

# Training Equipment Life Cycle Awareness for Combat Training Centers

**Julie A. Kent**  
**Raytheon Company**  
**Orlando, FL**  
**Julie\_A\_Kent@Raytheon.com**

## ABSTRACT

Combat training centers run large scale training exercises simulating force on force combat. These centers use a variety of simulation equipment to monitor and control the training exercises. Maintenance and upgrade of the equipment requires both time and money. Different pieces of equipment contribute to the training exercises in distinctive ways. Often training centers encounter time sensitive tasks and limited funding related to simulation equipment maintenance and upgrades. It is important to invest in the equipment which will provide the greatest advantage for training capabilities.

Enterprise Architecture (EA) tools have been used to show the relationships between technology and the business capabilities enabled by the technology. The reader will learn results of an investigation into the use of an EA tool to show the relationship between technology in use at a combat training center and the training capabilities available from that technology. The investigation focused on how technology obsolescence impacted training capabilities.

In conducting this study, the importance of software upgrades and advancements became apparent. Software often has a shorter life cycle than hardware, but provides an easier upgrade and replacement plan. So system managers benefit from a different view of software obsolescence planning. Through the increased use of virtualization, a single hardware component may run large numbers of software components. The architecture tools provide some visibility into the aggregation of software and the possible risks to training delivery from component interactions. The system manager has increased visibility allowing investments to be targeted in areas where they will achieve more impact on military training capabilities.

## ABOUT THE AUTHOR

**Julie Kent** has been working in systems integration for over 20 years. She moved to Orlando ten years ago taking a position with Raytheon. Julie worked database upgrades and integration in support of training exercises at the National Training Center (NTC), Joint Readiness Training Center (JRTC) and Joint Multinational Readiness Center (JMRC). She transitioned towards integrating Commercial-off-the-Shelf (COTS) products to create a Management Information System (MIS) supporting cross platform work order management and life-cycle upkeep. She currently serves as a Sr. Principal Systems Engineer in the Global Training Solutions Mission Area of Raytheon Intelligence, Information and Services (IIS). She recently completed a certificate in cognitive science from the University of Central Florida, exploring how availability of data may influence decision making. Previously she was granted an MBA from the University of Baltimore, Master's degree in Computer Science from UMBC, and a Bachelor's degree in Electrical Engineering from Virginia Tech.

This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

## Training Equipment Life Cycle Awareness for Combat Training Centers

The Joint Readiness Training Center (JRTC) provides the Army with realistic training for a wide variety of situations. In addition to combat engagement rehearsal, the JRTC provides training for support personnel including medical personnel, security forces, and logisticians. This training allows units to practice critical tasks as a team in a realistic setting improving readiness. Sensor data is used to track the status of players so hits and misses can be monitored and stored for later discussion. During After Action Reviews (AARs), the teams can assess their actual performance of these tasks, as captured on camera or by other sensors. AARs allow every participant the opportunity to learn from analyzing their own actions and the actions of others. This type of training serves as a mission rehearsal for real world application, which has become increasingly important as the Army integrates modular units from disparate locations into a single task force (Zacchino, 2012).

### BACKGROUND ON LARGE SCALE TRAINING EXERCISE SIMULATION EQUIPMENT

The Instrumentation Systems (IS) supporting combat training at JRTC are Information Systems with inputs from cameras, microphones, and a variety of sensors on Instrumentable- Multiple Integrated Laser Engagement System (I-MILES) gear. These systems are based on networked computers, similar to other large scale information systems. They provide communication systems, data collection systems, data repositories, and a variety of retrieval mechanisms. The importance of these systems for military training continues to expand as the Army seeks ways to reduce costs and maintain force readiness (Cole, 2016). Applying the architecture practices in-use of Information Technology (IT) has benefits for monitoring the long term instrumentation system maintenance needs and related computing devices. Reuse of best practices from other information technology industries supports the Army in making training system investments in a manner that provides the best value (Quartel, Steen, & Lankhorst, 2011).

#### Type of Equipment

Subsystem integration at JRTC dates back to the 1990s with the installation of the Motorola RDMS in 1994 and the documentation of interface controls between the subsystems ("Interface Control Document for the Data Communications Interface/Detection Device Interface for Combat Training Center - Instrumentation Systems (CTC-IS) ICD Revision E.1," 1999). The specifics of the devices and their interfaces change as subsystems are upgraded, producing the need for configuration control. These upgrades occur in response to changing technology, but also in response to changing training objectives of the military. Increased capabilities of tactical systems drive changes in training objectives and expansion of demands upon the training systems. As with other systems, additions and changes may occur based on feedback from the community using the devices (Simpson & Oser, 2003). Often, systems supporting military training are procured individually and then integrated to provide additional training features. For example, one system may be used to track ground vehicles during training maneuvers and later procurement is used to track aircraft. Integrating these systems allows for a more complete tracking display. However, the upgrade paths for these systems are typically independent, frequently corresponding to the tactical devices that are in-use (Rebarick, 2013, p. 15); although IT security and ATO may be combined across the different devices.

#### Process of Training Device Maintenance

Maintainers of training devices are often focused on the item requiring repair and how quickly the device can be fixed. It is a responsibility of the support teams to keep the equipment up and running so it can be used when rotations are scheduled. The time required to restore normal operations is tracked as the Mean Time to Repair (MTR) and maintenance personnel strive to minimize the MTR. An Enterprise Architecture (EA) approach focuses more on the impact of a particular piece of equipment. The overall health and availability of the system may be impacted by particular failures more than others. Certain equipment may have redundancy built into the architecture so the system can continue to function even if a particular component is no longer running.

Hardware maintenance is a combination of corrective repair and preventive actions. Corrective repair often involves swapping out a faulty component for one that is in good working order. Further investigation and lower level repair may require a depot or dedicated maintenance facility. Some equipment fails and is replaced with newer versions

while other instances may continue to run with no complications. This can result in different versions of hardware products being in-use at the same time in different parts of the system.

Alternatively, commonly failing components may be replaced as a form of preventive maintenance. Over time maintenance personnel build a database showing components that require more frequent repair. Planning an upgrade project to replace these components can result in decreased maintenance costs and improved system reliability. Other preventive actions are generally based on manufacturer recommendations. These can include tasks such as blowing out dust or cycling power. The manufacturer may recommend firmware upgrades or other service over the life of the system.

Software does not deteriorate as it ages and can continue to run indefinitely, provided the hardware continues to function and be maintained. Over time, the Army has procured many different software assets for the different Live Training Instruments in use at Combat Training Centers (CTCs). Starting in 2008, the government contracted to run the Live Training Transformation (LT2) product line to manage the different software components in the Live Training Systems. Companies that develop and maintain Live Training (LT) software adhere to architecture standards promoting interoperability. The LT2 community set up a portal on the Internet. The LT2 portal provides an electronic means for the community of developers to share software changes and notify others of those changes (Rivera, Samper, & Clinger, 2008). Through the portal new software releases for JRTC are packaged and delivered to the JRTC maintainers with installation instructions. There is also a mechanism for issues with software to be reported back to the community of users. Software updates are often driven by security needs. System security and Information Assurance (IA) may mandate that software is kept to certain patch levels. They may also insist upon unsupported software being secured because there is no organization equipped to respond if a security flaw in the software is discovered. These types of security issues are also worked through the LT2 system.

### **Purpose of Current JRTC EA Project**

The JRTC EA project stems from a need for forward planning regarding upgrades and improvements on training systems. Without a common road map, regarding component upgrades and system changes, the addition of new functionality to the training devices often does not take into account the lifespan of the existing equipment. In some cases this could lead to planned replacement of equipment that is still within its useful lifespan. In other instances it may result in the intention to use equipment to support improvements, without recognizing full extent of the software upgrades required to support the necessary drivers and to interface with the planned improvements. EA recognizes, “We also need a broader sense of time than just the ‘now’: we need to picture what’s happened in the past, and prepare and plan for the future.(Graves, 2008, p. 101)”

The JRTC EA project supplies ways to visualize the time spans on expected equipment life for multiple components thereby introducing a greater span of time to prepare for and perform maintenance activities. Maintenance activities are generally calculated in hours of labor while component lifespans are measured in months and years. Army training systems may exist for decades. Maintenance of the system focuses on sustainment and quick turnaround time to repair, enabling equipment use and for training. The need to look at factors impacting maintenance and equipment availability over a longer time period requires expanded views beyond individual work orders.

In addition, there is a goal to improve communication between discipline areas including management, network engineering, software engineering, and technical security. The many viewpoints available in architecture tools allow each discipline to look at familiar views while also sharing information others may find useful.

### **APPROACH TO ASSESSMENT OF EQUIPMENT**

The Open Group Architecture Framework (TOGAF) is useful for extending architecture from the systems view to the operational outcomes the system is intended to support. It provides a means of displaying the capabilities an organization has and the way the organization uses its resources, technology and people to deliver those capabilities. Often EA is used as a tool to motivate change. In this instance it is being used to measure the changes occurring based on projects that affect the existing equipment, but are not necessarily run by the community operating and maintaining the equipment. TOGAF was selected as an architecture framework in part because the Department of Defense (DoD) is familiar with some of the views available in the framework. “The original development of TOGAF Version 1 in

1995 was based on the Technical Architecture Framework for Information Management (TAFIM), developed by the US Department of Defense (DoD)” (The Open Group, 1999). TOGAF provides a robust set of relationships between different architecture components and viewpoints showing how changes in technology can affect operations. As an added advantage for this project, the TOGAF overview provides a means of showing at-a-glance what parts of the JRTC architecture are documented in the project.

### **Prior Raytheon Studies**

Raytheon had previously conducted an internal architecture project using many of the same tools. Lessons Learned (LL) from that project were reviewed by the project team at the commencement of this project and incrementally during the documentation of the architecture.

### **Collecting Work Order Information**

Following standard asset management practices, a record of work orders on simulation equipment is captured. A robust asset management system has been used to collect decades of repair and upgrade activity on system components. At regular intervals, this data is analyzed using reliability engineering practices to locate parts repaired or replaced frequently. The asset management system is also used to determine the costliest parts and those requiring the most labor and time to replace. Filters can be applied to this data isolating different areas of interest.

### **Collecting Life Cycle Information**

Prior to this project manufacturer life cycle information had been collected via text entry in the asset management system or maintained in a spreadsheet by life cycle engineers. The information was collected from emails supplied to by the manufacturer or by accessing the manufacturer website. On other occasions the manufacturer was contacted by telephone to solicit end of life information and to discuss potential upgrade paths. At times, one manufacturer was acquired by another manufacturer, causing a ripple effect of product replacements and model changes.

In addition to life cycle announcements from manufacturers, other organizations supply life cycle information to the JRTC. Army, DoD and Army directives regarding the need to phase in or phase out different versions of software or particular models of hardware must be followed. This information is considered internal life cycle information because it is generated by the customer community rather than the suppliers. Direction comes in the form of emails and electronic announcements from the customer and other supporting agencies. Previous to this project lifecycle information was manually stored in a spreadsheet and accessible to maintenance personnel on site.

### **Previous use of Troux**

The Raytheon business unit supporting the JRTC had previously used Troux for internal EA projects obtaining experience using different tools within the software. Previous efforts had included the use of TOGAF for EA modeling so that module was included in the JRTC project. The organization uses Troux Architect to create dynamic models showing the relationships between business processes, management information systems, information technology components, and business objectives. Recent projects included the use of Troux Source repository to provide storage for model data in a manner, making it available to multiple users.

### **Current Project**

The project started with a Troux Enterprise Investment Planning (EIP) success planning workshop. Going into this workshop three business questions were identified to collect data on. These three questions are:

- 1) In terms of capability to deliver flawless training opportunities, what subsystems are most critical for training delivery?
- 2) What technology is the most expensive to sustain and maintain?
- 3) What technology is at most risk of failure due to obsolescence?

These three questions were used as a basis to determine information to populate in the Troux Repository (*JRTC Success Plan*, 2015).

All members of the team were given background and training in the Troux Enterprise Portfolio Management (EPM) suite of tools. These tools include the Troux Source repository which serves as a centralized database for architecture

data, Troux Navigate a web based set of data management tools, Troux Architect ("Troux Architect," 2016) a desktop software tool for creating views and models of the data in the repository, and Troux Insight ("Troux Insight," 2015) a web based reporting tool providing interactive perspectives to assist in answering business questions.

### **Selected Architecture Perspectives**

Based on previous experience with TOGAF, GTS decided to use the architecture overview as a framework to show the areas where data had been collected and was being used for analysis. Four specific Insight perspectives, detailed below, were selected for data gathering and reporting purposes. In support of this data collection, a number of edit screens were developed to allow information entry through Insight in addition to batch data loaded from Excel using Navigate.

The four Insight perspectives desired:

- 1) **Capability Risk Assessment** report was expected to show training capabilities in use at JRTC that are important to the Army and depend on growing/obsolete technology or is no longer supported by the equipment manufacturer. In order to produce this display a strategic value for the training capability has to be assessed.
- 2) **Application Costs – Technology** rollup was selected to determine cost of maintenance and sustainment. This perspective totals the hardware, software, and operational expenses of each application and groups them by the business capabilities.
- 3) **Software and Hardware Product Importance** was designed to display the number of applications using a particular software product or a particular model of hardware. The idea of this perspective is to show where a particular product is in use across multiple systems, therefore garnishing more attention than would be apparent from any single system review.
- 4) **Technology Risk Assessment** perspective was established to show the impact of software lifecycles on the ability to deliver training. This approach provides a heat map with applications grouped by the capabilities support and colored by the current life cycle stage of the software. When multiple software products are used for an application the color reflects the product furthest along the life cycle continuum.

### **Data Collection**

Initially, three sources were used for data collection: the existing asset management system, automated network scan results, and manual inventory of the system. Manual methods were primarily used to resolve questions or conflicts related to data collection from the automated systems. Data conflicts were identified regarding specific versions and names in-use for particular components.

The team held weekly status meetings to review progress. Consolidated monthly status reports with updates on reporting capabilities were presented to senior management incorporating feedback and guidance for the direction of the project.

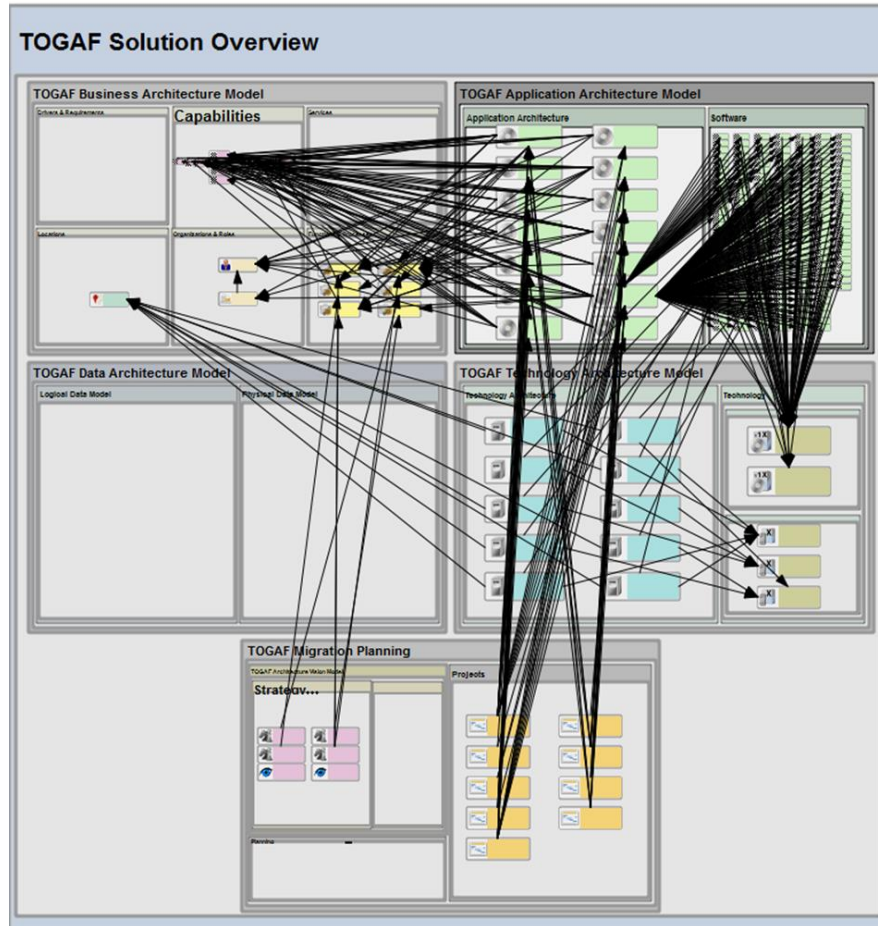


Figure 1 TOGAF overview for one subsystem shows the many relationships between architecture components

## RESULTS OF EA PROJECT

Data entry for this effort took significantly more time than was anticipated. At the start of the project it was estimated that data entry would take about 80 hours over the course of several weeks. After the project was completed data entry was determined to have taken over 300 hours and many months. These hours include a period of trial and error about how to best organize the software components. The majority of the time was spent loading software components. These components are not stored in another enterprise system and there is little automation available to support the tracking of these components.

As data entry progressed, TOGAF overviews were compiled to show the progress of loading components. An example of this type of overview is shown in Figure 1. These diagrams made visible the many relationships between architecture components. Changes to a hardware or software product may affect many other parts of the organization. This illustrates the need for prompt maintenance in order to support training system availability. It also indicates a need for ongoing updates to legacy equipment to promote interoperability with new components. The robust nature of these relationships is shown in Figure 1 which ties back to specific capabilities in the upper left hand side and specific training delivery strategies in the lower left hand side.

The TOGAF is useful for showing relationships, but does not provide an overview of system life. The view could be modified to include more references to the overall system status, but that would require the architect to pull a new report each time an update occurred. Trous Insight provides dynamic screens which pull new data each time

information is loaded. The Troux tool was selected to showcase life cycle updates on a regular basis so viewers would always have the most up-to-date information.

The third business question was conceived as a way to locate equipment reaching end of life by either the manufacturer producing the equipment or the customer making use of it standards. In creating displays to examine hardware and software components approaching end of life, viewers asked about the operational state of elements which were had a longer lifespan. A desire for an overall picture of system health was requested. This led to the creation of a fourth business question: What is the current state of the JRTC Lifecycle? Figure 2 shows an example of the summary view created in response to the question.

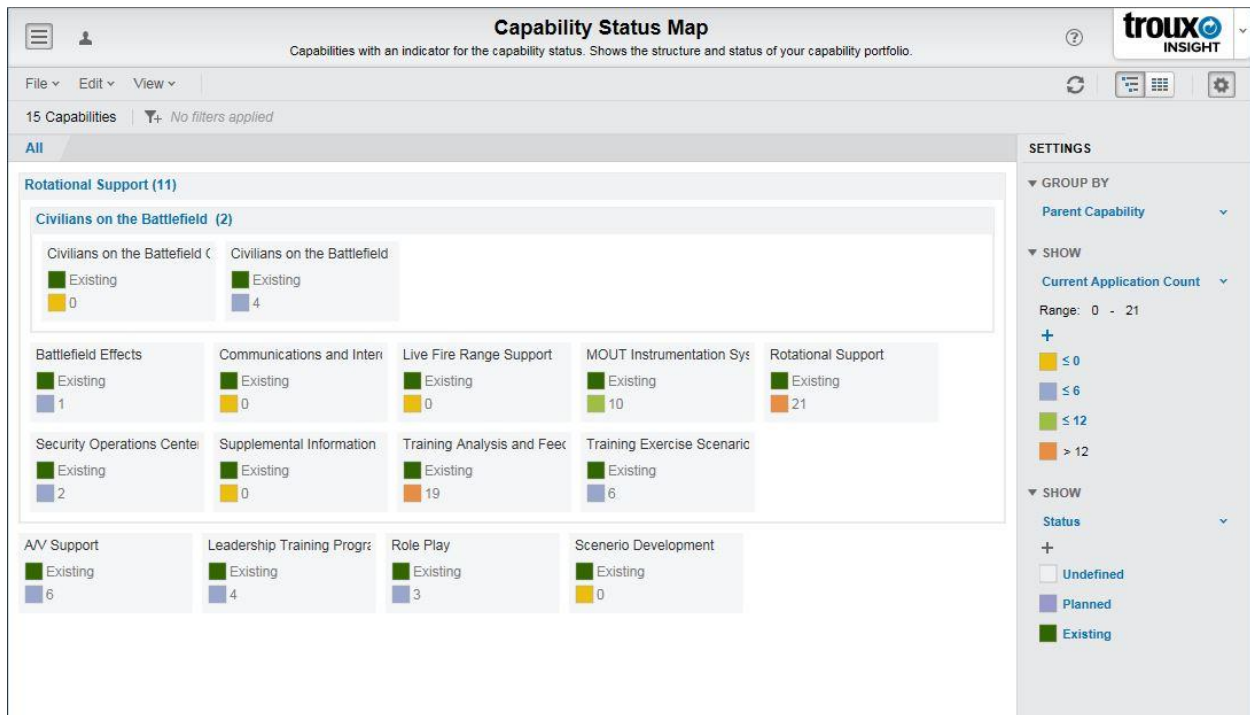
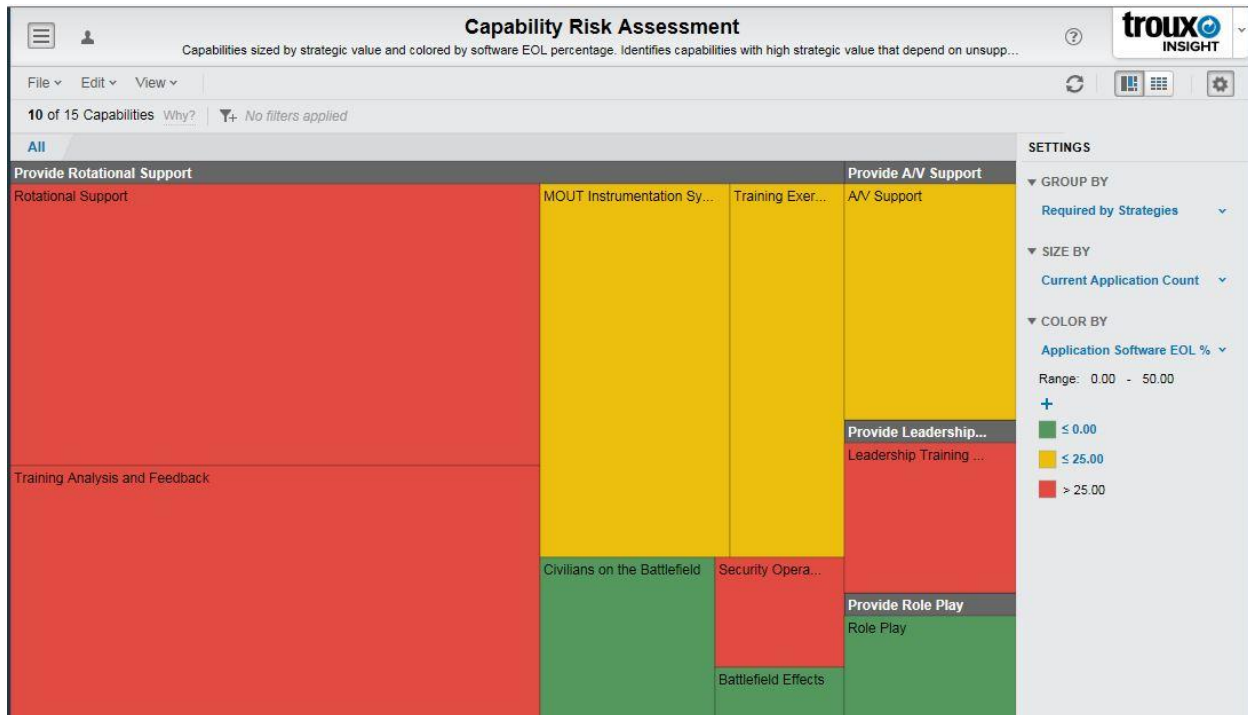


Figure 2. Capability Status display shows lifecycle summary of applications supporting each capability

Grouping by parent capability allows for all to be shown using a single standard screen. They are colored so that capabilities requiring a large number of applications can be recognized. Capabilities without supporting applications or where the supporting applications have not yet been entered in the system are shown in yellow.

### Well Received Perspectives

After loading data over the course of several months, the attempt to display screens from the initial success plan was explored. The Capability Risk Assessment did not show any data despite the numerous applications which were loaded. It appeared none of the capabilities had an average strategic value and it was not evident how to create one. At that point an alternate perspective was developed to present the relationship between capabilities and the software lifecycle risk. The perspective uses business functions to tie the capabilities to applications. The connections are shown in Figure 3. The size block is based on the count of applications supporting each capability. The color of the block is based on the percentage of the software in-use the manufacturer defines as at end of life.

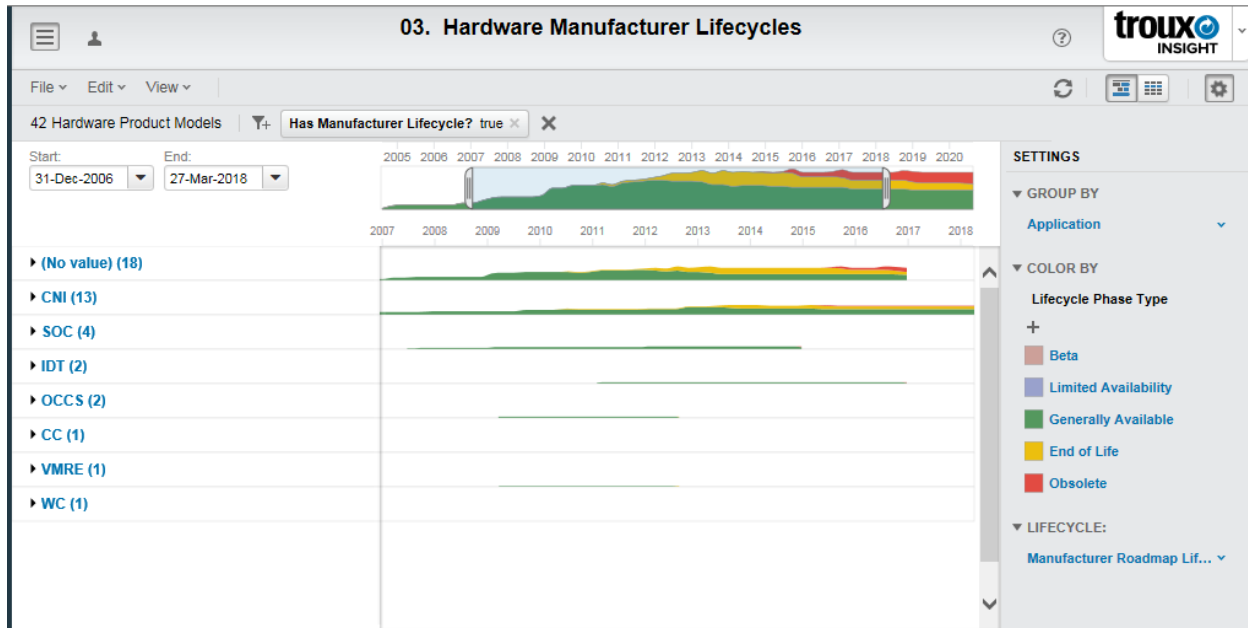


**Figure 3. Capability risk assessment is sized by the number of applications supporting each capability.**

While this view displays the overall risk for any given capability, viewers often request to see the particular software in-use for an application. More importantly, there is a desire to see the overall life cycle for both software and hardware. The question of when products currently supported by the manufacturer would cease to be supported was raised along with when would it be supported, but at end of life and being phased out by the manufacturer.

In order to display this information, data to sustain it was needed in a manner more efficient than individual product lookup on the manufacturer website. To accomplish this BDNA® Technopedia® IT was licensed for use of the project along with software to support loading the results into the Troux repository. Perspectives were then developed to show the current state of hardware or software grouped by application. This provided a forward planning ability allowing life cycle engineers to anticipate when replacement products would be needed for hardware.





**Figure 4. Hardware Manufacturer Life Cycle grouped by Application**

Figure 4 shows years of life cycle data with a consolidation for all the hardware used across the subsystems and for each subsystem. The arrows to the left of each subsystem can be toggled to open up an application list by name along with its particular life cycle plan. The screen is filtered so that only hardware with life cycle information in the repository is shown. This allows engineers to focus on the particular hardware of interest. The slider at the top of the screen, or the date boxes on the upper left, may be used to reduce the time frame in view. Thus the engineers can drill in to specific points in time as well as products.

A corresponding screen is in-place to view internal life cycle data. An editor allows updates of individual internal life cycle data and the entry of waivers or exceptions. In addition, spreadsheets with life cycle data can be entered directly into the repository without having to manually manipulate the data.

As shown in Figure 5, similar data is available for some of the software products in-use at JRTC. The software is not updated in the same manner as hardware. Hardware is updated when a particular product is swapped for a replacement. Software updates generally leave the same product in-place, but update the version number or the patch level. Therefore, software life cycles generally correspond to a particular version or patch level. Manufacturers may cease to support a particular version of the software product while continuing to support newer ones. Maintenance personnel do not necessarily have access to all of the software in-use on a system. Software updates may be released as a batch or part of other entities and installed without supplying the operators with specifics on what product versions are updated and in-use. In most cases, the software information available to support the EA project came from the results of IA scans.

Unlike hardware which wears out after a period of use, software will continue to run over time. Historically software is often considered inoperable if the hardware to run it on is no longer be procured. Through virtualization of machines, software can continue to run even when the hardware for which it was originally created is no longer available. This means software can continue to operate long after its manufacturer's end of life date and quite often it can run when the developers are no longer available to support modifications. Although the software does not pose a risk to regular operations, it does cause an additional challenge for new integration and for updates of other software components which interface with it. Being aware of software which is past the manufacturer's support date, allows future development and upgrade efforts to locate newer versions, switch to other software products, or build an interface to isolate the older components. As a result, the process enables new software to be installed with fewer issues.

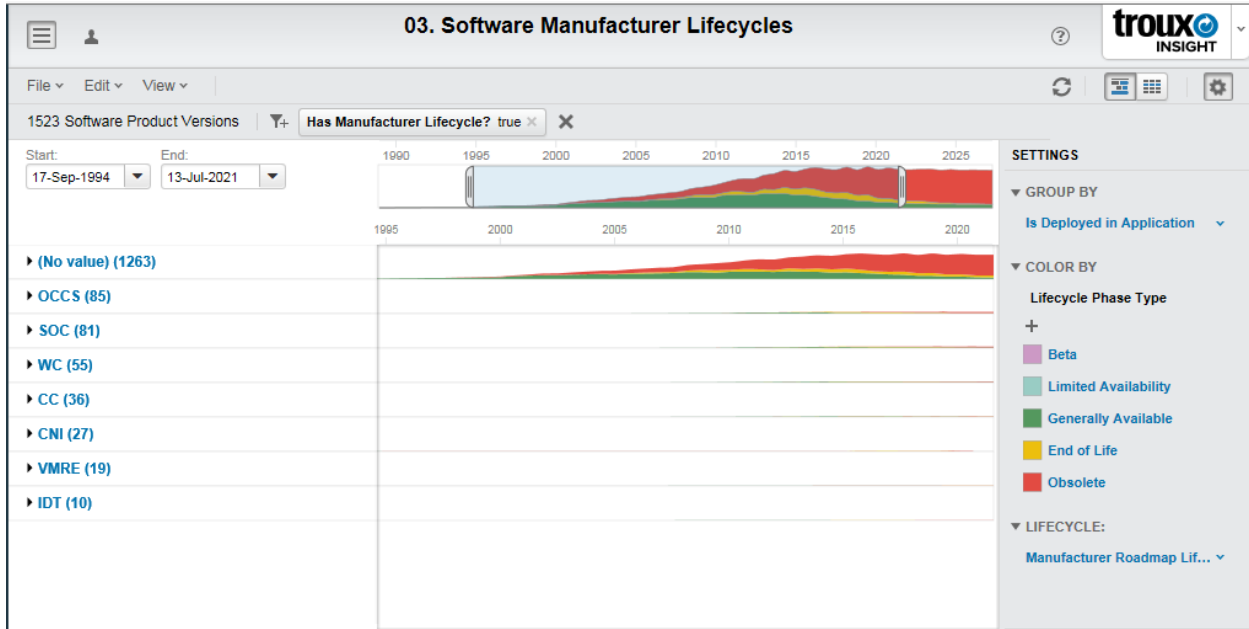


Figure 5 Software Manufacturer Life Cycle over Time

**Objectives Needing Further Investigation**

Some success was achieved in producing the Application Costs – Technology Rollup view, but there was not sufficient detailed data to support a breakdown of cost to the application level. In general, the same labor pool supported multiple applications. Software license costs were not available to the personnel working on the project. Hardware costs were available, but with virtualization often the same hardware supports multiple applications. As one organization is procuring and maintaining all the applications, a methodology for distributing costs across applications had not been developed.

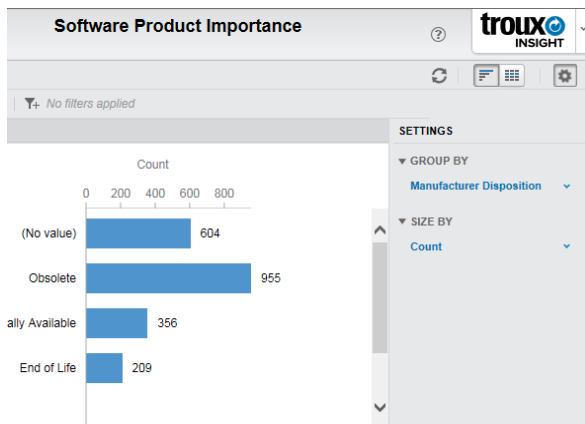


Figure 6. Count of Software Products by Life Cycle Phase

The Software Product Importance perspective is shown in Figure 6. From this display it is apparent that a number of software products are considered obsolete by their manufacturers. Although these products have been secured and the systems have authority to operate, additional integration and ongoing changes to the software may be hampered by running code which is no longer supported and may be difficult for modern software to interface with. The frustration with this perspective is the inability to drill into each life cycle stage and see the list of software product versions contributing to the count. Although it is possible to set up an alternate view which filters all the software product versions by life cycle stage and locations of use, moving from one view to the other is not intuitive for people looking for information on current software status.

**Future Directions**

While BDNA® Technopedia® IT provides an efficient means of collecting manufacturer data for IT equipment, there are many components in-use for simulation which are not considered IT equipment. There are several other repositories of manufacturer data that may be integrated to provide a more complete life cycle picture. In particular it would be useful to have information related to audio visual equipment.

Project data has been added to the repository manually. The intent is to automate the collection records and provide additional relationships between the projects and the particular technology affected by each project.

Having successfully demonstrated the utility of EA at JRTC, there are many other training devices which may benefit from additional life cycle planning. Use of this tool at other CTCs, especially the National Training Center (NTC) at Fort Irwin is being planned.

## Conclusion

The independence of hardware and software in modern technology impacts the obsolescence planning process. Historically, when an object was obsolete it was replaced in total. Now hardware can become obsolete, but the software on it is still current, supported, and useful. Alternatively, software may lack manufacturer support and cease to be interoperable even while the hardware running it remains supported by the manufacturer. Hardware may continue to run for many years past the manufacturer end of life date and without any manufacturer sustainment. Software may continue to run long after its publisher has handed it over.

Organizations may decide they no longer want to use certain hardware based on historical reliability. Organizations may choose to no longer want to run certain software based on security issues. In order to make these decisions in an informed manner, organizations need to be aware of all hardware and software installations. This serves as a starting place for determining the cost of phasing out the particular product or products.

The project served as an exploratory journey into some of the ways that legacy hardware and software continue to support live training exercises. Views available through an EA project illustrate how business operations are dependent upon different pieces of equipment and a wide variety of both customer and commercial products used to support the custom software. EA tools prove useful in depicting relationships between hardware and software assets and the mission capabilities delivered.

## REFERENCES

- BDNA® Technopedia® IT. Mountain View, CA: BDNA Corporation. Retrieved from <http://www.bdna.com/products/technopedia/>
- Cole, W. E. (2016). Modelling and Simulation Targeted to Meet Number One Priority of Army's Top General. *Military Technology*, 40(12), 20-20.
- Graves, T. (2008). *Bridging the silos: Enterprise Architecture for IT-architects*. Colchester, Essex, England: Tetradian Books.
- Interface Control Document for the Data Communications Interface/Detection Device Interface for Combat Training Center - Instrumentation Systems (CTC-IS) ICD Revision E.1. (1999). 1999-08.
- JRTC Success Plan*. (2015). Raytheon Internal Document.
- Quartel, D., Steen, M. W. A., & Lankhorst, M. M. (2011). Application and project portfolio valuation using enterprise architecture and business requirements modelling. *Enterprise Information Systems*, 6(2), 189-213. doi:10.1080/17517575.2011.625571
- Rebarick, B. (2013). *Improving DoD Acquisitions Through Performance-Based Contracting*. (3630962 D.Mgt.), University of Maryland University College, Ann Arbor. Retrieved from <https://login.ezproxy.net.ucf.edu/login?url=http://search.proquest.com/docview/1564232794?accountid=10003> ProQuest Dissertations & Theses Global database.
- Reyes, J. J. (2012). Logistics Training at the Joint Readiness Training Center. *Army Sustainment*, 44(5), 46-47.
- Rivera, J., Samper, W., & Clinger, B. (2008). Live Training Transformation (LT2) product line applied standards for reusable integrated and interoperable solutions. *MILCOM 2008 - 2008 IEEE Military Communications Conference*, 1.
- Simpson, H., & Oser, R. L. (2003). Evaluating Large-Scale Training Simulations. *Military Psychology (Taylor & Francis Ltd)*, 15(1), 25-40.
- The Open Group. (1999). *TOGAF Version 9.1*. T. O. Group (Ed.) Retrieved from <http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap01.html>
- Troux Architect (Version 9.3). (2016). Austin, TX: Planview. Retrieved from <http://www.troux.com/enterprise-architecture/products>

- Troux Insight (Version 2.0). (2015). Austin, TX: Planview. Retrieved from <http://www.troux.com/enterprise-architecture/products>
- Troux Navigate. (2015). Austin, TX: Planview.
- Troux Source. (2015). Austin, TX: Planview.
- Zacchino, J. J. J. (2012). Refining the CSSB at JRTC. *Army Sustainment*, 44(6), 16-19.