BRIEFING PAPERS® SECOND SERIES

PRACTICAL TIGHT-KNIT BRIEFINGS INCLUDING ACTION GUIDELINES ON GOVERNMENT CONTRACT TOPICS

The Agile Contractor Guide To DoD Data Rights

By David L. Bodner[*]

Agile software development is the current dogma for the U.S. Department of Defense (DoD) software procurements and by 2018 nearly all DoD software development projects were declared to be "agile."[1] Agile software development breaks development into smaller segments of working functionality, which are iteratively provided to DoD users for feedback. However, this iterative user feedback can lead to providing proprietary software to the DoD without clarity as to the DoD's license right. Contractors and the DoD should mitigate this uncertainty or no man's land of license rights through use of an access agreement or Agile license agreement.

While the method the DoD uses to develop software has changed, the need for contractors to understand the Defense Federal Acquisition Regulation Supplement (DFARS) computer software regulations has not. Contractors that wish to grow and protect their DoD business must understand how the DFARS regulations apply to Agile software development to protect their proprietary software while also providing essential capability to their DoD customers. This BRIEFING PAPER provides a basic background on Agile software development, a primer on the DFARS licensing regulations, and then pre-award and post-award tips for protecting proprietary software when performing Agile software development.

Software Development Methods: From Waterfall To Agile

Starting in the 1980s, the DoD used the waterfall software development method.[2] Under the waterfall method, the DoD specifies detailed and fixed software requirements up front for the contractor to progress through an orderly series of sequential stages: design, test, and deployment to the DoD customer only upon completion.[3] Since the DoD does not get the software until it is finished, the DoD would require significant Contract

---

[*]*David L. Bodner is an associate with Blank Rome LLP. Mr. Bodner's practice encompasses all areas of government contracting with a focus on regulatory compliance, data rights, claims, government investigations, and bid protest litigation.*

THOMSON REUTERS®

Data Requirements List (CDRL) deliverables to track progress. Due to the fixed requirements and sequential stages, the waterfall method assumes very few changes during the entire software development lifecycle.[4] In practice, since requirements are not always fully documented, understood, or even fixed, the waterfall method resulted in significant changes to schedule, performance, and cost.[5]

Understanding the problems associated with waterfall helps set the stage for Agile. In contrast to the waterfall method, the Agile software development method does not assume the user can understand all the requirements up front but rather assumes that change is constant.[6] Unlike waterfall where the requirements are fixed, in the Agile method, the schedule and cost are fixed and the requirements are variable for each iteration.[7] The U.S. Digital Service's TechFAR Handbook defines Agile as "a method of software development that is based on iterative and incremental processes and collaboration among a team."[8] More specifically, the Agile method is how the contractor develops software. Agile is defined by four values and 12 principles.[9] The DoD Innovation Board translated the Agile values to its own maxims in the following chart:[10]

Agile value	DIB maxim
Individuals and interactions over processes and tools	"Competence trumps process"
Working software over comprehensive documentation	"Minimize time from program launch to deployment of simplest useful functionality"
Customer collaboration over contract negotiation	"Adopt a DevSecOps culture for software systems"
Responding to change over following a plan	"Software programs should start small, be iterative, and build on success - or be terminated quickly"

At its core, the DoD has a desired high-level software end-state and contracts for a collaborative and iterative method without specifying the detailed requirements up front. The high-level software vision is broken down into small segments of functionality. The Agile team then works on the lower-level software requirements and receives frequent feedback from the DoD users. The DoD user feedback is incorporated into the software development to refine the requirement and improve the software. This process is repeated for different functionalities until the high-level software end-state is achieved. There are different variations on the iterative development and continuous feedback process.[11]

Understanding the Agile method is complicated by buzzwords such as "product vision," "sprints," "user stories," and the "definition of done." A frequent analogy to describe the Agile method is building a house. The goal or "product vision" would be a single-family home that provides shelter and comfort. In Agile, the product vision is software that performs the functionality needed by the user.[12] Building a home is a large project, which is composed of thousands of smaller tasks. In Agile, the smallest user requirements are termed "user stories," which for the house could translate to a place to store silverware such as a cabinet. Similarly, in Agile development, the user story is the smallest user functionality such as getting the software to provide a notification when a user checks out a document.[13] To accomplish each user story requires a short cycle of work such as planning the cabinets, installing the cabinets, and testing that the cabinets open and close. In Agile, this is called a "sprint," which is a short unit of work focused on completing a defined subset of functionality using design, development, and testing to demonstrate the software functionality was accomplished.[14]

Finally, the family will want to have a shared understanding of when the cabinets are done, such as meeting

the local county building code or acceptable quality like closing evenly and softly. In Agile development, this is the "definition of done" or an understood definition of the actions for a given user story that must be completed.[15] While the cabinets may meet the definition of done, the family would not take delivery because the rest of the house is not ready. The family would be able to test the functioning of the cabinets and provide feedback before the entire kitchen was installed. Contrast this with a waterfall development where the family would need to specify the entire house from the very beginning down to the soft-close cabinets but would be unable to go into the house to check out the cabinets until the entire house was done.

DFARS: Licenses, Types Of Software, And Delivery

Agile software development may change the frequency of user feedback, but it does not change the DFARS rules for the DoD's license rights in software. DFARS Part 227 provides the prescriptive rules for DoD license rights and DFARS Part 252 provides the solicitation provisions and contract clauses. There are three key principles in the DFARS scheme: (1) the DoD obtains a license instead of ownership; (2) license rights are determined by the source of funds for development; and (3) delivery is necessary for the DoD to exercise license rights. Additionally, the DFARS treats noncommercial and commercial software differently.

The DoD Does Not Own The Software

First, the DoD does not own any of the software developed pursuant to the DFARS data rights clauses.[16] This is true even if the DoD paid for every cent of software development. The broadest right in the DFARS is an unlimited license right, which allows the DoD to share or disclose software "for any purpose whatsoever, and to have or authorize others to do so."[17] However, even with this broad license grant, this does not mean the DoD owns the software. When consumers purchase *My Cousin Vinny* on Prime video or on a DVD, they obviously do not own all the intellectual property (IP) for *My Cousin Vinny* but rather possess a license to watch it. The same is true for the DoD's license under DFARS 227.7203-4(a), which provides a "Grant of

license" and "[t]he contractor or licensor retains all rights in the software or documentation not granted to the Government." Therefore, the DoD gets a license in the software and the license scope determines what the DoD can do with the software.

Development Funding Determines License Rights

Second, the extent that DoD-funded development determines its license rights in the noncommercial software (NCS). The life cycle of an item starts with conception, then proceeds through experimentation and design and eventually production and support.[18] During this process and potentially throughout, software is developed. "Developed" under the DFARS means whether someone skilled in the art can reasonably expect the software or program to perform as intended.[19] This does not require perfection, 100% completion, or even ready for sale. Thus, a software program that is improving reliability in beta testing is likely already developed under the DFARS even though it is not released to all end users. In an Agile development effort where functionality is broken down into smaller segments, each subroutine or functionality will be developed before completion of the entire software.

Importantly, determining who funded development occurs at "the lowest practicable segregable portion of the software."[20] This means finding the smallest portion of functionality (e.g., a subroutine) and seeing whose funds paid to develop that functionality. As one commentator put it, determining funding at the lowest level makes sense because that is how development happens.[21] "A new rocket engine, for example, does not spring fully developed out of the mind of a developer, nor does software . . . . Development does not occur all at once, but rather at these discrete segregable levels."[22]

A common misconception is license right assertions based on the total amount paid rather than considering what the funding paid for. For example, while the DoD may have paid billions to develop the P-8 reconnaissance aircraft, the contractor may have exclusively paid $50,000 to develop the pilot's rotating seat. When looking at it from a high level, the DoD paid a substantially higher sum for development of the P-8 but that does not mean the DoD has unlimited rights in the entire P-8 or

the contractor's rotating seat design. The key is analyzing the funds used for each area of development and not focusing on the total amount paid for development.

Once development funding is determined at the lowest segregable level, the DFARS NCS clause, DFARS 252.227-7014, establishes four Government licenses for NCS: (1) unlimited rights; (2) Government purpose rights; (3) restricted rights; and (4) specifically negotiated license rights. The DoD's license is determined by whether the Government exclusively, partially, or did not fund development at the lowest segregable level.

Source of Funds at the Lowest Level	Government License Rights
Developed Exclusively at Private Expense	Restricted/Limited Rights
Developed with Mixed Funding	Government Purpose Rights
Developed Exclusively at Government Expense	Unlimited Rights
Negotiated	Specially Negotiated Rights

Delivery Is Necessary For DoD To Exercise License Rights

Third, the DoD needs delivery pursuant to a CDRL to exercise its license rights. This is because the DFARS rules separate the acquisition of license rights in software from the delivery of the software. This matches the DFARS policy, which states that the "DoD policy is to acquire only the computer software and computer software documentation, *and* the rights in such software or documentation, necessary to satisfy agency needs."[23] While the Government may have paid for development and have a license right in the NCS, if the Government does not have the NCS to exercise its license, its rights are described as "inchoate" or incomplete.[24]

The Government realizes or can exercise its license rights upon delivery of the software. Contractors deliver software pursuant to express contract terms in exchange for money or other consideration. If the contract does not require delivery of the software, then it is not part of the bargain.[25] The contractor does not include delivery in its price and the contractor does not deliver the software to the DoD. This results in situations where the DoD may have a license right in software without delivery of the same software. The DFARS deferred ordering clause, DFARS 252.227-7027, helps exhibit the concept. When there is no requirement for delivery, this clause provides the Government the opportunity to

order delivery but "the Contractor shall be compensated for . . . reproduction and delivery."[26]

The Government is required by the DFARS and DoD policy to require delivery to exercise its license rights. DFARS 227.7203-1 mandates that solicitations and contracts "shall" "[s]pecify the computer software or computer software documentation to be delivered under a contract and the delivery schedules for the software or documentation."[27] The DFARS policy also requires separate contract line item numbers (CLINs) for computer software delivery so contractors can price it.[28] DoD's recent data rights instruction describes the delivery requirement as a "core principle[]" stating:

> Clearly identify and match data deliverables with the license rights in those deliverables. Data or software deliverables are of no value unless and until the license rights to use it are attached, and the U.S. Government actually obtains and accepts those deliverables.[29]

The DoD uses CDRLs to specify data delivery requirements. A CDRL is a standard format for identifying potential data requirements in a solicitation and deliverable data requirements in a contract.[30] DFARS 215.470(b) instructs: "When data are required to be delivered under a contract, include DD Form 1423, Contract Data Requirements List, in the solicitation." The DoD policy requires the use of a CDRL by stating: "The CDRL provides a contractual method to direct the contractor to prepare and deliver data that meets specific approval and acceptance criteria."[31] CDRLs provide critical terms to remove ambiguity. The CDRL incorporates a Data Item Description (DID) that describes the content, format, and intended use of data.[32] Block 4 of the CDRL provides the DID code that can be used to retrieve the full document on the Defense Logistics Agency's (DLA's) website.[33] Without a CDRL, the contract may not include critical details of the requirement and contain an ambiguous term such as "deliver developed software." However, this creates a significant ambiguity as to what constitutes the required software or version, the scope of the software (object versus source code), the format, and many other details critical to reaching a meeting of the minds.

Putting it all together, in an Agile development effort, to exercise license rights defined in DFARS 252.227-7014, the DoD needs delivery of the NCS pursuant to a

CDRL. The Agile method changes the process for developing software, but it does not change the DFARS Part 227 prescriptions, DFARS Part 252 clauses, the DoD policy, or the necessity for delivery of NCS via a CDRL. However, for Agile development efforts under DoD's other transactions authority,[34] the DFARS are inapplicable, permitting the DoD and the performer to negotiate license rights under the agreement from scratch.

Commercial computer software has different DFARS rules than NCS. However, if the commercial software on its own satisfies the DoD's needs, the DoD would not need the Agile method to develop software that already exists. Thus, it is likely to form a part of the development due to the quality of commercial software and prevalence. The DFARS does not provide a specific contract clause governing the Government's rights in commercial computer software.[35] Rather, the DFARS instructs the DoD personnel to acquire commercial computer software "under the licenses customarily provided to the public unless such licenses are inconsistent with Federal procurement law or do not otherwise satisfy user needs."[36] Because the standard commercial license is typically written by the developer to benefit that developer, they are almost always more protective of the developer's proprietary software than either the unlimited rights or Government purpose rights license.[37] The limit is that a commercial license cannot contain terms inconsistent with federal procurement law.[38]

If the Government needs a greater license than what is "customarily provided to the public," the DoD can negotiate with the contractor to obtain greater license rights.[39] There is a clear policy preference for the DoD to fulfill its requirements with commercial products.[40] Additionally, contractors should seek to classify their software as commercial to the greatest extent possible to reap the benefits of writing their own license as opposed to the DFARS required license terms.

Open Source Software (OSS) may also be used in a DoD Agile software development project. OSS is "software for which the human-readable source code is available for use, study, re-use, modification, enhancement, and re-distribution by the users of such software."[41] Nearly all OSS is commercial computer software because if it has at least one non-Governmental use and is licensed to the public, it is commercial software.[42] What makes OSS unique is that it is publicly available. Some OSS license terms require as a condition of the license that the derivative work is also made available to the public at no charge. These "viral" clauses can cause issues for DoD and contractors through public disclosure of any derivative work using the OSS. Therefore, the OSS license terms should be carefully evaluated before creating derivative software from it.

Agile Development Pre-Award Tips

Protecting a company's proprietary software starts at development even before a solicitation is released. The following provides pre-award practical considerations for protecting company proprietary software for an Agile development solicitation.

Negotiating Access To Remove The Need For CDRLs

In an Agile development effort, the Contractor provides a DoD user source code or software artifacts so the user can provide feedback. Often, the contractor provides the DoD user the software, which was not required by a CDRL. The DoD's license rights are unclear, and the exchange may occur outside the DFARS structure. Since there was no required CDRL delivery, there is no DFARS 252.227-7017 assertion intended to provide notice of any restrictions before award.[43] The contractor may place a proprietary company marking on the source code while the DoD asserts that it exclusively paid for development and has unlimited rights to share the source code with whoever it pleases. This impending dispute could happen hundreds of times over the course of an Agile development effort. This potential dispute is unnecessary and avoidable. The DoD and the contractor should negotiate an access agreement or an Agile license agreement providing clear terms for the DoD's use of source code or software artifacts it gains access to during the contract.

DFARS Case 2018-D018: Alternatives To Formal CDRL Deliveries

The DoD proposed a change to the DFARS and the DFARS Procedures, Guidance, and Information (PGI) to implement § 871 of the National Defense Authoriza-

tion Act for Fiscal Year 2018, Pub. L. No. 115-91.[44] This proposed change seeks to improve DoD planning and negotiation of software deliverables and license rights before contract award through several "alternatives to the delivery of source code and related software design details."[45] One major alternative is providing the DoD access to the software rather than a formal delivery. The proposed DFARS PGI states:

> (C) For each technical data or computer software requirement, the requiring activity must determine whether the Government's needs can be better satisfied through access or formal delivery of the technical data or software (e.g., physical or electronic transfer of the data into Government custody). The Government should consider access to technical data or software when—
>
> (1) The delivery of technical data or software is not cost-effective or feasible for technical, legal, or contractual reasons; and
>
> (2) Access meets the Government's needs for such technical data or software throughout the life-cycle of the program.[46]

This proposed DFARS change provides valuable options to DoD personnel to find creative solutions to computer software licensing issues. However, the DoD and contractors do not need to wait until finalization of the DFARS change to implement Congress' 2018 authorization.

Access Agreements Remove Uncertainty And Foster Collaboration

An "access agreement" is a negotiated agreement between the DoD and contractor whereby the contractor provides some form of access to technical data or computer software stored within a contractor-controlled repository or facility.[47] The scope of DoD access is subject to negotiation but the agreement can permit the "Government to view, print, download, annotate, modify, or make derivatives of technical data or computer software" and should be incorporated into the contract.[48]

An Agile development effort generates significant software artifacts, and a repository can serve as a centralized storage location that controls access and versions of software. However, since the DoD may need to view the software or software artifacts to provide feedback, the DoD and the future contractor can negotiate the DoD's access to the contractor's repository. This access agreement will mitigate the need to develop hundreds of CDRLs for iterative data the DoD does not need and may become obsolete. The agreement can also provide certainty and limits on the DoDs use for when the DoD user provides feedback during the contract.

The access agreement can set forth what artifacts or software the DoD can access, how the DoD can access the software, what the DoD can do with the software accessed, limits on derivative work, and the duration of the access pursuant to contract. The DoD and the contractor can negotiate tailored license rights that fit the project and appropriately limits the DoD's ability to disclose the data it has access to. The access agreement provides set limited license rights to a defined set of data that is incorporated into the contract. The DoD must follow the set license or run afoul of the Trade Secrets Act.[49] Critically, the access agreement should define that access is not delivery under the contract and the contractor should mark the accessed data with a company proprietary marking rather than a DFARS legend to distinguish between data the DoD accesses and data that is delivered under the DFARS.

For instance, a contractor could negotiate a license that provides the DoD broad access to view but not download artifacts within the repository for feedback purposes for the duration of the contract. Using an access agreement can facilitate collaboration while also satisfying the DFARS policy of the DoD only acquiring rights "necessary to satisfy agency needs."[50]

Using An Agile License Agreement Can Clarify DoD License Rights

During an Agile software development effort, the DoD may not need CDRL deliveries of each iterative piece of source code if the user has access to provide feedback. The DoD may have its own repository for continuous integration and continuous delivery but may not want to specify a CDRL delivery for every single iteration of code. Moreover, the DoD Agile guidance contemplates saving administrative effort and cost through reduced CDRLs because the DoD receives "evidence of performance in the form of working software."[51] However, without an agreement defining

the DoD's license rights, there could be disputes about the DoD's ability to use or disclose software artifacts it receives outside of a CDRL during development. Moreover, due to the speed of Agile development, customer relationships, or employee training, a contractor risks failing to mark the software artifacts provided to the DoD, which may result in dissemination beyond what was intended.

An Agile license agreement can provide a negotiated solution for a wide swath of designated software or even de facto license rights for not yet identified software or software artifacts the DoD receives access to outside of a CDRL delivery. The agreement can define the Government's use, disclosure (e.g., no disclosure or a limited group of other contractors on the project), rights in derivative work, return or destruction of accessed software, and even include a set marking to distinguish from the DFARS legends. Critically, the Agile license agreement can remove uncertainty and foster collaboration with the Government and contractor. These clear negotiated terms can be incorporated into the contract to provide a meeting of the minds on license rights.

In the event a contractor does not enter into an Agile license agreement, at a minimum, a contractor should conspicuously place the company proprietary markings on the software provided to the Government outside of a CDRL. A base marking subject to further company customization can include the following:

> Information contained herein was not delivered under the DFARS 252.227-7013, DFARS 252.227-7014, a Contract Data Requirements List, or under any term in any contract with the U.S. Government. Additionally, the information contained herein is proprietary to Company A, is submitted in confidence, and is privileged and exempt from disclosure by the U.S. Government under paragraph (b) of the Freedom of Information Act (5 U.S.C. § 552) and subject to 18 U.S.C. § 1905).

Failure to mark the software might result in the giveaway of the company's proprietary rights to the DoD and third parties, which can happen in an instant. First, the marking limits the DoD's ability to disclose the software. Second, the marking can help protect the likely trade secret and/or copyright status of the software. Third, the marking should not be a DFARS legend since the software was not delivered pursuant to a CDRL and the contractor marking should state it was not delivered under the contract.

The DoD may not have funded any portion of development, have no CDRL requirement, and is entitled to restricted license rights. However, the DoD may request receipt of the data as part of a definition of done or just to check that the code works. An Agile license agreement can remove the ambiguity regarding the DoD's license rights in the software when it receives the software. This can help mitigate potential miscommunications, unauthorized uses, or extra contractual agreements. Moreover, it serves the DOD goal of providing notice to the parties of the license rights early in the process. Contractors also benefit from the certainty, protection of their proprietary software, and open collaboration within set license right ground rules.

Understand Nonstandard Clauses That May Affect The License Grant

The FAR and DFARS clauses are considered standard because they went through the regulatory notice and comment process. However, a solicitation is not limited to FAR and DFARS clauses. Offerors should carefully review an Agile solicitation for any nonstandard or special clauses that may affect the contractor's license grant to the DoD. Three common areas where solicitations may contain special contract terms are the definition of done, commercial software clauses, and requirements for a minimum license grant.

First, an Agile software development solicitation may provide a definition of done. This definition of done defines the documentation for user stories, acceptance criteria, code quality, and standards compliance.[52] Contractors should carefully evaluate the definition of done term to determine the relation to a CDRL delivery, which is necessary for the DoD to exercise its DFARS license rights for NCS. The definition of done term may operate independently of CDRLs or may include a CDRL delivery for the definition of done. If the definition of done term is tied to a CDRL, contractors need to assert DFARS 252.227-7017 restrictions, mark all definition of done deliveries with DFARS legends, and price this iterative CDRL delivery requirement. Additionally, the CDRL DID should align with the definition of done such as defining the same content requirements to meet the definition of done.

If the definition of done is not associated with a

CDRL delivery, the DoD is creating uncertain license rights, if any, in viewing the software to determine if it meets the definition of done. A contractor may want to propose an Agile license agreement to provide certainty and limitations to the DoD's license right as part of the definition of done. There is no bright-line rule that the definition of done requires a CDRL delivery or not, which is why it is so important for offerors to carefully evaluate the definition of done term and provide feedback on the solicitation when provided an opportunity.

Second, the solicitation may include a custom provision or clause governing commercial software since the DFARS provides for relying on commercial terms available to the public rather than prescribing a DFARS clause.[53] Depending on the DoD program needs, the solicitation may request to review the commercial license agreement, notice of all the planned use of commercial software, or adapting DFARS 252.227-7017 to provide assertions for the commercial software. In a situation adapting the DFARS 252.227-7017 clause, a contractor would want to assert its restrictions but also provide the DoD's license grant in the commercial license terms. While DFARS 252.227-7014 limits the situations in which an offeror would be able to make new assertions after award, the restrictions in this clause would be inapplicable to commercial software so a contractor could provide new assertions after award.

Critically, there may be language in the solicitation that impacts an offeror's ability to receive award for commercial software license terms that the DoD interprets as contravening federal procurement law.[54] While the exact terms that contravene federal procurement law do not have a DoD-wide definition, general areas include disputes clauses using state courts or arbitration as opposed to the Armed Services Board of Contract Appeals or U.S. Court of Federal Claims, automatic renewal or indemnity clauses that may violate the Anti-Deficiency Act, and payment clauses that conflict with the Prompt Payment Act to name a few. Offerors need to carefully dissect special clauses regarding commercial software license rights to protect the commercial software from inadvertent license grants.

Finally, an Agile solicitation may include a contract clause requiring delivery of software with no less than unlimited rights and elsewhere the solicitation requires unconditional assent to the terms and conditions of the solicitation to be eligible for award. The DoD goal is to only receive unlimited rights data for ease of administration and program sustainment. However, these terms go against the DFARS policy to avoid discouraging offerors from providing computer software with restricted rights.[55] Second, this combination of terms may violate 10 U.S.C.A. § 3771(b)(8) or more specifically DFARS 227.7203-1(c), by requiring as a condition of award, agreement to deliver unlimited license rights, when the DoD may be entitled to less. The DoD and offerors are better served without a bright-line term setting a minimum threshold for the DoD's license rights. For the DoD, there is a risk of a pre-award protest. More likely, the terms deprive the DoD of delivery of valuable software with less than unlimited rights, such as Government purpose or restricted rights software, which could still satisfy the DoD's mission. The clause is also problematic for a contractor that spent valuable time and resources developing the software. The contractor may not want to grant the DoD the ability to share its development achievements with competitors for commercialization. As a result, the company may forgo submitting a proposal, limiting one of the DoD's prime goals of competition and the company's goal of growing its business. Asking a timely solicitation question to resolve the issue before proposal submission to correct and highlight the term could result in an adjusted term and increased competition.

Engage The DoD To Remove Unnecessary Clauses, Requirements, And Agile BS

As the DoD attempts to shift its mindset toward Agile software development, contractors and the DoD should leverage collaborative aspects already available in the acquisition process. The DoD can employ an Agile mindset early by using draft solicitations for industry feedback. Pre-solicitation feedback directly from industry is not prohibited. Rather, it is encouraged. FAR 15.201 encourages exchanges of information to include contract type, terms and conditions . . . statements of work, and data requirements. The OFPP Myth-Busting series provided:[56]

4.	Misconception - "Agencies generally have already determined their requirements and acquisition approach so our impact during the pre-RFP phase is limited."
	Fact - Early and specific industry input is valuable. Agencies generally spend a great deal of effort collecting and analyzing information about capabilities within the marketplace. The more specific you can be about what works, what doesn't, and how it can be improved, the better.

The OFPP specifically advised agencies to issue a draft solicitation to obtain comments and suggestions from potential vendors on how to improve the solicitation. It also busted another myth by stating: "Simply providing suggestions and comments prior to formal requirements development will not trigger an organizational conflict of interest, as long as the vendor is not then hired to develop the requirements."[57]

Industry should seize on the opportunity to provide feedback on a DoD draft solicitation. For instance, industry can identify CDRLs made obsolete by the Agile method.[58] A Software Test Report CDRL, Data Item Description DI-IPSC-81440A, may not be the most effective deliverable when the performance work statement requires testing software configuration items as part of a continuous integration continuous delivery with instantaneous results. These CDRLs can slow and detract from software development. The GAO Agile guidance specified that "[d]ue to the anticipated close and continuous work coordination between the government and contractors, the number of formal deliveries for contract data requirements list may be less than what is collected for a traditional IT acquisition."[59] Offerors can advise on other CDRL tweaks such as minimizing the frequency of delivery to provide the DoD only the code it needs and not the iterative and soon to be obsolete versions. This early collaboration can result in a solicitation focused on software development as opposed to redundant administration.

Industry can also advise the DoD on clauses pre-award that are used inappropriately, limit competition, or inapplicable to the acquisition. For instance, industry should notify the DoD when it includes DFARS 252.227-7030, "Technical Data—Withholding of Payment," and DFARS 252.246-7001, "Warranty of Data—Basic," that these clauses will only apply to technical data and not computer software. Additionally, industry can advise DoD on the appropriate use of DFARS 252.227-7026, "Deferred Delivery of Technical Data or Computer Software." Rather than simply inserting the clause into a solicitation without specifying the intended data for deferred delivery, industry can advise the DoD that it needs to identify the software or technical data for which DoD is seeking deferred delivery. Additionally, an appropriately structured deferred delivery clause can limit industry from delivering software for which the DoD does not need because the iterative Agile development process resulted in improved versions of the code.

Offerors as the industry experts can provide valuable redlines to the solicitation to make the solicitation a better product for both parties. Offerors could identify a proposed six-month sprint as too long or suggest corrections to make a project Agile.[60] Offerors can advise the DoD of any outdated standards in the solicitation. Offerors can also advise of any ambiguity in the solicitation terms such as the relation of any special clauses to CDRL requirements. Offerors can advise of missing teams or technical expertise to accomplish the product vision. If the DoD has an Agile framework attachment, industry can provide feedback to improve it. Industry input is critical to setting up a fair solicitation and successful contract performance.

DoD is moving the requirement to plan IP strategies earlier so DoD should engage industry, the entity whom the DOD seeks a license from, as early in the process as possible. As DoD shifts its mindset toward Agile, the acquisition process can leverage the collaborative aspects permitted by statute and regulation.

Effectively Use The Data Rights Assertions List

DFARS Part 227 requires early identification of any software or software documentation to be delivered to the DoD with restrictions on use.[61] The DFARS 252.227-7017 (Data Rights Assertion List) provision is completed by the offeror and its subcontractors as part of the proposal, to identify any computer software or computer software documentation with restrictions on use, modification, release, disclosure.[62] Critically, the provision only applies to data which will be *delivered* under the contract.[63] This list of contractor assertions on intended delivered software is then incorporated into the contract.[64]

In an Agile development solicitation, appropriately

completing the DFARS 252.227-7017 list provides the contractor the ability to use restrictive markings on CDRL deliveries. Offerors should first scan the solicitation for any CDRLs, carefully unpacking the CDRL DIDs to understand the content of the requested delivery to evaluate any proprietary information implicated. Offerors then should assert appropriate restrictions at the lowest practicable segregable portion of the software.[65] Notably, the DFARS advises against DoD challenges to asserted restrictions prior to a competitive contract award unless resolution of the assertion is essential for successful completion of the procurement.[66]

In an Agile solicitation, an agency's high-level product vision should not change. However, "Agile software development expects and anticipates changing technical requirements within the agency's high-level vision or need."[67] Offerors can look to the product vision to help understand the scope of the project and deliverables for which assertions are required.

Due to the less prescriptive nature of an Agile solicitation technical requirement, "new information" after award may cause a contractor to provide new assertions. The solicitation may contain a broadly written CDRL that the DoD in contract administration asserts applies to newly developed code that no one envisioned at the outset of the Agile project. Amending the Data Rights Assertion List after award is limited to "new information" or "inadvertent omissions" for NCS.[68] Depending on the circumstances, it may be the case that this unanticipated CDRL application or unanticipated scope constitutes "new information" allowing the contractor to provide new assertions on the Data Rights Assertions List.[69]

Also, important to an Agile development solicitation is to obtain a meeting of the minds on delivery. The Data Rights Assertion List only applies to data that is delivered under the contract. Therefore, the DoD may anticipate based on a Data Rights Assertion List that it is getting unlimited rights in a sought-after source code, when the contractor does not believe there is a delivery of the source code under the contract. Contractors need to understand the CDRL delivery terms under the solicitation and to the extent it is unclear, should ask a question during the solicitation's question and answer period.

Agile Development Post-Award Tips

An offeror's hard pre-award work to protect its proprietary software can be thrown away due to a misstep in contract performance. The following provides tips to protect proprietary software during contract performance.

Marking CDRL Deliverables And Non-CDRL Deliverables

The importance of marking cannot be overstated because the marking places the DoD on notice of its license rights. Marking is important because failing to do so can result in the loss of protection for that software. The DFARS instructs that unmarked NCS and documentation "delivered or otherwise provided under a contract without restrictive markings shall be presumed to have been delivered with unlimited rights and may be released or disclosed without restriction."[70] While the DFARS provides an opportunity to retroactively mark the deliverable within six months, the contractor must prove the lack of marking was inadvertent and agree to relieve the Government of liability for the unmarked software.[71]

First, the DFARS provides specific instructions for the marking and placement of markings for delivered software. The DFARS provides for use of prescribed legends that match the license grant for the software.[72] The DoD can reject a marking that is not in the format required by the DFARS as nonconforming. The DFARS markings also convey notice of the license grants. For instance, a restricted rights legend requires any person other than the Government to promptly notify the contractor if it encounters the marking, whereas a Government purpose rights legend contains no such restriction. The DoD can reject a marking in the appropriate format that does not match the license grant as unjustified. The DoD uses the Data Rights Assertion List for reviewing delivered markings. This is another reason accurately completing the Data Rights Assertion List is important for performance. Markings are not limited to the DFARS legend, as the DFARS permits appropriate copyright notices under 17 U.S.C.A. §§ 401–402.[73] Contractors should work to implement internal controls and train relevant personnel so appropriate markings are placed on CDRL deliverables.

Second, marking is equally important for software

not delivered pursuant to the DFARS and distinguishing from CDRL deliverables. For non-CDRL or DFARS deliverables, a contractor is not required to use the DFARS prescribed legends. In fact, a contractor should not use the DFARS legends but rather the company's standard proprietary markings and a specific disclaimer that the software is not delivered pursuant to the DFARS or any other terms in the contract. Importantly, this signals to the DoD that the software was not delivered pursuant to the DFARS and further disclosure is limited by the proprietary marking. Additionally, by protecting commercial trade secrets through proprietary legends, nondisclosure agreements (NDAs), the company can obtain court injunctions against wrongful misappropriation or use.[74] The proprietary marking on software is critical even for software outside the DoD because the DFARS clauses specify that information placed in the public domain is unlimited rights material, per se whether it occurs in a company's Government business or its commercial activities.[75] Since the software is not delivered pursuant to the DFARS, the DFARS legend format, marking placement, and license grant requirements are inapplicable but protecting the information with company proprietary markings is just as important.

Notably, there are no DFARS legends for commercial software. Therefore, even if the software delivery is contemplated by the contract, the contractor is free to use and should use a company proprietary marking.

Applicable to both DFARS and non-DFARS marking, especially in Agile, is marking every version of the deliverable and the materials housing the deliverable such as an external hard drive, thumb drive, or the transmittal document.[76] The source code may get separated from the transmittal documents so the source code needs to bear the marking as well.

Placing a marking on the software is not so straightforward for electronic delivery of source code. Contractors should insert the marking into the source code via a comment. The marking should be on the first line and potentially before every function. Using a marking via a comment should be ignored by the compiler to not infringe on functionality. Since object code is not human readable, the transmittal document should bear the marking and the file name should begin with the marking to signal the restriction.

Additionally, contractors should use a splash page to display the marking. However, for any combat software or combat simulators, the DFARS prohibits including any software "instructions that interfere or delay the operation of computer in order to display a restrictive rights legend or other license statement at any time prior to or during use of the computer software, or otherwise cause such interference or delay."[77] The combat weapon software used to control the missiles, radars, and other combat elements on an aircraft carrier would prohibit a splash page unless prior written permission is obtained from the contracting officer. Contractors can use a comment to insert the applicable DFARS legend into the source code itself without interfering or delaying the operation of the software.

Use DFARS Clauses To Limit Exposure And Use Of Delivered Data

In an Agile development environment, there may be multiple developers, multiple integrators, users, and contracted support services all working towards the Government's desired end state. There are a few areas of fine print to the DFARS clauses that may arise in an Agile development framework.

First, under the DFARS clauses the DoD can share restricted rights deliverables with "covered Government support contractors" (CGSCs).[78] These are separate contractors providing "independent and impartial advice or technical assistance directly to the Government" to support management and oversight of a program rather than directly furnishing the end item or service.[79] This could be a University Affiliated Research Center all the way to an acquisition support contractor helping the Government formulate its follow-on requirement. The DoD is required to notify the contractor of the disclosure to a CGSC.[80] However, this could be a good topic to bring to the contracting officer's attention at the start of the award such as a post-award conference. Although the CGSC should have DFARS 252.227-7025, "Limitations on the Use or Disclosure of Government-Furnished Information Marked With Restrictive Legends," in their contract, the clause requires the CGSC to "enter into a non-disclosure agreement with the party whose name appears on the legend, if required to do so by that party."[81] The clause purports to create a third-party ben-

eficiary right in the party whose name appears on the restrictive legend; however, an ounce of prevention is worth a pound of cure. Having the CGSC employees who will handle another contractor's proprietary information sign an NDA, can impart the weightiness for which those employees are about to partake.[82]

Second, one of the many asserted reasons for the Government to require delivery of NCS is to maintain competition. In essence, the DoD would like to share the software on future solicitations with competitors. While there may be other restrictions on sharing the data on a solicitation such as International Traffic in Arms Regulations/Export Administration Regulations, if the Government is sharing Government purpose rights data, there should be an NDA to access the data. Often the DoD relies on the inclusion of DFARS 252.227-7025, however at the solicitation stage there is not a contract with the *offerors* yet. Therefore, in accordance with DFARS 252.227-7014(b)(2)(iii)(A), offerors should be required to enter an NDA before accessing the data. The reason is that if an offeror is not the awardee that offeror should not be able to use the data further for its own benefit and should be required to destroy the data under the NDA because it is no longer being used for a Government purpose. While easily overlooked, this is an important step considering many competitors receive access via a solicitation even if they ultimately do not submit a winning offer. Contractors should maintain a record of the data delivered to the DoD and at worst, look on the DFARS 252.227-7017 to catalogue the software delivered. By knowing which command has a company's data, while not dispositive, it can help track down the contracting officer to remind them of their obligation with respect to Government purpose rights data.

Guidelines

These *Guidelines* are intended to assist you in understanding the application of the DFARS data rights regulations to Agile software development. They are not, however, a substitute for professional representation in any specific situation.

1. Software is developed when a person skilled in the art can reasonably expect the software to perform its intended purpose. Development is likely to occur before the software ever leaves the company and should be analyzed at the lowest segregable portion. Document development by describing the software, the funding used, tests performed, and any commercial applications.

2. The DoD uses CDRLs to specify data delivery requirements. Understand what software the solicitation requires delivery of by examining the CDRL content requirements described in the DID. The DID number in Block 4 of the CDRL should be used to retrieve the full DID content at https://quicksearch.dla.mil/qsSearch.aspx.

3. A data rights assertion list requires contractors to propose restrictions on the DoD's use of data that will be delivered. Offerors and their subcontractors should review the CDRLs and DIDs to understand the content of the required deliverables to make applicable restrictions with the submission of their offer.

4. When using a contractor repository, consider negotiating an access agreement that limits the need for CDRL deliverables and increases collaboration. The access agreement can define what data the DoD can access, how the DoD can access the data, the DoD's license in the data, how long the DoD can access the data, destruction/return of the data, delivery status under the contract, and a marking for access data that is different than the DFARS legends.

5. Negotiate an Agile license agreement that also limits the need for formal CDRL deliverables. The Agile license agreement can specify the same limits on the DoD's use of accessed data but when a contractor data repository is not used.

6. Maximize collaborative efforts such as redlining draft solicitations to limit unnecessary CDRLs, clarify uncertain terms, and propose changes based on industry best practices. Remove ambiguities through participation in Q&As

7. For software the DoD encounters outside of a CDRL, mark with company restrictive legends, specify the data is not delivered under the DFARS/contract, and do not use a DFARS legend. Under the DFARS, unmarked data is presumed to have been delivered with unlimited rights and may be released or disclosed without restriction.

8. Place markings on software in multiple ways. Use comments to insert markings that will not interfere with compiled code. Program a splash screen for non-combat software.

9. Contact the DoD to understand any CGSCs that receive access to proprietary data. Once the CGSCs are identified, execute an NDA with them.

10. For any Government purpose rights data delivered to the DoD, monitor solicitations to ensure the DoD is making offerors execute the DFARS 252.227-7014(b)(2)(iii)(A) NDAs.

ENDNOTES:

[1]U.S. Dep't of Defense Innovation Board, DIB Guide: Detecting Agile B.S. 1 (Oct. 9, 2018). But see U.S. Gov't Accountability Office, GAO-21-105298, DOD Software Acquisition: Status of and Challenges Related to Reform Efforts 14 (Sept. 2021) ("Fewer than one-third of the weapon programs we reviewed that reported using Agile (11 of 36 programs) also reported delivering software to users in less than 6 months. Further, only one-sixth of the programs (6 of 36) told us they deliver software in less than 3 months, which is closer to recommended industry standards.").

[2]Daniel E. Shoeni, "Long on Rhetoric, Short on Results: Agile Methods and Cyber Acquisitions in the Department of Defense," 31 Santa Clara High Tech. L.J. 385, 408 (May 2015) (citing U.S. Dep't of Defense, DoD-STD-2167, Military Standard: Defense System Software Development (1985)).

[3]U.S. Dep't of Defense OUSD (A&S), Agile 101: An Agile Primer 9 (2019).

[4]U.S. Gov't Accountability Office, GAO-20-590G, Agile Assessment Guide: Best Practices for Agile Adoption and Implementation 69 (Sept. 2020).

[5]Daniel E. Shoeni, "Long on Rhetoric, Short on Results: Agile Methods and Cyber Acquisitions in the Department of Defense," 31 Santa Clara High Tech. L.J. 385, 399 (May 2015).

[6]Dean Leffingwell, Scaling Software Agility: Best Practices for Large Enterprises 26 (2007).

[7]U.S. Gov't Accountability Office, GAO-20-590G, Agile Assessment Guide: Best Practices for Agile Adoption and Implementation 69 (2020).

[8]U.S. Digital Services, TechFAR Handbook for Procuring Digital Services Using Agile Processes 2 (2021).

[9]Kent Beck et. al., Manifesto for Agile Software Development (2001), https://agilemanifesto.org.

[10]U.S. Dep't of Defense Innovation Board, DIB Guide: Detecting Agile B.S. 1 (2018).

[11] U.S. Dep't of Defense, DoD Digital Modernization Strategy: DoD Information Resource Management Strategy Plan FY 19-23, at 44–45 (July 12, 2019); U.S. Dep't of Defense OUSD (A&S), Agile 101: An Agile Primer 15 (Nov. 18, 2019).

[12]U.S. Digital Services, TechFAR Handbook for Procuring Digital Services Using Agile Processes 9–10 (2021) (A product vision "establishes a high level definition of the scope of the project, specifies expected outcomes, and produces high level budgetary estimates.").

[13]U.S. Dep't of Defense OUSD (A&S), Agile 101: An Agile Primer 12 (2019) (A user story means the "smallest unit of requirements written from a user's perspective of how they will use the software.").

[14]U.S. Dep't of Defense OUSD (A&S), Agile 101: An Agile Primer 13 (2019) (A "sprint" means a "short cycle of work (notionally two to four weeks in duration) that focuses on completing a defined subset of project deliverables, or usable functionality. Each sprint includes planning, design, development, integration, testing, and demonstrating working software to the product owner, users, and other stakeholders.").

[15]U.S. Dep't of Defense OUSD (A&S), Agile 101: An Agile Primer 13 (2019) (The "definition of done" means a "shared and understood definition of the activities that must be completed for a given user story to be considered complete. The definition of done is used to provide clarity around the expected quality of work to meet the users' needs.").

[16]Boeing Co. v. Sec'y of Air Force, 983 F.3d 1321, 1324 (Fed. Cir. 2020), 63 GC ¶ 8 ("The [DFARS 252.227]-7013 clause also makes clear, however, that the contractor retains all rights not granted to the government."); Cubic Def. Applications, Inc, ASBCA No. 58519, 18-1 BCA ¶ 37,049, 60 GC ¶ 189 ("In general, a contractor (developer of IP) retains title to its developed IP and DoD receives from the contractor a nonexclusive license to use, reproduce, modify, release, perform, display or disclose the technical data or software.").

[17]DFARS 252.227-7014(a)(16).

[18]Matthew S. Simchak, "Protecting Rights in Technical Data and Computer Software: Applying the Ten Practical Rules and Their Corollaries," 33 Pub. Cont. L.J. 139, 148 (Fall 2003).

[19]DFARS 252.227-7014(a)(7).

[20]DFARS 227.7203-4(b).

[21]W. Jay DeVecchio, "Taking the Mystery Out of Data Rights," 18-8 Briefing Papers 1, 3 (July 2018).

[22]W. Jay DeVecchio, "Taking the Mystery Out of Data Rights," 18-8 Briefing Papers 1, 3 (July 2018).

[23]DFARS 227.7203-1(a) (emphasis added).

[24]IHS Global Inc. v. United States, 106 Fed. Cl. 734, 739 n.5 (2012) (The Air Force in support of its sole-source award argued that even though it may have paid for the data "in the absence of an express delivery contract provision, it could possess, at most only 'inchoate' data rights.").

[25]Matthew S. Simchack & David A. Vogel, Licensing Software and Technology to the U.S. Government: The Complete Guide to Rights to Intellectual Property in Prime Contracts and Subcontracts 338–39 (2000) (asserting that without a contractual requirement for delivery, "giving the government rights in data it has not ordered is an unenforceable exchange; the contractor has not received any 'consideration' for the grant of broad rights to the Government in the unordered data").

[26]DFARS 252.227-7027.

[27]DFARS 227.7203-1(b)(1).

[28]DFARS 227.7203-1(b)(3).

[29] U.S. Dep't of Defense OUSD (A&S), DoDI 5010.44, Intellectual Property (IP) Acquisition and Licensing 4 (Oct. 16, 2019).

[30]Assistant Sec'y of Defense, Production & Logistics, DoDM 5010.12-M, Procedures for the Acquisition and Management of Technical Data 8 (May 1993) (updated on Aug. 31, 2018).

[31]Assistant Sec'y of Defense, Production & Logistics, DoDM 5010.12-M 26, Procedures for the Acquisition and Management of Technical Data 26 (May 1993) (updated on Aug. 31, 2018) ("With the exception of data specifically required by Subpart 52.2 of the FAR or Subpart 252 of the DFARS (references (j) and (b)), or specifically excepted by Subpart 227.405.70 of reference (b), all data-generating or record-keeping data requirements shall be listed on the DD Form 1423.").

[32]Raytheon Co. v. United States, 160 Fed. Cl. 428, 431 (2022), 64 GC ¶ 224.

[33]Defense Logistics Agency, Assist Quick Search (Nov. 7, 2022, at 8:56 pm), https://quicksearch.dla.mil/qsSearch.aspx.

[34]10 U.S.C.A. § 4021; 10 U.S.C.A. § 4022.

[35]DFARS 227.7202-4.

[36]DFARS 227.7202-1(a).

[37]DFARS 252.227-7014(a)(15)(ii) ("[t]ransfer a computer program to another Government agency without the further permission of the Contractor. . .").

[38]DFARS 227.7202-1(a).

[39]DFARS 227.7202-3(b).

[40]10 U.S.C.A. § 3453.

[41]John S. McCain National Defense Authorization Act for Fiscal Year 2019, Pub. L. No. 115-232, § 1655(h)(6), 132 Stat. 1636, 2151 (2018).

[42]DFARS 252.227-7014(a)(1) ("Commercial computer software" means software developed or regularly

used for non-governmental purposes which—(i) Has been sold, leased, or licensed to the public; (ii) Has been offered for sale, lease, or license to the public; (iii) Has not been offered, sold, leased, or licensed to the public but will be available for commercial sale, lease, or license in time to satisfy the delivery requirements of this contract; or (iv) Satisfies a criterion expressed in paragraph (a)(1)(i), (ii), or (iii) of this clause and would require only minor modification to meet the requirements of this contract.").

[43]DFARS 252.227-7017(b) ("The identification and assertion requirements in this provision apply only to technical data, including computer software documentation, or computer software to be *delivered* with other than unlimited rights." (emphasis added)).

[44]Defense Federal Acquisition Regulation Supplement: Noncommercial Computer Software (DFARS Case 2018-D018), 85 Fed. Reg. 2101 (Jan. 14, 2020) (to be codified at 48 C.F.R. pts 227, 237, 239, 252); Defense Federal Acquisition Regulation Supplement: Noncommercial Computer Software (DFARS Case 2018-D018), 87 Fed. Reg. 4546 (Jan. 28, 2022) (to be codified at 48 C.F.R.pts 227, 237, 239, 252); see 10 U.S.C.A. § 4576 (formerly cited as 10 U.S.C.A. § 2322a; added by National Defense Authorization Act for Fiscal Year 2018, Pub. L. No. 115-91, § 871, 131 Stat. 1283, 1496 (2017)).

[45]Proposed DFARS PGI 227.7203-2(b)(2)(ii)(B) (2018-D018 (p) PGI Text LILO.docx (live.com)).

[46] Proposed DFARS PGI 227.7203-2(2)(ii)(c).

[47]Proposed DFARS PGI 227.7203-2(2)(ii)(B).

[48]Proposed DFARS PGI 227.7203-2(2)(ii)(B).

[49]18 U.S.C.A. § 1905.

[50]DFARS 227.7203-1(a)

[51]U.S. Dep't of Defense OUSD (A&S), Contracting Considerations for Agile Solutions: Key Agile Concepts and Sample Work Statement Language, Version 1.0, at 39 (Nov. 18, 2019).

[52]U.S. Digital Services, TechFAR Handbook for Procuring Digital Services Using Agile Processes 12 (2021).

[53]DFARS 227.7202-1(a).

[54]DFARS 227.7202-1(a).

[55]DFARS 227.7203-1(d) ("Offerors and contractors shall not be prohibited or discouraged from furnishing or offering to furnish computer software developed exclusively at private expense solely because the Government's rights to use, modify, release, reproduce, perform, display, or disclose the software may be restricted.").

[56]Office of Federal Procurement Policy, Myth-Busting 2: Addressing Misconceptions and Further Improving Communication During the Acquisition Pro-

cess 8 (May 7, 2012).

57 Office of Federal Procurement Policy, Myth-Busting 2: Addressing Misconceptions and Further Improving Communication During the Acquisition Process 8 (May 7, 2012).

58DFARS 227.7203-1(a).

59U.S. Gov't Accountability Office, GAO-20-590G, Agile Assessment Guide: Best Practices for Agile Adoption and Implementation 104 (2020).

60The Dep't of Defense Innovation Board: Detecting Agile B.S. 1 (2018).

61DFARS subpt. 227.71.

62DFARS 227.7203-3(a).

63DFARS 252.227-7017(b) ("The identification and assertion requirements in this provision apply only to technical data, including computer software documentation, or computer software to be *delivered* with other than unlimited rights." (emphasis added)).

64DFARS 252.227-7017(f) ("If the Offeror is awarded a contract, the assertions identified in paragraph (d) of this provision shall be listed in an attachment to that contract.").

65DFARS 227.7203-4(b).

66DFARS 227.7203-13(d)(1).

67U.S. Digital Services, TechFAR Handbook for Procuring Digital Services Using Agile Processes 8 (2021).

68DFARS 252.227-7014(e)(3) ("In addition to the assertions made in the Attachment, other assertions may be identified after award when based on new information or inadvertent omissions unless the inadvertent omissions would have materially affected the source selection decision.").

69DFARS 252.227-7014(e)(3).

70DFARS 227.7203-10(c)(1).

71DFARS 227.7203-10(c).

72DFARS 252.227-7014(f).

73DFARS 252.227-7014(f).

74Matthew S. Simchak, "Protecting Rights in Technical Data and Computer Software: Applying the Ten Practical Rules and Their Corollaries," 33 Pub. Cont. L.J. 139, 159 (Fall 2003).

75DFARS 252.227-7014(b)(1)(iv); Matthew S. Simchak, "Protecting Rights in Technical Data and Computer Software: Applying the Ten Practical Rules and Their Corollaries," 33 Pub. Cont. L.J. 139, 159 (Fall 2003).

76DFARS 252.227-7014(f)(1).

77DFARS 252.227-7014(f)(1).

78DFARS 252.227-7014(b)(3)(iii).

79DFARS 252.227-7014(a)(6).

80DFARS 252.227-7014(a)(15)(v).

81DFARS 252.227-7025(b)(5)(iv).

82DFARS 227.7103-7.

BRIEFING PAPERS