The Use of Technology Readiness Levels for Software Development

Dr. John Niemela and Dr. Matthew Fisher

The rapid growth of technology is clearly evident in our daily lives, *and* its use is increasing in every aspect of acquisition and development within DOD. Technology enables the Acquisition, Logistics and Technology (AL&T) Workforce to create superior communication and weapon systems that provide warfighters with battle dominance. So strong is the appeal to reap the benefits of technology that it is being inserted before the risk associated with using it has been thoroughly tested and certified. This concern was brought out in a General Accounting Office (GAO) report titled *Better Management of Technology Development Can Improve Weapon System Outcomes.* The report explains how commercial "best practices" ensure that new technology is sufficiently mature to eliminate the possibility of inordinate risk on a product acquisition or development. This article outlines the process developed by the Army to integrate these best practices into the software development process.


Connectivity versus risk? So strong is the appeal to reap the benefits of technology that it is being inserted before the risk associated with using it has been thoroughly tested and certified.

The GAO report cited two conditions that were absolutely critical to reducing resource and schedule risk attendant with the use of new technology. First, a science and technology organization is where the technology management should be located. It provides the environment for maturing technologies as opposed to an organization that concentrates on the cost, schedule and performance aspects of producing products. Second, technology and program managers must be supported with the discipline, processes, readily available information, readiness standards and authority to ensure technology is ready for integration into the system acquisition or development.

The GAO report recommended that DOD adopt methods to assess the maturity and readiness of technology prior to commitment to system acquisition and development. One method recommended to DOD was using Technology Readiness Levels (TRLs) as a means of managing new technologies when incorporating them into system acquisition and development. The Army responded to this imperative by informing its research and development (R&D) centers that the TRLs would serve as yardsticks for assessing technology maturity and potential use in system development and demonstrations. The general notion of TRLs in the context of the technology transition process is shown in the figure on Page 12. As noted in this figure, the Army Research, Development and Engineering Center's (RDEC's) critical mission is to manage technologies from applied research to facilitate technology transition to the systems development and acquisition community at a minimum TRL of 6.

The *Better Management of Technology Development Can Improve Weapon System Outcomes* report explains how commercial "best practices" ensure that new technology is sufficiently mature to eliminate the possibility of inordinate risk on a product acquisition or development.

TRLs for Software

Coincidentally, with the emphasis on lessening the risk in technology transition, revolutionary technical and operational concepts were rapidly emerging — to include network-centric warfare — that we are heavily dependent on computer software. The TRLs in place at the time were difficult to apply to technology that was primarily based on software — they were quite platform-centric.

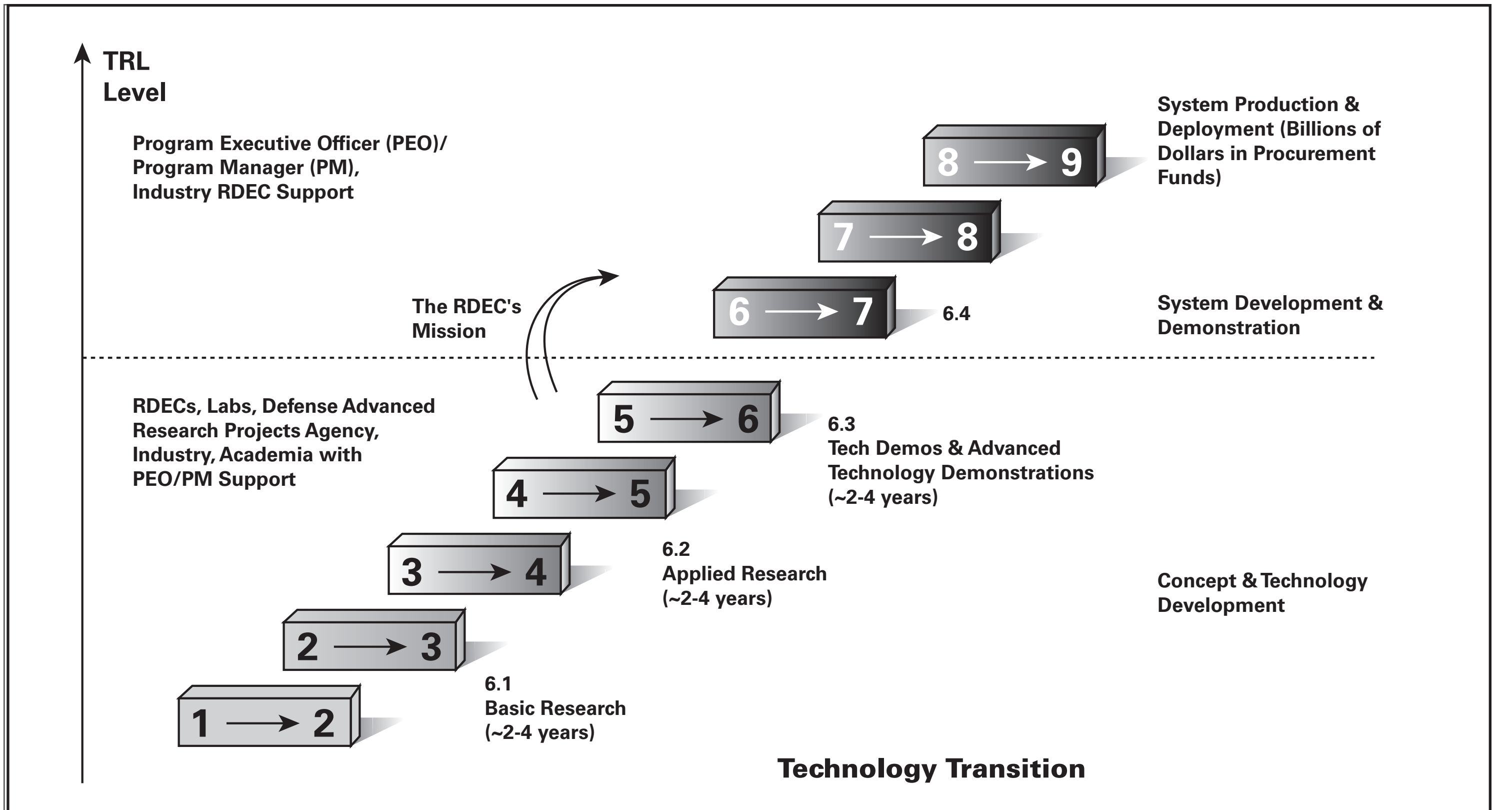
To address this difficulty, the Communications-Electronics Command's (CECOM's) Research, Development and Engineering Center (CERDEC) was requested by HQDA to conduct an intensive investigation and propose a solution that would allow TRL concept applications to systems employing software. A team was assembled, in virtual space, with representation from the Software Engineering Institute (SEI), Army Research Laboratory, Simulation and Training Command and CECOM Software Engineering Center. The result was a set of TRLs for software — compatible with those for hardware — that were completed in time to be applied to the extensive analyses preceding the Future Combat Systems Milestone B Decision in May 2003.

As shown in the following text, TRLs for both hardware and software systems are measured along a scale of one to nine, starting with basic concept studies proceeding to laboratory demonstrations and ending with technology that has proven itself in the military and/or operational environment.

TRLs for Hardware (HW) and Software (SW)

TRL 1. Basic principles observed and reported.

HW/System or subsystem (S): Lowest TRL. Scientific research begins to



be translated into applied R&D. Examples might include paper studies of a technology's basic properties.
SW: Lowest level of software readiness. Basic research begins to be translated into applied R&D. Examples might include a concept that can be implemented in SW or analytic studies of an algorithm's basic properties.

TRL 2. Technology concept and/or application formulated.
HW/S/SW: Invention begins. Once basic principles are observed, practical applications can be invented. Applications are speculative and there is no proof or detailed analysis to support the assumptions. Examples are limited to analytic studies.

TRL 3. Analytical and experimental critical functions and/or characteristic proof of concept.
HW/S: Active R&D is initiated. This includes analytical studies and laboratory studies to physically validate analytical predictions of separate technology elements. Examples include components that are not yet integrated or representative.
SW: Active R&D is initiated. This includes analytical studies to produce code that validates analytical predictions of separate SW elements. Examples include SW components that are not yet integrated or representative but satisfy an operational need. Algorithms run on a surrogate processor in a lab environment.

TRL 4. Component and/or breadboard validation in lab environment.
HW/S: Basic technological components are integrated to establish that they will work together. This is relatively "low fidelity" compared to the eventual system. Examples include integration of "ad hoc" hardware in the lab.
SW: Basic SW components are integrated to establish that they will work together. They are relatively primitive with regard to efficiency and reliability compared to the eventual system. System SW architecture development initiated to include interoperability, reliability, maintainability, extensibility, scalability and security issues. SW integrated with simulated current/legacy elements as appropriate.

TRL 5. Component and/or breadboard validation in relevant environment.
HW/S: Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements so that they can be tested in a simulated environment. Examples include high-fidelity lab integration of components.
SW: Reliability of SW ensemble increases significantly. The basic SW components are integrated with reasonably realistic supporting elements so that they can be tested in a simulated environment. Examples include high-fidelity lab integration of SW components. System SW architecture established. Algorithms run on a processor(s) with characteristics expected in the operational environment. SW releases are "Alpha" versions and

configuration control initiated.
VV&A initiated.

TRL 6. S model or prototype demonstration in a relevant environment.
HW/S: Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in technology's demonstrated readiness. Examples include testing a prototype in a high-fidelity lab environment or in a simulated operational environment.
SW: Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in SW demonstrated readiness. Examples include testing a prototype in a live/virtual experiment or in simulated operational environment. Algorithm run on processor or operational environment integrated with actual external entities. SW releases are "Beta" versions and configuration controlled. SW support structure in development. VV&A in process.

TRL 7. System prototype demonstration in an operational environment.
HW/S: Prototype near, or at, planned operational system. Represents a major step up from TRL 6, requiring demonstration of an actual system prototype in an operational environment, such as an aircraft, vehicle or space. Examples include testing the prototype in a test bed aircraft.
SW: Represents a major step up from TRL 6, requiring the demonstration of an actual system prototype in an operational environment, such as in a command post or air/ground vehicle. Algorithms run on processor of the operational environment integrated with actual external entities. SW support structure in place. SW releases are in distinct versions. Frequency and severity of SW deficiency reports do not significantly degrade functionality or performance. VV&A completed.

TRL 8. Actual system completed and "flight qualified*" through test and demonstration.
HW/S: Technology has been proven to work in its final form and under expected conditions. In almost all cases, TRL represents the end of true system development. Examples include developmental test and evaluation (T&E) of the system in its intended weapon system to determine if it meets design specifications.
SW: Software has been demonstrated to work in its final form and under expected conditions. In most cases, this TRL represents the end of system development. Examples include T&E of the SW in its intended system to determine if it meets design specifications. SW releases are production versions and configuration controlled, in a secure environment. SW deficiencies are rapidly resolved through support structure.

* Qualification attributes include reliability, maintainability, extensibility, scalability and security.

TRL 9. Actual system "flight proven" through successful mission operations.
HW/S: Actual application of the technology in its final form and under mission conditions, such as those encountered in operational T&E. In almost all cases, this is the end of the last "bug-fixing" aspects of system development. Examples include using the system under operational mission conditions.
SW: Actual application of the SW in its final form and under mission conditions, such as those encountered in operational T&E. In almost all cases, this is the end of the last bug-fixing aspects of system development. Examples include using the system under operational

> The GAO report recommended that DOD adopt methods to assess the maturity and readiness of technology prior to commitment to system acquisition and development.

> However, TRLs should not be considered the panacea to eliminate technical risks associated with acquisitions or developments. Simply, TRLs provide additional information to allow managers to make more informed, programmatic decisions for their projects/programs.

mission conditions. SW releases are production versions and configuration controlled. Frequency and severity of SW deficiencies are at a minimum.

System functionality is demonstrated in environments of increasing realism. Initially, at the basic research level, the possibility of new capabilities may only be that of conjecture. As the concept matures, its feasibility is demonstrated in "laboratory" environments approaching that of actual field environments. Software integration is successively accomplished with other system hardware and software components as follows:

- *Verification, Validation and Accreditation (VV&A).* VV&A, often very resource-intensive, is staged as it becomes evident that the software is to be fielded. Verification and validation helps improve software quality and maturity. This cannot be accomplished without documenting and "baselining" the software products.
- *Configuration Management.* Is essential for tracking and coordinating development of all software components on a common baseline, as well as preventing unauthorized access and automatically alerting users when a component has been altered.
- *Software Deficiency Reports.* The frequency and severity of software deficiencies are documented in reports that result in corrective actions.
- *Software Release Documentation.* Knowledge gained from user experience with beta and alpha software version releases is documented and "fed back" to technology development for incorporation into subsequent releases.
- *Software Development.* Early in the process, a surrogate processor that may have functional characteristics such as throughput, but not form factor or environmental characteristics, can be used. As the software matures, the "run-time" software is executed on the intended processor to take advantage of the size, weight, power, cost and performance benefits of "Moore's Law."

TRLs are important tools that the R&D community can use to plan, prioritize and allocate resources to assure that their technology — hardware and software — is suitable for transition to systems level development. Similarly, TRLs are valuable to the AL&T Workforce allowing more complete assessments of, and better decisions on, the technology that underlies their system acquisition and development projects.

However, TRLs should not be considered the panacea to eliminate technical risks associated with acquisitions or developments. Simply, TRLs provide additional information to allow managers to make more informed, programmatic decisions for their projects/programs.

The U.S. Army has seriously embraced the management practice of making major programmatic decisions based on disciplined approaches for assessing technology maturity, and recommending only that technology which is ready for transition to proceed to system level development. With an ever-increasing dependency on software, acquisition program managers cannot overlook the risks associated with software development. TRLs for software are critical for program managers to make informed, programmatic decisions that lower acquisition and development risks.

DR. JOHN NIEMELA was the Chief Scientist of the Integrated Battle Command Directorate at CERDEC when this article was written, but has since retired. He earned a B.S. in electrical engineering from Carnegie Mellon University and an M.S. and Ph.D. in electrical engineering from the University of Pennsylvania.

DR. MATTHEW FISHER is a Visiting Scientist at SEI. He earned a B.S. and Ph.D. in electrical engineering from Drexel University and an M.S. in engineering from the University of Pennsylvania.