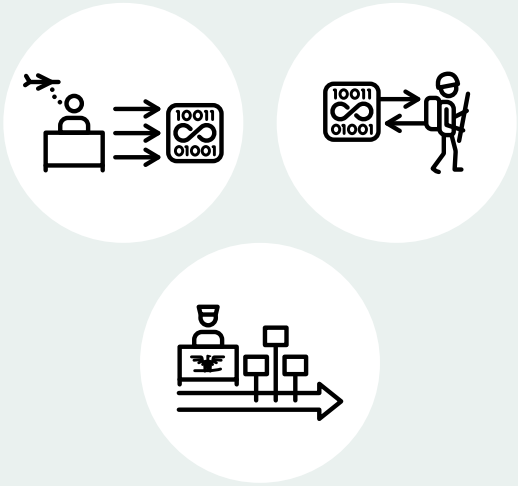
Software Acquisition *Go Bag*

Tactical Guide

02

Cracking the CNS Code



Who should use this guide?

This guide is for those of you involved in developing the Capability Needs Statement (CNS) for programs that are or will be on the Software Acquisition Pathway (SWP). You may be a **Sponsor**, **User**, or the **Program Management Office** (PMO).

See our User's Guide to learn more about roles.

What will I learn from this guide?

This guide provides helpful tips and tricks that enable you to craft an effective CNS to address your program's unique objectives. If you (or your team) need more details on the SWP, start with our Supplement *What Is the Software Acquisition Pathway?*

Acronyms

ADM	Acquisition Decision Memo
CA	Certification Authorities
CDD	Capability Development Document
CNS	Capability Needs Statement
CONEMP	Concept of Employment
CONOPS	Concept of Operations
DA	Decision Authority
DT	Developmental Testing
DoDI	Department of Defense Instruction
DoW	Department of War
OT	Operational Testing
PM	Program Manager
PMO	Program Management Office
PT	Product Team
SW-ICD	Software Initial Capabilities Document
SWP	Software Acquisition Pathway
T&E	Test and Evaluation
UAV	Unmanned Aerial Vehicle
WAU	Warfighting Acquisition University

Why should I care?

Getting the CNS right unlocks substantial benefits for you:

- Your CNS will be the guidepost for your architecture work and will help you develop your acquisition strategy and cost estimates.
- An effective CNS allows you—whether you're a member of the PMO, a User, or member of the Product Team—to decompose CNS capabilities into lower level requirements and to prioritize those requirements in your program's backlog.
- Your CNS provides a trade space where you and others on your program can develop novel solutions.
- Your CNS is also the foundation that your project will use to establish the value statements it needs for periodic value assessments.
- Developing your CNS helps to foster a collaborative environment among stakeholders, including the PMO, Users, and the Sponsor.

Taking the time to engage effectively with Users and other stakeholders to establish a well-crafted CNS is critically important. Getting it wrong can create pitfalls such as the following:

- Prioritizing the backlog will be more difficult because some items may be at the capability level and some items may be at lower levels.
- You will lock in the lower level requirements too soon, which will prohibit Users from updating them as needs arise. In traditional system development, changing requirements was discouraged and was usually a difficult process—which included elements such as Engineering Change Proposals, Rough Order of Magnitude cost estimates, a Configuration Change Board, Integrated Master Schedule changes, etc. The conceptual basis of Agile approaches is that as circumstances change (and they always do!), the project can adapt and reprioritize to add new requirements. Less important requirements can be dropped to make room in the budget for new requirements, and more funding can be added if necessary.
- It will be difficult to calculate a top-level cost estimate because a mix of high-level and low-level requirements makes it more difficult to develop a consistent cost estimate.

We provide some tips and examples to help you develop a great CNS at the right level of detail for your program!

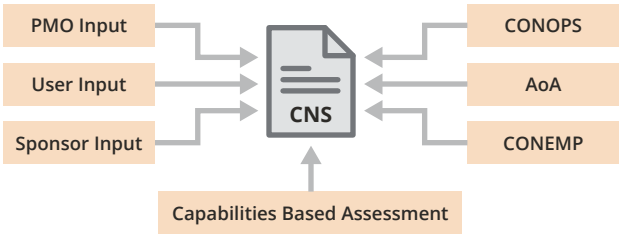
What do I need to know?

Are you confused about what should be in your CNS? You are not alone! One of the most frequent requests we receive is for a sample CNS. If only it was that easy! Never fear; this guide will provide you with the practical insight you need to develop a CNS that will be effective for your program.

What *is* a CNS?

A CNS is a key acquisition artifact that documents the high-level capabilities provided by the software. It is a living document and should be updated as needed after each value assessment. You must have a draft CNS before you can get an Acquisition Decision Memorandum (ADM) from your Decision Authority (DA) to enter the SWP in either the planning phase or the execution phase. Your Sponsor is responsible for creating a draft CNS with input from Users. The draft CNS can be based on a Concept of Operations (CONOPS), a Concept of Employment (CONEMP), or information from a Capabilities Based Assessment or Analysis of Alternatives (AoA).

Figure 1. Draft CNS inputs



> If the PMO is already established, its personnel should help develop the CNS.

Your CNS should be clear and concise. It must describe the software's desired mission outcomes and should be fewer than 10 pages long. The typical content of a CNS includes the following information. (See the CNS template[1] for more descriptions.)

Typical CNS Content[2]

Operational Context/Threat Summary

Describe the mission objectives, current and potential future threats, and legacy systems that will be replaced.

Capabilities Needed

Describe key software capabilities that are needed to achieve the mission(s). These should be high-level capabilities that allow for a trade space during development.

Capability Performance Attributes

Provide quantitative and qualitative attributes for each capability and, if possible, a target level for each measure.

Interoperability

Outline the major systems, services, and networks the software solution must interoperate with and how interfaces and data will be governed.

Requirements Management

Briefly outline the plan for how the Sponsor, operational commands, and Users will capture, prioritize, and continuously refine the lower level needs that will guide software development.[2]

> If you are transitioning from an existing program, you may be able to use your existing requirements document, but we highly recommend a good "scrub" of it as soon as possible to ensure the requirements are at the right level of detail to allow Agile concepts to be used in developing your software.

What does the policy say?

The SWP policy, DoDI 5000.87, states the following:[3]

> *The planning phase will be guided by a draft CNS developed by the operational community. The Sponsor will approve the CNS before the execution phase starts. […]*

> *All required capabilities in the CNS should be prioritized to effectively guide the software development.*

> *Periodic review of the CNS should occur at least as often as each value assessment to determine if updates are warranted. […]*

> *The Sponsor will oversee development of a draft CNS to support the initiation of a software acquisition and use of this pathway. The CNS should be clear and concise. Programs using the embedded path will align the CNS with the requirements documents of the system(s) the software will be embedded. […]*

> *Insights gained during the planning phase will be incorporated into the CNS before approval. The PM will actively engage with the Sponsor during the CNS development to ensure operational and technical feasibility. The Sponsor will approve the CNS before entry into the execution phase. Approval of the CNS should be delegated to the lowest level practical based on the size, risk, complexity, and interdependency of the software needs. […]*

> *The Sponsor will periodically update the CNS throughout development as required to reflect the current high-level operational needs for the software solution.*

Guidance on strategies and approaches

First, consider a few broad guidelines about creating a CNS:

- If you are in the PMO and are spearheading the development of the CNS, you must coordinate with your PMO personnel, Sponsor, Users, and their requirements personnel (ultimately, the Sponsor signs the CNS).
  - Collaborating on the CNS can be the start of a great relationship with your Users, so be sure to be very receptive to User input! It is critical to establish this collaboration on day one!
- If you are the Sponsor and are developing the draft CNS, be sure to keep its content at a high level of detail and to get Users involved. If there is a PMO established, get them involved as well. Developing the CNS really should be a group effort.
  - Ensure that the operational context and threats accurately reflect the environment for the needed capabilities and ensure that the CNS is updated if these elements change.
- The CNS should be a high-level document that lists the *capabilities* warfighters need that the software will contribute to. See Table 1 for examples.
  - Include enough detail to allow the system architecture work to begin.
  - Be sure the requirements are not too detailed and are at a high enough level of detail to leave room for a trade space. In other words, make sure the CNS describes capabilities and not lower level requirements.
  - Include PMO personnel to be sure that you capture all the capabilities needed to enable mission execution, including training, logistics, performance, etc.
  - Be sure to consider aspects like cybersecurity, testing, certifications, modeling, etc.
  - Include interoperability requirements, including transitions from legacy systems.
- During the execution phase, the CNS will be broken down into lower level epics, features, and stories by the Product Team with help from the PMO and Users. The Users really need to be involved during increment planning activities! See the Supplement for this Tactical Guide called *How Does a CNS Drive the Process for Creating Working Software?* for an example of how to break down the CNS.

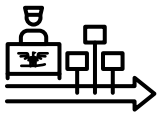
Table 1 is an example that compares requirements you would see in an older requirements document, such as a Capability Development Document (CDD), and requirements you should expect in a CNS. As you can see, the CNS example requirement is expressed at a higher level. Don't worry; you'll see more examples later in this guide.

Table 1. Example CDD vs. CNS Wording

CDD Requirements (more detailed)	CNS Requirement (capability level detail)
The air vehicle **shall** have a time to climb from 100 to 20,000 feet of 30 seconds or less.	The Air Force **needs** an air vehicle that is **capable of completing** all the sorties listed in the CONOPS.
The air vehicle **shall** have a total flight time of 2 hours or more.	
The air vehicle **shall** be able to return to flight no more than 30 minutes after landing.	

Roles and Their Associated CNS Responsibilities

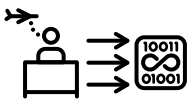
The following role cards describe the responsibilities of the roles involved in creating and executing the CNS:



1. Program Management Office

The PMO should be part of a collaborative team made up of the Sponsor, Users, and PMO personnel, who are jointly developing the CNS.

- The PMO needs to ensure that the CNS is written at a high enough level to allow a trade space during Agile software development.
- The PMO also must ensure that areas such as logistics, training, cybersecurity, and certifications are included in the CNS.
- The PMO, along with the Sponsor, must ensure that any changes to needed capabilities or threats trigger an update to the CNS.

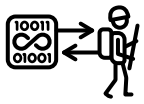


2. Sponsor

(sometimes called Requirements Owner)

The Sponsor is the individual or organization holding the authority and advocating for needed end-user capabilities and associated resource commitments.

The Sponsor should be part of a collaborative team made up of the Sponsor, Users, and PMO personnel, who are jointly developing the CNS.



3. User

The User is the person who ultimately uses the software solution.

- A User (or end user) conveys operational concepts, requirements, and needs to the collaborative group; participates in prioritizing capabilities and continuous testing activities; and provides feedback on developed capabilities.
- The Users should be part of a collaborative team made up of the Sponsor, Users, and PMO personnel, who are jointly developing the CNS.

Unclear about all the groups and roles involved in the SWP? Grab our *Software Acquisition Go Bag User's Guide* from the Pack Your Go Bag webpage.

What about joint programs?

The current SWP policy calls for using a Software Initial Capabilities Document (SW-ICD) for SWP programs that have joint equities.[4] An updated Manual for the Joint Requirements Oversight Council and the Joint Force Requirements Process was released January 15, 2026.[5] This document no longer requires a SW-ICD or JROC approval of joint requirements. All SWP programs should use a CNS going forward and requirements will be validated by the program's Service.

What if my program is part of a larger portfolio or a system of systems?

If all your program's systems or applications have the same Sponsor, Users, threats, interoperability needs, etc. you can create a single CNS with sections for each system or application.

What classification level should the CNS be?

Due to the importance of the CNS to key elements of the SWP, keep the CNS unclassified where possible. Create a classified annex or appendix when needed.

What are some additional examples of CDD vs. CNS requirements?

Table 2 provides examples using different system and requirements types.

> All of these examples would be a part of a larger CNS that describes other functions of the capability.

Table 2. Examples of CDD vs. CNS Requirements

Example Type	CDD Requirements	CNS Requirement
Cybersecurity	The system **shall** protect against the introduction of malicious software into any element and component. The system **shall** verify that all software and data distributed by the system is free of false or deceptive information or malicious code. The system **shall** protect against the unauthorized modification and corruption of information and network data/services. The system **shall** protect against the unauthorized traceability and traffic disclosure or analysis of links connecting with or within the system.	The Space Force **requires** a capability that **will** incorporate DoW Cybersecurity Standards to ensure the data and system are safe from malicious code, unauthorized access, and other cyber threats.
Airplane	The air vehicle **shall** have a time to climb from 100 to 20,000 feet of 30 seconds or less. The air vehicle **shall** have a total flight time of 2 hours or more. The air vehicle **shall** be able to return to flight no more than 30 minutes after landing.	The Air Force **needs** an air vehicle that is **capable of completing** all the sorties listed in the CONOPS.
Drone	The drone **shall** have an intuitive display. The drone controls **shall** be simple to operate. The drone operator **shall** require minimal training to operate the drone.	The Army **requires** a tactical drone **capability that can be operated** by an E-4 with basic drone training. > The intuitiveness of the display and simple controls are issues that should be worked out during demonstrations.
Navy Unmanned Aerial Vehicle (UAV)	The UAV **shall** be capable of detecting and classifying surface vessels at a minimum range of 200 nautical miles, utilizing its onboard sensors. The UAV **shall** have a minimum endurance of 12 hours and a 200 nautical mile radius of the host ship. The UAV **shall** provide a secure, real-time data link of 5 Mbps to the host ship for sensor data and command and control signals.	The Navy **requires** a ship-launched Unmanned Aerial System capability **that provides extended-range surveillance** within a 200 nautical mile radius of the host vessel for the purpose of detecting and classifying surface vessels while maintaining secure and reliable communications with the ship.

When do I need to update an existing CNS?
The SWP requires periodic review of the CNS to determine if updates are warranted. The following types of missions or environment changes might warrant an update:

- **Changes in the Threat:** One of the biggest drivers of a CNS update is a change to the threat, especially if that threat drives a change in system capabilities. The CNS must accurately describe the threats the system will face so that software developers and Users have a common understanding.
- **Changes in the Capabilities Needed:** Capability changes should also drive CNS updates. In the past, these changes could come from something like Engineering Change Proposals, but depending on your contract type, these changes might be easier to implement when using the SWP! (Look for our future guide on contracting strategies!) Capability changes could be due to feedback from Users during sprint or increment demonstrations, engineering design changes, or a change in the threat. Any of these can drive changes in needed capabilities.
- **Changes Needed Due to Test Results:** Test results might also drive the need for CNS updates. If testing for required certifications shows that the system is not meeting a capability, then the CNS may need to be updated to better describe that capability or to add additional capabilities to better describe the system. Not all test results will drive a CNS change; some may just require software updates.

The CNS is meant to be a living document. You need to plan time to update it in collaboration with your Sponsor, Users, and the Product Team after each value assessment.

Building Your CNS

Now you're prepared. You're on your way. What do you do next?

If you are a new program, do these things:

- Meet with your Sponsor and determine the best way to collaborate with them on the CNS. This collaboration is crucial and should begin on day one or as soon as possible.
- Once you are in the planning phase, refine the draft CNS with input from the Sponsor, all Users, the PMO, and the Product Team, who may already be working on the project.

If you are an existing program that already has a requirements document (CDD, etc.), do these things:

- Review your existing requirements documents to determine if they can be used to guide Agile development on the SWP as we've discussed earlier (see Go Bag's *What is the Software Pathway?* for more information).
- It is **very important** that the existing requirements documents be suitable for the SWP. The existing requirements could be good enough to enter the planning phase, but most (if not all) existing programs should convert to a CNS prior to entering the execution phase. Most existing requirements documents include lower level requirements that do not leave enough trade space to prioritize and adapt requirements as needed during software development.
- See the examples we've provided in Table 1 and Table 2 for preferred CNS wording.
- If the existing requirements document is sufficient for now, plan time to rework it into CNS format later in the program.
- If the existing requirement document is not sufficient, meet with your Sponsor and determine the best way to collaborate on creating a CNS.
- In either case, you can use the CNS template to get started.[6]

Real Challenges from a Program That Did Not Convert to a CNS (see the *Why Should I Care* section earlier for more information on pitfalls and consequences)

- Current Users were not involved in developing the existing requirements document, so it was difficult for them to assess the priorities and complete the value assessment.
- The existing documents were written at too low a level, making the Agile prioritization process difficult.

What's next?

Head on over to Pack Your Go Bag on our website to download the Supplement for this Tactical Guide called *How Does a CNS Drive the Process for Creating Working Software?*

Future Tactical Guides and Supplements are in the works that will provide more help with software acquisition.

Stay tuned!

How do I learn more?
On the Adaptive Acquisition Framework's website, refer to the following:[7]

- Learn more about the SWP.
- Read about how to Define Capability Needs.
- Download the SWP Artifact Templates for a CNS.

Visit our Software Acquisition Go Bag website:[8]

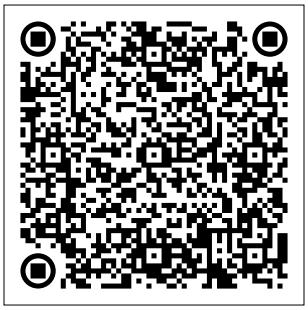
- Unclear about all the groups and roles involved in the SWP? Grab our *Software Acquisition Go Bag User's Guide* from the Pack Your Go Bag webpage.
- Need more background on the SWP? Grab our *What is the SWP?* Supplement from the Pack Your Go Bag webpage.

What if I have questions or want to provide feedback?
We hope you do because your feedback will help us support your journey! Send us questions or feedback on the Contact page.

What if I need more help?
Reach out to us at our Contact page so that we can discuss how the SEI might assist with your specific challenges.

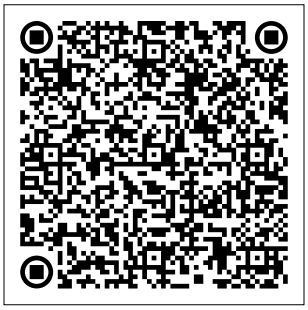
How can I stay engaged with the Software Acquisition Go Bag?
We don't want you to miss out on anything! Subscribe to our mailing list to participate in future Go Bag webinars and be notified about Tactical Guide and Supplement launches.

Don't take the journey alone! Invite your program office, leadership, stakeholders, and other co-workers to join you on this journey by recommending that they also subscribe.



Learn more about the Go Bag project »

Send questions or feedback to our Contact page »



Subscribe to our mailing list for updates »

Endnotes

1 Warfighting Acquisition University (WAU). SWP Artifact Templates. *WAU Website*. November 10, 2025 [accessed]. aaf.dau.edu/aaf/software/templates

2 Warfighting Acquisition University (WAU). Define Capability Needs. *WAU Website*. November 10, 2025 [accessed]. aaf.dau.edu/aaf/software/define-capability-needs

3 Department of War (DoW). *Operation of the Software Acquisition Pathway*. DoDI 5000.87. DoW. October 2, 2020. esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500087p.PDF?ver=virAfQj4v_LgN1JxpB_dpA%3D%3D

4 Department of War (DoW). *Operation of the Software Acquisition Pathway*. DoDI 5000.87. DoW. October 2, 2020. esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500087p.PDF?ver=virAfQj4v_LgN1JxpB_dpA%3D%3D

5 Joint Staff, *Manual for the Joint Requirements Oversight Council and the Joint Force Requirements Process*. January 15, 2026. jcs.mil/Portals/36/Documents/Library/Manuals/CJCSM%205123.01.pdf

6 Warfighting Acquisition University (WAU). SWP Artifact Template. *WAU Website*. November 10, 2025 [accessed]. aaf.dau.edu/aaf/software/templates

7 Warfighting Acquisition University. Software Acquisition. *Adaptive Acquisition Framework Website*. October 7, 2025 [accessed]. aaf.dau.edu/aaf/software

8 Software Engineering Institute (SEI). Software Acquisition Go Bag. October 7, 2025 [accessed]. *SEI Website*. sei.cmu.edu/projects/gobag

Software Acquisition *Go Bag*

About the SEI

The Software Engineering Institute (SEI) advances software as a strategic advantage for national security through research, development, and deployment of tools, technologies, and practices in software engineering, artificial intelligence (AI), cyber, and acquisition transformation. We serve the nation as a federally funded research and development center (FFRDC) sponsored by the U.S. Department of War (DoW).

Contact Us

CARNEGIE MELLON UNIVERSITY
SOFTWARE ENGINEERING INSTITUTE
4500 FIFTH AVENUE; PITTSBURGH, PA 15213-2612

sei.cmu.edu
412.268.5800 | 888.201.4479
info@sei.cmu.edu

This material is based upon work funded and supported by the Department of Defense under Air Force Contract Nos. FA8702-15-D-0002, and FA870225DB003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM26-0195