

DOD 5000.4-M-2 SOFTWARE RESOURCES DATA REPORT (SRDR) MANUAL

CHAPTER 1. SCOPE AND PROCESS

1.1 Introduction

This manual describes and explains the collection and reporting of software element data on major DoD software intensive systems. Data is required from ACAT IA, ACAT IC or ACAT ID programs containing software effort with a projected value greater than \$25M (FY 2002 dollars). The data collection and reporting applies to developments and upgrades whether performed under a commercial contract or internally by a government Central Design Activity (CDA) under the terms of a memorandum of understanding (MOU).¹ This manual is divided into six chapters. This first chapter provides background and a general description of the data categories, the government's intended use of the data, and the process by which each project defines and submits its project-specific data for archiving and future analysis.

Subsequent chapters of this manual contain samples of the DD Form 2630-1, 2630-2, and 2630-3, showing example data items, collectively known as the Software Resources Data Report or SRDR (Chapter 2), instructions to accompany the sample forms (Chapter 3), suggested language to include in any software Request for Proposal (RFP) that includes this reporting requirement (Chapter 4), the Contract Data Requirements List (CDRL) that references DD Form 2630-2 and DD Form 2630-3 (Chapter 5), and Data Item Descriptions (DIDs) referencing this manual and the relevant forms (Chapter 6).

1.2 Rationale

The purpose of this data collection is to improve the Department's ability to estimate the costs of software intensive programs. Representatives from the Service Cost Centers collaborated with the Office of the Secretary of Defense, Program Analysis & Evaluation (OSD/PA&E) to identify appropriate data to collect from DoD software intensive systems. Software intensive systems covered by the data collection include major automated information system (MAIS) programs, i.e., those that are classified as Acquisition Category IA (ACAT IA) programs, and major defense acquisition programs, i.e., those that are classified as Acquisition Category IC and ID (ACAT IC and ID) programs. Data collected from applicable programs will be limited to the type and size of the software application, the schedule and labor resources needed for its development, and optionally, the quality of the delivered software.

¹ Within this manual, the term *contract* is used to refer either to a formal contract or to a memorandum of understanding. Data reporting occurs in either case, i.e., whether the software development or upgrade is done by a commercial concern, by a CDA within the government, or by a combination of both.

All data will be requested under non-financial DD Form 2630 data items. (Financial information and project status data are specifically excluded from the required software data.) The DD Form 2630-1 records a government program manager's estimates-at-complete for a software element. This report, known as the *Initial Government Report*, is due as part of the Cost Analysis Requirements Document, or CARD. After contract award (or MOU), Defense Materiel Developers (or government CDAs) use DD Form 2630-2 to report software element estimates within 60 days of project start. This form, known as the *Initial Developer Report*, is also used to report developer estimates of any subsequent release of software within 60 days of the start of work on that release. The DD Form 2630-3, known as the *Final Developer Report* is used to report actual values within 60 days of any software release to the government as well as within 60 days of final delivery.² The developer-completed DD Form 2630-2 and DD Form 2630-3 are to be included in each appropriate contract through a Contract Data Requirements List (CDRL). Access to the data will be limited to government personnel with a need to know.

For particularly small or large software developments, the project office may shorten or lengthen the submission deadlines, accordingly. Also, the program office may choose to combine a set of smaller releases within a contract into a single release for reporting purposes. Separate software element developments within a single contract may be reported on separately or, at the discretion of the government, may be aggregated. Data for subcontracts for less than \$25 million (FY 2002) in software development may also be aggregated onto one or more reports. Software development subcontracts for more than \$25 million (FY 2002) are to be reported to the government separately, by either the subcontractor or the prime, as mutually agreed between the subcontractor and prime.

The requested data consist of brief descriptions of software *size, schedule, effort, and quality*, the minimum needed for cost estimating. These data categories are based on and limited to the core set identified by the Software Engineering Institute (SEI) and by Practical Software and Systems Measurement (PSM) for best practices of software development organizations. Chapter 2 contains sample SRDR forms showing examples of the basic data items. These forms must be customized to reflect the service or command cost estimators' and PM's agreed-upon measurement requirements and reporting format.

1.3 Background

DoD cost analysts estimate the resources required for software systems using a variety of methods. Many analysts rely on tools that require inputs such as the estimated

² If a contract covers only a single delivered software release, only one initial and one final report form, each describing the overall project, are required. However, if software is delivered to the government in two or more releases, then a separate pair of submissions is required to give initial estimates and final measurements for each release. For example, a software element delivered in three increments would have a CARD submission and an overarching initial and final SRDR submission. It would also have three release-specific initial and three release-specific final SRDRs, for a total of nine associated SRDRs.

size and type of an application, the language used, the experience of the development team, and the required reliability. These methods and tools typically yield resource and schedule estimates based on relationships derived from the past performance of a set of programs. A less formal estimating methodology that is also commonly employed depends on analogy – using historical data of similar projects to predict outcomes of future programs. In either case, cost analysts need historical data that reflect actual experience.

An experience base of software development data within OSD will become particularly important as new development methods and processes are used on software programs. Without knowledge of other similar projects, analysts are unable to judge the relevance of their estimating methods to the new regimes of software development. The centralization of data from new development methods will enable more analysts to make use of the results. Accordingly, the DoD service cost center managers requested that the Cost Analysis Improvement Group (CAIG) within PA&E research how the DoD cost analysis community could obtain better measurements of the Department's software projects in order to improve their software cost estimates.

The Contractor Cost Data Report Project Office (CCDR-PO), a subordinate organization within OSD, PA&E, established a Software Metrics Working Group (SMWG) and held numerous meetings with representatives from PA&E, the CAIG, and the service cost centers between 1999 and 2001. The SMWG also invited and consulted with representatives from defense materiel developers. Although the SMWG was established by the CCDR-PO, there is no intent to combine the SRDR with the CCDR financial report. In fact, industry direction has indicated that, because of the potential difficulty obtaining approval for reporting dollar amounts on the same form as software management data, better measurements will accrue from a software data collection process that is separate from any financial reporting.

The proposed data items are a subset of those found on the to-be-cancelled DD Form 2630, a four-page data collection form that was the predecessor to the current two-page DD Form 2630-1. DD Form 2630-2 and DD Form 2630-3 are variants of the DD Form 2630-1 and are also two pages each. The data items were specifically selected to be directly measurable and relevant to cost estimating but insensitive to the acquisition strategy used on the project. Sample SRDR forms are shown in Chapter 2.

The applicable software projects include new developments and major upgrades or re-developments of existing systems. Because all software development efforts behave in fundamentally similar ways that can be measured at a high level by size, schedule, effort, and quality attributes, both new and upgrade developments are applicable to the proposed reporting. Maintenance-only activities or post-deployment software support (PDSS) are not a part of this data collection.

1.4 General Description of Data To Be Collected

This section provides a general description of the data to be collected and why each element was chosen. Chapter 2 contains the sample SRDR forms (DD Form 2630 series) and Chapter 3 contains customization and completion instructions as well as proposed definitions of the data elements contained on the three variants in the DD Form 2630 series.

1.4.1 Project Identification and Description

The Sample SRDR begins with context information that identifies the product, developer and report. Project identification information includes: the project name, the version or release of the product, the developing organization, the report as-of date, contract number or other identifier, and reporting event (initial government report, initial contract or release report, or final contract or release report). The initial government report, DD Form 2630-1, does not include any information about the developing organization.

The initial contract or release report, DD Form 2630-2, and the final contract or release report, DD Form 2630-3, also contain project-level information that describes the process used to develop the software application. These data include: the type of application and the associated development process used, a capability rating of the developer, and a list of previous similar projects the developer has completed. It also requests information on the primary and secondary languages used, and the extent to which existing commercial off-the-shelf (COTS) or government off-the-shelf (GOTS) applications were used. All these data are used to help analysts understand the context of the product and may be used as inputs to various commercial software estimation models to refine the effort and schedule estimates.

1.4.2 Key Measurement Data

The software data that comprise the remainder of each report include measures of project size, schedule, effort, and quality. These are each discussed below.

1. Project Size

Project size is the major cost driver for most software developments and is the key quantifying dimension of the delivered product. On the Sample SRDR, project size is described by the number of functional and interface requirements, and by some measure of the amount of new, modified, and reused code that will be delivered as part of the final product. One data item is reserved to explain the units used to measure project size, such as number of lines of code, function points, forms, screens, etc. The specific size metric used in any project-specific customization will be determined by the Cost Working-level Integrated Product Team (CWIPT), which may include contractor participation.

2. Project Schedule

Schedule data reflects the time required to develop the product. The project schedule is defined by the start and end months of the software development activity (either estimated or actual). For consistency, software activity is defined to start with the identification of software requirements and not to include any earlier system requirements effort. Software activity is considered to end at delivery to the government, presumably after a test and evaluation milestone. If software activities also include installation, data conversion, or other non-developmental effort, these activities may be included in the schedule as part of the customization of the form.

3. Development Effort

Total development effort reflects the amount of staffing in hours needed to deliver the product. The form has fields for estimated or actual labor hours (depending on the reporting event) needed to develop the software product. The sample form allows data providers to input these data by software development phase or activity. Again, if software activities include installation, data conversion, or other non-developmental activities, they would appear in the project-specific customization.

4. Quality

The most commonly used measures of software quality are failure rate and defect density. On the DD Form 2630-1 submission with the CARD, the program office is asked to estimate the delivered software quality either in terms of the expected Mean Time to Defect (MTTD) or by some other means such as by analogy with the operational quality of other systems. For DD Form 2630-3, data providers either report MTTD or define and report a different operational measure of quality. DD Form 2630-2 does not contain the quality section. The quality section may be tailored out of the project-specific forms at the discretion of the CWIPT and with the concurrence of the CAIG Chairman.

1.5 How Data Will Be Used

DD Form 2630-1 and DD Form 2630-2 serve to record estimates of a project's size, effort, schedule, and quality. DD Form 2630-3 reports the actual results of a project using the same units of measure. Collecting both estimated and actual data on the size, effort, schedule and quality of various kinds of projects will allow analysts to study life cycle trends for projects in each category. This will help analysts study project growth and perform uncertainty analyses for the probable outcomes of new projects. In particular, expected schedule and effort can be put in perspective with actual experience on similar projects.

Over time, cost analysts will be able to improve their predictions of project efforts and schedules by developing relationships relating size, schedule, effort, and quality for various application types, development environments, and other project characteristics. Commercial software estimating models are also widely used by DoD analysts and the

accuracy of these can be improved through calibration with the actual experiences reported under the proposed data collection. Using historical data from similar systems, analysts will be able to make realistic projections of the expected sizes of new systems. More realistic size estimates will, in turn, result in better effort and schedule estimates.

MTTD or other quality data reported through this mechanism will help analysts understand the product quality obtained within a given schedule and effort. These measures can be used to develop estimating relationships that relate quality to size, effort, and schedule. (At least one commercial cost model uses MTTD to predict delivery date, reliability, and remaining defect density.) These results can be compared with initial expectations of mean time to failure (MTTF) and other quality goals to determine what investment is required to obtain systems of a given quality.

1.6 Data Definition and Reporting Process

All software intensive systems requiring a projected software development effort greater than \$25 million are subject to software data reporting under this proposal. Software intensive systems include MAIS (ACAT IA) programs, and major defense acquisition (ACAT IC and ID) programs. This section provides further details of the process used to specify and approve the specific software data elements that each project will report. Figures are used to illustrate the process.

For all programs, the CWIPT identifies specific data that satisfy the SRDR template and that are meaningful for the subject program. Using this guidance, the government program manager (PM) and the CWIPT develop a customized SRDR together with a set of data definitions and instructions. Chapters 2 and 3 provide the basis for a customization. The PM also develops Request For Proposal (RFP) language and a draft Contract Data Requirements List (CDRL). The PM summarizes the elements for which software resource measurement data are desired in a software resources measurement plan. The plan, including the customized SRDR, the data definitions, the draft RFP, CDRL, and DID, are to be provided to prospective developers for comments. The PM and the CWIPT will finalize the plan and submit it to the CAIG Chairman for approval. Suggested RFP language, as well as a proposed CDRL and Data Item Description (DID) appear in chapters 4, 5, and 6 respectively.³ This planning process is depicted in Figure 1, below.

³ In the case of developments of upgrades conducted by a government CDA, the CDRL and DID do not apply. Instead, the use of the SRDR would be adopted as part of the agreement or signed MOU. The suggested RFP language can be adapted for this purpose.

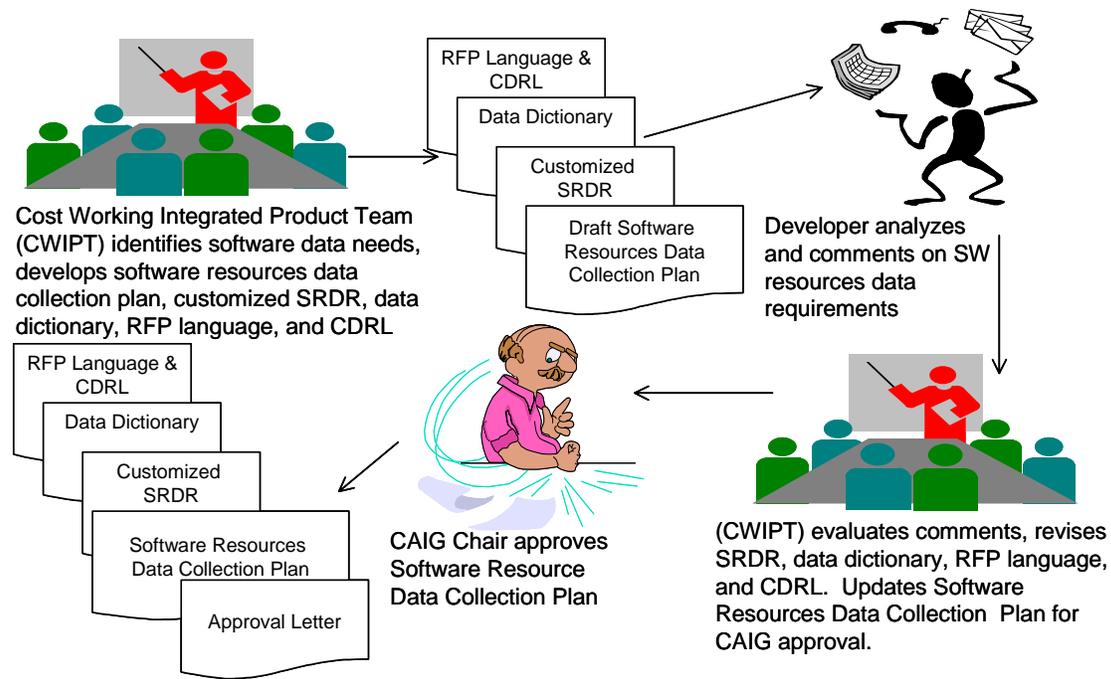


Figure 1. Software Measurement Planning Process

Contractors responding to the RFP are provided with the approved software resource measurement plan and are instructed to submit proposal-specific Software Development Plans and Software Measurement Plans that comply with the approved software measurement plan for the program. Details, such as the exact definition of software size to be used, must be included in any proposal. Small changes may be made during contract negotiations to satisfy the PM and the CWIPT as illustrated in Figure 2.

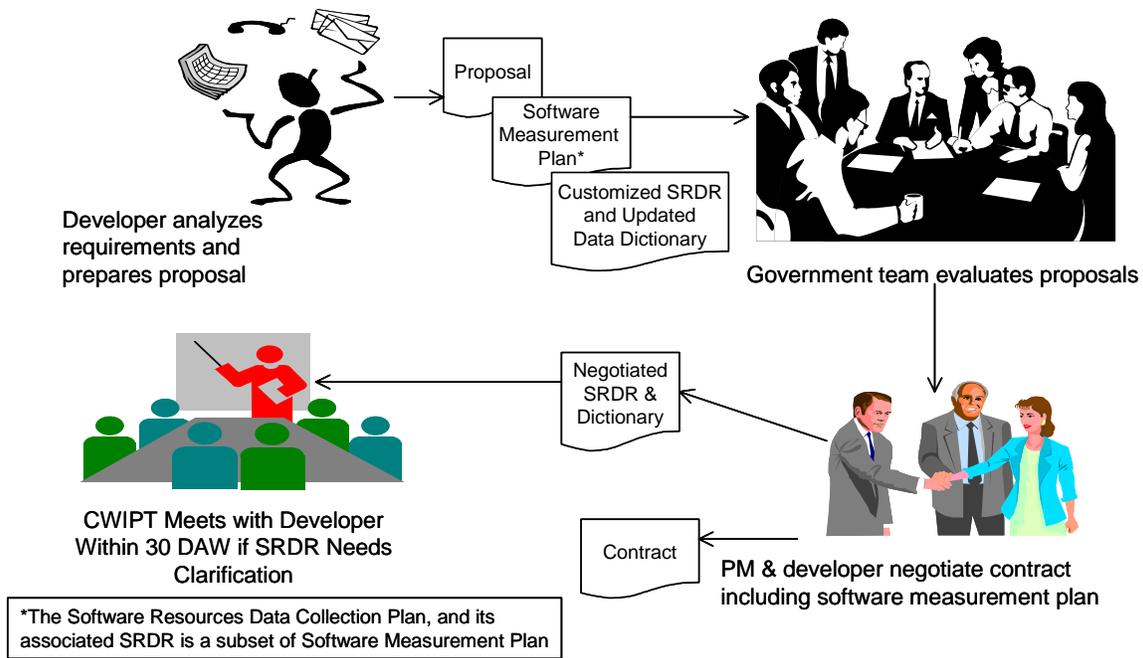


Figure 2. Software Measurement Planning Process (concluded)

Within 60 days of contract award, the software developer must submit an initial SRDR (DD Form 2630-2) for the entire software product, customized as agreed to by the CWIPT. The developer must also submit an initial SRDR for each software release or element within 60 days of its initiation. Within 60 days after development, and within 60 days after each software release or element is delivered to the government, the software developer must submit a final “as built” SRDR (DD Form 2630-3), customized as agreed to by the CWIPT. Developers must submit a final SRDR for the entire software product upon contract completion. Developers submit SRDRs to the Defense Automated Cost Information Management System (DACIMS) using established encryption technology.⁴ Government program managers may choose to receive reports for prior approval and may retrieve filed reports from DACIMS. This process is depicted in figure 3.

⁴ The web site address for the cognizant office is <http://ccdr.pae.osd.mil/>. Encryption certificates can be obtained by accessing the registration page at this site. After registering, data files can be e-mailed as attachments to CCDRPO@osd.pentagon.mil.

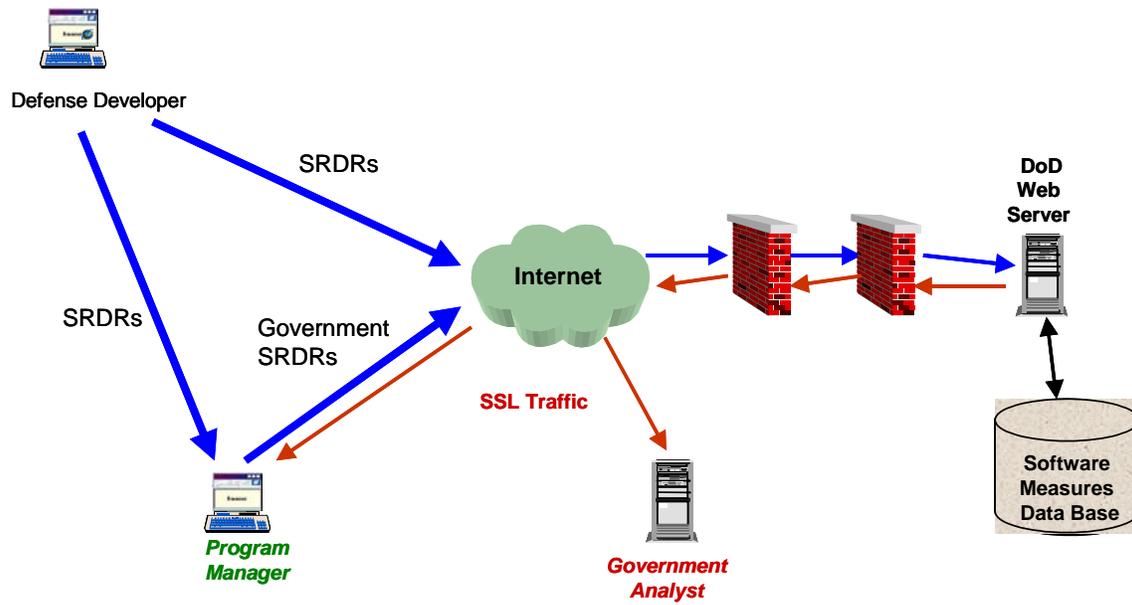


Figure 3. Software Measurement Data Collection Process

DOD 5000.4-M-2 SOFTWARE RESOURCES DATA REPORT (SRDR) MANUAL

CHAPTER 2. SAMPLE DD FORM 2630

2.1 Introduction

This chapter shows a sample Software Resources Data Report for each of the three variants of the DD Form 2630 series (DD Form 2630-1, DD Form 2630-2, and DD Form 2630-3). The sample reports are contained on the following pages.

2.2 Sample DD Form 2630-1

This is the initial government form for use by the program manager to establish expectations about the software project. (See following pages.)

2.3 Sample DD Form 2630-2

This is the initial developer report form providing estimates at complete, to be submitted by the developer within 60 days of contract award (covering the entire project) or within 60 days of the start of work for any deliverable build, release, or increment of software covered by the contract or MOU. (See following pages.)

2.4 Sample DD Form 2630-3

This is the final developer report form providing actual as-built data for each delivery of software (release, version, build, etc.), due within 60 days after each delivery (covering just that deliverable), and at contract completion (covering the entire project). (See following pages.)

DOD 5000.4-M-2
SOFTWARE RESOURCES DATA REPORT
(SRDR) MANUAL

CHAPTER 3. INSTRUCTIONS FOR THE DD FORM 2630 SERIES
SOFTWARE RESOURCES DATA REPORT (SRDR)

3.1 Introduction

The forms in the DD Form 2630 series are used to describe the development or upgrade of a major software element. The DD Form 2630 series is collectively titled the Software Resources Data Report (SRDR). Any submission of a report in the DD Form 2630 series must be accompanied by an explanatory document, known as a SRDR Data Dictionary, which explains data definitions and any details required to correctly interpret the responses. The described software development or upgrade effort can be the subject of a single software contract, a deliverable release within a larger software effort, or a software component of a larger system contract. The subject development or upgrade can be performed commercially or as an internal (“organic”) DoD effort.⁵ The DD Form 2630 is designed to record both the expectations and actual results of new software developments or upgrades. It is not designed for reporting on, nor should it be used for, software maintenance or software operation and sustainment efforts. Similarly, the reporting form should not be used for collecting management tracking measures during the course of a project since the sample data items are not designed to record partial progress or interim results.

This document explains the content of the DD Form 2630 series by describing each data item contained in the sample forms shown in Chapter 2. The data items shown on the sample forms are only examples and must be customized to be consistent with data that the development organization normally maintains to manage a project and also to be in accordance with the approved Software Resources Data Collection Plan, developed by the Cost Working-level Integrated Process Team (CWIPT). Thus, the sample forms illustrate but do not mandate the data items needed to satisfy the basic requirement to estimate and report software size, effort, schedule, and (optionally) quality at the beginning and end of a major software development or upgrade.

This chapter constitutes a set of instructions for the sample forms, showing the level of detail that would be needed to explain any customized or added data items. As such, the sections of this chapter can be used as a point of departure for a customized

⁵ For convenience, the term contract is used in this document to mean the authorizing vehicle or agreement that describes the software development or upgrade project whether or not it is in the form of a formal contract.

SRDR Data Dictionary. Other than deferring to the CWIPT, these instructions do not specify a process for customizing, completing, or submitting DD Form 2630 forms.

Three instances of the DD Form 2630 are required to record the customer's and developer's expectations as well as the actual outcome of a project: a planning report completed by the program office at the time of solicitation (DD Form 2630-1), an initial report completed by the developer at the beginning of development (DD Form 2630-2), and a final report completed by the developer at the end of development (DD Form 2630-3). Additional forms are required if the contract consists of multiple releases or constituent elements of software. In this case, separate forms are required prior to development (DD Form 2630-2) and after delivery (DD Form 2630-3) of each release or element.

The government program management office for a reporting project submits an Initial Government Report, DD Form 2630-1, customized as necessary, before contract award (e.g., as part of the Cost Analysis Requirements Document or CARD, due 180 days before contract award). The development organization (e.g., contractor or CDA) submits an Initial Developer Report DD Form 2630-2, customized as agreed upon with the program management office, within 60 days after contract award. The development organization should submit a Final Developer Report, DD Form 2630-3, customized as agreed upon, within 60 days of final delivery describing the as-delivered software product and its development process. In the case of multiple incremental deliveries (builds, releases, versions, elements, etc.), the development organization should submit, within 60 days of the start of any increment, and additional DD Form 2630-2 containing estimates for that increment. The development organization should then submit, within 60 days of delivery of an increment, an additional DD Form 2630-3 describing the as-built product and its development process.

It is assumed that forms will be submitted as computer files (Excel readable) in order to allow convenient customization of the names and numbers of data items. Each DD Form 2630 series form must be submitted with a similarly customized SRDR Data Dictionary. The sign-off area on page two includes space to identify the file name and revision for the associated SRDR Data Dictionary.

Each sample DD Form 2630 series form is divided into two pages. Page one has three sections (Section I, II, and III). Page two has two additional sections (Section IV and V) plus a sign-off area at the end. Space for brief comments, explanations, or context information is provided after each part. More extensive comments should be documented as part of the associated data dictionary.

3.2 Instructions for Part 1: Report Context

Items 1 through 4 of Part 1 should be completed for all three submissions of the DD Form 2630. Additional items (5 through 10) are to be completed after the development organization has been identified (DD Form 2630-1 and DD Form 2630-2, only).

1. System/Element Name (version/release)

This is the name used to refer to the software product being developed, including any applicable version, release, build, or other identifier. Include the name of the work breakdown structure (WBS) element and its associated WBS number.

2. Report As Of

This is the date as of which all other answers are meaningful for this submission of the form. If a subsequent report supersedes a previous report, for example to correct an error, this date would be the retroactive date of the superseded report rather than the current date.

3. Authorizing Vehicle (MOU, contract/amendment, etc.)

This is the 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.

4. Reporting Event

The event that drives this submission of the DD Form 2630 is already shown in the sample customization. Possible choices are, "CARD," "Project/Release Start," or "Contract/Release End" corresponding to the DD Form 2630-1, 2630-2, or 2630-3, respectively. Space is provided to indicate the specific submission number of this form, so as to identify it in the event that a subsequent form is needed to correct or revise an earlier submission.

5. Development Organization

For report submissions after contract award, this is the name of the company or organization that is the responsible developer of the software product being developed. The associated SRDR Data Dictionary should be used to explain the mapping of development organizations, software components and DD Form 2630 forms submitted. As with any other customization of this form, agreement on the level of aggregation must be reached between the developer and program office.

6. Certified CMM Level (or equivalent)

This is the Software Engineering Institute (SEI) Capability Maturity Model (CMM) number of the level (1 through 5) at which the primary development organization has been formally certified. If no formal certification has been conducted, leave the item blank. If a single submission is used to represent the work of multiple organizations, enter the level of the organization that will be expending the most amount of effort on the development project (not necessarily the prime contractor) and note this in the associated SRDR Data Dictionary. If the government has accepted an alternate assessment mechanism, such as the SDCE (Air Force) or ISO-15504, enter a pointer to the results here and explain the meaning of the assessment in the SRDR Data Dictionary. It is possible for this assessment to change between an initial developer and a final developer submission.

7. Certification Date

If the answer to item 7 is non-blank, this is the date when the formal assessment associated with the indicated level was conducted.

8. Lead Evaluator

If the answer to item 7 is non-blank, this is the name of the person that lead the formal SEI CMM assessment and determined the maturity level indicated.

9. Affiliation

This is the affiliation of the Lead Certifying Analyst in the previous item.

10. Precedents

Up to five analogous systems that have been developed by the same software organization or development team are listed here.

3.3 Instructions for Part 2: Product Description

Most of the items in Part 2 are included on all three forms of the DD Form 2630 series. Only the development process and developer experience are omitted from DD Form 2630-1 (initial government report). The numbers for these items are skipped in the sequence on that form so that other items have numbers that correspond to their counterparts.

1. Primary Application Type

Using one or more domain names from the list in section 3.7 of this chapter, when possible, describe the primary application type being developed. The primary type may be the only application type listed, but any number of application types may be listed. (Space for four is provided on the form but submissions may include any number.) If none of the examples shown in the list of application types are appropriate, enter a phrase to describe the application type and define it in the associated SRDR Data Dictionary. When there are internal development efforts within a program that are large and independent, respondents may choose to report each using a separate DD Form 2630 instead of as various application types within a single report.

2. Percent of Product

This is the approximate percentage of the product size that is of the indicated primary application type, up to 100%.

3. Development Process

For the initial developer DD Form 2630-2 and final developer DD Form 2630-3 submissions, this is the name of the development process planned or followed for the primary application of the system. Use common industry terms, such as waterfall, spiral, or RAD, rather than proprietary names that are internal to the development organization. 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.

4. Upgrade or New

This indicates 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.).

5. Secondary Application Type

If the development contains a major secondary application type, indicate it here.

6 - 8. Secondary Application Type Details

This indicates the system percentage of the secondary application type, its development process and whether it is new or an upgrade.

9 - 12. Third Application Type and Details

This indicates the third application type, its percentage of the system, its development process and whether it is new or an upgrade.

13-16. Fourth Application Type Details

This indicates the fourth application type, its percentage of the system, its development process and whether it is new or an upgrade. If a project includes more than four application types, extend the form or submit additional sheets as required.

17. Primary Language

This is the computer language in which most of the development is expected to be (or was) conducted. This can be a compiled language, such as FORTRAN, Ada, or C, or it can be an interpreted language, such as Forté. Use the 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.

18. Percent of Product Size

This shows the approximate amount of the final development effort that is expected to be (or was) involved with producing code in the Primary Language. This may differ somewhat from the percent of the final physical product that will be written in this language since a large portion of the delivered product might use generated code or COTS products that are not directly developed.

19. Secondary Language

This shows the secondary language used in the development (if any), using the same definitions given under the Primary Language.

20. Percent of Product Size

This shows the approximate amount of the final development effort that will be (or was) involved with producing code in the Secondary Language. This may differ somewhat from the percent of the final physical product that will be written in this language since a large portion of the delivered product might use generated code or COTS products that are not directly developed.

21. List COTS/GOTS Applications

This shows the names of the applications or products that will (or do) participate in the final delivered product, whether they are commercial off-the-shelf (COTS) or Government off-the-shelf (GOTS) products. If a proprietary application or product that is not generally commercially available will be (or was) included, identify it here and include any necessary explanation in the associated SRDR Data Dictionary.

22. Peak staff (team size in FTE) expected to work on and charge to this project

This is the expected or actual peak team size, measured in full-time equivalent staff. Only include direct labor in this calculation unless otherwise explained in the associated SRDR Data Dictionary.

23. Percent of Personnel by experience level in domain

For the initial and final reports, this is the percent of project personnel that is expected to be (or was) highly experienced in the domain (three or more years of experience), nominally experienced in the project domain (one to three years of experience), and entry level (zero to one year of experience). The percentages reported at each level should take into consideration the duration each person works on the project (so that, for example, a single highly experienced person who works on the project for two years constitutes the same percentage of the total as two entry level people who each contribute a year of effort). The experience level of a person is rated as he or she begins work on the project or the increment being reported, so that experience gained between the initial and final reports of a project or increment is not counted towards the rating.

3.4 Instructions for Part 3: Product Size Reporting

Part 3 asks for quantitative information about the size of the software development. If this is an initial, DD Form 2630-2, provide estimates-at-complete for the relevant release or delivery. If this is a final, DD Form 2630-3 then provide actual values for the delivery or release covered by this report.

1. Number of Requirements, not including External Interface Requirements

This is the number of requirements satisfied or to be satisfied by the developed software product. In the initial reports (DD Form 2630-1 and 2630-2), provide estimates of the total number of requirements to be implemented by the software being developed. In the final DD Form 2630-3, provide the actual number of requirements implemented by the developed software using the same counting method as was used in the estimating reports. Do not count requirements concerning external interfaces not under project control. Explain any details about the requirements counting methods in the SRDR Data Dictionary.

2. Number of External Interface Requirements

This is the number of *external* interface requirements 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. In the initial reports (DD Form 2630-1 and 2630-2), provide estimates of the total number of interface requirements to be handled by the software to be developed. If the developed system interfaces 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. In the final DD Form 2630-3, provide the actual number of interface requirements handled by the developed software using the same counting method as was used in the initial reports. Explain any details about the external interface requirements counting methods in the SRDR Data Dictionary.

3. Amount of Requirements Volatility encountered during development

As part of the final DD Form 2630-3 report, indicate the amount of requirements volatility 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 how initial expectations were or were not met during the course of the software development.

Code Size Measures

This unnumbered block is used to define the code size measure used in items 4 through 6. A measure other than those listed may be indicated if none of those shown are applicable. The preferred size measures are total physical source lines of code or carriage returns (to be indicated below by “S”), noncommented and nonblank source lines of code (to be indicated by “Snc”), or number of logical source statements (to be indicated by “LS”). If another size measure is being used, provide an abbreviation for it and briefly explain it. For example, unadjusted function points, adjusted function points, object points, feature points, classes, algorithms, or other functional measures could be indicated. Use the SRDR Data Dictionary for longer explanations, if required.

The size measure chosen should allow independent verification of the project size by examining the software products produced by the development. For this reason, unless a post-hoc analysis of functional size will be conducted to compare with estimated function points or other functional size estimates, one of the source code counting methods is preferred as a size measure, where “code” can refer to any hand-edited product such as lines of a computer language or lines in tables used to configure a reusable product. Many models normalize to SLOC, which is a convenient common denominator for describing product size, even if the initial planning is done using another measure, such as function points, objects, classes, screens, algorithms, etc. However, developed code size may be expressed in other terms if SLOC is a meaningless measure of the output for the majority of the programmer effort (such as when developing a web page using an iconographic tool interface). As with other customizations, the selected size measure should be in accordance with the approved Software Resources Data Collection Plan, developed by the CWIPT.

The next three items are intended to capture the size of the system under development by partitioning (exhaustive with no overlaps) the code into three categories. (Any customization of this form should maintain a partitioning categorization to avoid double counting or omissions in the delivered code size measurement.) The configuration control system is assumed to be the repository for completed code. (Unless otherwise explained in the associated SRDR Data Dictionary, code that is developed but not maintained under a configuration control system is not to be considered part of the developed system.) Only the most recent version of each code unit should be counted. For each of the next three items, indicate the size measure abbreviation in the blank provided.

4. New Code

Most software projects utilize a combination of new, reused, and generated code to accomplish the required function. Any code that was developed specifically for this project, or was reused or generated by tools but then extensively modified (more than 25% of the lines changed or added), is considered new code. Code generator inputs prepared by hand, such as tables or scripts, are also counted as new code.

5. Modified Code

Source code that was generated by tools or obtained from outside the project (even if within the same organization) and was then reused with minor modifications (less than 25% modified) by this project is reported under this item. If modifications were substantial (more than a notional 25%), the code is counted as new (item 4). This assessment should be done at the code unit level and not across the whole project.⁶

6. Reused Code

Source code that was obtained from outside the project (even if within the same organization) or that was generated by tools and not modified at all is reported under item 6.

3.5 Instructions for Part 4: Resource and Schedule Reporting

Project development is typically broken down into phases or activities. This form can be customized to include the names of the phases or activities that are appropriate for the subject development.

1 - 6. Software Development Activities

Items 1 through 6 under Part 4 are taken from the activity definitions used in ISO12207 and are intended to be generic to any software development (though they may not be strictly associated with development phases by the same names). These activities may be performed simultaneously, sequentially, or both. The two initial reports (the DD

⁶ As a simplistic example, if a 100,000-line project consists of 100 units of 1,000 lines each, and 30 of those units each have 100 modified lines (each unit being 10% modified), then that entire collection of 30,000 lines should be considered modified code. However, if another 20 units each have 300 modified lines (each unit being 30% modified), then that entire collection of 20,000 lines should be considered new code.

Form 2630-1 and the DD Form 2630-2) include estimates of the schedule and total effort applied to each activity. The final report contains actual schedules and total efforts for each activity. Many of the activities will overlap, even in a waterfall style of development. In an iterative or spiral development, activities may start and stop. To the extent that is sensible for the approach used (or expected), the dates are the earliest and latest that each activity occurred (or is estimated to occur). Month numbers, starting with month 1 at the time of Contract Award, are shown in the first two columns.⁷

7. Other Direct Software Engineering Development Effort

Item 7 is for any direct project hours that are not accounted for in the previous six items. (Schedule is not applicable to this item.) In the text space provided, summarize the kinds of activities included, such as project management, IV&V, configuration management, quality control, problem resolution, library management, process improvement, measurement, training, documentation, data conversion, or supporting a customer-run acceptance test. Also include software delivery, installation, deployment and/or implementation, to the extent these activities are included in the development contract. If any *allocated direct* charges are applied to a project, they should be included in this item.

The contribution of any *indirect* hours is described in the comment block or in the SRDR Data Dictionary (e.g., training, process improvement, methodology research) but not included in these totals.

3.6 Instructions for Part 5: Product Quality Reporting (optional)

Desired quality is requested on the program office CARD (DD Form 2630-1) report at part 5, item 1a or 1b. Actual quality of the delivered system is requested on developer final reports (DD Form 2630-3) at part 5, item 2a or 2b. No reporting of estimated quality is needed for the developer's initial reports (DD Form 2630-2). The sample DD Form 2630 suggests quantifying quality operationally (through failure rate and defect discovery rate). However, other methods may be used if appropriately explained in the associated SRDR Data Dictionary. Quality reporting may be deemed inappropriate by the CWIPT. If so, a project may tailor Part 5 out of its DD Form 2630 series reports.

1a. Required Mean Time to Defect (MTTD) at Delivery

The required MTTD at time of delivery is one method by which a customer can specify nominal product quality. The definition of this measure must include whether minor or only major (mission compromising) defects are counted, and whether recurring known defects or only new ones are counted. Also, the operational time basis must be clarified, such as by indicating whether a system is only operational eight hours a day or continuously, or whether a system operates in a single instance or in multiple instances at different locations simultaneously. Use the associated SRDR Data Dictionary to clarify the counting method.

⁷ For builds or releases that do not begin at the start of a project, such as a build subsequent to an initial build, the starting month number can be greater than 1 for schedule estimation or reporting purposes.

1b. Analogy with Similar Systems

An alternative method to specify nominal quality is to compare the required reliability of this system with typical reliability for systems of this type. For example, if the system is an operational flight program (as noted in Part 2, item 1), higher than nominal reliability might be expected for the OFP of a fly-by-wire aircraft.⁸ On the other hand, if the OFP were to control a pilotless vehicle, such as a surveillance or drone aircraft, the required reliability might be lower than average for OFP systems. A customization of this item could allow the response to be in terms relative to other similar systems, for example a scale such as “much higher,” “somewhat higher,” “nominal,” “lower,” or “much lower” would be appropriate. As with any customization, the explanation of the data must be included in the SRDR Data Dictionary.

2a. Measured or Computed Mean Time to Serious or Critical Defect (MTTD)

At Contract End, an actual measure of software quality can be reported. The DD Form 2630-3 includes items 2a and 2b as two examples of how delivered product quality may be reported. Item 2a is an example of a quantitative measure of quality using the observed or computed interval between serious or critical defect discoveries. (An example of five defect categories can be found in the superseded MIL-STD-498. Developers may customize these definitions to conform to their existing definitions.) Developers should use existing procedures for distinguishing defects from routine development changes, such as problems found after an inspection, after a configuration control baseline, or after advancement to the next state of a development process.

2b. Analogy with Similar Systems

Item 2b is an example of a qualitative measure of product quality using analogy to other similar systems. Use the SRDR Data Dictionary to document details of this or any other quality measure used.

Filename and Revision Date of Applicable Software Resources Data Report Data Dictionary

The definitions of any customized item or any other clarifying definitions of metrics reported on a submitted DD Form 2630 should be contained within a SRDR Data Dictionary. Submitters are encouraged to submit both the DD Form 2630 and the SRDR Data Dictionary as electronic files. The name and date of the file containing the data definitions should appear here.

Point of Contact and Sign Off

The form concludes with a sign-off line for the name, phone, and e-mail of the contact person to handle any inquiries about the data submitted, plus the date of completion (which would usually be later than the as-of date in part 1).

3.7 Application Types

Use the following domain names (mission and function areas) in Part 2 of the DD Form 2630 to specify the application type(s) for the software system under development.

⁸ See also section 3.7 Application Types, at the end of these instructions.

If none of these domain areas are applicable, enter a phrase that describes the application type and define it in the associated SRDR Data Dictionary. The following list contains descriptions of overlapping areas; it is not an attempt to partition the possible domain space. It is preferable to identify sufficiently general domains in Part 2 of the DD Form 2630 to avoid the need to use a large number of different application types to describe the system.

3.7.1 Warfare Mission Areas

- 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

3.7.2 Mobility Mission Areas

- Air Mobility
- Land Mobility
- Sea-Surface Mobility
- Undersea Mobility
- Space Mobility

3.7.3 Communications, Command & Control/Intelligence Mission Areas

- Communications, command & Control
- Intelligence, including Reconnaissance

3.7.4 Mine and Obstacle Mission Areas

- Land Mine/Obstacle/Countermeasures
- Sea Mine/Countermine

3.7.5 Mission and System Support Mission Areas

- Logistics
- Manpower, Personnel and Training
- Mission/System Support

3.7.6 Weapon Systems 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)

3.7.7 Defensive Systems Functions

- Hit Avoidance
- Signature Control/Suppression Reduction
- Armor, Infantry and Crew Protection
- EMP Hardening/Survivability from Nuclear Weapons
- Damage Control
- Chemical/Biological Defense
- Deterrence

3.7.8 Mine Functions

- Mine Mooring
- Mine Neutralization/Destruction

3.7.9 C3I Functions

- Information Management
- Communication
- Guidance/Navigation/Position Location
- Avionics/Vetronics/Display Systems

3.7.10 Electronic Warfare Functions

- Electronic Countermeasures
- Jamming
- Deception
- Cryptography
- Electronic Counter Countermeasures

Low Probability
Electromagnetic Signal Measurement/Intelligence
Jam Resistance

3.7.11 Assessment/Analysis Functions

Simulation
Weapons and Munitions Effects/Target Kill Assessment
Vulnerability Analysis

3.7.12 RDT&E Functions

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

3.7.13 Miscellaneous Functions

Multi-Function Applications
Robotics
Human Factors/human Engineering
Artificial Intelligence/Adaptive Systems
Basic Scientific Research/University Interactions

3.7.14 Supply/Support/Construction Functions

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

3.7.15 Management/Personnel Functions

RDT&E Management
Acquisition Management
Financial Management
Medical/Casualty Care
Performance Appraisal

3.7.16 Other Embedded Functional Areas

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

3.7.17 Other Software System Functions

Decision Support
Financial, Accounting, Bookkeeping, Payroll, etc.
Information System
Management Information System
Personnel, Human Resources, etc.
Operating System
Online training or education software

DOD 5000.4-M-2
SOFTWARE RESOURCES DATA REPORT
(SRDR) MANUAL

CHAPTER 4. SUGGESTED RFP AND SOW LANGUAGE

4.1 Introduction

This chapter contains suggested language referencing DD Form 2630 (the Software Resources Data Report or SRDR) software data reporting for inclusion in any request for proposal (RFP). Paragraphs are shown for sections L (instructions) and M (evaluation) of the RFP. It also contains suggested language to convey the statement of work (SOW) to the contractor or developer.

4.2 RFP Language for Section L, Instructions:

The government desires software measurement data on the elements identified within the attached Work Breakdown Structure. The data desired for each marked element are contained on the attached sample DD Form 2630 forms (SRDR) and associated definitions and instructions. The government desires to collect a subset of the same data that the contractor normally collects to oversee and manage software development efforts. Therefore, the government expects the contractor to customize or tailor the draft DD Form 2630 forms to be consistent with data it normally collects. The contractor shall propose the software measurement data within a Software Resources Data Collection Plan, which may be part of either a Software Development Plan or a separate Software Measurement Plan. The contractor shall provide a SRDR Data Dictionary with the customized DD Form 2630 forms.

The contractor shall submit a completed DD Form 2630-2 within 60 days after contract award for the entire software product, and within 60 days after initiation of each software release or build. The contractor shall submit a completed DD Form 2630-3 within 120 days of delivery of each delivered software release. The contractor shall submit a completed DD Form 2630-3 for the entire software product within 120 days of delivery of the final software element. Report format and other delivery requirements are specified in the attached CDRL.

4.3 RFP Language for Section M, Evaluation:

The contractor's customized SRDR and Data Dictionary will be evaluated

- a) on the extent to which the report captures the government's stated need, and
- b) on the extent to which the data provided is integrated with the contractor's normal oversight and management procedures.

4.4 SOW Language for Software Resources Data Report Compliance

The DD Form 2630 series (SRDR) is needed to supply the government with basic information about the size, effort, schedule, and quality of a developed software product. The DD Form 2630-2 and DD Form 2630-3 are authorized by the associated CDRL. To minimize the cost and maximize the meaningfulness of the data reported, the DD Form 2630 forms and their instructions must be customized so as to conform as closely as possible to measures customarily used by the software development organization while still satisfying the basic government requirements. To serve as a point of departure, the

government has provided sample DD Form 2630 forms and an associated set of instructions. These data elements represent the data the government desires. However, the government is aware that not all entities manage their software efforts using the same metrics. Therefore, the development organization is free to propose its set of software metrics that meets the overall intent of the data contained with the DD Form 2630.

The government has pre-determined – based on a targeted work breakdown structure – those elements within the WBS on which it desires a DD Form 2630. These elements are indicated within Attachment [x]. The customized DD Form 2630-2 must be submitted before and the customized DD Form 2630-3 must be submitted after development for each of the identified software elements within this WBS. The sample DD Form 2630 shows desired data items that are appropriate for the DD Form 2630-2 (prior to development) and DD Form 2630-3 (end of development) submissions. The developer shall submit a SPDR Data Dictionary with specific data item definitions for the proposed customized DD Form 2630-2 and DD Form 2630-3 as part of the software development proposal.

A DD Form 2630-2 must be submitted within 60 days of contract award and must contain estimates at complete for measures of size, effort, and schedule. Estimated values should be at the 50% confidence level, i.e., the probability that the actual value will be lower than the estimated value should equal the probability that it will be higher. A DD Form 2630-3 must be submitted within 120 days of software product delivery. A DD Form 2630-3 contains actual values at complete for measures of size, effort, schedule, and quality. If a contract anticipates multiple deliveries of a product, such as successive builds or releases, a DD Form 2630-2 must be submitted to provide estimates at complete for each expected delivery and a DD Form 2630-3 must be submitted after each software element is delivered. DD Form 2630-2 and DD Form 2630-3 must also be submitted for any elements developed by a subcontractor that exceed \$25 million (FY 2002) in software development cost. Subcontracted software development for less than \$25 million (FY 2002) may be reflected (rolled-up) in the data provided on the primary DD Form 2630 or may be reported separately, as appropriate or reasonable. A DD Form 2630-3 is required at contract completion that covers the entire software product.

The customized DD Form 2630-2 and the customized DD Form 2630-3 must reflect the same measures and definitions so that the two reports will give corresponding views of the expected and actual size, effort, schedule, and quality of a developed product.

CHAPTER 5. CONTRACT DATA REQUIREMENTS LIST (CDRL)

5.1 Introduction

This chapter contains the proposed CDRLs that reference the Software Resources Data Report (SRDR) and the associated Data Item Descriptions (DIDs) for the DD Form 2630-2 (initial developer report) and the DD Form 2630-3 (final developer report).

5.2 CDRL for DD Form 2630-2

(See following pages.)

5.3 CDRL for DD Form 2630-3

(See following pages.)

DOD 5000.4-M-2 SOFTWARE RESOURCES DATA REPORT (SRDR) MANUAL

CHAPTER 6. DATA ITEM DESCRIPTIONS (DIDS)

6.1 Introduction

This chapter contains the proposed Data Item Descriptions (DIDs) for the developer-completed DD Form 2630-2 and DD Form 2630-3.

6.2 Data Item Description for DD Form 2630-2

(See following pages.)

6.3 Data Item Description for DD Form 2630-3

(See following pages.)

DATA ITEM DESCRIPTION

Title: Software Resources Data Report: Initial Developer Report (DD Form 2630-2)

Number:

Approval Date: Draft

AMSC Number:

Limitation:

DTIC Applicable: No

GIDEP Applicable: No

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

Applicable Forms: Sample Software Resources Data Report: Initial Developer Report (DD Form 2630-2).

Use/relationship: The DD Form 2630-2 is used to obtain the expected (estimates-at-complete) characteristics of a software product and its development process. These data will be used to compile a database of software product sizes, schedules, effort, and quality that government analysts can draw upon to help predict the cost of new systems.

- a. Information to be acquired through these data will include the developer's estimates of software product size, development schedule, peak staff, and direct labor hours.

- b. The definitions of the data items are negotiable but must include the three categories of size, schedule, and effort. The contractor must provide a dictionary that defines the data elements contained on the negotiated DD Form 2630-2.
- c. The definition of the software product is negotiable but should be a named, controlled, testable, and deliverable program, subsystem, or system. A reportable product can be an incremental version, release or full operating capability, whether or not it will complete the overall system or whether or not some requirements will be deferred to a future delivery or upgrade.

The format and specific contents of this report must be tailored to reflect the negotiated data elements, data definitions, and software system definition to enable relevant and low cost data reporting. Applicable programs are all Major Defense Acquisition Programs (MDAP) that contract for (or write an MOU for) more than \$25 million (FY 2002) for software. Subcontracts for more than \$25 million (FY 2002) in software development should be reported on separate DD Form 2630-2 submissions, either by the prime contractor or the directly by the subcontractor. Subcontracts for less than \$25 million (FY 2002) in software development should be included (rolled-up) in the data reported for the prime contract DD Form 2630-2.

Requirements:

- 1. *Reference documents.* Interim guidance DODI 5000 *Defense Acquisition* provides mandatory acquisition procedures for MDAP and MAIS programs (30 October 2002). Attachment 2 of this guidance, *Operation of the Defense Acquisition System*, Tab C (Table 3), summarizes contract reporting requirements. Detailed instructions for preparing the DD Form 2630-2, the Software Product Development Report - Initial, are contained in Chapter 3 of the SRDR Manual, DoD 5000.4-M-2.
- 2. *Format.* The DD Form 2630-2 shall be in the format agreed to by the contractor and the Government as specified in the contractor's Software Development or Measurement Plan.
- 3. *Content.* The DD Form 2630-2 shall contain estimated software measurement data as described in the contractor's software development plan and software measurement data element dictionary.

DATA ITEM DESCRIPTION

Title: Software Resources Data Report: Final Developer Report (DD Form 2630-3)

Number:

Approval Date: Draft

AMSC Number:

Limitation:

DTIC Applicable: No

GIDEP Applicable: No

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

Applicable Forms: Sample Software Resources Data Report: Final Developer Report (DD Form 2630-3).

Use/relationship: The DD Form 2630-3 is used to obtain the actual (at complete) characteristics of a software product and its development process. These data will be used to compile a database of software product sizes, schedules, effort, and quality that government analysts can draw upon to help predict the cost of new systems.

- d. Information to be acquired through these data will include the developer's measures of software product size, development schedule, peak staff, direct labor hours, and quality.
- e. The definitions of the data items are negotiable but must include the four categories of size, schedule, effort, and quality. The contractor must provide a dictionary that defines the data elements contained on the negotiated DD Form 2630-3.
- f. The definition of the software product is negotiable but should be a named, controlled, testable, and deliverable program, subsystem, or system. A reportable product can be an incremental version, release or full operating capability, whether or not it completes the overall system or whether or not some requirements have been deferred to a future delivery or upgrade.

The format and specific contents of this report must be tailored to reflect the negotiated data elements, data definitions, and software system definition to enable relevant and low cost data reporting. Applicable programs are all Major Defense Acquisition Programs (MDAP) that contract for (or write an MOU for) more than \$25 million (FY 2002) for software. Subcontracts for more than \$25 million (FY 2002) in software development should be reported on separate DD Form 2630-3 submissions, either by the prime contractor or the directly by the subcontractor. Subcontracts for less than \$25 million (FY 2002) in software development should be included (rolled-up) in the data reported for the prime contract DD Form 2630-3.

Requirements:

4. *Reference documents.* Interim guidance DODI 5000 *Defense Acquisition* provides mandatory acquisition procedures for MDAP and MAIS programs (30 October 2002). Attachment 2 of this guidance, *Operation of the Defense Acquisition System*, Tab C (Table 3), summarizes contract reporting requirements. Detailed instructions for preparing the DD Form 2630-3, the Software Product Development Report - Final, are contained in Chapter 3 of the SRDR Manual, DoD 5000.4-M-2.
5. *Format.* The DD Form 2630-3 shall be in the format agreed to by the contractor and the Government as specified in the contractor's Software Development or Measurement Plan.
6. *Content.* The DD Form 2630-3 shall contain actual software measurement data as described in the contractor's software development plan and software measurement data element dictionary.

