, typical for evolutionary software developments	Minor changes to requirements caused by design reviews or changing mission requirements	Frequent Major Redirections	
6			Evolutionary software development with significant			

			user interface requirements			
7	Some significant changes expected (none late in development phase)	Technology exists, unfamiliar to developer		Some significant changes expected (none late in development phase)		Changing requirements
8			Frequent moderate & occasional major changes			
9						New Hardware
10	Expect major changes occurring at different times in development phase	Technology is new, frequent major redirections	Frequent major changes	Expect major changes occurring at different times in development phase		Parallel hardware development

Following the groundwork laid by the 2008 SDCE Handbook the 2011 SRDR Data Item Description (DID) Guide supplied more guidance for RV and its reporting. Regarding RV this regulation provides two key items. First, it specifically outlines how contractors should be recording RV and it defines when they should be measuring from. The 2011 SRDR DID Guide states, “Indicate the amount of requirements volatility expected during development as a percentage of requirements that will change after the Software Requirements Review” (2011, p.10). Secondly, the 2011 DID Guide ask for the “contractor’s internal definitions used for classifying requirements volatility” (2011, p.15). Calling for a specific measure and definition of RV is necessary for the allowance of future analysis of past data. These guidebooks are continually modified and updated to try and maximize the usefulness of the dataset being built.

The SRDR DID Guide was updated from the 2011 version in 2017. The RV guidance put in place in 2017 is the most recent guidance issued. This guidance defines

RV as, “the amount of requirements volatility expected during development as a percentage of requirements at the Software Requirements Review (SRR) that will change or be added thereafter” (2017, p.16). Additionally, this latest guide asks for two other RV related items, they are looking for a percentage of External Interface Requirements Volatility as a percentage on the same submissions, as well as, “For each release and priority, provide a count of the number of unplanned software changes that were added/changed/deleted from the release after the release began” (2017, p.17, 35).

Overall, how the DoD views and approaches RV has been and is in line with how the literature has described it. The DoD does not take perhaps the most difficult or demanding approach but is direct and proper execution may be helpful for future estimates. RV is a difficult yet important parameter for estimating software development as it offer a. Properly assessing RV could merit strong estimates, failure to assess RV properly is likely to lead to cost and schedule overruns. The discussion of RV and its utility is not complete, and this study aims to aid that discussion.

III. Methodology

The SRDR dataset from CADE provides the historical data that will be used for the analysis. The dataset is a collection of contractors' inputs from all Acquisition Category I and IA Programs, Major Defense Acquisition Programs (MDAP), and Major Automated Information Systems (MAIS) programs following Milestone A (OSD CAPE, 2019), this SRDR dataset has 78 columns and includes 5074 project submissions. Submissions are tied directly to a program and Computer Software Configure Items (CSCIs) which often also identifies the work breakdown structure (WBS) element. Submissions are reported at multiple timeframes of a project but the important submissions for this research are the ones identified as "*Initial*" or "*Final*."

For the analysis to be complete, several steps are needed. The first task is to identify what trends if any exists for the reporting of RV on *Initial* and *Final* submissions. This will include observing the RV reporting trends over time, while identifying the primary reporting schemes and when DoD regulations where updated. Following this deep dive into the composition of the overall data the individual schemes must be further analyzed. A histogram will be taken of each of the primary schemes as a way of further examination. These distributions and overall observations of reporting trends will be included in chapter 4.

The testing vessel for this exploratory study will be contingency analysis. The study will test relationships between the multiple schemes of RV in the SRDR, a variety of growth measurements, and program attributes. Each of these variables will be carefully selected and made binary for the building of contingency tables. Odds Ratios will be calculated when the Pearson test shows significance less than 0.1.

Data

This analysis uses the SRDR dataset and breaks it into many subsets depending on which analysis is being conducted. For the observational analysis, which is looking at the reporting trends over time the 5,074 submissions are reduced to 3,018. Of these remaining 3,018 submission 1,072 were identified as *Initial*, 1,946 were *Final*. Because past guidance asked for RV reporting for both *Initial* and *Final* submissions the data must be viewed independently when seeing the data trends. Each time RV is analyzed the *Initial* and *Final* submissions will be looked at separately.

The next set of data will be utilized for contingency analysis between RV and the growth measures. For analysis in growth a measurement from an *Initial* and *Final* submissions are needed, the CADE analysts on the SRDR compilations dataset identify these as "Data Pairs" and have compiled a list of 408 pairs on a separate tab in the dataset. Table 5 breaks down the data found in these 408 pairs. For the *Initial* submissions 288 did not provide any input for RV, while only 43 provided no RV input for *Final* submissions. Of the 120 that did provide RV input on *Initial* submissions 43 were reported in the Adjective scheme, 38 as a percentage, 35 as a 1-5 numerical rating, and 4 inputs were zeros. On the *Final* submissions the 365 RV inputs were broken out as 95 in an Adjective scheme, 40 as a percentage, 221 as a 1-5 numerical rating, and 10 as zeros. In the final submissions one entry was provided as, "Low, 8.58%" and was counted as both a percentage and Adjective.

Table 5: Volatility Inputs Categories and Number of Inputs for Paired Initial and Final Submissions

	Volatility Input:	Number of Inputs
Initial	Pairs	408
	No Input	288
	Remaining	120
	Adjective	43
	Percentage	38
	Numerical	35
	Zero	4
Final	Pairs	408
	No Input	43
	Remaining	365
	Adjective	95
	Percentage	40
	Numerical	221
	Zero	10
	*1 input was given as: Low, 8.58% so it has been counted in Adjective and Percentage	

To try and determine what conceptually undergirds the SMEs attempt to quantify RV, we will conduct contingency analysis comparing those quantifications of RV to a series of variables which conceptually relate to RV. A larger dataset can be used for this study, since we do not need to assess *change* from *Initial* to *Final*. For these tests, the following exclusion process took place. We begin with 5,074 submissions. On the SRDR dataset the Validation and Verification team have labeled certain program inputs as "Impossible Schedules" based on improbable measures of the variables, Peak Head Counts and Hours. Removing those programs from the dataset reduces the submissions available from 5,074 to 3,954. The remaining data will be further separated into *Initial* or *Final* inputs, so as to conduct separate tests. Not all submissions are labeled as *Initial* or *Final*, and those that are not will be excluded. Table 6 shows the data that is left after excluding the impossible schedules and *non-Initial* or *Final* submissions. The *Initial* data allows us to test what a submitter may consider when providing RV inputs that are predictions of potential change. For these we have 861 data points remaining of various RV schemes as shown in Table 6. The *Final* data allows us to test what a submitter may consider or what may impact providing RV input. This leaves us with 1,549 submissions to analyze, broken in various schemes, as shown in Table 6.

Table 6:Volatility Inputs Categories and Number of Inputs for Initial and Final Submissions

Initial	Initial Submissions	861
	No Input (N/A or Unknown)	505
	Remaining	356
	Adjective	130
	Percentage	133
	Numerical	45
	Zero	34
	Input as "Normal"	14
Final	Final Submissions	1549
	No Input (N/A or Unknown)	297
	Remaining	1257
	Adjective	331
	Percentage	327
	Numerical	522
	Zero	77
	*5 inputs were given as: Low, 8.58% and were counted in Adjective and Percentage	

RV Input Schemes

RV will be tested in a variety of ways for this study. First, RV will be tested with the growth measures and then later tested with the selected program attributes. The three RV input schemes of Adjective, Percentage, and Numerical will all be tested for statistical differences in growth outcomes and program attributes. Because there are many ways that data submitters can provide inputs in an Adjective scheme, this scheme must also be classified to Low, Medium and High. This will be done using Adjective inputs found in Table 7. In addition to the three primary schemes a conjoined scheme and testing inputs vs blanks will be tested. The conjoined scheme maximizes the applicability of the dataset by allowing for the inclusion of zeros and percentages over 100% and tries to normalize all schemes to one. It will allow for an all inclusive view of RV and of the trends in ratings. Table 7 shows the methods used for conjoining the various input schemes to one simplified system. For the percentage scheme, inputs of 0-33% will classified as Low. The percentage inputs were split input thirds because they are being classified into 3 categories. For the Adjectival schemes Nominal is most often referred to as "small non-critical changes" and therefore will be considered a Low input. The other attempt to maximize the available data, is done by testing the projects which provided RV submissions against those which did not. This will also allow for a look into who and perhaps what characteristics are more indicative of receiving RV input. The distributions, and trends overtime for the RV reporting can all be found in chapter 4.

Table 7: Conjoined RV Conversion Method

Low	0-33%, Low, Very Low, Nom, Nominal,0, 1, 2, and yellow
Medium	3, medium, med, and 34-66%
High	67-100+%, 4,5, high, very high

Growth Measures

The literature made indications that RV may impact the performance of a project in a multitude of ways, for this reason we will be testing out RV variables against the measures identified in Table 8. To test growth, we must have the *Initial* and *Final* submissions for our CSCI. Table 8 shows the number of submissions available for each growth category to be assessed. The median will be used to separate the upper distribution of percent growth from the lower half for contingency analysis. The median was the best choice for splitting the data because the distributions were heavily skewed right with means, often the mean was found to be three to ten times greater than the median. Using the median weights all inputs the same and does not let the potential outliers distort the analysis by showing most projects performing under the mean. Distributions and descriptive statistics of the growth factors can be found in the Appendix.

Table 8: Growth Measures, Number of Inputs, and Medians

Growth Measures	N	Median
Equivalent Source Lines of Code (ESLOC)	408	35.4%
Schedule in Months	402	19.4%
Peak Head Count	408	0.0%
Total Hours	408	24.7%
Phase 1: Requirements Analysis	324	5.2%
Phase 2: Architecture & Design	309	12.8%
Phase 3: Coding & Testing	354	26.7%
Phase 4: Software and System Integration	278	23.5%
Phase 5: Qualification Testing	204	21.8%
Phase 6: Development Test & Evaluation	127	41.8%
Internal Requirements	408	0.0%
Staffing Mix Change Yes or No	404	N/A

Performances and Growth Measures

The contingency tables will evaluate various breaks in RV measuring schemes compared to a median spilt in the growth measures. Each test will tell us how the RV schemes may or may not predict program changes. For tracking the program changes the SRDR Compilation dataset tracks many values that are measured and will inevitably be different in the *Final* report. It is known that the *Initial* report is an estimation for the project, but it is assumed that the info on the *Final* report is the actual values. Of the measured inputs, twelve of them will be used to test against the RV inputs. The percent change will be calculated for eleven of the twelve values. For one of the values a simple measure of change or no change will be used. The percent change will be calculated for the eleven variables using Equation 3. The median will be used to split the percent change values categories of greater than the median or less than or equal to the median. This will allow us to determine if there is a relationship between the RV categories or the top or bottom half of the growth measure.

Equation 3: Percent Change

$$\%Change = \frac{Final - Initial}{Initial}$$

The first growth measure to be examined will be the equivalent source lines of code (ESLOC). ESLOC is commonly used as a reference or assessment of the total size of a project. RV is the changing of requirements and this has been known to potentially cause a project to grow over time, in estimation it is most often used to adjust a projects size. An Analysis of RV values from *Initial* submissions and ESLOC will tell us if the RV values are at all predictive of ESLOC growth. For RV inputs on *Final* submissions a relationship between RV inputs and ESLOC growth will be indication that submitters may consider change in size as part of their RV assessment.

Next, the schedule measured in months will be examined. The schedule of a project is repeatedly acknowledged in the literature to be positively correlated to RV. This analysis will assess if this notion is in fact true for the DoD dataset. If the level of RV is proven to be properly aligned with the upper or lower distribution of Schedule in Months growth than we can prove that this ideal holds true for the way these DoD projects have been assessed in the past.

Peak Head Count can be viewed as a measure of multiple aspects of a project. It may be a sign of how much a software project changed over time, a measure of increased labor (cost) and a measure of size growth. It is also a good measure for analysis because all 408 projects supplied input. Being a measure for cost, size and general level of change proper correlation with RV for *Initial* reporting will show if RV is able to predict growth in team size. For *Final* submissions finding a relationship could indicate that submitters may be considering Peak Head Count for assessing project volatility.

The SRDR does not provide actual cost values, but the hours are tightly related to the labor for software development, which is a known driver of costs. Total Hours has been known to be used as a proxy for cost, and the change in total hours thus as a proxy for cost growth. The literature explicitly draws a link between cost growth and RV. Significant relationships in the proper direction could confirm this relationship in the DoD data.

The literature made it apparent that RV can have impacts on growth in the different phases of development. Therefore, we will use the individual phase data from the SRDR to analyze the percent change in hours for each of the six phases tracked. Requirements Analysis is the first phase found on the SRDR and change in this category could lend itself to higher reporting of RV. The second phase is Architecture and Design the literature does not spend much time addressing this phase but the fact that it is earlier in development leads us to believe more change in this phase may not affect cost and schedule like later phases may. Coding and Testing is recognized as Phase 3 and like Phase two it is not widely discussed in the literature being in the front half of the development effort, we should not expect significant change to demand the highest levels of growth but perhaps have a stronger effect than Phase Two. The next three phases are Software and System Integration, Qualification Testing, and Development Test and Evaluation are all in the latter half of development and from what the literature tells us requirements changes in these phases should have greater impact on cost and schedule growth.

The SRDR supplies four columns in addition to volatility on requirements. Total Internal Requirements, New Internal Requirements, Total External Requirements and New External Requirements, New Internal Requirements will be assessed as the measure of requirements change. RV is a measure or assessment of requirements change, and managers and developers have more control over internal requirements than external, for this reason the change in New Internal Requirements will be analyzed with RV inputs. It should be expected that higher levels of RV correlate with the top distribution of New Internal Requirements change. If not, this may be indication that submitters have considered the volatility rating on the SRDR differently than what has been asked or is currently being asked by the DoD in the DID guidelines.

The literature repeatedly indicates that the skill level of the developers and the quality of the software is directly tied to successful builds and abilities to stay on target for costs and schedules. The best measure available for the quality of a team on the SRDR is the Staffing Mix. The Staffing Mix shown, is assessed across five skill levels ranging from highly experienced to entry level, the percentage of the team that makes up each skill level is indicated. A change in this mix may be a sign of an initially poor or overmatched team or even perhaps a change in scope making the project more or less complex. Rather than trying to quantify the level of change that occurred within a team, the identification of change will suffice for this exploratory study. For those reasons it was decided to test Staffing Mix Change as a Yes or No against the RV inputs.

Program Attributes

The SRDR compiles many characteristics of the CSCIs that supply submissions for the dataset. Some of these characteristics could aid in the quest to assess what affects the way a SME provides input, or what they may be considering for their input. Twelve characteristics will be analyzed using contingency analysis with RV as the theoretical dependent variable. Next, the program attributes will be tested with the growth measures to verify the accuracy of the significant RV relationships. In identifying these twelve characteristics, three items were considered, one, how definite is the characteristic? Two, when can the characteristic be known? and three, should it be considered or related to RV? The definite aspect is important because items that do not change, but may be predictive, have value, and their input is not subjective. Knowing the characteristic at the time of the estimate is important because that is when RV data is to be reported. Lastly, items that inherently seem like they should be considered for RV will be tested to determine if SMEs are considering them. Table 9 shows the characteristics and elements that have been identified for testing. When necessary, histograms of the program attribute will be used to find which elements should be analyzed. The values displayed in the histograms are the count of both *Initial* and *Final* submissions when impossible schedules have been removed.

Table 9: Software Development Characteristics and Elements for RV and Growth Analysis

Program Attribute	Elements
Service	Air Force, Navy, Army
Contractor	Raytheon, Boeing, Northrup Grumman Company, BAE, General Dynamics, Lockheed Martin, Northrup Grumman Information Systems
Operating Environment	Air Vehicle, Manned; Surface Fixed, Manned; Air Vehicle, unmanned; Surface Mobile, Manned
Application Domain	Command & Control, Vehicle Control, Other Real Time Embedded
Super Domain	Engineering, Real time, Automated Information System, Unknown
Development Process	Waterfall, Spiral, Incremental, Agile
New vs Upgrade	
Location	East Coast, West Coast, Central, Southeast
Primary Software Language	Java, C++, Ada
New Requirements	New Internal Reqs ≥ 50%, New External Reqs ≥ 50%
Size by Total Hours and ESLOC	Total Hours, ESLOC
Experience Level	Highly Experienced ≥ 50%, Inexperienced ≥ 50%

Service

An examination into the relationship between Service branches and Volatility inputs will identify any possible differences that may exist. The style of RV input as well the strength of relationship and direction will be determined through analysis. The histogram of the DoD service branches in Figure 5 shows the distribution of Volatility inputs by DoD service branch. The Air Force, Navy and Army will each be analyzed using contingency tables against the RV variables and the growth measures.

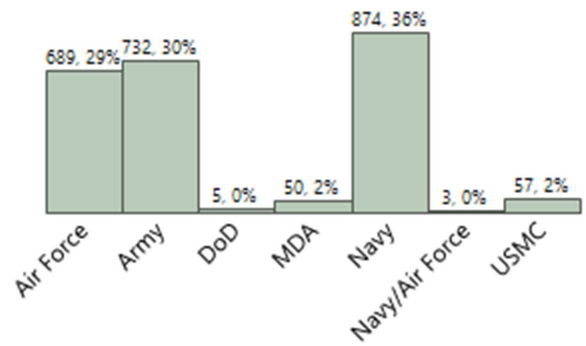


Figure 5: Distribution and Frequency of DoD Services on the SRDR

Contractor

If any category should be expected to show significant difference in RV inputs, it is Contractors. Large contractors will have their own culture, training, organizational standards that could all impact the way they asses the RV of a project. Many contractors have provided inputs for the SRDR. Not all contractors will have provided enough inputs to be considered for individual analysis of RV. Within the SRDR Contractors have been reduced to their "Short Name" which is an abbreviated name of the performing organization. This column has been used to create the histogram in Figure 6. The seven companies that provided the data most often account for nearly 70% of all data and will be used for the analysis.

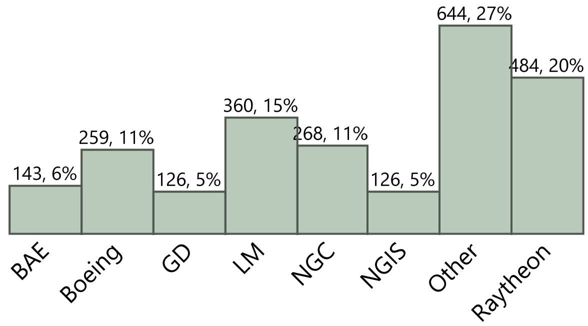


Figure 6: Histogram of Contractors by RV Inputs on the SRDR

Operating Environment

For the purpose of future analysis, the SRDR has a column that identifies the operating environment for the project. This information is not provided by a contractor but is updated by the analyst who is updating the SRDR Compilation dataset. The dataset identifies 15 Operating Environments, one of those being "Other" meaning not one of the 14 normally tracked. The Operating Environment is the environment the software will perform its function. Things like, air, sea, surface, fixed, mobile, manned, and unmanned are a few items used to describe a particular Operating Environment. The acronyms are identified in the general order of; where will it operate, will it move, and is it manned. The histogram in Figure 7 is used to identify Manned Fixed Surface (SFM), Manned Air Vehicle (AVM), and Unmanned Air Vehicle (AVU) as the Operating Environments which provided the most RV inputs making up over 60% of all inputs.

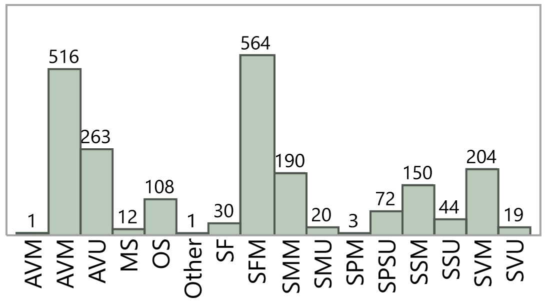


Figure 7: Distribution of Operating Environment on the SRDR

Application Domain

Application domain is another item that analyst inputting data into the SRDR will identify and update. There are roughly 20 domains that are identified in the SRDR. These could be described as the type of mission or function the software will support. A few examples are Command and Control (C&C), Training (TRN), and Mission Planning (MP). The histogram in Figure 8 is used to decide which domains are most prevalent and will be used for analysis. C&C, Other Real Time Embedded (RTE) and Vehicle Control (VC) account for half of all inputs and will be used for analysis.

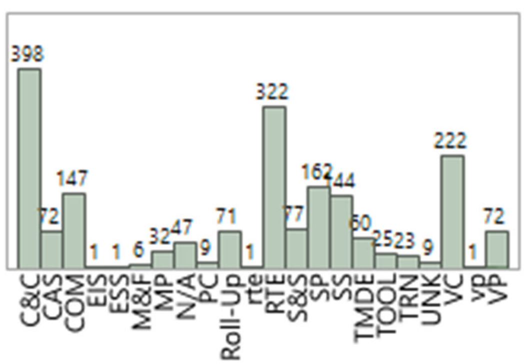


Figure 8: Distribution of Application Domain on the SRDR

Super Domains

Super Domains are four key domains the most projects can be assigned to. This is a label assigned to projects for the purpose of future analysis. The four areas looked at are Automated Information Systems (AIS), Engineering (ENG), Support (MS), and Real-Time (RT). If the super domain is unable to be figured out it will be given the tag UNK meaning unknown. Figure 9 shows the five elements commonly entered in the SRDR and their respective counts for how often they occur. Analysis will be conducted looking at AIS, ENG, RT, and UNK, these four tags account for 98% of inputs.

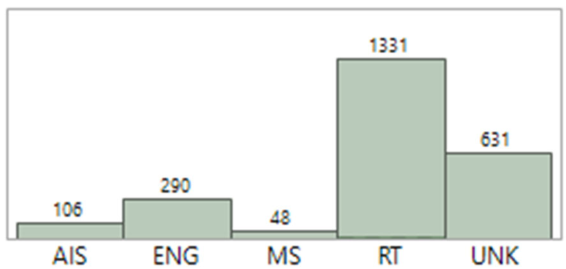


Figure 9: Distribution of Super Domain on the SRDR

Development Process

The process being used for each software project is to be identified by the contractor. Some of the well-known and frequently identified methods include, Agile, Spiral, Waterfall, and Incremental. This category possesses some extra complications for analysis due to the variability of process inputs and the ability of multiple process being used or input. To clean up this category, a new one was created which converted all similarly described inputs to one keyword process. Also, any projects that identified multiple processes were simply considered Mixed. What was left where categorized as other. The histogram in Figure 10 shows the distribution of the conjoined category. Based on the histogram Agile, Incremental, Spiral, and Waterfall will be used for closer analysis on their relationship to RV inputs and growth factors.

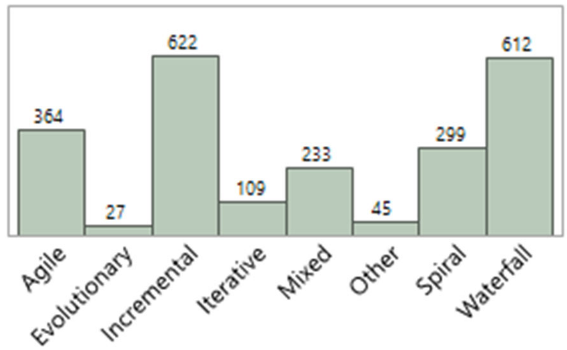


Figure 10: Distribution of Software Development Processes on the SRDR

New vs Upgrade

Every software project on the SRDR is identified as New meaning it is a new product or Upgrade meaning it is a upgrade of an existing software. Occasionally, projects are identified as New/Upgrade, these projects were excluded for the contingency table analyzing its role for RV input. Intuition would lead us to expect new projects to have less defined requirements and therefore high RV and growth.

Location of the Worksite

The company location information was used to see if the location of where the work occurred was in any way predictive. Many contractors have multiple locations so an analysis of geographical location could prove to be valuable. This was done by taking the locations provided and classifying them into four areas, The East Coast, West Coast, Central and Southeast. The East Coast included projects in the states of Virginia, Massachusetts, Maryland, New York, New Jersey, Connecticut, Pennsylvania, and New Hampshire. The West Coast includes the projects that named California or Washington as their location. The Central Region includes the states of Texas, Iowa, New Mexico, Illinois, Michigan, Colorado, Utah, Indiana, Missouri, Minnesota, Ohio, Kansas, and Arizona. The Southeast includes the states of Alabama, Georgia, and Florida. Figure 11 gives a visual representation of the states represented in the SRDR and calculated regions of the worksites to be analyzed.

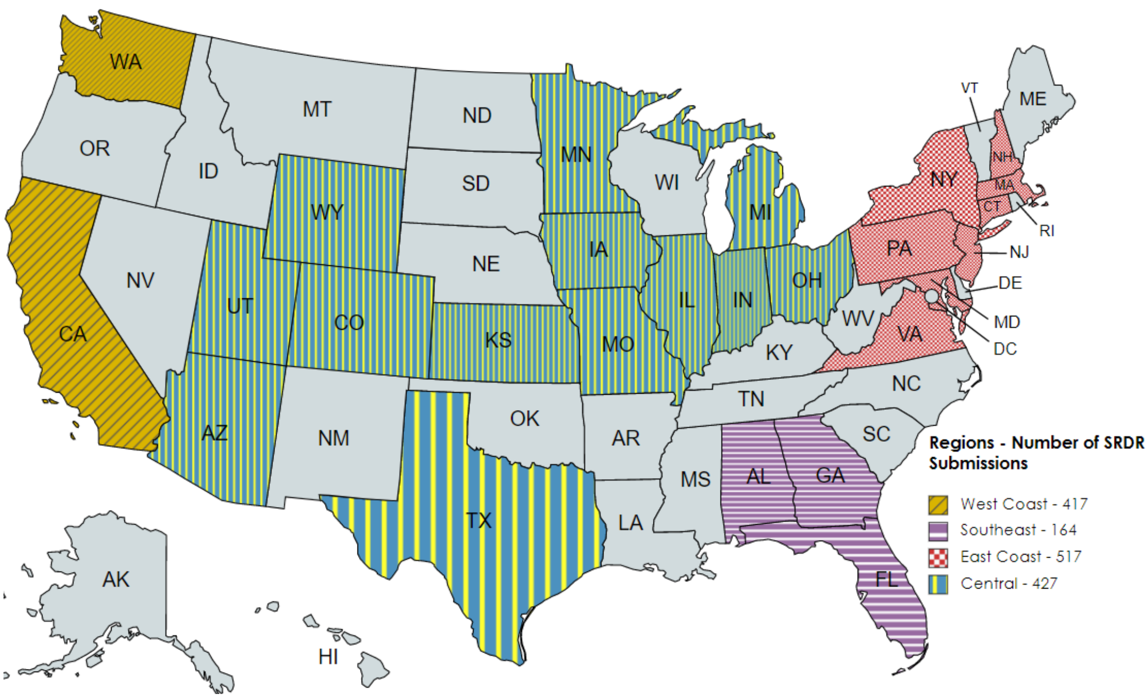


Figure 11: US Map Highlighting Regions for Analysis

Software Language Used

Software language should be considered important for anything software related as it has direct impacts on the availability of ready to use code, the number of people or firms capable of building, and the level of security available or necessary. This analysis will only consider the primary language of a project. The histogram in Figure 12 shows the distribution of the Primary Software Languages that are most popular in the SRDR and that will be used for analysis.

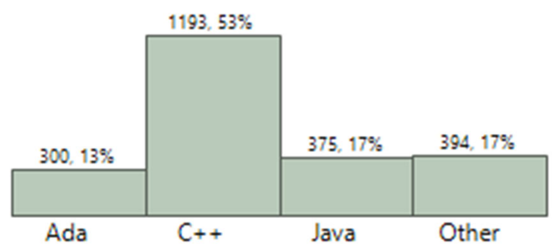


Figure 12: Distribution of Software Language Adjusted

Measured Attributes:

New Requirements

The SRDR supplies four columns in addition to volatility on requirements. Total Internal Requirements, New Internal Requirements, Total External Requirements, and New External Requirement. These are provided as a count by the contractor. To Normalize this data between projects the percentage of new requirements to total requirements were found for both internal and external requirements. For this analysis the aim is to find whether the new requirements impact the Volatility input. The cutoff for building the contingency table will be 50%. This value was chosen for the cutoff to decide if RV is assessed differently when over half of the requirements are new. The analysis will compare internal and external requirements separately, and against RV inputs and the growth factors. For the growth analysis tests, only information from the *Initial* submissions will be used.

Size by ESLOC and Total Hours

ESLOC is a calculation on the SRDR that is determined through other information that is provided by the contractor. ESLOC is often used to estimate the relative size of a project. For this analysis it will be split at the median, to make a binary categorical variable. This is done so the top half and bottom half can be assessed against the RV variables and growth factors. For the growth analysis the ESLOC of Initial Submissions will be used. As a second factor of size and a proxy for cost Total hours will also be used for analysis. Total Hours are found by the summation of all the phases and other development efforts. This will be split into top and bottom halves at the median for analysis.

Experience of Development Staff

The dataset provided multiple categories which describe the staff of the contractor's team that is completing the build. The key categories for this analysis are the categories about Experience level. For the experience inputs two categories will be formed, one accounting for highly experienced teams, and one for inexperienced teams. This aims to find if highly experienced and or inexperienced teams behave differently for giving RV Data and in performance. Experienced teams will be decided by teams which the Very High percentage plus the High percentage are greater than 50%. A team will be considered inexperienced if the percentage of entry level staff plus low experience is greater than 50%. For the growth analysis only values from the *Initial* submission will be used as this is the data that can be known at the time of *Initial* RV input.

Conducting the Contingency Analysis

Once all the RV inputs, growth measures, and project characteristics have been converted to binary variables contingency analysis using contingency tables, Pearson test and an Odds Ratio will be conducted. The test will first be performed between RV and growth, then software characteristics and RV, and lastly software characteristics and growth. For this exploratory contingency analysis, a Pearson Test, with an alpha of 0.10 will be used to identify significant relationships. When a significant relationship is determined an Odds Ratio will be calculated to determine the strength of the relationship. The Odds ratio is a measure of association. Odds ratios which are greater than one indicate a higher likelihood for the association while values less than one represent a reduced likelihood of the association. For the test between RV and the growth measures, the Odds Ratio will tell us if higher RV inputs are associated with higher measures of growth. More specifically a higher likeliness for greater than median growth. When testing RV and program attributes the Odds Ratio will tell us if certain attributes are associated with higher RV inputs. And when testing program attributes and growth measures the Odds Ratio will describe the association between the attribute and higher chance for greater than median growth. The software package of JMP a Statistical Discovery tool will be used to conduct all test. An example of a Mosaic Plot, Contingency Table, Pearson Test and Odds Ratio as displayed in JMP can be found in Figures 13 and 14.

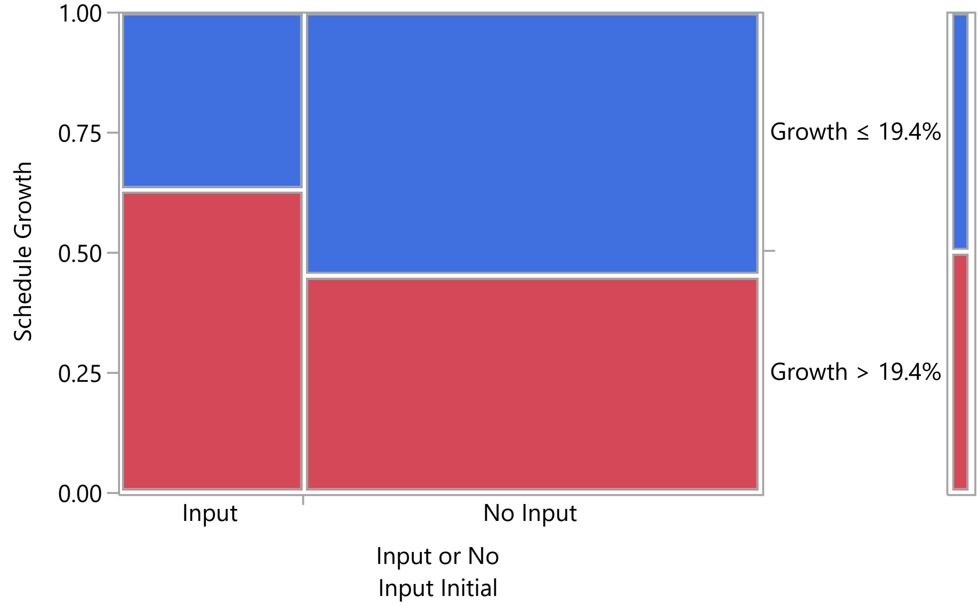


Figure 13: Mosaic Plot of RV Input on Initial Submissions and Schedule Growth

A Mosaic Plot is the graphical representation of the values found in the Contingency table in Figure 14. We can see from the "break" in horizontal white line that there is some association between providing RV input and Schedule Growth above 19.4%. The median value of the 402 projects that provided schedule information is 19.4%. The Pearson chi-squared test is used to test for independence. In this example the value of 0.0012* indicates that that there is a statistically significant dependence between the tested values. The Odds Ratio, calculated using the Contingency table, shows the strength of the association in the data set for providing Input and experiencing greater than 19.4% growth in schedule is 2.1. This was calculated by taking the count of the submissions that supplied an RV input and grew more than 19.4% (73), multiplied by those that provided no submission and grew less than or equal to 19.4% (157). This value was then divided by the count of submissions that supplied RV input and grew less than or equal to the median (43), multiplied by the count of submissions that provided no

RV input and grew more than the median ((73x157)/(43x129) = 2.066). This Odds Ratio is telling us that, “When RV input was received on *Initial* SRDR submissions, the project is 2.1 times more likely to fall in the upper distribution of growth by Schedule in Months.” For each set of theoretical independent and dependent variable relationships determined to be important to this study a Contingency Table, Pearson Test and for the chi squared values less than our alpha of 0.10 an Odds Ratio will be examined. For the analysis portion of the study, only the relationship being tested, and the Odds ratio will be displayed. In total 978 contingency tests will be conducted and analyzed for this exploration into the value of RV in the SRDR dataset.

Contingency Table

Input or No Input Initial	Count Total % Col % Row %	Schedule Growth Growth > 19.4%	 Growth ≤ 19.4%	Total
Input		73	43	116
		18.16	10.70	28.86
		36.14	21.50	
		62.93	37.07	
No Input		129	157	286
		32.09	39.05	71.14
		63.86	78.50	
		45.10	54.90	
Total		202	200	402
		50.25	49.75	

Test	ChiSquare	Prob>ChiSq
Pearson	10.490	0.0012*

Odds Ratio

Odds Ratio	Lower 95%	Upper 95%
2.066162	1.326762	3.217626

Figure 14: JMP Output for RV Input on Initial Submissions and Schedule Growth

IV. Analysis and Results

Description of the Data

Numerous instructions have been provided since this SRDR Compilations dataset was erected, and these instructions have influenced how we can evaluate the data. Four bar graphs will be used to assess the trends in reporting and to characterize the data used in the contingency analysis. The first two bar graphs are reflective of the groups of data that were analyzed for the growth analysis section. The latter two graphs Figures 17 and 18 are reflective of all the Initial and Final submissions on the SRDR. The datasets for the growth analysis are not perfect subsets of the larger set as some of the "*Initial*" submissions are in fact interim reports that were used to allow for more data points by CADE analysts. They are also not perfect reflections of what is used for our analysis as we removed submissions with impossible schedules, these lager bar charts are aimed at recognizing trends in submissions not the quality of each submission.

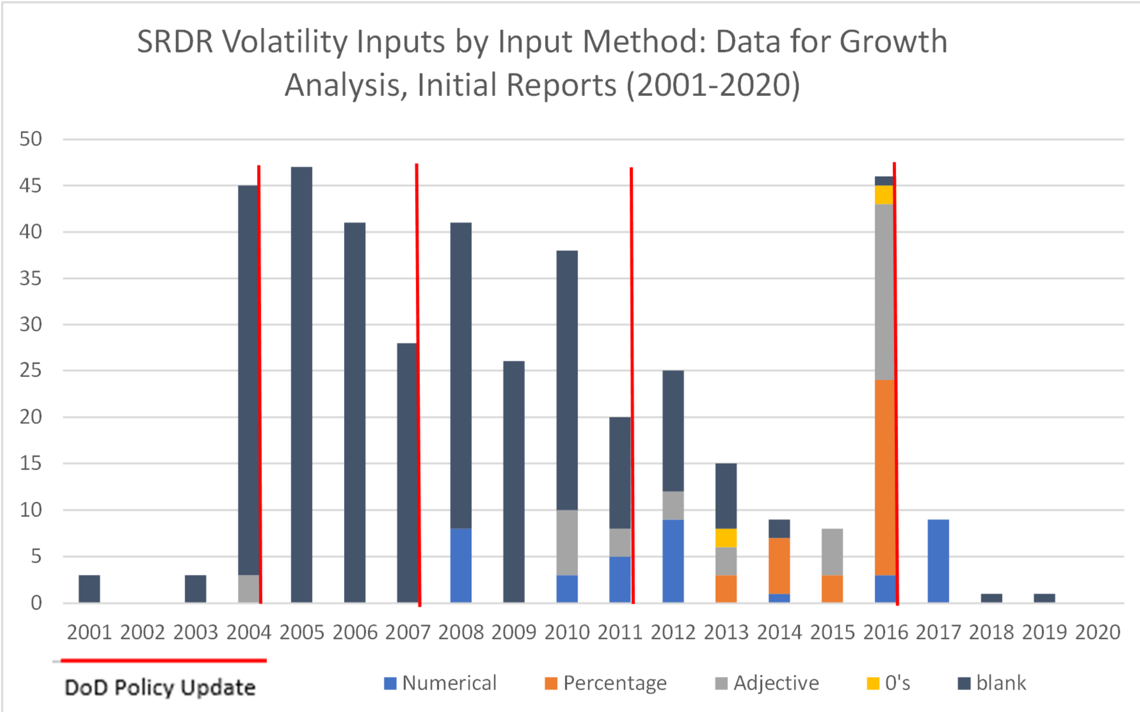


Figure 15: SRDR RV Inputs by Year, Method, with Policy Updates Identified: Initial Reporting used for Growth Analysis

This first bar chart is displaying the breakout of the 408 Initial submissions that are used in the growth analysis portion of the contingency analysis. For this bar chart, and each of those to follow, the y-axis is a count related to the RV inputs. Figure 15 shows that very few inputs were provided on Initial SRDR submissions until 2008. The lack of Initial submissions is likely due to 2004 DoD 5000.4M SRDR guidance which only called for submission on the final report. From 2007 onward, guidance requires RV on both reports. However, it does not appear that this guidance gained traction right away. The 2011, guidance also called for reporting on both reports yet we don't see a clear majority of submissions providing RV input until 2014. The year 2016 accounts for over one-third of all RV inputs on Initial reports that were provided. Although the total dataset covers a 20-year window, the analysis of Initial RV inputs to predict growth are primarily the more recent 7-year window of the data. Also, while we cannot but be enticed to compare their relative ability to predict growth, the largely derive from different years and significantly different N counts.

The second set of contingency tables will be conducted using Final RV measures, and the available data is displayed in Figure 16. As should be expected from the guidance, there are far fewer blanks in the data.

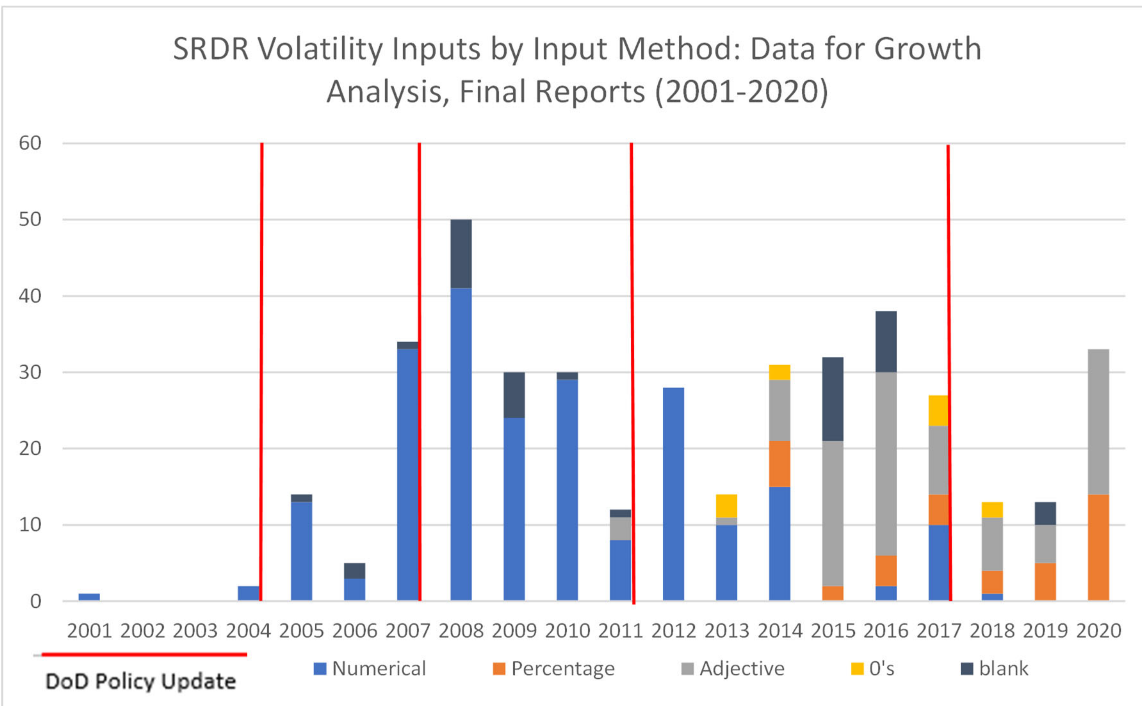


Figure 16: SRDR RV Inputs by Year, Method, with Policy Updates Identified: Final Reporting Used for Growth Analysis

Figure 16 shows us that for RV inputs on Final reports the numerical scheme was by far the dominant input scheme for a 10-year window from 2004-2014. This pattern is alarming, in that guidance directed an adjectival categorical scheme. Further alarming is that an adjectival system only starts to make a presence when the percentage scheme is directed. Since that guidance, both percentage and categorical dominate.

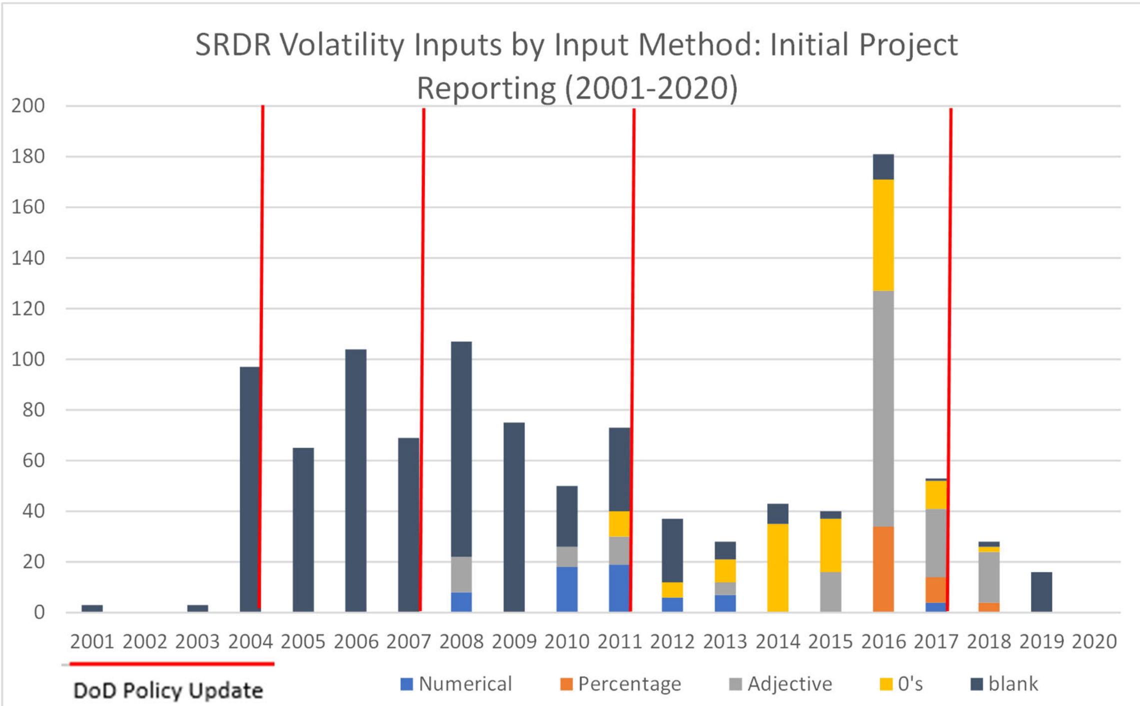


Figure 17: SRDR RV Inputs by Year, Method, with Policy Updates Identified: Initial Reporting

Figure 17 expands our view to the entirety of Initial submissions on the SRDR. The analysis of RV input and program attributes will be a subset of this set. The more recent years, particularly the post 2011 and 2017 guidance are of the most interest for two reasons. First, because again that is where most of the data comes from, and secondly, from 2011 onward the guidance clearly called for and had been calling for reporting on Initial submissions. The reporting of "0's" on Initial reports is nearly as popular as an input as any other scheme, a thorough explanation for this trend was not found. Most of the other trends mirror what is seen in the smaller growth analysis dataset, lack of early reporting and spike in 2016 submissions being the most notable.

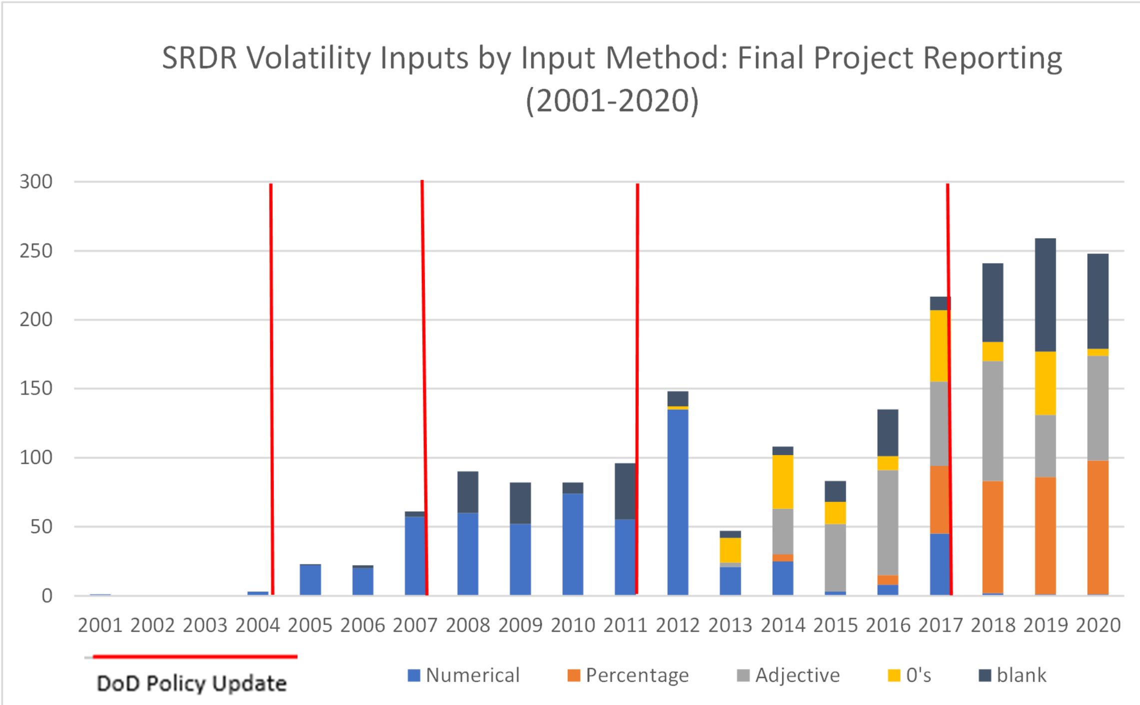


Figure 18: SRDR RV Inputs by Year, Method, with Policy Updates Identified: Final Reporting

For the *Final* submissions this chart identifies some trends previously found, we again see that 1-5 reporting is the most popular until around 2013. It is clear that the guidance may have encouraged reporting but the details within the guidance were either unknown or ignored by the submitters of the data. Either scenario is a poor reflection of the DoD data collection efforts. Again, in this set the later guidance is unable to change or persuade the RV reporting behaviors. It takes 6 years for any real traction on RV reporting by percentage and it again never wins out in popularity.

Proportion of Reponses for Growth Analysis

For the Conjoined system prescribed in chapter 3, we see that overall, inputs tend to be most commonly classified as Low by our system. Figures 19 and 20 show the responses for Initial and Final entries. It is important to note that in Final inputs, High values are submitted 3 times more often than in Initial inputs. Knowing that projects do tend to grow, a more frequent occurrence of a high post-mortem assessment makes sense.

The histograms in Figures 19 and 20 both show that Low is by far the most common area for entry for RV. It is important to note that in *Final* inputs High values are seen three times more often than in *Initial*. Knowing that projects do tend to grow this is expected. However, our SMEs should be expected to also have this knowledge and are perhaps including it in their assessments. A contingency table assessing the association between Medium and High inputs to higher than median for growth or likelihood for staffing change will be completed for analysis.

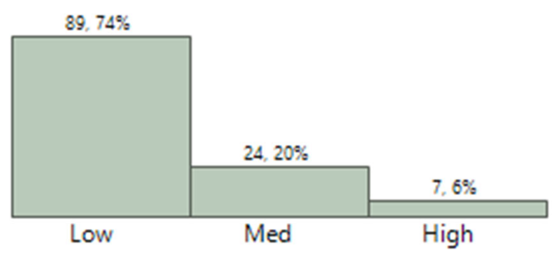


Figure 19: Distribution of Initial Conjoined Inputs for Growth Analysis

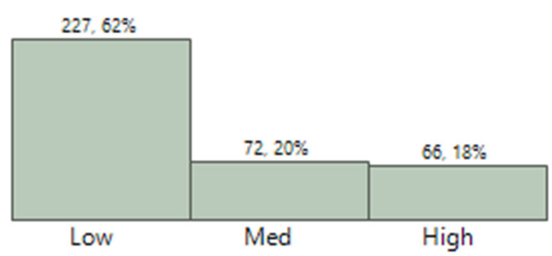


Figure 20: Distribution of Final Conjoined Inputs for Growth Analysis

Of the 408 completed projects, only 35 Initial reports provided inputs in the 1-5 method of reporting RV data. From Figure 15, we know that these inputs are mostly from 2008 to 2012. Figure 21 shows us that there were zero inputs for 5, and only one input for 1.

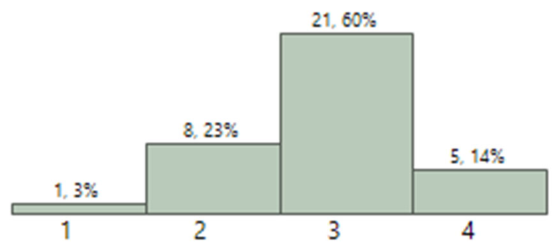


Figure 21: Distribution of Initial Numerical Inputs for Growth Analysis

Of the 408 completed projects, 221 *Final r*eports provided inputs in the 1-5 schema. Like the *Initial* report, 3 is the most common number reported, but the distribution is much more uniform in nature. Moreover, there are now many inputs of 5, again showing that managers do input higher values after the fact.

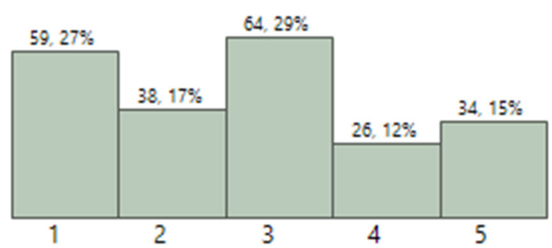


Figure 22: Distribution of Final Numerical Inputs for Growth Analysis

For the Initial submissions, 43 inputs were provided in an adjective scheme to describe them. Only the words Nominal, Low and Very Low were used. In our Conjoined scheme for analysis, are all classified as Low. To still conduct a test the Very Low entry will be classified as Low and the analysis will be conducted using Low vs Nominal. Nominal will be considered the higher of the two values for testing.

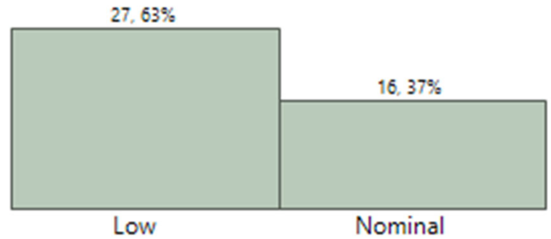


Figure 23: Distribution of Initial Adjective RV Inputs for Growth Analysis

There were 95 inputs provided in some form adjective scheme for the final reports. These 95 inputs were reclassified to Low, Medium, and High. Inputs of Low were left alone and inputs of Very Low, Nom, and Nominal were converted to Low. Low inputs made up 92% of all inputs, this is substantially different than what is found for the other categorical measure. This can be attributed to a number of potential factors. The first being that perhaps the conversion of reducing to the Low measure is overstated by considering nominal as Low. The second possibility would be the perhaps the psychological impact of naming a programs volatility by words has a bias towards low ratings where numbers may be less emotional. The analysis will conducted testing the High and Medium values to their likeliness towards greater than median growth.

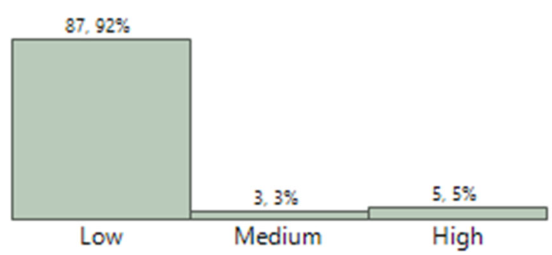
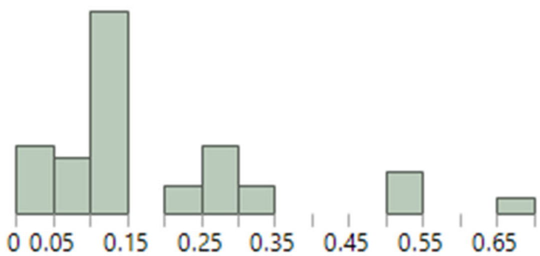


Figure 24: Distribution of Final Adjective RV Inputs for Growth Analysis

For percentage scheme, both *Initial* and *Final* inputs have distributions that are skewed to the right, with means are much higher than the median values. The upper three quartiles will be tested for association with higher levels of growth and program attributes. This is chosen to try and determine if the lowest quarter grows significantly different than the upper three fourths of the data. Our research in Chapter 2 lends to the intuition that a percentage method is likely to understate the real impact of RV, so if these more extreme low values do not indicate more significant relationships than other schemes that intuition may be supported.



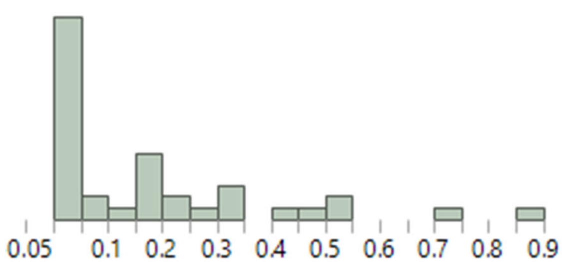
Quantiles

100.0%	maximum	0.68
99.5%		0.68
97.5%		0.68
90.0%		0.5
75.0%	quartile	0.25
50.0%	median	0.149
25.0%	quartile	0.075
10.0%		0.04
2.5%		0.02
0.5%		0.02
0.0%	minimum	0.02

Summary Statistics

Mean	0.1901622
Std Dev	0.1492698
Std Err Mean	0.0245398
Upper 95% Mean	0.2399312
Lower 95% Mean	0.1403931
N	37
Median	0.149

Figure 25: Distribution, Quantiles and Summary Stats for Inputs Entered as Percentages on Initial Submissions for Growth Analysis



Quantiles

100.0%	maximum	0.858
99.5%		0.858
97.5%		0.85505
90.0%		0.4959
75.0%	quartile	0.255
50.0%	median	0.0565
25.0%	quartile	0.0237
10.0%		0.0237
2.5%		0.00415
0.5%		0.004
0.0%	minimum	0.004

Summary Statistics

Mean	0.1732075
Std Dev	0.2070112
Std Err Mean	0.0327314
Upper 95% Mean	0.2394129
Lower 95% Mean	0.1070021
N	40
Median	0.0565

Figure 26: Distribution, Quantiles and Summary Stats for Inputs Entered as Percentage on Final Submissions for Growth Analysis

Proportion of Responses for Program Attributes

For the analysis with RV as the theoretical dependent variable we are assessing software project characteristics and how they contribute to or may affect what is found for RV inputs. Because pairs are not necessary the number of inputs is much larger and the trends in RV inputs need to be looked into for the new dataset. The larger dataset will create a stronger feeling of what the global trends of RV may be.

Looking at whether an input into volatility column was provide will highlight what variables contribute to the likelihood of input. This analysis will aim to identify the biggest contributors and offenders by analyzing RV inputs compared to items like Service, Contractors, Location, Development Style and the other previously identified characteristics. For this variable, inputs recorded as a 0 were counted as inputs where those received as “N/A” or “Unknown” were considered the same as blanks.

Histograms of the conjoined inputs are displayed in Figures 27 and 28. With these histograms we can see that Low is the most often reported measure. Seeing that Low is by far the most common entry for both *Initial* and *Final* Inputs of all entries, analysis will be geared toward differentiating between Low entries and all others.

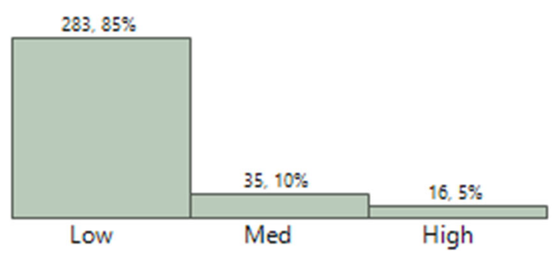


Figure 27: Distribution and Frequencies of RV Initial Inputs Conjoined to Low, Med, High

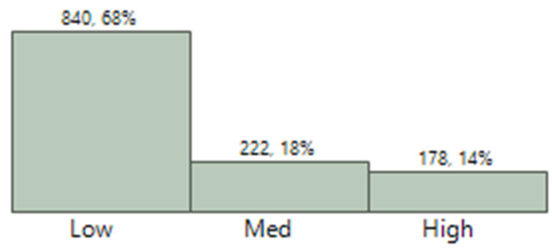


Figure 28: Distribution and Frequencies of RV Final Inputs Conjoined to Low, Med, High

The Numerical scheme for RV input has been a common and preferred method since the SRDR has been collecting data. However Figure 30 shows us that there were far more inputs in this scheme for Final submissions when compared to Initial. Again, it can be identified that *Final* reports contain much high frequencies of the higher RV inputs, this may be indication that estimators are typically under assessing RV. The testing will focus on distinguishing if 1 is reported differently between program elements.

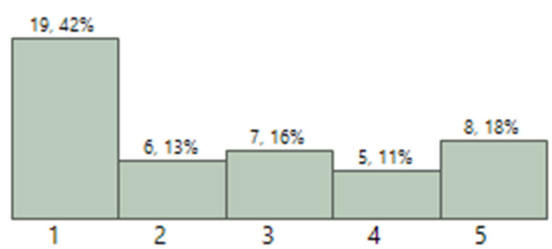


Figure 29: Distribution and Frequencies of RV Initial Inputs Entered as 1-5

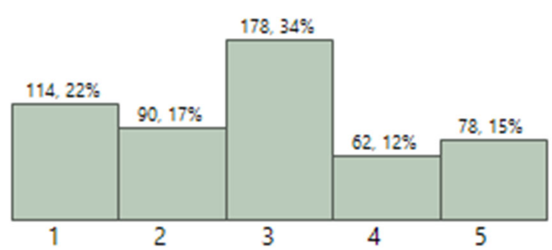


Figure 30: Distribution and Frequencies of RV Final Inputs Entered as 1-5

As stated in Chapter 3 the Adjective inputs were classified as Low, Medium, or High in the same fashion as the Conjoined method. There are many inputs on the SRDR that are input as Nominal, these will be converted to Low. On the SRDR, in the volatility column, some inputs have notes that identify the scheme used for the input, these are helpful for making the determination of where an input should fall. The distributions in Figures 31 and 32 show that Low vs All will be the best choice for testing associations between RV reporting and program characteristics.

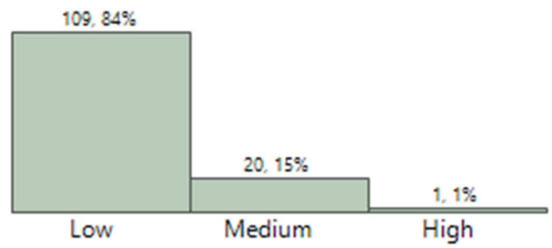


Figure 31: Distribution and Frequencies of RV Initial Adjective Inputs Classified as Low, Med, High

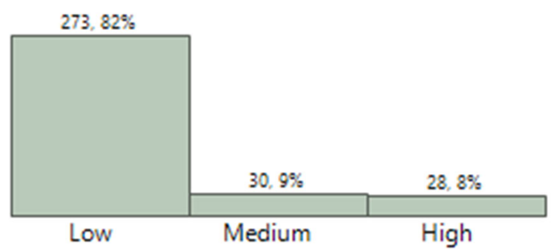
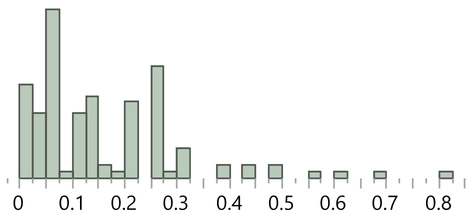


Figure 32: Distribution and Frequencies of RV Final Adjective Inputs Classified Low, Med, High

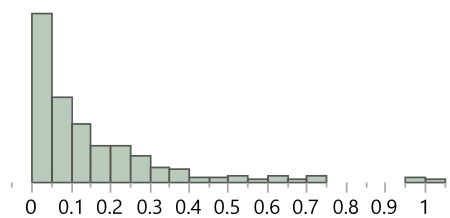
The Percentage scheme include many of the most recent inputs and is the current system that is requested and needed from contractors for input. To analyze this column the distribution and descriptive stats for all percentage can be seen in Figures 36 and 37. Based on the distribution shape and prior finding a method will be chosen to split the data into two categories. For closer analysis at the lower inputs the 1st Quartile has been chosen to split the data for both the *Initial* and *Final* submission types.



Quantiles		
100.0%	maximum	0.8
99.5%		0.8
97.5%		0.5895
90.0%		0.3
75.0%	quartile	0.25
50.0%	median	0.11
25.0%	quartile	0.05
10.0%		0.02
2.5%		0.01
0.5%		0.01
0.0%	minimum	0.01

Summary Statistics	
Mean	0.1536541
Std Dev	0.1425376
Std Err Mean	0.0123596
Upper 95% Mean	0.1781026
Lower 95% Mean	0.1292057
N	133

Figure 33: Distribution and Descriptive Statistics of RV Initial Inputs Entered as Percentages



Quantiles		
100.0%	maximum	1
99.5%		0.98125
97.5%		0.6
90.0%		0.315
75.0%	quartile	0.198
50.0%	median	0.07
25.0%	quartile	0.0237
10.0%		0.02
2.5%		0.01
0.5%		0.003625
0.0%	minimum	0.003

Summary Statistics	
Mean	0.1345528
Std Dev	0.1635759
Std Err Mean	0.0090875
Upper 95% Mean	0.152431
Lower 95% Mean	0.1166745
N	324

Figure 34: Distribution and Descriptive Statistics of RV Final Inputs Entered as Percentages

Associations Between RV Schemes and Program Growth

The key purpose of measuring and reporting volatility is to identify potential cost growth. Therefore, our first test aims to see if RV measures relate to measures of growth. There are two views in this section. The first tests the *Initial* RV inputs to eventual growth. These results show the ability of RV schemes to be early indicators, or predictors of growth or ultimately fulfill its mediating role between the endogenous program attributes and program growth. The next look relates the *Final* RV inputs to growth. This is the post-mortem look. These measures should be more broadly and convincingly associated with growth if, in fact, contractors are earnestly fulfilling their responsibility of inputting RV. They also reveal which aspects of growth may figure into the manager's conceptual measure of RV. Such added associations may indicate more conceptual elements of RV that the Monday morning quarterback considers but which may not be easy to define in advance.

The tables below display the relationships of the many contingency tests conducted. The contingency tests split both the independent and dependent variables into binary forms, removing continuous values, and seeking looser associations. Each cell shows the Odds Ratio of the association of the intersection of the column variable and the row variable. The columns with the light blue headers represent the different constructs of our independent variable, RV. As throughout, there are multiple RV schemes, therefore there are multiple columns here. The first table is a simple look at whether input was provided or not. For this all inputs are valued the same, as to say a 0, yellow, or 95% would all simply be considered an input. For the following tables the actual values of RV inputs will be considered. They are all set up so that the odds ratio depicted will display the association between the higher end of RV reporting and greater than median growth or change for the Staffing Mix measure. The columns, in order, are: Conjoined, Adjective, Numerical and Percentage.

The yellow rows on the left-hand side represent the dependent variables. Each of these variables were measured as percent growth from *Initial* estimate to *Final* input, and then split into a binary variable based on the median. They show the high end, such that relationship of *high* independent variable to *high* dependent variable is revealed in each cell. The variables are the percent growth in ESLOC, in months of schedule, in Peak Head Count, in Total Hours (which can be independent of schedule length), in Phase length for each of the six phases, and in the measure of Internal Requirements. The last column is a simple yes or no indication of Staffing Mix Change.

Odds ratios were only recorded for statistically significant relationships based on the Pearson Test. We used an alpha value of 0.10 to determine significance. The number in a cell is the Odds Ratio of the independent variable (RV input) landing in the upper distribution of the dependent (growth measure). In extreme relationships of all items relating all or none relating, the Pearson test cannot technically be conducted. You can consider that 9 of 9 would conceptually represent an Odds Ratio of infinity, while a "0 of 9" would represent an Odds Ratio of "0." But since it falls outside the technical parameters of the test, we have simply written "X of X" or "0 of Y." "None" in a cell shows there was no statistical relationship between the RV value and the growth measure. "N/A" shows there was not enough data (N<20) to determine if a significant relationship was present.

In each case, the tables are presented in the following order, first the table displays the Odds Ratios for Initial and Final reports simply on input and the relation to growth, in the second table, the Initial RV inputs and the growth measures will be displayed and analyzed, followed by the *Final* RV inputs and growth measures. These tables are set up and should be read in the same manner but their analysis aims to find differing conclusions.

Please reference the examples below for reading the tables:

Ex 1) Growth Measure = Schedule in months, Odds Ratio = 2.1, RV value = Input

This cell should be read as saying, **"When RV input is received, the project is 2.1 times more likely to experience greater than median growth in Schedule."**

EX 2) Growth Measure = Internal Requirements, Odds Ratio = 2.5, RV Value = Conjoined (Med, High)
This cell should be read as saying, **"When all RV inputs have been conjoined to one scheme, the projects associated with the conjoined "Medium or High" inputs, were found to be .4 times as likely to have greater then median growth in Internal Requirements."**

Table 10: Odds Ratios for Submissions that provided RV Input to Experience Higher than Median Growth or Change

The Odds Ratios in the table represent the likelihood that submissions which provided RV input would fall in the upper distribution of the growth measure on the left-hand column.		
Percent Change split by median	**Input Received on Initial Submission**	**Input Received on Final Submission**
Equivalent Source Lines of Code (ESLOC)	None	None
Schedule in Months	**2.1**	**2.4**
Peak Head Count	**1.6**	**2.1**
Total Hours	None	None
Phase 1: Requirements Analysis	**0.6**	None
Phase 2: Architecture & Design	None	None
Phase 3: Coding & Testing	**1.5**	None
Phase 4: Software and System Integration	None	None
Phase 5: Qualification Testing	**0.5**	None
Phase 6: Development Test & Evaluation	None	None
Internal Requirements	**1.7**	**6**
Staffing Mix Change: No Change	**1.7**	**1.8**

The first thing that is apparent is that the Initial submissions have many more significant relationships than for the Final submissions. Secondly, the general trend for both Initial and Final is towards higher than median growth. In Initial reporting that relationship is a good sign that managers properly sensed that the project necessitated a volatility discloser. For the Final reporting perhaps it is of a similar indication, but for a after completion disclosure rather than predictive.

Table 11: Odds Ratios for the Association Between the Higher RV Inputs and Higher than Median Growth or Change for *Initial* Submissions

The Odds Ratios in the table represent the likelihood of the RV value in the blue cells to fall in the upper distribution of the growth measure in the left-hand column.				
Percent Change split by median	**Conjoined RV (Med & High)**	**Adjective (Nominal)**	**Numerical (3,4,5)**	**Percentage (Upper 3 Quartiles)**
Equivalent Source Lines of Code (ESLOC)	None	None	None	**10.5**
Schedule in Months	None	None	**--**	None
Peak Head Count	None	**8.6**	None	None
Total Hours	None	None	None	None
Phase 1: Requirements Analysis	None	None	**0.1**	None
Phase 2: Architecture & Design	None	None	None	N/A
Phase 3: Coding & Testing	None	None	None	None
Phase 4: Software and System Integration	None	None	None	N/A
Phase 5: Qualification Testing	None	N/A	N/A	None
Phase 6: Development Test & Evaluation	**2 of 2**	**10**	N/A	N/A
Internal Requirements	**0.4**	None	None	**8.8**
Staffing Mix Change: Change	None	None	**15.8**	**16.1**
(--) Spurious relationship odds ratio of 0				

Overall, the finding for RV as a predictor for growth are limited. The strongest results are from the Percentage scheme and Adjective scheme. Interestingly the only strong growth measure to be associated in more than one scheme is Staffing Mix Change, the finding was that for the higher RV reporting in the Numerical or Percentage scheme were determined to be far more likely to experience a change in there Staffing Mix. Percentage found strong predictive associations between the Internal Requirements and

ESLOC. For the Adjective scheme Nominal inputs were generally found to grow more than Low inputs but these levels of growth do seem to be uncharacteristic for the typical Nominal description of "small non-critical redirections." This finding perhaps warrants a deeper dive into how to best treat Nominal inputs relative to Low or High inputs. The 5-item numerical scale had 3 associations out of 12, but 2 of them were in the wrong direction, and vastly so. Early use of the 1 and 2 in the numerical scheme did not prove to correlate to a likelihood to experience lower growth.

Table 12: Odds Ratios for the Association Between the Higher RV Inputs and Higher than Median Growth or Change for *Final* Submissions

The Odds Ratios in the table represent the likelihood of the RV value in the blue cells to fall in the upper distribution of the growth measure in the left-hand column.				
Percent Change split by median	**Conjoined (Med & High)**	**Adjective (Med & High)**	**Numerical (3,4,5)**	**Percentage (Upper 3 Quartiles)**
Equivalent Source Lines of Code (ESLOC)	None	**8 of 8**	None	**0.03**
Schedule in Months	**0.5**	**8 of 8**	**0.6**	**5.3**
Peak Head Count	**2.3**	**7.5**	**3.8**	None
Total Hours	**1.8**	**6.8**	**2.3**	None
Phase 1: Requirements Analysis	**1.9**	None	**2.2**	**10.5**
Phase 2: Architecture & Design	**1.7**	**4.0**	None	None
Phase 3: Coding & Testing	**1.6**	None	**2.2**	**0.2**
Phase 4: Software and System Integration	**1.8**	None	**2.0**	N/A
Phase 5: Qualification Testing	None	None	None	None
Phase 6: Development Test & Evaluation	**1.9**	**2 of 2**	None	N/A
Internal Requirements	**1.7**	None	**1.8**	**0.2**
Staffing Mix Change: Change	None	**0.2**	**1.8**	**--**
(--) Spurious relationship odds ratio of 0				

The first thing to notice is that there are substantially more significant relationships than seen in Table 11, and that across all RV schemes the trends are mostly in the correct direction, that is Higher RV assessment meaning a likelihood towards Higher levels of growth. This is important for affirming that RV is in fact positively correlated with growth.

Perhaps the most substantial finding in this table is that the conjoined RV scheme had the most significant relationships. The Med & High category for the conjoined inputs were significant in 9 of the 12 growth measures and in the correct direction for 8 of those 9. This is important because the ability to accurately normalize inputs would be useful for future RV research. The Adjective scheme shows us two very strong relationships, low inputs in this category are 6.8 and 7.5 times more likely to be in the upper half of growth in Total Hours and Peak Head Counts respectively. These are two of the more common proxies for costs. Although not as many significant relationships are found as the other schemes these findings are substantial in the relationship to potential cost growth. The Numerical or Conjoined inputs are not correlated in the correct direction, or at least the direction the literature or our intuition would have us expect to see for Schedule Growth. The higher Percentage inputs followed intuition for Schedule Growth but were seemingly backwards for Phase 3, ESLOC and Internal Requirements.

Lastly, we examine our growth measure to get the feel for what the SMEs likely find most important to capture in RV inputs. Peak Head Count, Total Hours, and Requirements Analysis Phase were all showed 3 significant relationships with the RV schemes, and all were correlated in the appropriate direction. Schedule was the only measured growth factor that was found to be backwards in more than one scheme. This builds the case that perhaps RV reporting in the DoD dataset is more cost focused opposed to schedule. It is evident through the lack of stronger associations, Internal Requirements have not been of great importance for determining RV inputs.

Program Analysis

This section will provide a combination of what is traditionally considered a deeper dive, as well as further associations of the whole model. It looks at who is providing RV inputs, and under what circumstances. And then it looks at who is experiencing the various kinds of growth we have already looked at, and under what circumstances. The objective is to try to determine if RV is doing its intended job of translating volatile circumstance into a global RV measure which is then predictive. Figure 35 (block and arrow diagram) repeats the theoretical relationships of RV, as discussed in chapter 1. In the last section, we showed that RV does have some relationship to growth measures. It can predict growth, and as a post-mortem, it seems personnel are quite capable of aligning it to growth. Of that figure, then we find partial support for the arrow labeled A. In this section, we will test the relationships shown by arrows B and C. B is testing whether underlying knowable aspects (agents, circumstances, characteristics) relate to RV, and C is testing whether they also relate to these growth measures. In more formal models, these relationships as well as the previous one would be tested simultaneously, to determine if RV "mediates" or captures the relationship between these the aspects of the project and its eventual growth. Here, owing to the poor quality of the data, we can do piecemeal associations, and thus only suggest RV as the mediator.

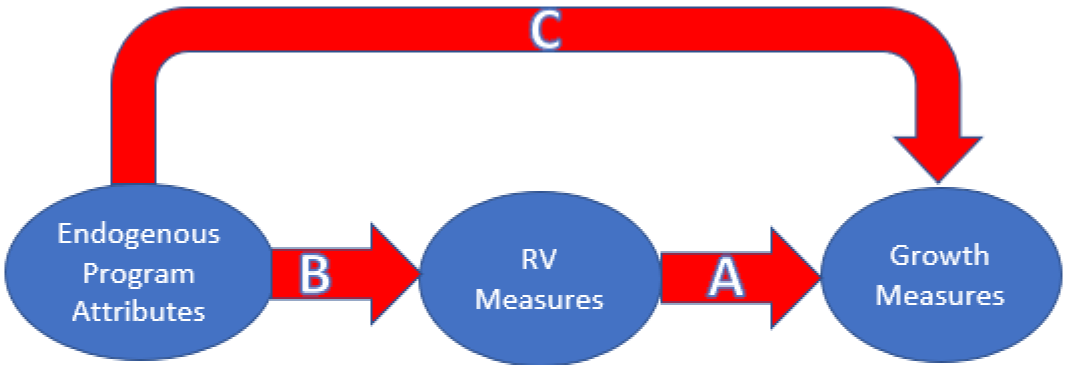


Figure 35: Mediation Model for Program Attributes, RV measures, and Growth Measures

Going forward each subsection breaks down the data into different categories. The first program attributes are definite by nature and are known at the beginning of a project. Next, a few estimated measures will be evaluated. There are nine definite categories to analyze, they are Service, Contactor, Operating Environment, Application Domain, Super Domain, Development Process, New or Updated Software, Worksite Location, and Primary Software Language. For the estimated aspects of the programs, we will analyze New requirements, two aspects of size (ESLOC and Total Hours), and Personnel experience level. Each subsection will have three tables. The first two tables will treat various agents, circumstances, and characteristics as the independent variables that might relate to RV. The first of these uses *Initial* RV inputs. The second, the *Final* RV inputs. The benefit of two looks is that we can view RV inputs that are to be predictive by nature and then reflective. The third table will then relate these same agents, circumstances, and characteristics to the previous study's measure of growth. The idea is to slowly bring into focus where RV and where there is growth. The identification of both of those relationships serves as support for the potential for RV, in its current usage, to predict cost growth and perform its role as a mediating variable. Each section will contain a brief description of the trends and relationships present.

The first two tables of each 3-table set, display the association between the Higher RV values and various software characteristics using Odds Ratios. The table rows down the left-hand side (light blue) represent the dependent variable, RV Input vs No Input, Conjoined RV, Numerical, Percentage, and Adjective, and the orange column headers are the independent variables. Odds Ratios will display the strength of the significant relationships that are present. The tables follow the same general rules as Tables 10-12.

Please reference these examples for reading the first two tables in each section:

Ex 1) RV value = Input, IV (orange cell) = Air Force, Odds Ratio =1.3

This cell should be read as saying, **"Air Force projects are 1.3 times more likely to provide RV input on initial submissions than the other services."**

Ex 2) RV value = Cat. by Words-Low, IV (orange cell) = Air Force, Odds Ratio = 0 of 41

This cell can be read as saying, **"All 41 Air Force projects submitted on initial reports in the Adjectival scheme were Low."**

EX 3) RV value = Numerical (2,3,4,5), IV (orange cell) = Navy, Odds Ratio = 50

This cell should be read as saying, "**Navy projects reported in the Numerical scheme are 50 times more likely to provide RV input of greater than 1 on initial submissions compared to the other services."**

For the third table in each section the relationship between the software characteristic and the various items chosen to assess growth in prior sections. This analysis is used as a way to evaluate if the RV inputs follow the growth trends of the characteristic being examined. The growth analysis tables display the relationship between the software characteristics and various measures of growth using an Odds Ratio to show the strength of significant relationships. The table columns (orange) represent the independent variable (software characteristic) and the yellow rows on the left-hand side represent the dependent variables (growth measures).

Please reference the examples below for reading the software characteristic growth tables:

Ex 1) Growth Measure = ESLOC, Characteristic = AF, Odds Ratio = 0.5

This cell should be read as saying, **"Air Force projects are 0.5 as likely to end up in the upper distribution of all services ESLOC growth."**

Ex 2) Growth Measure = Staffing Mix Change, Characteristic = Army, Odds Ratio = 2.3

This cell should be read as saying, **"Army projects are 2.3 times more likely to see a change in staffing mix compared to other service's projects."**

Service

Table 13: Odds Ratio for RV Inputs by Service *Initial*

The Odds Ratios in the table represent the greater likelihood that the Service branch relative to other branches employed higher RV.			
RV Measurement	**AF**	**Navy**	**Army**
Input	**1.3**	**0.7**	**0.6**
Conjoined (Med & High)	**0.05**	**5.0**	**0.5**
Numerical (2,3,4,5)	None	**50.0**	**0.03**
Percentage (Upper 3 Quartiles)	None	**0.5**	**3.3**
Adjective (Med & High)	**0 of 41**	**0.2**	**4.8**

Table 14: Odds Ratio for RV Inputs by Service *Final*

The Odds Ratios in the table represent the greater likelihood that the Service branch relative to other branches employed higher RV.			
RV Measurement	**AF**	**Navy**	**Army**
Input	**0.6**	**1.6**	None
Conjoined (Med & High)	**0.6**	**1.7**	None
Numerical (2,3,4,5)	None	**0.6**	**2.0**

Percentage (Upper 3 Quartiles)	**5.0**	**1.7**	**0.3**
Adjective (Med & High)	**0.2**	**3.3**	None

Air Force Projects have been found to be more likely than the other services to provide RV inputs for *Initial* submission. On *Initial* submissions Navy projects typically report higher levels of RV. RV inputs on final submissions are provided more often for all services. These trend in a similar fashion as the *Initial* inputs, that is Naval projects tend to report higher levels of RV than the AF and Army projects. These finding are substantiated by the finding in the growth table below. Naval projects experience the higher tendencies toward change. While AF projects trend towards lower levels of change.

Table 15: Odds Ratio for Getting Higher than Median Value of Growth by Service

The Odds Ratios in the table represent the greater likelihood that the specific Service branch relative to other branches experiences above median growth for the given measure			
Percent Change split by median	**AF**	**Navy**	**Army**
Equivalent Source Lines of Code (ESLOC)	**0.5**	**1.5**	None
Schedule in Months	None	**1.9**	**0.5**
Peak Head Count	**0.6**	None	None
Total Hours	None	None	None
Phase 1: Requirements Analysis	**0.5**	**1.7**	None
Phase 2: Architecture & Design	None	None	None
Phase 3: Coding & Testing	None	None	None
Phase 4: Software and System Integration	None	None	None
Phase 5: Qualification Testing	None	None	None
Phase 6: Development Test & Evaluation	None	None	None
Internal Requirements	None	None	None
Staffing Mix Change: Change	**0.6**	None	**2.3**

Contractor

Table 16: Odds Ratio for Low RV Inputs by Contractor *Initial*

The Odds Ratios in the table represent the greater likelihood that the contractor relative to other contractors employed higher RV.							
RV Measurement	A	B	C	D	E	F	G
Input	0.4	None	None	0.6	3.0	1.7	None
Conjoined (Med & High)	None	None	0 of 20	None	0 of 35	0.4	3.3
Numerical (2,3,4,5)	N/A	N/A	None	None	None	0.2	None
Percentage (Upper 3 Quartiles)	None	10 of 10	6 of 6	0.2	None	None	14 of 14
Adjective (Med & High)	None	None	None	None	None	0 of 15	10.0

Table 17: Odds Ratio for Low RV Inputs by Contractor *Final*

The Odds Ratios in the table represent the greater likelihood that the contractor relative to other contractors employed higher RV.							
RV Measurement	A	B	C	D	E	F	G
Input	None	1.5	None	None	0.6	0.5	None
Conjoined (Med & High)	3.3	0.7	0.6	2.5	None	0.4	0.3
Numerical (2,3,4,5)	32 of 32	2.5	None	2.0	None	0.5	None
Percentage (Upper 3 Quartiles)	None	10.0	10.0	10.0	3.3	0.5	0.1
Adjective (Med & High)	5.0	0 of 56	None	10.0	None	0.3	2.5

For RV submissions on *Initial* reports E, and F are better than other contractors at providing an input, A and D are statistically less likely to provide inputs compared to the other contractors. For *Final* reports E and F are found to be less likely to provide RV input, perhaps this is indication that they more closely follow the CSDR reporting guidelines, as the RV reporting is only called for an initial or interim report. A and E see statistically higher growth in schedule, perhaps this is a measure they consider for their final RV submissions. D sees higher growth in Peak Head Count and in Phase 4, but lower in ESLOC and Schedule, it is possible that Phase 4 and increasing staff contribute to their RV ratings. Lockheed only sees statically high change in Internal Requirements but still report lower level of RV. If internal requirements are considered for their RV inputs, they are understating its value. G reports low and grows less in Phase one and schedule, these may likely be drivers for their RV input Level.

Table 18: Odds Ratio for Getting Higher than Median Value of Growth by Contractors

The Odds Ratios in the table represent the greater likelihood that the specific contractor relative to other contractors experiences above median growth for the given measure.							
Percent Change split by median	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Equivalent Source Lines of Code (ESLOC)	None	None	None	**0.6**	None	None	**1.7**
Schedule in Months	**2.7**	None	**0.5**	**0.5**	**3.4**	None	**0.4**
Peak Head Count	None	None	None	**1.8**	None	None	None
Total Hours	None	None	**2.1**	None	**0.5**	None	None
Phase 1: Requirements Analysis	None	None	**2.1**	None	None	None	**0.5**
Phase 2: Architecture & Design	None	None	None	None	None	None	None
Phase 3: Coding & Testing	None	None	None	None	None	None	None
Phase 4: Software and System Integration	None	None	**0.3**	**2.2**	None	None	None
Phase 5: Qualification Testing	None	None	None	None	None	None	None
Phase 6: Development Test & Evaluation	None	None	None	None	None	None	None
Internal Requirements	**0.5**	None	None	None	**2.1**	**1.8**	None
Staffing Mix Change: Change	None	**0.4**	None	None	None	None	**2.1**

Operating Environment

Table 19: Odds Ratio for Low RV Inputs by Operating Environment *Initial*

The Odds Ratios in the table represent the greater likelihood of the operating environment relative to other operating environments employed higher RV.

RV Measurement	Air Vehicle, Manned	Surface Fixed, Manned	Air Vehicle, Unmanned	Surface Mobile, Manned
Input	None	**0.7**	**1.5**	None
Conjoined (Med & High)	None	**0.4**	None	**2.5**
Numerical (2,3,4,5)	None	**0.2**	**0 of 7**	**5 of 5**
Percentage (Upper 3 Quartiles)	None	None	None	None
Adjective (Med & High)	None	None	**5.0**	None

Table 20: Odds Ratio for Low RV Inputs by Operating Environment *Final*

The Odds Ratios in the table represent the greater likelihood of the operating environment relative to other operating environments employed higher RV.				
RV Measurement	**Air Vehicle, Manned**	**Surface Fixed, Manned**	**Air Vehicle, Unmanned**	**Surface Mobile, Manned**
Input	**1.4**	**0.4**	None	None
Conjoined (Med & High)	None	**0.7**	None	**0.7**
Numerical (2,3,4,5)	None	None	None	None
Percentage (Upper 3 Quartiles)	**2.0**	None	**2.5**	**0.5**
Adjective (Med & High)	**0.4**	None	**3.3**	**0.3**

Operating Environments describe the environment of the hardware which the software will operate on. We can see that for *Initial* RV reporting less input is provided for SFM, and more is provided for AVU. Surface operating environments appear to experience less RV and mobile may experience more than fixed. Air Vehicles tend report higher RV and unmanned higher than manned. The growth trends below substantiate the RV reporting seen on final reporting. Software operating in the air has higher growth trends than on the ground, and unmanned trends higher than manned. This aligns with our intuitive since of complexity and reinforces what the literature exclaimed about complexity leading to higher volatility and RVs use in models to help capture complexity.

Table 21: Odds Ratio for Getting Higher than Median Value of Growth by Operating Environment

The Odds Ratios in the table represent the greater likelihood that the specific Operating Environment relative to other Operating Environments experiences above median growth for the given measure.				
Percent Change split by median	**Air Vehicle, Manned**	**Surface Fixed, Manned**	**Air Vehicle, Unmanned**	**Surface Mobile, Manned**
Equivalent Source Lines of Code (ESLOC)	None	None	**1.7**	None
Schedule in Months	None	None	**2.5**	None
Peak Head Count	None	None	None	None
Total Hours	None	None	None	None
Phase 1: Requirements Analysis	None	**0.4**	None	**0.2**
Phase 2: Architecture & Design	None	None	None	None
Phase 3: Coding & Testing	None	None	None	None
Phase 4: Software and System Integration	None	None	**1.9**	None
Phase 5: Qualification Testing	None	**0.6**	None	None
Phase 6: Development Test & Evaluation	**0.5**	None	**2.2**	None
Internal Requirements	**2.0**	None	None	None
Staffing Mix Change: Change	None	None	None	None

Application Domain

Table 22: Odds Ratio for Low RV Inputs by Application Domain *Initial*

The Odds Ratios in the table represent the greater likelihood that the contractor relative to other contractors employed higher RV.			
RV Measurement	**Command & Control**	**Other Real Time Embedded**	**Vehicle Control**
Input	**0.7**	None	**1.5**
Conjoined (Med & High)	None	None	**3.3**
Numerical (2,3,4,5)	None	**5 of 5**	**10.0**
Percentage (Upper 3 Quartiles)	None	None	None
Adjective (Med & High)	None	None	**5.0**

Table 23: Odds Ratio for Low RV Inputs by Application Domain *Final*

The Odds Ratios in the table represent the greater likelihood that the contractor relative to other contractors employed higher RV.

RV Measurement	Command & Control	Other Real Time Embedded	Vehicle Control
Input	**2.1**	None	**3.0**
Conjoined (Med & High)	None	None	**1.4**
Numerical (2,3,4,5)	None	None	None
Percentage (Upper 3 Quartiles)	**2.0**	None	**0.4**
Adjective (Med & High)	None	None	**5.0**

Application domain describes the "what" the software being developed is intended to do. Vehicle Control (VC) projects are identified as being more likely than other types of projects to provide an RV input. When VC is reported it tends to be for higher levels of RV, this is in line with the fact that VC projects are more volatile in terms of growth. Command and Control and Real Time application types did not show particular RV reporting or growth associations.

Table 24: Odds Ratio for Getting Higher than Median Value of Growth by Application Domain

The Odds Ratios in the table represent the greater likelihood that the specific Application Domain relative to other domains experiences above median growth for the given measure.			
Percent Change split by median	**Command & Control**	**Other Real Time Embedded**	**Vehicle Control**
Equivalent Source Lines of Code (ESLOC)	None	None	None
Schedule in Months	None	**0.6**	None
Peak Head Count	None	None	None
Total Hours	None	None	**2.8**
Phase 1: Requirements Analysis	None	**1.7**	**4.2**
Phase 2: Architecture & Design	None	None	**3.3**
Phase 3: Coding & Testing	None	None	**2.2**
Phase 4: Software and System Integration	None	None	None
Phase 5: Qualification Testing	None	None	**2.0**
Phase 6: Development Test & Evaluation	None	None	**7.9**
Internal Requirements	None	None	None
Staffing Mix Change: Change	**2.0**	None	None

Super Domain

Table 25: Odds Ratio for Low RV Inputs by Super Domain *Initial*

The Odds Ratios in the table represent the greater likelihood that the contractor relative to other contractors employed higher RV.				
RV Measurement	Engineering	Real time	Automated Information System	Unknown
Input	1.8	None	2.1	0.7
Conjoined (Med & High)	0.3	None	None	None
Numerical (2,3,4,5)	None	None	0 of 5	None
Percentage (Upper 3 Quartiles)	None	None	None	None
Adjective (Med & High)	None	2.5	None	None

Table 26: Odds Ratio for Low RV Inputs by Super Domain *Final*

The Odds Ratios in the table represent the greater likelihood that the Super Domain relative to other domains employed higher RV.				
RV Measurement	Engineering	Real time	Automated Information System	Unknown
Input	0.4	1.7	None	None
Conjoined (Med & High)	None	1.3	0.4	None
Numerical (2,3,4,5)	None	0.6	None	None
Percentage (Upper 3 Quartiles)	None	None	0.4	2.0
Adjective (Med & High)	None	3.3	None	0.4

Super Domains are the key groups of software domains that application types fall under. They can be thought of as the activity the software will support. For *Initial* RV reports Engineering and Automated Information System (AIS) software projects tend to report more often. AIS software projects trend toward low levels of RV while Real Time projects trend towards higher measures in RV reporting. Real Time software projects are likely to experience greater than median growth in ESLOC while, AIS are more likely to see lower than median ESLOC and Phase 1 growth.

Table 27: Odds Ratio for Getting Higher than Median Value of Growth by Super Domain

The Odds Ratios in the table represent the greater likelihood that the specific Super Domain relative to other domains experiences above median growth for the given measure.				
Percent Change split by median	**Engineering**	**Real Time**	**Automated Information System**	**Unknown**
Equivalent Source Lines of Code (ESLOC)	None	**2.2**	**0.3**	No Data
Schedule in Months	None	None	None	No Data
Peak Head Count	None	None	None	No Data
Total Hours	None	None	None	No Data
Phase 1: Requirements Analysis	None	None	**0.4**	No Data
Phase 2: Architecture & Design	None	None	None	No Data
Phase 3: Coding & Testing	None	None	None	No Data
Phase 4: Software and System Integration	None	None	None	No Data
Phase 5: Qualification Testing	None	None	None	No Data
Phase 6: Development Test & Evaluation	**0.6**	None	None	No Data
Internal Requirements	None	None	None	No Data
Staffing Mix Change: Change	None	None	None	No Data

Development Process

Table 28: Odds Ratio for Low RV Inputs by Development Process *Initial*

The Odds Ratios in the table represent the greater likelihood that the Development Process relative to other processes employed higher RV.				
RV Measurement	**Waterfall**	**Spiral**	**Incremental**	**Agile**
Input	**0.6**	**0.3**	**1.6**	**10.0**
Conjoined (Med & High)	**5.0**	None	**0.5**	**0.1**
Numerical (2,3,4,5)	**5.0**	None	**0.1**	None
Percentage (Upper 3 Quartiles)	None	**10.0**	**5.0**	**0.3**
Adjective (Med & High)	**3.3**	None	**0 of 50**	None

Table 29: Odds Ratio for Low RV Inputs by Development Process *Final*

The Odds Ratios in the table represent the greater likelihood that the Development Process relative to other processes employed higher RV.				
RV Measurement	**Waterfall**	**Spiral**	**Incremental**	**Agile**
Input	**1.3**	**1.6**	**1.8**	**0.3**

Conjoined (Med & High)	**1.3**	**2.0**	None	**0.2**
Numerical (2,3,4,5)	None	None	None	None
Percentage (Upper 3 Quartiles)	None	**5.0**	**0.2**	**0.2**
Adjective (Med & High)	None	None	None	None

DoD software has been identified as being developed in several processes, these are the most popular used on the SRDR. It is common for hybrid of these processes to be used but projects with hybrid processes were not examined for this analysis. It is evident that projects that use Incremental and Agile style of development are much more likely to report RV inputs on *Initial* submissions than other methods. Projects that use the Waterfall or Spiral process of development have been found to report RV inputs on *Initial* reports less often than other processes. For *Final* reporting Incremental and Agile are likely to report lower levels of RV while Waterfall and Spiral report higher levels of RV. For actual growth the Waterfall and Agile processes RV reporting is substantiated by the growth trends.

Table 30: Odds Ratio for Getting Higher than Median Value of Growth by Development Process

The Odds Ratios in the table represent the greater likelihood that the specific Development Process relative to other processes experiences above median growth for the given measure.				
Percent Change split by median	**Waterfall**	**Spiral**	**Incremental**	**Agile**
Equivalent Source Lines of Code (ESLOC)	None	None	None	None
Schedule in Months	None	**0.6**	None	None
Peak Head Count	None	None	**1.4**	None
Total Hours	None	None	None	None
Phase 1: Requirements Analysis	None	None	None	**0.3**
Phase 2: Architecture & Design	None	None	None	None
Phase 3: Coding & Testing	None	None	None	None
Phase 4: Software and System Integration	**1.9**	None	None	None
Phase 5: Qualification Testing	None	None	None	**0.2**
Phase 6: Development Test & Evaluation	None	**0.3**	**2.3**	None

Internal Requirements	**1.9**	None	None	None
Staffing Mix Change: Change	**0.5**	None	**3.7**	None

New or Upgrade Software

Table 31: Odds Ratio for Low RV Inputs by New Software *Initial*

The Odds Ratios in the table represent the greater likelihood that new software relative to upgraded software employed higher RV.	
RV Measurement	**New**
Input	**0.7**
Conjoined (Med & High)	**0.4**
Numerical (2,3,4,5)	None
Percentage (Upper 3 Quartiles)	None
Adjective (Med & High)	**0.1**

Table 32: Odds Ratio for Low RV Inputs by New Software *Final*

The Odds Ratios in the table represent the greater likelihood that new software relative to upgraded software employed higher RV.	
RV Measurement	**New**
Input	**0.5**
Conjoined (Med & High)	**1.7**
Numerical (2,3,4,5)	**1.4**
Percentage (Upper 3 Quartiles)	**5.0**
Adjective (Med & High)	None

A New or Upgrade classification on the SRDR indicates whether the primary development is a new software or an upgrade of existing software. Each project is identified as being one or the other, there were a handful of projects which indicated both, and these were not considered in the analysis which produced the Odds Ratios. New projects are found to report RV less often than Upgrades. New projects were also found to report RV at higher measures. The higher reporting is not substantiated. When significant differences in growth were found New Software was actually found to be more likely to fall in the lower level of growth distributions.

Table 33: Odds Ratio for Getting Higher than Median Value of Growth by New Software

The Odds Ratios in the table represent the greater likelihood that New software relative to upgraded software experiences above median growth for the given measure.	
Percent Change split by median	**New**
Equivalent Source Lines of Code (ESLOC)	**0.7**
Schedule in Months	None
Peak Head Count	None
Total Hours	None
Phase 1: Requirements Analysis	None
Phase 2: Architecture & Design	None
Phase 3: Coding & Testing	None
Phase 4: Software and System Integration	**0.6**
Phase 5: Qualification Testing	None
Phase 6: Development Test & Evaluation	None
Internal Requirements	**0.7**
Staffing Mix Change: Change	None

Location

Table 34: Odds Ratio for Low RV Inputs Worksite Location *Initial*

The Odds Ratios in the table represent the greater likelihood that the worksite location relative to other locations employed higher RV.				
RV Measurement	**East Coast**	**West Coast**	**Central**	**Southeast**
Input	**1.7**	**0.7**	None	**0.3**
Conjoined (Med & High)	None	None	None	None
Numerical (2,3,4,5)	None	None	**10.0**	None
Percentage (Upper 3 Quartiles)	**0.4**	None	None	None
Adjective (Med & High)	None	None	None	None

Table 35: Odds Ratio for Low RV Inputs by Worksite Location *Final*

The Odds Ratios in the table represent the greater likelihood that the worksite location relative to other locations employed higher RV.

RV Measurement	East Coast	West Coast	Central	Southeast
Input	**0.7**	**0.7**	**2.2**	None
Conjoined (Med & High)	**0.7**	None	**1.4**	None
Numerical (2,3,4,5)	None	None	**1.7**	**0.6**
Percentage (Upper 3 Quartiles)	**2.5**	None	None	**0.3**
Adjective (Med & High)	None	None	None	None

The DoD hires contractors which perform work across the United States. Location indicates the location of the worksite where the software will be built. Our analysis has indicated that worksites on the East Coast are associated with a more willingness to provide RV input for *Initial* submissions but less willing for *Final* submissions. The West Coast was found to be less willing to provide RV input for either submission. Centrally located worksites showed a tendency for reporting RV at higher levels, while the East Coast worksites were more likely to report low. The growth associations do not solidify these relationships. The Central worksites trended towards higher than median growth for requirements analysis but lower than median for schedule and internal requirements. The East Coast leaned towards higher than median internal requirements change but lower than median for Phase 6.

Table 36: Odds Ratio for Getting Higher than Median Value of Growth by Worksite Location

The Odds Ratios in the table represent the greater likelihood that the general Worksite Location relative to other locations experiences above median growth for the given measure.				
Percent Change split by median	**East Coast**	**West Coast**	**Central**	**Southeast**
Equivalent Source Lines of Code (ESLOC)	None	None	None	None
Schedule in Months	None	None	**0.6**	None
Peak Head Count	None	None	None	None
Total Hours	None	None	None	None
Phase 1: Requirements Analysis	None	**0.4**	**2.1**	None
Phase 2: Architecture & Design	None	None	None	None
Phase 3: Coding & Testing	None	None	None	None

Phase 4: Software and System Integration	None	None	None	None
Phase 5: Qualification Testing	None	None	None	None
Phase 6: Development Test & Evaluation	**0.5**	None	None	None
Internal Requirements	**1.9**	None	**0.6**	None
Staffing Mix Change: Change	None	None	None	**0.2**

Primary Software Development Language

Table 37: Odds Ratio for Low RV Inputs by Software Language *Initial*

The Odds Ratios in the table represent the greater likelihood that the Software Language relative to other languages employed higher RV.			
RV Measurement	**Java**	**C++**	**Ada**
Input	**1.7**	None	None
Conjoined (Med & High)	**0.3**	**1.7**	**0.3**
Numerical (2,3,4,5)	None	**0.1**	None
Percentage (Upper 3 Quartiles)	None	None	**3.3**
Adjective (Med & High)	None	None	None

Table 38: Odds Ratio for Low RV Inputs by Software Language *Final*

The Odds Ratios in the table represent the greater likelihood that the Software Language relative to other languages employed higher RV.			
RV Measurement	**Java**	**C++**	**Ada**
Input	**1.9**	**0.5**	**1.9**
Conjoined (Med & High)	**1.3**	None	**0.4**
Numerical (2,3,4,5)	None	None	None
Percentage (Upper 3 Quartiles)	**2.0**	**3.3**	**0.1**
Adjective (Med & High)	**5.0**	None	**0 of 24**

Projects with Java as the Primary software language are more likely to provide RV input for either submission. Java also showed an association to lower rating of RV for *Initial* submissions but higher rating for final submissions, and no notable associations for growth. Few relationships between the software languages and the growth measures were found and those that were, were contradictory or not well supportive to the RV rating trends.

Table 39: Odds Ratio for Getting Higher than Median Value of Growth by Software Language

The Odds Ratios in the table represent the greater likelihood that the specific Software Language relative to other languages experiences above median growth for the given measure.			
Percent Change split by median	**Java**	**C++**	**Ada**
Equivalent Source Lines of Code (ESLOC)	None	None	None
Schedule in Months	None	None	**0.5**
Peak Head Count	None	None	None
Total Hours	None	None	None
Phase 1: Requirements Analysis	None	None	None
Phase 2: Architecture & Design	None	None	None
Phase 3: Coding & Testing	None	None	None
Phase 4: Software and System Integration	None	None	None
Phase 5: Qualification Testing	None	None	None
Phase 6: Development Test & Evaluation	None	**0.5**	None
Internal Requirements	None	**0.5**	**2.0**
Staffing Mix Change: Change	**2.0**	**0.6**	None

Estimated or Measured Values

The next three attributes for analysis, are characterized by being estimates on *Initial* submissions and actuals for *Final* submissions. For testing with RV inputs both submissions will be used, but for the tests with the growth measures only the *Initial* inputs will be used. Some of these elements are used as measures for calculating growth. For the growth measures, percent change from *Initial* to *Final* was calculated, where the analysis with RV the estimated or measured inputs will only be using a single value.

New Requirements

Table 40: Odds Ratio for Low RV Inputs by New Requirement *Initial*

The Odds Ratios in the table represent the greater likelihood that the percentage of new requirements relates to higher RV.

RV Measurement	New Internal Reqs ≥ 50%	New External Reqs ≥ 50%
Input	None	None
Conjoined (Med & High)	None	None
Numerical (2,3,4,5)	None	None
Percentage (Upper 3 Quartiles)	None	None
Adjective (Med & High)	None	None

Table 41: Odds Ratio for Low RV Inputs by New Requirements *Final*

The Odds Ratios in the table represent the greater likelihood that the percentage of new requirements relates to higher RV.		
RV Measurement	**New Internal Reqs ≥ 50%**	**New External Reqs ≥ 50%**
Input	None	**2.7**
Conjoined (Med & High)	**0.5**	**0.6**
Numerical (2,3,4,5)	**0.3**	None
Percentage (Upper 3 Quartiles)	**10.0**	**5.0**
Adjective (Med & High)	None	None

New requirements greater than 50%, Internal or External showed no significant associations in reporting on *Initial* submissions. For *Final* submissions, when more than 50% of the External Requirements were new, there was an increased likelihood for receiving RV input. These inputs were typically lower ratings of RV, which is supported by the growth associations found for ESLOC, Phase 4, and Internal Requirements. New Internal Requirements greater than 50% also has trends towards low ratings in RV, yet the associations for growth are toward the high side for Phase 1, 2 and Schedule of the growth measures.

Table 42: Odds Ratio for Getting Higher than Median Value of Growth by New Requirements

The Odds Ratios in the table represent the greater likelihood that more than 50% New Requirements leads to above median growth for the given measure.

Percent Change split by median	New Internal Reqs ≥ 50%	New External Reqs ≥ 50%
Equivalent Source Lines of Code (ESLOC)	**0.4**	**0.2**
Schedule in Months	**3.5**	None
Peak Head Count	None	None
Total Hours	None	None
Phase 1: Requirements Analysis	**2.6**	None
Phase 2: Architecture & Design	**2.9**	None
Phase 3: Coding & Testing	None	None
Phase 4: Software and System Integration	**0.2**	**0.3**
Phase 5: Qualification Testing	None	None
Phase 6: Development Test & Evaluation	None	None
Internal Requirements	None	**0.2**
Staffing Mix Change: Change	None	None

Size by Total Hours and ESLOC

Table 43: Odds Ratio for Low RV Inputs by Size *Initial*

The Odds Ratios in the table represent the greater likelihood that the project size relates to higher RV.		
RV Measurement	**Total Hours > Median**	**ESLOC > Median**
Input	**0.6**	None
Conjoined (Med & High)	None	None
Numerical (2,3,4,5)	None	**0.2**
Percentage (Upper 3 Quartiles)	None	**2.0**
Adjective (Med & High)	None	**3.3**

Table 44: Odds Ratio for Low RV Inputs by Size *Final*

The Odds Ratios in the table represent the greater likelihood that the project size relates to higher RV.		
RV Measurement	**Total Hours > Median**	**ESLOC > Median**
Input	**1.5**	None
Conjoined (Med & High)	**2.5**	**1.7**
Numerical (2,3,4,5)	**3.3**	**2.0**
Percentage (Upper 3 Quartiles)	**2.5**	**1.7**
Adjective (Med & High)	None	None

The larger projects measured by total hours were found less likely to provide RV input for *Initial* submissions and more likely for *Final* submissions. On *Final* submissions larger projects were generally found to report higher levels of RV. However, the growth relationships do not substantiate those relationship except for when it comes to Peak Head Count. These results may not tell the whole story, that is larger projects can experience massive growth, but the percent change may be lower.

Table 45: Odds Ratio for Getting Higher than Median of Growth by Size

The Odds Ratios in the table represent the greater likelihood that the larger projects experience above median growth for the given measure.		
Percent Change split by median	**Total Hours > Median**	**ESLOC > Median**
Equivalent Source Lines of Code (ESLOC)	None	**0.6**
Schedule in Months	**0.6**	None
Peak Head Count	**1.7**	**1.6**
Total Hours	**0.6**	None
Phase 1: Requirements Analysis	None	None
Phase 2: Architecture & Design	None	None
Phase 3: Coding & Testing	**0.7**	None
Phase 4: Software and System Integration	None	None
Phase 5: Qualification Testing	**0.4**	**0.5**
Phase 6: Development Test & Evaluation	None	None
Internal Requirements	None	None
Staffing Mix Change: Change	None	None

Experience Level

Table 46: Odds Ratio for Low RV Inputs by Personnel Experience Level *Initial*

The Odds Ratios in the table represent the greater likelihood that the Personnel Experience Level relates to higher RV.		
RV Measurement	**Highly Experienced ≥ 50%**	**Inexperienced ≥ 50%**
Input	**1.8**	**3.5**
Conjoined (Med & High)	None	**10.0**
Numerical (2,3,4,5)	None	None

Percentage (Upper 3 Quartiles)	None	None
Adjective (Med & High)	**0.4**	**10.0**

Table 47: Odds Ratio for Low RV Inputs by Personnel Experience Level *Final*

The Odds Ratios in the table represent the greater likelihood that the Personnel Experience Level relates to higher RV.		
RV Measurement	**Highly Experienced ≥ 50%**	**Inexperienced ≥ 50%**
Input	None	**0.4**
Conjoined (Med & High)	**0.6**	None
Numerical (2,3,4,5)	**0.6**	None
Percentage (Upper 3 Quartiles)	**0.5**	None
Adjective (Med & High)	None	None

It is important to distinguish that the teams that are not categorized as being highly experienced are not necessarily, Inexperienced, and visa versa. This is possible because there are 5 experience levels and the upper two or lower two are combined to create there respective category for testing. When providing RV input for *Initial* submissions both highly experienced and inexperienced teams were found more likely to provide input. Highly experienced teams tended to report lower levels of RV while Inexperienced teams provided higher inputs. The few trends that are present do not support the RV reporting trends for either experience level.

Table 48: Odds Ratio for Getting Higher than Median Value of Growth by Personnel Experience Level

The Odds Ratios in the table represent the greater likelihood that the Personnel Experience Level experiences above median growth for the given measure.		
Percent Change split by median	**Highly Experienced ≥ 50%**	**Inexperienced ≥ 50%**
Equivalent Source Lines of Code (ESLOC)	None	**0.1**
Schedule in Months	None	None
Peak Head Count	None	None
Total Hours	None	None

Phase 1: Requirements Analysis	None	None
Phase 2: Architecture & Design	None	None
Phase 3: Coding & Testing	**1.8**	None
Phase 4: Software and System Integration	None	None
Phase 5: Qualification Testing	None	None
Phase 6: Development Test & Evaluation	None	None
Internal Requirements	**2.0**	None
Staffing Mix Change: Change	**0.7**	None

V. Conclusions

Findings:

RV Collection

This study set out to conduct a deep investigation into requirements volatility (RV) and its reporting in the DoD's SRDR Compilation Dataset. In doing so, it has been shown that the efforts to collect RV data have been a continuous struggle. For nearly two decades and four policy updates the DoD has been unable to get the same level of consistency in reporting and overall buy in that are found in other reported metrics. This study did not find and did not dig to uncover the underlying cause of this RV reporting avoidance. Perhaps it is the uncertain and dynamic aspects of software which RV ultimately is trying to capture, that steers the contractor away from even wanting to attempt such a task. Regardless, if the DoD is going to continue to require that RV be assessed, enforcement of what is required must be improved.

Figure 36 is visual depiction of this struggle for collecting RV data from contractors. The percentages on the vertical of axis are representative of the percentage of total submissions that correctly followed the guidance for that particular year. The figure was created by applying the RV guidelines per the DoD instructions from the years of 2004, 2007, 2011, and 2017 to the SRDR dataset, then counting how many followed the prescribed format for that year. Over the 16-years which we have submissions and specific guidance available, the DoD rarely saw the correct reporting of RV over 40%, and often it was far worse.

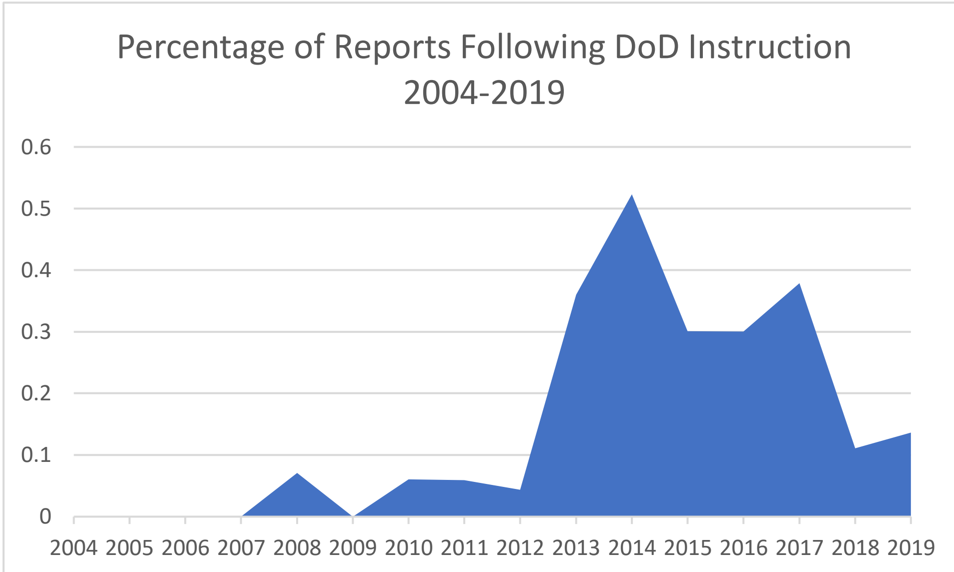


Figure 36: Percentage of Reports Following DoD Instruction 2004-2019

From the beginning, RV reporting has been troubled. The 2004 DoD 5000.4 CSDR Manual was truly calling for more of a global assessment of a program. That is, it was asking the contractors to assess the overall program performance not just the requirements after completion. This is perhaps why today we see the requirements volatility input column on the SRDR simply labeled "Volatility." Through time, however, the DoD clearly molded its asking toward a more requirements specific assessment and wished to track its ability as an early indicator for future performance. In 2007, the DoD added an *Initial* assessment, or prediction as well as a *Final* assessment. These were specifically noted to be qualitative and not absolute measures. This lends one to believe that it was once understood that the complexities and nature of change in requirements in DoD software did not comply with a more calculated approach. And may pose a reasonable explanation to the reluctance of program manager today to provide of percentage-based inputs.

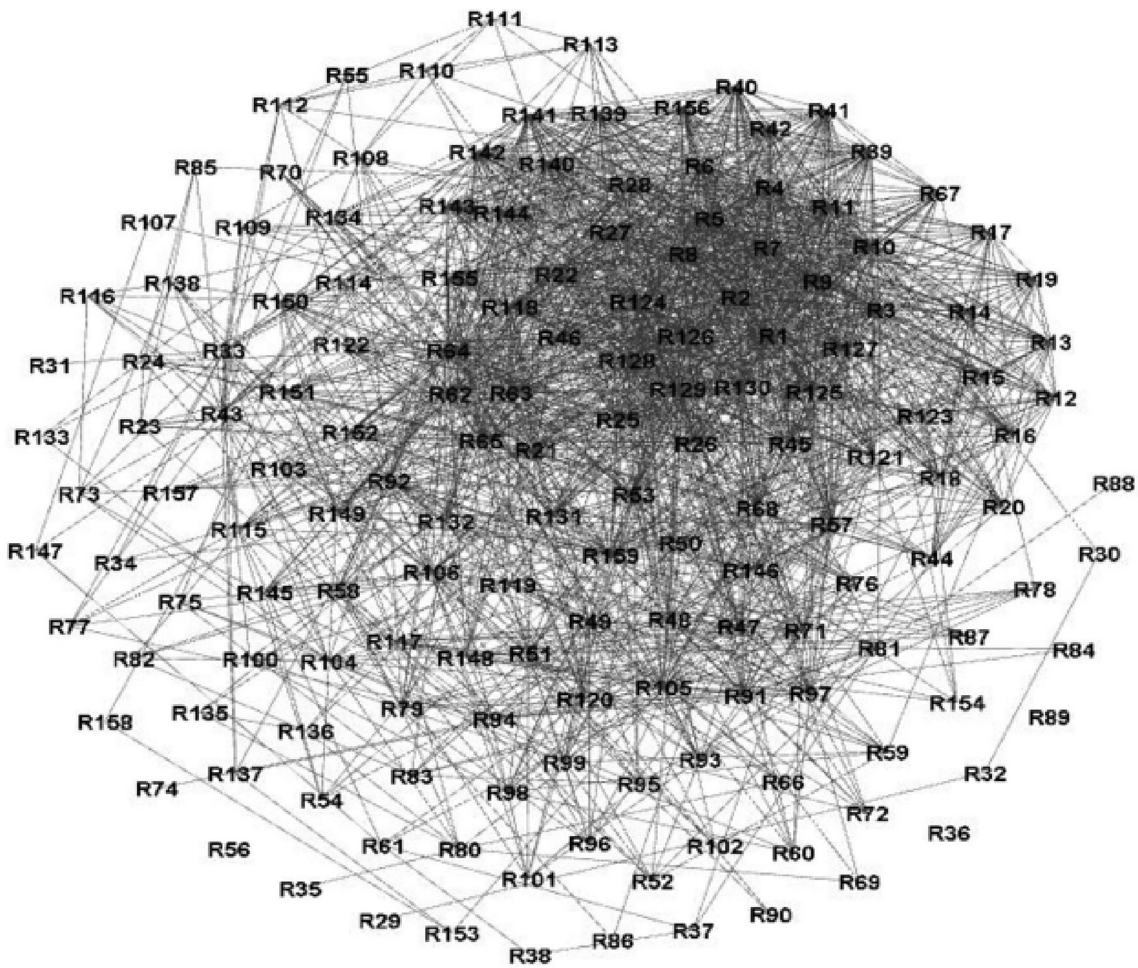


Figure 37: Example of Requirement Network (Hein, 2021)

Because of the interconnectedness of requirements, as well as the compounding nature of assessing uncertainty, volatility should not be expected to have a linear relationship with performance. Figure 37 is the requirement network for a machine that has under 200 requirements and in terms of technology would be considered far less complex than projects found on the SRDR. The network is used as visual representation of what a contractor should be considering as they estimate RV in the manner we ask, they must not only consider the requirements that are likely to change they should consider the second or third order effects those changes may have. Not all requirements are equal, so when accounting for changing requirements, it is illogical to assume they should then be treated equally. This interconnected nature and compounding effects of change makes it questionable as to why the 2011 and 2017 SRDR DIDs ask for input to be switched to percentages. From the estimation perspective, it seems an impossible task. Furthermore, the percentage-based answer equally if not more so relies on the gut feelings of the inputter, only rather than a total project assessment it forces a repeated assessment of individual requirements. For these reasons it is probable that a percentage-based approach will likely understate the RV when used as an estimation. A percentage would be much more proper for final reporting as the recorded measure of change as it is often used in the research in the field. The same 2017 guidance which directs how RV is to be input calls for a count of Added/New, Modified, Deleted, and Deferred requirements for SRDR reporting. An assessment of these counts would perhaps be the most proper way to asses the requirements volatility of a project at completion. Because all these changes are likely to have different impacts to the growth of a project a weighting system should be considered when creating the RV measure.

RV Input and Performance

Providing any input at all for RV was proven to be more often associated with upper distributions of our growth measures, in fact for *Initial* submissions it was found to have more predictive ability than any of the RV measurement systems. The limited inputs on *Initial* submissions may contribute to this relationship but perhaps it is strengthened by managers properly sensing that the project necessitated a volatility discloser.

RV Schemes Proven as Predictors

Ultimately the exploration into which of the methods was the best predictor for *Initial* reporting should be considered inconclusive. The associations presented in the Adjectival scheme contingency tests contradicted what should have been expected for a test between low and nominal projects, but yet were some of the strongest relationships found. The percentage scheme yielded a couple of promising relationships but without a larger dataset it cannot be decided truly a better predictor than the other methods. The 1-5 method was not found to be a good predictor and for these small sets was the worst of the three schemes. Because the conjoined is a mashup of the other three schemes with the addition of the 0 inputs it is no surprise that its results were mostly found to have no statistically significant associations. This all being said, without one scheme truly proving itself more worthy, perhaps a consideration for a scheme that is easier for the submitter and can attempt to capture the complexities of RV is warranted.

For *Initial* reporting it would be best to frame inputs to the style of the commonly used cost estimation models. Categorical entry with an associated continuous RV output range, with prescribed characteristics of the ratings may be best. As a very rudimentary example, a 1-5 numerical scale, where each number represented an expected 20% range of RV, and each range was had checklist like descriptions that would guide a contractor to where the project may fall. Overtime, the guidance for characterizing entries would be shaped by the actual results of completed projects. A system like this would be less taxing on the submitter, while still allowing for comparison between estimates and actual RV and allowing submitters to consider the nonlinear aspects of RV in initial assessments.

What RV Captures

When contractors are supplying a postmortem look for RV it is clear that cost is more important than schedule. Stronger relationships in the appropriate direction were found for Peak Head Count and Total Hours, which are two known cost drivers, than were found for schedule. This is not surprising as typically one of the primary drivers behind concern for schedule is that projects that take too long end up also costing too much. A driving force behind software data collection, and RV is to improve the ability to estimate costs. Therefore, it is most logical that the two best proxies for costs that were considered as growth measures in this study were found to be of the highest concern for the inputter.

Program Attributes

The program attributes are what characterize the software being created, because of this it is expected to see differing relationships in both RV and growth. What was perhaps not so expected was the associations that were found in merely providing RV input. Tables 49 and 50 show the program attributes and the individual elements which proved to be indicative of reporting trends for *Initial* and *Final* submissions. These elements were all tested independently so it should not be assumed that when together these would compound the likelihood for projects to input or not input RV, further analysis is needed to make that determination. More general judgments and observations however can be made.

Table 49: Program Attributes which Indicate a Higher or Lower Likelihood of Providing RV Input on *Initial* Reports

Program Attribute:	**More Likely**	**Less Likely**

Service	Air Force	Navy, Army
Contractor	E, F	A, D
Operating Environment	AVU	SFM
Application Domain	VC	
Super Domain	ENG, AIS	UNK
Development Process	Incremental, Agile	Waterfall, Spiral
New vs Upgrade		New
Location	East Coast	West Coast, Southeast
Primary Software Language	Java	
Size by Total Hours and ESLOC		Total Hours
Experience Level	Highly Exp ≥ 50%, Inexp ≥ 50%	

Table 50: Program Attributes which Indicate a Higher or Lower Likelihood of Providing RV Input on *Final* Reports

Program Attribute	**More Likely**	**Less Likely**
Service	Navy	Air Force
Contractor	B	E, F
Operating Environment	AVM	SFM
Application Domain	C&C, VC	
Super Domain	RT, AIS	ENG
Development Process	Waterfall, Incremental	Agile
New vs Upgrade		New
Worksite Region	Central	East Coast, *West Coast
Primary Software Language	Java, Ada	C++
New Requirements	Int Reqs ≥ 50%, Ext Reqs ≥ 50%	
Size by Total Hours and ESLOC	Total Hours	

Our exploratory study discovered some trends in both *Initial* and *Final* reporting. The first trends to discuss are between the service branches, the Air Force was found to be more likely to report on *Initial* submissions but less likely for *Final.* From a general understanding this could be indication of the closest following of RV guidance, if we only considered the 2017 guidance this would be true, however sense we know that much of our data is pre 2017 we should not consider this. Rather than the lens of a particular guidance being a sign of proper following, we are more identifying if certain program attributes identify RV as more or less important regardless of guidance. From this aspect we see that, the application type of Vehicle Control, the Super Domain of Automated Information System, Incremental Development, projects whose primary language is Java are all characteristics of programs that regard RV as more important. This is determined because these attributes were associated with statistically higher levels of RV input for both submission types. Potential explanations could perhaps be these attributes align with the projects that render a certain level of complexity that has made SMEs report RV. Another possible explanation is that some of these program attributes require an extra level of reporting detail causing higher likelihood for RV reporting. Just like complexity may drive a feeling for a need to disclose volatility simplicity may negate this feeling. Some attributes like the operating environment of SFM (Surface, Fixed, Manned) may be not as complex and developers may feel less of a need to disclose the expectation of 0 or low RV. The location of West Coast being significantly related to lower chances for inputting is interesting, merely because of the loose stereotype of the West Coast namely California being more "laid-back," this more relaxed culture could contribute to less concern with what is uncertain.

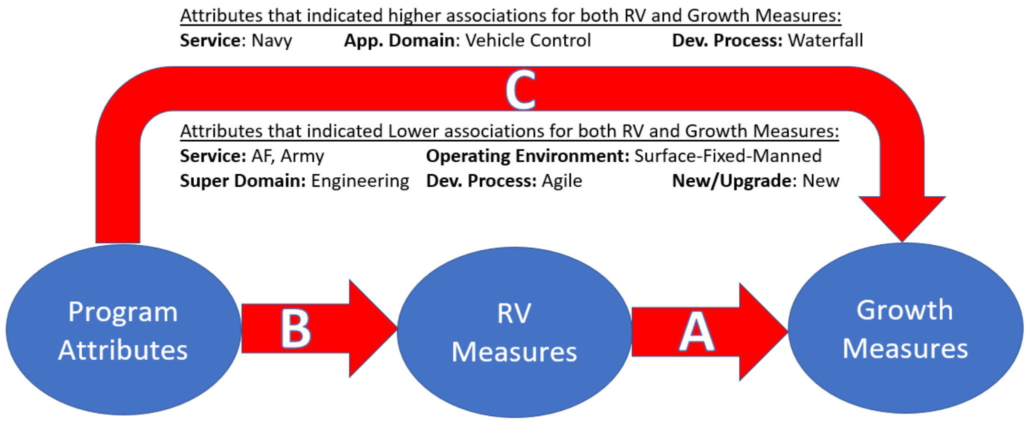


Figure 38: Mediation Model with Significant Program Characteristics for *Initial* Submissions

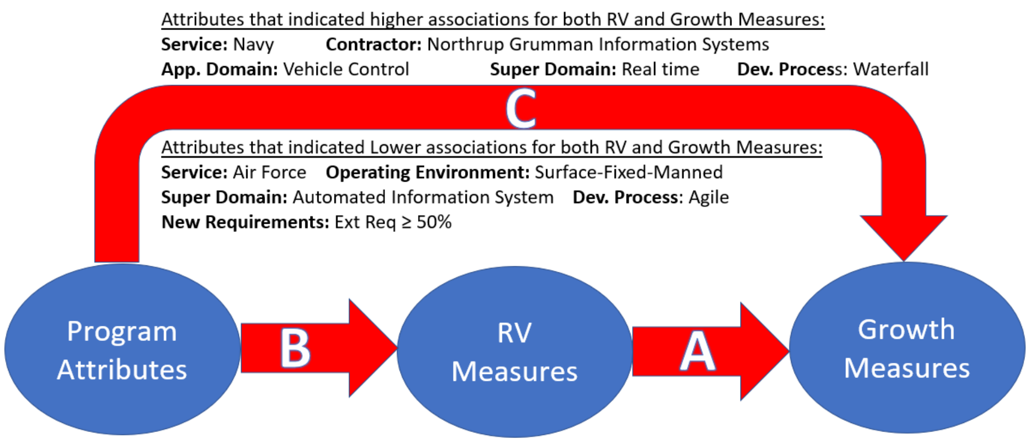


Figure 39: Mediation Model with Significant Program Characteristics for *Initial* Submissions

We conducted an exploratory deeper dive in an aim to determine what associations if any may suggest RVs role as a mediator. This deeper dive was able to identify a few relationships that were found to not only correlate repeatedly with the RV values but with some measures of growth as well. All three service branches were found to have relationships to RV and growth that were in the appropriate directions. This is suggestive that the Navy had taken on projects that experienced higher volatility, while the AF and Army were more likely to experience less volatility in their projects. One of the more interesting findings was that a new software build was more likely to have lower levels of growth than upgrades. Initial intuition wants to tell us that new should be more complicated than an upgrade. However, in software this is not always the case, oftentimes upgraded systems will have legacy requirements that can complicate a build by requiring integration between new and old systems. Overall, this model should serve as a way of sanity check for *Initial* submissions, as quick glance of what the past has proven. These program attributes have been tested independently and found to trend toward the stated direction of volatility which was proven to indicate a level performance at least, loosely.

The program attributes and their significant associations to both RV and growth measures allowed for some understanding to be gained on RVs role as a mediating variable. When an attribute was proven to have the same directional relationships with both RV and growth, RV was being validated as having a mediating role. The exploratory techniques and simple tests conducted were necessitated by messy data perhaps overtime a more sophisticated study of this mediation role may be conducted.

Recommendation for the Future of RV in the SRDR

The nature of the available data, the past and continued struggle for receiving RV inputs calls for no action which aims to further complicate its collection. The nature of the existing dataset is grossly categorical, and this study highlighted it may be of more use perhaps as a global assessment of Volatility, as that is how our data suppliers tend to view RV, especially at completion. The current data does not make a strong case for a precise measure, and the past performance of RV as a predictor of requirements or cost growth does not indicate a precise measure is feasible. RV was found far more likely to associate with attributes other than requirements, so a transition back to a more global assessment of change and uncertainty may be called for. Using Volatility as global assessment, which considers, requirements, among other software attributes like, operating environment and application domain, or nature of the technology may prove to be conceptually easier to assess and thus encourage input.

In the attempts to conjoin the past inputs of RV data it was determined that access to the general cost estimation model used would have been beneficial. Not all models weight categorical values the same so knowledge of the models could have aided in better categorization. If RV collection continues to struggle to gain contractor buy in for a standard reporting system we recommend adding a request for the cost model for the SRDR.

Lastly, one of the struggles with this research was understanding the goals and intentions behind RV reporting in the SRDR dataset. More specifically, what does the DoD want or want analyst to gain through viewing RV data. The unique position for the resource data collection, is that analysts are able to observe and look at a wide range and variety of projects. This range is wide in the size, application, and status of the projects. Perhaps from this view, the strictly analyst view, RV may be more useful as an attempt to quantify or judge the requirements management abilities of developers. Requirements management is known to be a critical indication for both quality software and for software development effort to remain on time and budget. Strong abilities in requirements management are often noted by how well developers handle, avoid, and track requirements change. Switching RV away from its current role as a variable known to be used as a product sizing aspect for estimations, is no small undertaking. Contractors, and estimators are going to hold on to that association, as that is the role RV has played for so long. But from the DoD's analysts perspective a scoring of requirements management by way of looking at RV data may be more useful.

Limitations

This research was limited by the dataset. The dataset is what ultimately drove the exploratory nature of the study and necessitated, a less formal or conventional approach. The dataset itself, was found to be most limited by the changing, enforcement and compliance of DoD policy. Although RV has been a familiar term and documented for years the lack of a more structured industry standard hinders research in the field and more specifically the SRDR dataset. This led to gross inconsistencies across time and difficulty in knowing how RV inputs should best be valued. The many schemes in which RV is input led to a great amount of small assumptions to be made to even consolidate them into what was believed to be the initial scheme. For the assessment of a few values this may not be of major concern, but a lot the inputs needed at least a minor adjustment.

The number of and timeline of the available paired data is also a limitation for this research. Software is one of the most dynamic environments in terms of both technological advancement and cost estimation. So when many of the RV inputs are reaching 10+ years in age this can challenge the applicability for any findings. In a perfect scenario the most data would be available for the most recent projects but that is not the case while looking at the paired data. The paired data must rely on older estimates and uses many submissions that contain impossible schedules and interim reports which hinder the reliability of findings.

Future Research

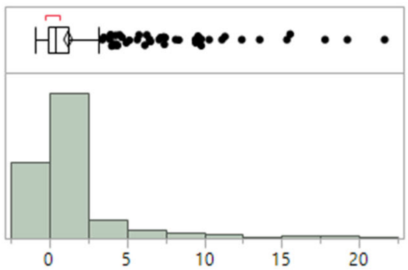
Many areas of further research have been indicated throughout this paper. The most impactful additional research for this study would be a dive into what is the driving force behind the lack of compliance for RV reporting. There are a number of factors that could be leading causes, but I believe the most impactful deep dive would be into the communication aspect. Primarily answering general questions like how is DoD policy disseminated for cost data collection, do services differ in strategies? Or What happens when policy is not followed? Thoroughly, investigating these questions could shine further light on why the Volatility column on the SRDR Data Compilation is the way it is.

Another avenue for future research would be to expand the program attributes examined, to continue to identify items which may act as indicators for future performance. This study was focused on the more basic elements, a deeper dive into all the measured inputs could perhaps further potential predictive attributes. Future analysis may also consider measuring growth simply as the *Final* minus *Initial* rather than percent change, especially while looking at the large projects which are understated by percent change. Additionally, a commitment to a consolidated RV system would allow for more sophisticated test for the ability for RV to act as a mediator. For commitment to such system the relationship between nominal and Low inputs must be more thoroughly investigated.

Appendix

Distributions and Summary Stats of Growth Measures

%Change ESLOC



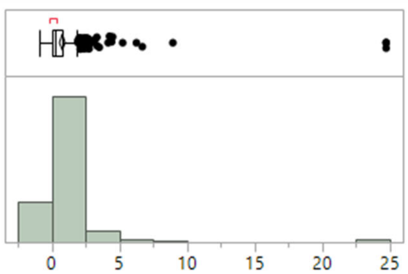
Quantiles

100.0%	maximum	21.666833
99.5%		19.19064771865
97.5%		10.20994649525
90.0%		3.5541217882
75.0%	quartile	1.2271610235
50.0%	median	0.3541829235
25.0%	quartile	-0.11229626525
10.0%		-0.4990621346
2.5%		-0.7944434604
0.5%		-0.90027030761
0.0%	minimum	-0.944735763

Summary Statistics

Mean	1.2328203
Std Dev	2.8896558
Std Err Mean	0.1430593
Upper 95% Mean	1.5140476
Lower 95% Mean	0.9515929
N	408
Median	0.3541829

% Change Schedule in Months



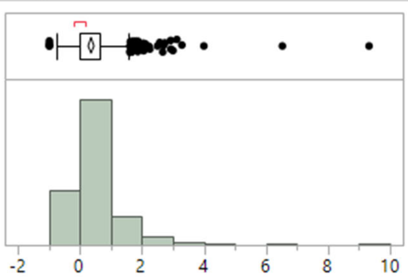
Quantiles

100.0%	maximum	24.76271186
99.5%		24.76271186
97.5%		4.238673969325
90.0%		2.0063856181
75.0%	quartile	0.74803827725
50.0%	median	0.194267516
25.0%	quartile	0
10.0%		-0.1667905304
2.5%		-0.476134762275
0.5%		-1
0.0%	minimum	-1

Summary Statistics

Mean	0.7455742
Std Dev	2.3504764
Std Err Mean	0.1172311
Upper 95% Mean	0.9760386
Lower 95% Mean	0.5151099
N	402
Median	0.1942675

%
Change: Peak Head Count



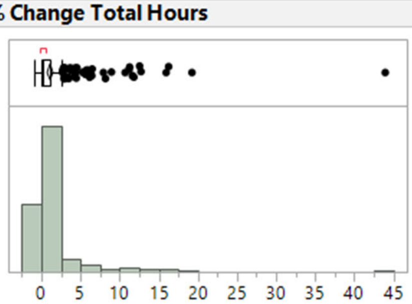
Quantiles

100.0%	maximum	9.345
99.5%		6.42423076879
97.5%		2.688625000075
90.0%		1.4686046511
75.0%	quartile	0.635294118
50.0%	median	0
25.0%	quartile	0
10.0%		-0.2643421055
2.5%		-0.5945454546
0.5%		-1
0.0%	minimum	-1

Summary Statistics

Mean	0.3666977
Std Dev	0.9113073
Std Err Mean	0.0451164
Upper 95% Mean	0.455388
Lower 95% Mean	0.2780074
N	408
Median	0

% Change Total Hours



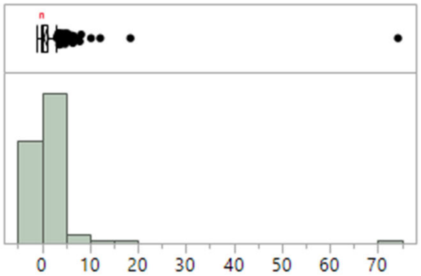
Quantiles

100.0%	maximum	43.97064504
99.5%		19.1072486237
97.5%		11.08451413475
90.0%		2.6561563597
75.0%	quartile	1.048952768
50.0%	median	0.247253315
25.0%	quartile	-0.027705648
10.0%		-0.3294324195
2.5%		-0.702428019975
0.5%		-0.84148154998
0.0%	minimum	-0.944165812

Summary Statistics

Mean	1.124919
Std Dev	3.2366362
Std Err Mean	0.1602374
Upper 95% Mean	1.4399152
Lower 95% Mean	0.8099229
N	408
Median	0.2472533

% Change P1: Requirements Analysis



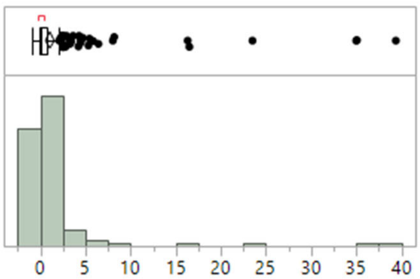
Quantiles

100.0%	maximum	74.2037037
99.5%		39.3498263875
97.5%		6.633063815125
90.0%		2.877760578
75.0%	quartile	1.01820678625
50.0%	median	0.0520674195
25.0%	quartile	-0.3395332375
10.0%		-0.62345977
2.5%		-0.90885442275
0.5%		-0.964682714875
0.0%	minimum	-0.975265018

Summary Statistics

Mean	0.9762336
Std Dev	4.563367
Std Err Mean	0.2535204
Upper 95% Mean	1.4749933
Lower 95% Mean	0.4774739
N	324
Median	0.0520674

% Change P2: Architecture & Design



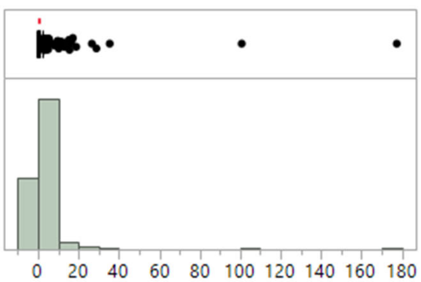
Quantiles

100.0%	maximum	39.4
99.5%		37.024
97.5%		8.059731111
90.0%		2.313959391
75.0%	quartile	0.724839264
50.0%	median	0.127573231
25.0%	quartile	-0.17081891
10.0%		-0.567898153
2.5%		-0.890346122
0.5%		-0.9467844783
0.0%	minimum	-0.95703125

Summary Statistics

Mean	1.02429
Std Dev	4.1830536
Std Err Mean	0.2379656
Upper 95% Mean	1.492534
Lower 95% Mean	0.556046
N	309
Median	0.1275732

% Change P3: Coding & Testing



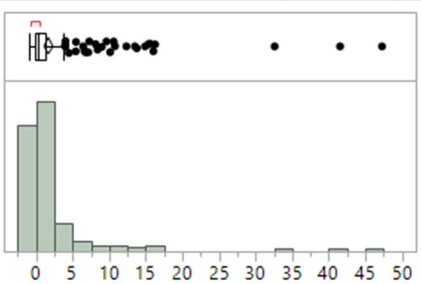
Quantiles

100.0%	maximum	177.543379
99.5%		118.0496069375
97.5%		15.990909985
90.0%		2.7703437605
75.0%	quartile	0.9955225595
50.0%	median	0.267453221
25.0%	quartile	-0.109724242
10.0%		-0.463183928
2.5%		-0.779633089125
0.5%		-0.99044782385
0.0%	minimum	-0.993157895

Summary Statistics

Mean	2.024268
Std Dev	11.413253
Std Err Mean	0.6066076
Upper 95% Mean	3.2172873
Lower 95% Mean	0.8312486
N	354
Median	0.2674532

% Change P4: Software and System Integration



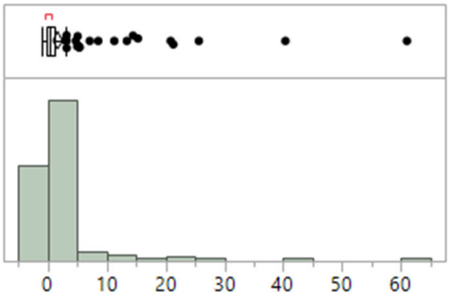
Quantiles

100.0%	maximum	47.3
99.5%		45.0346864155
97.5%		15.38561450475
90.0%		3.9821186596
75.0%	quartile	1.341244582
50.0%	median	0.2354506065
25.0%	quartile	-0.251904007
10.0%		-0.633174285
2.5%		-0.8889098777
0.5%		-0.929537287035
0.0%	minimum	-0.934657659

Summary Statistics

Mean	1.6656897
Std Dev	5.1337299
Std Err Mean	0.3079007
Upper 95% Mean	2.2718122
Lower 95% Mean	1.0595673
N	278
Median	0.2354506

% Change P5:Qualification Testing



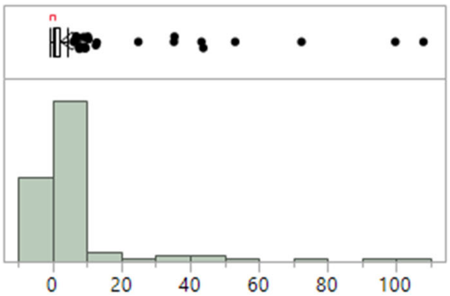
Quantiles

100.0%	maximum	61.07142857
99.5%		60.55416666525
97.5%		20.176670395
90.0%		3.1047942875
75.0%	quartile	1.102628134
50.0%	median	0.2177434645
25.0%	quartile	-0.1750275885
10.0%		-0.7025281025
2.5%		-0.935214331375
0.5%		-0.97775024015
0.0%	minimum	-0.97827028

Summary Statistics

Mean	1.6614787
Std Dev	6.1059474
Std Err Mean	0.4275018
Upper 95% Mean	2.5043921
Lower 95% Mean	0.8185653
N	204
Median	0.2177435

% Change P6: Development Test & Evaluation



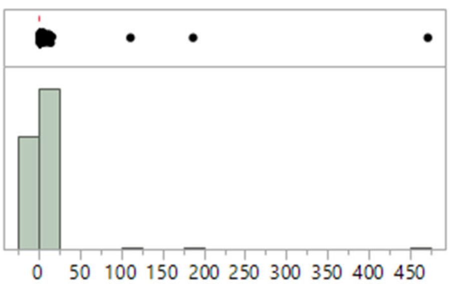
Quantiles

100.0%	maximum	108.0765027
99.5%		108.0765027
97.5%		68.649290548
90.0%		10.209009012
75.0%	quartile	2.015625
50.0%	median	0.418184855
25.0%	quartile	-0.2
10.0%		-0.6607997112
2.5%		-0.8737667154
0.5%		-0.963768116
0.0%	minimum	-0.963768116

Summary Statistics

Mean	5.3785271
Std Dev	16.49469
Std Err Mean	1.463667
Upper 95% Mean	8.2750812
Lower 95% Mean	2.4819731
N	127
Median	0.4181849

%Change Internal Requirements



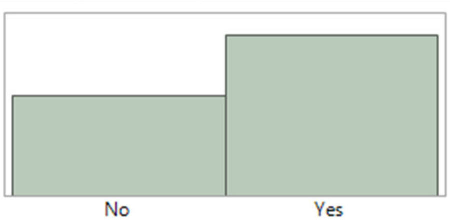
Quantiles

100.0%	maximum	472.3888889
99.5%		183.58
97.5%		9.304575664275
90.0%		2.4823113204
75.0%	quartile	0.1447348485
50.0%	median	0
25.0%	quartile	-0.38592332025
10.0%		-1
2.5%		-1
0.5%		-1
0.0%	minimum	-1

Summary Statistics

Mean	2.3668211
Std Dev	25.763965
Std Err Mean	1.2755063
Upper 95% Mean	4.8742239
Lower 95% Mean	-0.140582
N	408
Median	0

Staffing Mix Change Yes or No



Frequencies

Level	Count	Prob
No	154	0.38119
Yes	250	0.61881
Total	404	1.00000

N Missing 15
2 Levels

Bibliography

Aaramaa, S., Dasanayake, S., Oivo, M., Markkula, J., & Saukkonen, S. (2017). Requirements volatility in software architecture design: An exploratory case study. *Proceedings of the 2017 International Conference on Software and System Process*. https://doi.org/10.1145/3084100.3084105

Baron, R. M., & Kenny, D. A. (1986). The moderator–mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. *Journal of Personality and Social Psychology*, *51*(6), 1173–1182. https://doi.org/10.1037/0022-3514.51.6.1173

Barry, E. (2002). Software evolution, volatility and lifecycle maintenance patterns: A longitudinal analysis synopsis. *International Conference on Software Maintenance, 2002. Proceedings.* https://doi.org/10.1109/icsm.2002.1167806

CAPE. (2011). *Software Resource Data Reporting: Initial Developer Report and Data Dictionary.* Washington D.C.

CAPE. (2011). *Software Resource Data Reporting: Final Developer Report and Data Dictionary.* Washington D.C.

CAPE. (2017). *Software Resource Data Reporting: Development, Maintenance and Enterprise Resource Planning Development Reports, and Data Dictionary.* Washington D.C.

Defense Acquisition University (2013). *Defense Acquisition Guidebook*. Washington, DC.

Department of Defense. (1996). *DoD 5000.2–R: Management procedures for major defense acquisition programs.* Washington, DC.

Department of Defense. (1997). *Cost driver No. 3—Cost/schedule control system criteria (C/SCSC). DoD updated compendium of office of primary responsibility reports.* Washington, DC: DoD Regulatory Cost Premium Working Group.

Defense Contract Management Agency (2018). *DoD Earned Value Management Implementation Guide (EVMIG)*. Virginia: DCMA.

Department of Defense (2007). *Cost and Software Data Reporting (CSDR) Manual*. Washington DC.

Department of Defense, Chief Information Officer. (2019). *DoD Enterprise DevSecOps Reference Design*. Washington DC.

DoD 5000.04–M–1. Washington: DoD.

DoD 5000.04–M–2. (2004) *Software Resources Data Report (SRDR) Manual.* Washington D.C.

G. Stark, A. Skillicorn, and R. Ameele, "An Examination of the Effects of Requirements Changes on Software Releases," CROSSTALK, The Journal of Defence Software Engineering, December 1998.

Hein, P. H., Kames, E., Chen, C., & Morkos, B. (2021). Employing machine learning techniques to assess requirement change volatility. *Research in Engineering Design*, *32*(2), 245–269. https://doi.org/10.1007/s00163-020-00353-6

Henry, J., & Henry, S. (1993). Quantitative assessment of the Software Maintenance Process and Requirements Volatility. *Proceedings of the 1993 ACM Conference on Computer Science - CSC '93*. https://doi.org/10.1145/170791.170868

How to estimate, manage, and track performance on Modern Federal Software Development Programs. Galorath. (2021, July 10). Retrieved January 23, 2022, from https://galorath.com/how-to-estimate-manage-and-track-performance-on-modern-federal-software-development-programs-2/#_ftn1

Jama. (2022). *Four Fundamentals of Requirements Management*. www.jamasoftware.com. Retrieved November 18, 2022, from https://www.jamasoftware.com/requirements-management-guide/requirements-management/four-fundamentals-of-requirements-management/

Jones, C. (1996). Strategies for managing requirements creep. *Computer*, *29*(6), 92–94. https://doi.org/10.1109/2.507640

Jones, C. (2004, October). *Software project management practices: Failure versus success*. Retrieved October 20, 2021, from http://www.inf.ufsc.br/~dovicchi/pos-ed/pos/gerti/artigos/0410Jones.pdf

Jorgensen, M. (2005). Evidence-based guidelines for assessment of software development cost uncertainty. *IEEE Transactions on Software Engineering*, *31*(11), 942–954. https://doi.org/10.1109/tse.2005.128

Kulk, G. P., & Verhoef, C. (2008). Quantifying requirements volatility effects. *Science of Computer Programming*, *72*(3), 136–175. https://doi.org/10.1016/j.scico.2008.04.003

Luketic, D. P. (2020). *The utility of self-assessment in predicting program office estimate accuracy* (thesis). Air Force Institute of Technology.

Mundlamuri, S. (2005). *Managing the Impact of Requirements Volatility* (thesis).

Nidumolu, S. R. (1996). Standardization, requirements uncertainty and software project performance. *Information & Management*, *31*(3), 135–150. https://doi.org/10.1016/s0378-7206(96)01073-7

Nurmuliani, N., Zowghi, D., & Powell, S. (2004). Analysis of requirements volatility during software development life cycle. *2004 Australian Software Engineering Conference. Proceedings.* https://doi.org/10.1109/aswec.2004.1290455

Nurmuliani, N., Zowghi, D., & Williams, S. P. (2006). Requirements Volatility and Its Impact on Change Effort: Evidence-based Research in Software Development Projects. *AWRE*.

Osd Cape. (2019). *The Software Resources Data Report (SRDR) Implementation Guidance*. Washington DC.

OSD, PA&E, CAIG. (2007). *Software Resource Data Reporting: Initial Developer Report and Data Dictionary.* Washington D.C.

OSD, PA&E, CAIG. (2007). *Software Resource Data Reporting: Final Developer Report and Data Dictionary.* Washington D.C.

Singh, M. P., & Vyas, R. (2012). Requirements Volatility in Software Development Process. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*(4), 259–264.

Software engineering: Cocomo Model. GeeksforGeeks. (2020, June 8). Retrieved January 25, 2022, from https://www.geeksforgeeks.org/software-engineering-cocomo-model/

Standish Group, "Chaos," Standish Group Report, 1995.

Stutzke, R. D. (2012). Chapter 8: Estimating Software Size: The Basics, Chapter 17: Tracking Status. In *Estimating software-intensive systems: Projects, products, and Processes* (pp. 194–197-204–205, 521–523). essay, Addison-Wesley Educationa.

Valerdi, R. (2015). Pioneers of Parametrics: Origins and evolution of software cost estimation. *Journal of Cost Analysis and Parametrics*, *8*(2), 74–91. https://doi.org/10.1080/1941658x.2015.1070562

Zowghi, D., & Nurmuliani, N. (2002). A study of the impact of requirements volatility on Software Project Performance. *Ninth Asia-Pacific Software Engineering Conference, 2002.* https://doi.org/10.1109/apsec.2002.1182970

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
23-03-2022	Master's Thesis	October 2020 – March 2022

TITLE AND SUBTITLE	
Salvaging Value from the Measure of "Requirements Volatility" in the DoD's Software Resources Data Report (SRDR)	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	**5d. PROJECT NUMBER**
Walker, Robert T., 1 Lt, USAF	**5e. TASK NUMBER**
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 640 WPAFB OH 45433-7765	AFIT-ENV-MS-22-M-270
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**
Air Force Cost Analysis Agency 1111 Jefferson Davis Hwy, Arlington, Virginia	AFCAA
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRUBTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The DoD and the software development industry as a whole has long dwelt over the idea of requirements volatility (RV). The DoD has toiled with this concept so much so that its guidance was modified four times over a 13 year span. In these changes, its policy completely transformed the what, how, and when regarding RV information. As a result, the volatility data it has received is quite varied and seemingly useless for anything more than anecdotal analysis. This study takes several approaches to salvage value from this data. It begins with a survey of the uncertain concept of volatility, and provides an array of descriptive statistics to make clear what DoD currently has available for analysis. It then places volatility in its intended place as a mediator between problem characteristics and problem outcomes. It does this in two steps. First, it evaluates various volatility measurement schemes against an array of possible measures of growth where the impact of volatility may occur. This work is exploratory to determine which scheme may be most predictive and how so. Second, it identifies relationships between these various measures of volatility and the program attributes which program managers may be contemplating when they try to portray volatility. Both initial and final relationships are tested, capturing the ability of managers to assess volatility in any meaningful way at the start of the program and as a retrospective, or post-mortem. All tests are completed utilizing Contingency Analysis and the respective Odds Ratio to determine the strength of the relationships identified.

15. SUBJECT TERMS
Requirements Volatility

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Scott T. Drylie, AFIT/ENV
U	U	U	UU	120	**19b. TELEPHONE NUMBER** *(Include area code)* (937)-255-3636 x4441 scott.drylie@afit.edu

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18