

SRDR INITIAL GOVERNMENT REPORT

Title: Software Resources Data Reporting: Initial Government Report and Data Dictionary

Office of Primary Responsibility: (D)OSD/PA&E/CAIG

Applicable Forms: Software Resources Data Reporting: Initial Government Report (Sample Format 1)

These instructions describe the SRDR Initial Government Report process and its two deliverables. The first deliverable, the Initial Government Report, is used to obtain the government's estimates of the characteristics of a software product and its development process. The second deliverable is the Software Resources Data Reporting (SRDR) Data Dictionary, which defines each of the data elements within the Initial Government Report and describes the methods and rules to be used to perform the data measurement or estimation. Every submission of this SRD report shall contain both the SRDR Initial Government Report and an associated SRDR Data Dictionary.

The Initial Government Report is the first report of a series of SRD reports that describe major software developments or software upgrades conducted for OSD. Subsequent SRD reports are the Initial Developer Reports and the Final Developer Reports that document, respectively, the expected and the actual characteristics of a software development process and its delivered product.

The SRD report is not a management or software metrics report. It is not intended for tracking progress of the development during contract execution, nor is it intended to collect financial information. It does, however, collect the person-hours expended during software development.

The intent of the SRDR process is to collect objective, measurable data commonly used by industry and DoD cost analysts. These data are used to compile a repository of estimated and actual software product sizes, schedules, and effort that Government analysts can draw upon to build credible size, cost, and schedule estimates of future software-intensive systems.

Information to be acquired through these data will include descriptive information about the product and developer, and estimates of software product size, development schedule, peak staff, and direct labor hours.

A SRDR Data Dictionary that defines the data elements contained in the SRDR Initial Government Report must be submitted with the report. The definitions of the data items are negotiable but must include the following categories of data: Context, Project Description, Size, Effort, and Schedule. Reporting on estimated product quality is optional.

The format of the SRD reports is not constrained to the Sample Formats provided. Instead, the sample formats and the associated instructions (or DIDs in the case of Format 2 and Format 3) are intended to serve as guidelines for the categories of data expected. The data items shown in the sample formats are examples of the kinds of data that are commonly used to record these quantities. The SRDR Data Dictionary is required in

order to ensure that the definitions of the values reported in an SRD report are also documented and reported.

An SRDR submission shall be prepared in a Microsoft Excel-compatible electronic file format. For submissions that require discrete reporting of multiple WBS elements, the data shall be prepared and integrated into one electronic file.

The SRDR Data Dictionary shall be prepared in a readable electronic (digital) file format such as Microsoft Excel or Microsoft Word (e.g., PDF files are not acceptable).

Requirements:

1. *Reference Documents*. DoDI 5000.2, “Operation of the Defense Acquisition System,” establishes mandatory policies for requiring SRD reports. DoD 5000.04-M-1, “Cost and Software Data Reporting (CSDR) Manual,” prescribes procedures and instructions for stakeholders in the SRDR process. Detailed instructions for preparing the SRDR Initial Government Report and SRDR Data Dictionary are provided below.
2. *Format*. There is no prescribed data format for either the SRDR Initial Government Report or the SRDR Data Dictionary. The SRDR Initial Government Report shall be in a format agreed to by the program office and the Government. Software Resources Data Report: Initial Government Report (Sample Format 1), shown in Figure 1, serves as a starting point for developing a tailored report.
3. *Content*. The SRDR Initial Government Report shall contain estimates-at-complete of data items described in the SRDR Data Dictionary. The data shall reflect estimates of the entire software development project. The SRD report shall contain instances of each of the data elements outlined below and reflected in Sample Format 1 (Figure 1). Data elements reported beyond those outlined in this DID shall be agreed upon by the Cost Working-group Integrated Product Team (CWIPT) and approved by the OSD CAIG Chair.

3.1. Report Context and Development Organization.

3.1.1. *Security Classification*. The top and bottom of every page shall be marked with the security classification of the report which typically will be “Unclassified”. However, if the appropriate security classification based on the classification level of the data reported is classified, contact the DCARC for special processing instructions.

3.1.2. *System/Element Name*. The name of the system or element identified for reporting on the contract’s CSDR Plan. The name shall include any applicable version, release, build, or other identifier. The System/Element name shall also reference the name of the WBS element and its associated WBS number from the contract’s CSDR Plan.

3.1.3. *Report As-of Date*. Provide the date of the data, which is not necessarily the date the report was prepared.

3.1.4. *Authorizing Vehicle*. Provide the prime contract number (if applicable) and amendment number (if applicable), or reference to a memorandum of understanding or other documentation that authorizes the development of the subject software.

3.1.5. Event. A draft Initial Government Report is due 180 days before the OIPT. A second, revised or final, Initial Government Report is due 45 days before the OIPT. Indicate whether draft or final based on this schedule.

3.1.6. Submission and Supersedes Number. For each SRDR reporting event (draft or final), enter “1” the first time an SRD report is submitted, enter ”2” if the program office submits a corrected/revised SRD report, and so on. Indicate which report is being superseded by giving its submission number (if applicable).

3.1.7. Development Organization. Enter the name of the company or organization responsible for development of the software product, *if known*. The associated SRDR Data Dictionary must be used to explain the mapping of development organizations, software components, and SRD reports submitted *to the extent this information is known at the time of the Initial Government Report*.

3.1.8. SRDR Data Dictionary Filename and Last Revision Date. If the SRDR Data Dictionary is stored as a separate electronic file from the SRD report, provide the filename of the SRDR Data Dictionary file. Provide the date the associated SRDR Data Dictionary was last revised.

3.1.9. Comments. Provide any comments about report context and development organization. Include more detailed explanations in the associated SRDR Data Dictionary.

3.2. Product and Development Description. Product description data should be provided or estimated to the extent that this is possible at the time of the Initial Government Report. *The data elements shown here may include more detail than can be known at the time of this report*. If so, those details should be tailored out of the report.

3.2.1. Functional Description. For each element reported, provide a brief description of its intended function. What is it? What does it do?

3.2.2. Software Development Characterization. In general language, provide a brief description for each element reported that characterizes the software development work to be undertaken on that element. Examples might include completely new from-scratch development, rehosting of software to different processor/operating system, reengineering of legacy code into open architecture, translation of legacy code from Ada to C, and so on.

3.2.3. Application Type. Identify at least one application type (i.e., the intended end-user mission) to be developed using one or more domain names from those listed in Figure 2. A minimum of one primary application type shall be identified, but any number of application types may be listed. If none of the examples in Figure 2 are appropriate, enter a phrase to describe the application type and define it in the associated SRDR Data Dictionary. When internal development efforts within a program are large and independent, respondents may choose to report each using a separate SRD report instead of as various application types within a single report. For every application type reported provide:

3.2.3.1. Primary and Secondary Programming Languages. Enter the primary and secondary computer language in which most of the development is to be

conducted. This can be a compiled language, such as FORTRAN, Ada, or C, an interpreted language such as Basic, or a graphical or model-based language such as Rhapsody/UML or Simulink. Use the estimated amount of effort spent in development to determine the primary language rather than the amount of function delivered. Explain any interpretation of this item in the associated SRDR Data Dictionary.

3.2.3.2. Percent of Overall Product Size. Enter the approximate percentage (up to 100%) of the estimated product size that is of each application type. If relevant and appropriate, integrated Commercial Off-the-Shelf (COTS)/Government Off-the-Shelf (GOTS) packages may be included in this calculation. If so, an explanation must be placed in the SRDR Data Dictionary.

3.2.3.3. Planned Development Process. Enter the name of the development process to be followed for the development of the system (if known). Do not indicate a software architecture method (such as object-oriented development) or a development tool (such as Rational Rose), as these do not specify a process. Typical types of development processes adopted include waterfall, spiral, or Rapid Application Development (RAD).

3.2.3.4. Software Development Method(s). Identify the software development method or methods to be used to design and develop the software product (e.g., Structured Analysis, Object Oriented, Vienna Development Method, etc.).

3.2.3.5. Upgrade or New Development? Indicate whether the primary development is new software or an upgrade. A software system is considered new either if no existing system currently performs its function or if the development completely replaces an existing system. A software system that replaces part of an existing system (such as the replacement of a database) should be considered an upgrade. An existing software system that is being ported to a new platform or being reengineered to execute as a Web or distributed application (for example) would be considered an upgrade unless it is also being completely redeveloped from scratch (new requirements, architecture, design, process, code, etc.).

3.2.4. COTS/GOTS Applications Used.

3.2.4.1. Name. List the names of known applications or products that will constitute part of the final delivered product, whether they are COTS, GOTS, or open-source products. If a proprietary application or product that is not generally commercially available will be included, identify it here and include any necessary explanation in the associated SRDR Data Dictionary.

3.2.4.2. Integration Effort (Optional). If requested by the CWIPT, the SRD report shall contain an estimate of the effort required to integrate each COTS/GOTS application identified in Section 3.2.4.1. “Effort” may be expressed in terms of staff-hours, new/modified glue code, or a qualitative assessment of effort required (i.e., low, medium, high, etc.). The SRDR Data

Dictionary shall contain appropriate definitions of the chosen integration effort metric.

3.2.5. Staffing

3.2.5.1. Peak Staff. For the element reported, enter the estimated peak team size, measured in full-time equivalent (FTE) staff. Include only direct labor in this calculation unless otherwise explained in the associated SRDR Data Dictionary. The SRDR Data Dictionary shall include a definition of FTE that includes the hours per staff-month used to compute FTE.

3.2.5.2. Peak Staff Date. Enter the date when the estimated peak staffing is expected to be reached.

3.2.5.3. Hours per Staff-Month. Enter the estimated number of direct labor hours per staff-month. Indicate in the SRDR Data Dictionary whether the reported hours per staff-month reflect an accounting standard or a computation. If they reflect a computation, provide details on how the computation was performed.

3.2.6. Personnel Experience in Domain. If possible, estimate the project staff domain experience by experience level and estimate the percentage of project staff at each experience level identified. (Sample Format 1 identifies three levels: Highly Experienced, Nominally Experienced, and Inexperienced/Entry Level.) Provide a definition for each experience level (i.e., the number of years of experience) in the SRDR Data Dictionary. Also provide a definition of “domain experience” in the SRDR Data Dictionary (e.g., “Domain experience is defined as the number of years a project staff member has worked within a mission discipline such as real time fire control radar or missile guidance and tracking”).

3.2.7. Comments. Provide any comments about the product and development description. Include more detailed explanations in the associated SRDR Data Dictionary.

3.3. Estimated Product Size Reporting. Product size data should be estimated at the time of the Initial Government Report.

3.3.1. Number of Software Requirements. Provide the estimated number of software requirements. The method of counting estimated number of requirements implemented by the development software will be the same as that ultimately used for counting the actual, as-built requirements (as reported in the SRDR Final Developer Report). Do not count requirements concerning external interfaces not under project control (see next item, “Total Requirements”). Alternative requirements counts based on Use Cases are also permitted. The SRDR Data Dictionary shall provide both a definition of what types of requirements are included in the count (i.e., functional, security, safety, other derived requirements, etc.) and the units (e.g., “shall,” “sections,” paragraphs, etc.) and counting methods used.

3.3.1.1. Total Requirements. Enter the estimated number of total requirements to be satisfied by the developed software product at the completion of the increment or project. This count must be consistent with

the total size of the delivered software (i.e., it must not solely focus on new development, but must reflect the entire software product).

3.3.1.2. New Requirements. Of the total estimated number of requirements reported, the SRD report shall identify how many are new requirements.

3.3.2. Number of External Interface Requirements. Provide the estimated number of external interface requirements, as specified below not under project control that the developed system will satisfy. External interfaces include interfaces to computer systems, databases, files, or hardware devices with which the developed system must interact but which are defined externally to the subject system. If the developed system will interface with an external system in multiple ways (such as for reading data and also for writing data), then each unique requirement for interaction should be counted as an interface requirement. Provide the estimated number of interface requirements to be handled by the developed software. Explain any details about the counting methods for external interface requirements in the SRDR Data Dictionary.

3.3.2.1. Total External Interface Requirements. Enter the estimated number of total external interface requirements to be satisfied by the developed software product at the completion of the increment or project. This count must be consistent with the total size of the delivered software (i.e., it must not solely focus on new development, but must reflect the entire software product).

3.3.2.2. New External Interface Requirements. Of the total estimated number of external interface requirements reported, the SRD report shall identify how many are new external interface requirements.

3.3.3. Requirements Volatility. Indicate the amount of requirements volatility expected during development using a qualitative scale (very low, low, nominal, high, very high) relative to similar systems of the same type. This should be a relative measure rather than an absolute one in order to understand the expectations of the impact of requirements volatility during the course of the software development. Specific definitions for each rank in the qualitative scale and overall definition of what constitutes requirements volatility shall be provided in the SRDR Data Dictionary.

3.3.4. Estimated Total Delivered Code

3.3.4.1. Delivered Size: The SRD report shall capture the estimated delivered size of the product to be developed, not including any code that might be needed to assist development but that will not be delivered (such as temporary stubs, test scaffoldings, or debug statements). Additionally, the code shall be partitioned (exhaustive with no overlaps) into appropriate development categories. A common set of software development categories is newly developed code (either human generated or auto-generated), code reused from external sources (either with or without modification), and code reused from previous builds or predecessor projects (either with or without

modification). In all cases, the partitioning used for reporting shall not double count or omit any delivered software.

3.3.4.2. Sizing Units. The SRD report shall identify the size units of measure (i.e., logical Source Lines of Code (SLOC), physical lines, function points, etc.) used to count or estimate software size. A specific definition must be provided in the SRDR Data Dictionary. While SLOC is a prominent unit of software size, the SRD report may reflect different units of measure.

Alternative units, such as function points, are permissible so long as these units are also expected to be reported on the subsequent Initial Developer Report and the Final Developer Report. Units of measure that reflect weighted sum normalization of size into equivalent units, such as Equivalent New Lines of Code, shall not be used as a primary sizing unit of measure in the SRD report. This information may be reported in the SRD report's supplemental information.

3.3.5. Comments. Provide any comments about product size reporting. Include more detailed explanations in the associated SRDR Data Dictionary.

3.4. Estimated Resource and Schedule Reporting. The Initial Government Report shall contain estimates of schedule and effort for each software development activity.

3.4.1. Prime Contractor Development Effort. The units of measure for software development effort shall be reported in staff-hours. Effort shall be partitioned into discrete software development activities. The following activities are taken from the activity definitions used in standard ISO 12207 and are intended as an example of partitioning software development effort. Additional software support activities are also shown.

- software requirements analysis,
- software architecture and detailed design,
- software coding and unit testing,
- software integration and system/software integration,
- software qualification testing,
- software developmental test and evaluation, and
- other software support activities:
 - software quality assurance,
 - software configuration management,
 - software program management,
 - data,
 - software process improvement,
 - IV and V, and
 - problem resolution.

3.4.2. Subcontractor Development Effort. If possible, the effort data estimates in the SRD report shall be separated into two discrete categories and reported separately: Prime Contractor Only and All Other Subcontractors. If the reported subcontractor development effort cannot be partitioned by software development activity, then report only the total estimated effort and provide a definition in the SRDR Data Dictionary that explains what software development activities are included in the subcontractor-estimate of development effort. If it is not known if subcontractor effort will be applied, or if subcontractor-only development effort cannot be estimated, use 3.4.1 Prime Contractor Development Effort to report all effort and indicate in the SRDR Data Dictionary if it is known whether portions of the estimated effort will be expended by subcontractors.

3.4.3. Schedule. For each software development activity reported, provide the estimated start and end dates for that activity. Alternatively, month numbers, starting with month “1” at the time of Contract Award, can be used. If the schedule reflects multiple start and stop dates for the same activity, such as the case for iterative or spiral development, then to the extent that is sensible for the approach used (or expected), the earliest and latest end date that each activity occurred can be reported. If month numbers are used, provide the date that is equivalent to month “1” either in the SRDR Initial Government Report or in the SRDR Data Dictionary.

3.4.4. Comments. Provide any comments about resource and schedule reporting. Include more detailed explanations in the associated SRDR Data Dictionary.

3.5. Estimated Product Quality. The Initial Government Report may contain estimates of product quality for the delivered software product. Two examples of measures that can be used to express product quality are provided as guidelines. More detailed definitions should be provided in the Data Dictionary.

3.5.1. Mean Time to Defect (MTTD), as measured by execution time or actual elapsed hours.

3.5.2. Observed or Computed Quality as compared with nominal reliability of analogous systems.

3.5.3. Comments. Provide any comments about resource and schedule reporting. Include more detailed explanations in the associated SRDR Data Dictionary.

3.6. Point of Contact (POC) Information. Enter the following information for the person to be contacted for answers to any questions about this report, the data reported, or the associated SRDR Data Dictionary:

- name: last name, first name, and middle initial;
- department name;
- telephone number, including area code;
- e-mail address;
- fax number, including area code;

- signature (an electronic signature is acceptable); and
- date signed (usually later than the “as of” date).

**Figure 1. Software Resources Data Report: Initial Government Report
(Sample Format 1), Page 1**

Section 3.1.1 SECURITY CLASSIFICATION						
SOFTWARE RESOURCES DATA REPORTING: INITIAL GOVERNMENT REPORT (SAMPLE FORMAT 1) <i>Due 180 days before contract award as part of the Cost Analysis Requirements Description (CARD).</i>						
Section 3.1: REPORT CONTEXT AND DEVELOPMENT ORGANIZATION						
SYSTEM/ELEMENT NAME		Section 3.1.2		REPORT AS OF Section 3.1.3		
AUTHORIZING VEHICLE	Section 3.1.4	EVENT	Section 3.1.5	SUBMISSION #	Section 3.1.6	SUPERSEDES # <i>(if applicable)</i> Section 3.1.6
DEVELOPMENT ORGANIZATION (only if known) Section 3.1.7						
SRDR DATA DICTIONARY FILENAME		Section 3.1.8		LAST REVISION DATE		Section 3.1.8
COMMENTS Section 3.1.9						
Section 3.2 PRODUCT AND DEVELOPMENT DESCRIPTION						
FUNCTIONAL DESCRIPTION Section 3.2.1						
SOFTWARE DEVELOPMENT CHARACTERIZATION Report any known or estimated aspects of the software development or upgrade product and process) Section 3.2.2						
APPLICATION TYPE	PRIMARY PROGRAMMING LANGUAGE	SECONDARY PROGRAMMING LANGUAGE	PERCENT OF PRODUCT SIZE	PLANNED DEVELOPMENT PROCESS	SW DEVELOPMENT METHOD(S)	UPGRADE OR NEW?
Section 3.2.3	Section 3.2.3.1	Section 3.2.3.1	Section 3.2.3.2 %	Section 3.2.3.3	Section 3.2.3.4	Section 3.2.3.5
			%			
Section 3.2.4 COTS/GOTS APPLICATIONS USED:						
NAME	INTEGRATION EFFORT (OPTIONAL)		NAME	INTEGRATION EFFORT (OPTIONAL)		
Section 3.2.4.1	Section 3.2.4.2		Section 3.2.4.1	Section 3.2.4.2		
Section 3.2.5 STAFFING						
PEAK STAFF (<i>Maximum Team Size in FTE</i>)	Section 3.2.5.1	PEAK STAFF DATE	Section 3.2.5.2	HOURS/STAFF-MONTH	Section 3.2.5.3	
Section 3.2.6 PERSONNEL EXPERIENCE IN DOMAIN						
HIGHLY EXPERIENCED: Section 3.2.6	%	NOMINALLY EXPERIENCED: Section 3.2.6	%	INEXPERIENCED/ENTRY LEVEL: Section 3.2.6	%	
COMMENTS Section 3.2.7						
Section 3.3 ESTIMATED PRODUCT SIZE REPORTING						
3.3.1 NUMBER OF SOFTWARE REQUIREMENTS	TOTAL Section 3.3.1.1	3.3.2 NUMBER OF EXTERNAL INTERFACE REQUIREMENTS	TOTAL Section 3.3.2.1	REQUIREMENTS VOLATILITY	Section 3.3.3	
	NEW Section 3.3.1.2		NEW Section 3.3.2.2			
3.3.4 ESTIMATED TOTAL DELIVERED CODE			SIZING UNITS	Section 3.3.4.2		
Section 3.3.4.1 AMOUNT DEVELOPED NEW		HUMAN GENERATED				
		AUTO GENERATED				
Section 3.3.4.1 AMOUNT REUSED FROM EXTERNAL SOURCE		WITH MODIFICATIONS				
		WITHOUT MODIFICATIONS				
Section 3.3.4.1 AMOUNT INHERITED FROM PREVIOUS INCREMENT/BUILD or		WITH MODIFICATIONS				
		WITHOUT MODIFICATIONS				
		TOTAL DELIVERED CODE				
COMMENTS Section 3.3.5						

Section 3.1.1

SECURITY CLASSIFICATION

**Figure 1. Software Resources Data Report: Initial Government Report
(Sample Format 1), Page 2**

Section 3.1.1 SECURITY CLASSIFICATION				
SOFTWARE RESOURCES DATA REPORT: INITIAL GOVERNMENT REPORT (SAMPLE FORMAT 1)				
SECTION 3.4 ESTIMATED RESOURCE AND SCHEDULE REPORTING				
SOFTWARE ACTIVITY NAME	PRIME CONTRACTOR DEVELOPMENT EFFORT	SUBCONTRACTOR DEVELOPMENT EFFORT	Section 3.4.3 SCHEDULE	
			START MONTH	END MONTH
Section 3.4.1 (Example: SOFTWARE REQUIREMENTS ANALYSIS)	Section 3.4.1	Section 3.4.2	Section 3.4.3	Section 3.4.3
Section 3.4.1 (Example: SOFTWARE ARCHITECTURE AND DETAILED DESIGN)				
Section 3.4.1 (Example: SOFTWARE CODING AND UNIT TESTING)				
Section 3.4.1 (Example: SOFTWARE INTEGRATION AND SYSTEM/SOFTWARE INTEGRATION)				
Section 3.4.1 (Example: SOFTWARE QUALIFICATION TESTING)				
Section 3.4.1 (Example: SOFTWARE DEVELOPMENTAL TEST AND EVALUATION)				
ALL OTHER DIRECT SOFTWARE ENGINEERING DEVELOPMENT EFFORT				
TOTAL SOFTWARE DEVELOPMENT EFFORT				
COMMENTS				
Section 3.4.4				
Section 3.5 ESTIMATED PRODUCT QUALITY REPORTING (OPTIONAL)				
REQUIRED OR ACTUAL MEAN TIME TO SERIOUS OR CRITICAL DEFECT (MTTD) AT DELIVERY, IN EXECUTION TIME OR ACTUAL ELAPSED HOURS <i>(Provide the specific definition of the measure used in the associated Data Dictionary.)</i>				
Section 3.5.1				
OBSERVED OR COMPUTED RELIABILITY COMPARED WITH NOMINAL RELIABILITY OF ANALOGOUS SYSTEMS <i>(Provide details about the analogous systems and define nominal reliability in the associated Data Dictionary.)</i>				
Section 3.5.2				
COMMENTS				
Section 3.5.3				
Section 3.6 POINT OF CONTACT (POC) INFORMATION				
NAME (Last, First, Middle Initial)		DEPARTMENT	TELEPHONE NO. (<i>Include Area Code</i>)	
Section 3.6		Section 3.6	Section 3.6	
E-MAIL ADDRESS	FAX NO. (<i>Include Area Code</i>)	SIGNATURE	DATE SIGNED (YYYY-MM-DD)	
Section 3.6	Section 3.6	Section 3.6	Section 3.6	

Figure 2. Application Types

<u>Warfare Mission Areas</u>	<u>Defensive Systems Functions</u>	<u>Miscellaneous Functions</u>
Antiair Warfare Antisubmarine Warfare Naval Antisurface Ship Warfare Amphibious Warfare Chemical Warfare Biological and Radiological Defense Land Warfare Special Warfare Strategic Warfare Tactical Air Warfare Electronic Warfare Strategic Defense Initiative	Hit Avoidance Signature Control/Suppression Reduction Armor, Infantry and Crew Protection EMP Hardening/Survivability from Nuclear Weapons Damage Control Chemical/Biological Defense Deterrence	Multi-Function Applications Robotics Human Factors/Human Engineering Artificial Intelligence/Adaptive Systems Basic Scientific Research/University Interactions
Mobility Mission Areas	Mine Functions	Supply/Support/Construction Functions
Air Mobility Land Mobility Sea-Surface Mobility Undersea Mobility Space Mobility	Mine Mooring Mine Neutralization/ Destruction	Material Distribution and Payload Handling/Supply Systems Training Field Services (Water, Food, Tents, etc.) Bridging/Obstacles Support and Auxiliary Equipment Habitability Environmental Effects Facility Construction
Communications, Command and Control/Intelligence Mission Areas	C3I Functions	Management/Personnel Functions
Communications, Command and Control Intelligence, Including Reconnaissance	Information Management Communication Guidance/Navigation/Position Location Avionics/Vetronics/ Display Systems	RDT&E Management Acquisition Management Financial Management Medical/Casualty Care Performance Appraisal
Mine and Obstacle Mission Areas	Electronic Warfare Functions	Other Embedded Functional Areas
Land Mine/Obstacle/ Countermeasures Sea Mine/Countermine	Electronic Countermeasures Jamming Deception Cryptography Electronic Counter Countermeasures Low Probability Electromagnetic Signal Measurement/Intelligence Jam Resistance	Avionics Audio Signal Processing and Enhancement Command and Control Command, Control and Information Command, Control, Communications and Information Command, Control, Communications, Computers and Information Digital Signal Processing Guidance and Control Image Processing and Enhancement Operational Flight Program Simulation Telemetry Target Seeking Embedded Trainer Software Embedded Weapon
Mission and System Support Mission Areas	Assessment/Analysis Functions	
Logistics Manpower, Personnel and Training Mission/System Support	Simulation Weapons and Munitions Effects/Target Kill Assessment Vulnerability Analysis	
Weapon Systems Functions	RDT&E Functions	Other Software System Functions
Target Acquisition/ Search/Detect Threat Evaluation Target Tracking Weapon Assignment Fire Control Acquisition and Designation Launch Propulsion Control Flight Controls Conventional Munitions/ Weapons Directed Energy Weapons Hard Target Kill/Anti-Armor Fuzing Chemical Warfare (Offense)	Energetic Materials Manufacturing Technology Electronics Other Than Electronics Materials Development Metals, Ceramics, Organics and Composites Electronics Test Equipment/Technology Structural Electronics Reliability Maintainability Structures, Including Design and Manufacture Missile Aircraft Hull Body/Chassis	Decision Support Financial, Accounting, Bookkeeping, Payroll, etc. Information System Management Information System Personnel, Human Resources, etc. Operating System Online Training or Education Software