DATA ITEM DESCRIPTION

Title: SOFTWARE PRODUCT SPECIFICATION (SPS)

Number: DI-IPSC-81441A	**Approval Date:** 19991215
AMSC Number: N7366	**Limitation:**
DTIC Applicable:	**GIDEP Applicable:**

Office of Primary Responsibility: N/SPAWAR
Applicable Forms:
Use, Relationships:

The Software Product Specification (SPS) contains or references the executable software, source files, and software support information, including "as built" design information and compilation, build, and modification procedures, for a Computer Software Configuration Item (CSCI).

The SPS can be used to order the executable software and/or source files for a CSCI and is the primary software support document for the CSCI. Note: Different organizations have different policies for ordering delivery of software. These policies should be determined before applying this Data Item Description (DID).

This DID contains the format and content preparation instructions for the data product generated by specific and discrete task requirements as delineated in the contract.

This DID is used when the developer is tasked to prepare executable software, source files, "as built" CSCI design, and/or related support information for delivery.

This DID supersedes DI-IPSC-81441.

Requirements:

1. Reference documents. None.

2. General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this DID means a collection of data regardless of its medium.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required by this DID can be made more readable using these styles.

3. Format. Following are the format requirements.

The specification shall be in contractor format unless otherwise specified on the Contract Data Requirements List (CDRL)(DD 1423). The CDRL should specify whether deliverable data are to be delivered on paper or electronic media; are to be in a given electronic form (such as ASCII, CALS, or compatible with a specified word processor or other support software); may be delivered in developer format rather than in the format specified herein; and may reside in a computer-aided software engineering (CASE) or other automated tool rather than in the form of a traditional document.

4. Content. The specification shall contain the following:

a. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement; and signature blocks for the developer representative authorized to release the document, the acquirer representative authorized to approve the document, and the dates of release/approval. For data in a database or other alternative form, this information shall be included on external and internal labels or by equivalent identification methods.

b. Table of contents. The document shall contain a table of contents providing the number, title, and page number of each titled paragraph, figure, table, and appendix. For data in a database or other alternative form, this information shall consist of an internal or external table of contents containing pointers to, or instructions for accessing, each paragraph, figure, table, and appendix or their equivalents.

c. Page numbering/labeling. Each page shall contain a unique page number and display the document number, including version, volume, and date, as applicable. For data in a database or other alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

d. Response to tailoring instructions. If a paragraph is tailored out of this DID, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out." For data in a database or other alternative form, this representation need occur only in the table of contents or equivalent.

e. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

f. Standard data descriptions. If a data description required by this DID has been published in a standard data element dictionary specified in the contract, reference to an entry in that dictionary is preferred over including the description itself.

g. Substitution of existing documents. Commercial or other existing documents may be substituted for all or part of the document if they contain the required data.

The numbers shown designate the paragraph numbers to be used in the document.

1. Scope. This section shall be divided into the following paragraphs.

1.1 Identification. This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

1.2 System overview. This paragraph shall briefly state the purpose of the system and the software to which this document applies. It shall describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

1.3 Document overview. This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

2. Referenced documents. This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

3. Requirements. This section shall be divided into the following paragraphs to achieve delivery of the software and to establish the requirements that another body of software must meet to be considered a valid copy of the CSCI.

Note: In past versions of this DID, Section 3 required a presentation of the software design describing the “as built” software. That approach was modeled on hardware development, in which the product specification presents the final design as the requirement to which hardware items must be manufactured. For software, however, this approach does not apply. Software “manufacturing” consists of electronic duplication of the software itself, not recreation from design, and the validity of a “manufactured” copy is determined by comparison to the software itself, not to a design description. This section therefore establishes the software itself as the criterion that must be matched for a body of software to be considered a valid copy of the CSCI. The updated software design has been placed in Section 5 below, not as a requirement, but as information to be used to modify, enhance, or otherwise support the software. If any portion of this specification is placed under acquirer configuration control, it should be limited to Section 3. It is the software itself that establishes the product baseline, not a description of the software’s design.

3.1 Executable software. This paragraph shall provide, by reference to enclosed or otherwise provided electronic media, the executable software for the CSCI, including any batch files, command files, data files, or other software files needed to install and operate the software on its target computer(s). In order for a body of software to be considered a valid copy of the CSCI’s executable software, it must be shown to match these files exactly.

3.2 Source files. This paragraph shall provide, by reference to enclosed or otherwise provided electronic media, the source files for the CSCI, including any batch files, command files, data files, or other files needed to regenerate the executable software for the CSCI. In order for a body of software to be considered a valid copy of the CSCI's source files, it must be shown to match these files exactly.

3.3 Packaging requirements. This paragraph shall state the requirements, if any, for packaging and marking copies of the CSCI.

4. Qualification provisions. This paragraph shall state the method(s) to be used to demonstrate that a given body of software is a valid copy of the CSCI. For example, the method for executable files might be to establish that each executable file referenced in 3.1 has an identically-named counterpart in the software in question and that each such counterpart can be shown, via bit-for-bit comparison, check sum, or other method, to be identical to the corresponding executable file. The method for source files might be comparable, using the source files referenced in 3.2.

5. Software support information. This section shall be divided into the following paragraphs to provide information needed to support the CSCI.

5.1 "As built" software design. This paragraph shall contain, or reference an appendix or other deliverable document that contains, information describing the design of the "as built" CSCI. The information shall be the same as that required in a Software Design Description (SDD), Interface Design Description (IDD), and Database Design Description (DBDD), as applicable. If these documents or their equivalents are to be delivered for the "as built" CSCI, this paragraph shall reference them. If not, the information shall be provided in this document. Information provided in the headers, comments, and code of the source code listings may be referenced and need not be repeated in this section. If the SDD, IDD, or DBDD is included in an appendix, the paragraph numbers and page numbers need not be changed.

5.2 Compilation/build procedures. This paragraph shall describe, or reference an appendix that describes, the compilation/build process to be used to create the executable files from the source files and to prepare the executable files to be loaded into firmware or other distribution media. It shall specify the compiler(s)/assembler(s) to be used, including version numbers; other hardware and software needed, including version numbers; any settings, options, or conventions to be used; and procedures for compiling/assembling, linking, and building the CSCI and the software system/subsystem containing the CSCI, including variations for different sites, configurations, versions, etc. Build procedures above the CSCI level may be presented in one SPS and referenced from the others.

5.3 Modification procedures. This paragraph shall describe procedures that must be followed to modify the CSCI. It shall include or reference information on the following, as applicable:

a. Support facilities, equipment, and software, and procedures for their use

b. Database/data files used by the CSCI and procedures for using and modifying them

c. Design, coding, and other conventions to be followed

d. Compilation/build procedures if different from those above

e. Integration and testing procedures to be followed

5.4 Computer hardware resource utilization. This paragraph shall describe the "as built" CSCI's measured utilization of computer hardware resources (such as processor capacity, memory capacity, input/output device capacity, auxiliary storage capacity, and communications/network equipment capacity). It shall cover all computer hardware resources included in utilization requirements for the CSCI, in system-level resource allocations affecting the CSCI, or in the software development plan. If all utilization data for a given computer hardware resource is presented in a single location, such as in one SPS, this paragraph may reference that source. Included for each computer hardware resource shall be:

a. The CSCI requirements or system-level resource allocations being satisfied. (Alternatively, the traceability to CSCI requirements may be provided in 6.c.)

b. The assumptions and conditions on which the utilization data are based (for example, typical usage, worst-case usage, assumption of certain events)

c. Any special considerations affecting the utilization (such as use of virtual memory, overlays, or multiprocessors or the impacts of operating system overhead, library software, or other implementation overhead)

d. The units of measure used (such as percentage of processor capacity, cycles per second, bytes of memory, kilobytes per second)

e. The level(s) at which the estimates or measures have been made (such as software unit, CSCI, or executable program)

6. Requirements traceability. This section shall provide:

a. Traceability from each CSCI source file to the software unit(s) that it implements.

b. Traceability from each software unit to the source files that implement it.

c. Traceability from each computer hardware resource utilization measurement given in 5.4 to the CSCI requirements it addresses. (Alternatively, this traceability may be provided in 5.4.)

d. Traceability from each CSCI requirement regarding computer hardware resource utilization to the utilization measurements given in 5.4.

7. Notes. This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an

alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. Appendices. Appendices may be used to provide information published separately for convenience in document maintenance (e.g., charts classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes shall be lettered alphabetically (A, B, etc.).

END OF DI-IPSC-81441A