DATA ITEM DESCRIPTION

1. DID NUMBER: DID-ENG-SW-SWLIST-
2. TITLE: SOFTWARE LIST
3. DESCRIPTION AND INTENDED USE

The Software List (SWLIST) identifies and describes each Software product that forms part of the Supplies or would otherwise be developed or acquired under the Contract and includes additional attribute information for each Software product.

The Contractor uses the SWLIST to:

list the Software products to be developed or acquired under the Contract and those to be supplied to the Commonwealth; and

document key Software characteristics of interest to the Commonwealth.

The Commonwealth uses the SWLIST to:

achieve early visibility into the criticality, quantity and nature of the Software to be supplied and subsequently supported; and

understand the scope of Software to be delivered to the Commonwealth and Associated Parties, and the rights associated with that Software.

1. INTER-RELATIONSHIPS

The SWLIST is subordinate to the following data items, where these data items are required under the Contract:

Systems Engineering Management Plan (SEMP); and

Software Management Plan (SWMP).

The SWLIST inter-relates with the following data items, where these data items are required under the Contract:

System Architecture Description (SAD);

Mission System Technical Documentation Tree (MSTDT);

Contract Work Breakdown Structure (CWBS); and

Software Support Plan (SWSP).

The SWLIST inter-relates with the Technical Data and Software Rights (TDSR) Schedule.

1. ApPLICABLE DOCUMENTS

The following documents form a part of this DID to the extent specified herein:

|  |  |
| --- | --- |
| 1. Nil |  |

1. PREPARATION INSTRUCTIONS
   1. Generic Format and Content

The data item shall comply with the general format, content and preparation instructions contained in the CDRL clause entitled ‘General Requirements for Data Items’.

The SWLIST shall be provided in soft copy as a structured data file (eg, one or more databases, spreadsheets or other structured data format) that enables the SWLIST content to be accessed, queried, read, printed and used to generate soft copy tabulated text reports.

Except where the soft copy data file is compatible with a standard Software application defined elsewhere in the Contract, or otherwise agreed in advance and in writing by the Commonwealth Representative, the SWLIST shall be accompanied by any Software and Technical Data required to enable the functions identified in clause 6.1.2.

The data item shall include a traceability matrix that defines how each specific content requirement, as contained in this DID, is addressed by sections within the data item.

* 1. Specific Content
     1. Identity

The SWLIST shall identify each Software product or logical aggregation of Software products using a unique identifier.

Where the SWLIST is being used to report the content of a software build or increment, the build or increment shall be uniquely identified.

* + 1. Location in the System Hierarchy

The SWLIST shall identify the location of each Software product in the Materiel System hierarchy (eg, processing element, equipment, subsystem and system) using an indentured numbering system that provides traceability from the Software product to the top-level system. The indentured numbering system shall reflect the CWBS element numbers, unless specified otherwise in the Approved SWMP.

* + 1. Description

The SWLIST shall provide a brief description of the function or purpose of each Software product in terms of its contribution to functionality of the Mission System and / or Support System, cross-referring to the SAD where applicable.

* + 1. Software Criticality

The SWLIST shall identify the criticality of each Software product in accordance with the following table:

| Criticality | Effect on Materiel System | | Effect on Contract | |
| --- | --- | --- | --- | --- |
| Performance | Support | Cost | Schedule |
| 0 | 1. Software product is ‘safety critical’. Failure may result in loss of life, injury, or significant damage to property or the environment. | | 1. Not applicable. | |
| 1 | 1. Product is ‘mission critical’. Product failure results in mission failure due to: | | 1. Delays in schedule result in: | |
| 1. Major degradation of operational capability. | 1. Unresponsive support or unsupportable Software hinders system operation. | 1. Significant cost overrun, budget overrun likely or has occurred. | 1. Scheduled date for first System Acceptance is unachievable. |
| 2 | 1. Product failure results in degraded performance to a point where mission success is questionable due to: | | 1. Delays in schedule result in: | |
| 1. Significant reduction of operational capability. | 1. Software support, or work-around, delays or reduces system operation. | 1. Cost overrun with possible budget overrun. | 1. Possible slippage in scheduled date for System Acceptance. |
| 3 | 1. Product failure results in degradation of secondary mission due to: | | 1. Delays in schedule result in: | |
| 1. Minor reduction of operational capability. | 1. Software support, or work-around, delays or reduces secondary capability. | 1. Cost overrun but sufficient remaining budget. | 1. Compressed schedule, but scheduled Acceptance date is realistic and achievable. |
| 4 | 1. Product failure results in inconvenience with: | | 1. Delays in schedule result in: | |
| 1. No reduction in operational capability. | 1. No noticeable delays caused by Software support. | 1. Minor cost increase with negligible impact to budget. | 1. Negligible impact to the achievement of Acceptance. |

* + 1. Software Categories

Each Software product shall be categorised by a single category from the following table. Mission System and Support System Software may include both Bespoke Software (as defined in the table) and Commercial Software. Where a Software product is integrated from lower level Software products, which are of a different category, these lower level products need to be separately identified and reported in the SWLIST.

| Software Category | Description | Comments |
| --- | --- | --- |
| 1. Bespoke Software | 1. Software that is subject to software development or integration activities. | 1. Source Code may be available to the Commonwealth and allow the Commonwealth to modify and maintain the software independently of the original supplier. May integrate one or more subcomponents that are Commercial Items or Free and Open Source Software. |
| 1. Commercial Software that is not Free and Open Source Software (CNF) | 1. Commercial Software as defined in the Glossary, exclusive of Free and Open Source Software. | 1. Development is not required to meet the requirements of the system being acquired. Unless agreed in relation to a Key Commercial Item, the Commonwealth is unlikely to be able to acquire Source Code and/or the legal rights to modify or re-engineer the software. |
| 1. Commercial Software that is Free and Open Source Software (FOSS) | 1. Free and Open Source Software, as defined in the Glossary. | 1. Generally available to the public in Source Code and may also include compiled form. Subject to a variety of open source licences. Ongoing support may be provided from an open source community. |

* + 1. Other Software Attributes

The SWLIST shall identify whether each of the attributes, in the following table, applies to each Software product (ie, yes or no for each).

|  |  |  |
| --- | --- | --- |
| Software Attribute | Description | Comments |
| 1. Software as Firmware (SAF) | 1. Firmware is a combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control. | 1. SAF has not always been recognised as software but treated as hardware or as a component of a hardware item (eg, software-controlled electronics such as radios and GPS). SAF may not always be identified as a supportable item independent of the hardware item that contains it. |
| 1. Non Deliverable Software (NDS) | 1. Software that is not required to be delivered to the Commonwealth or any other person under the Contract because the Commonwealth does not need it for operation or support of the system (eg, unit test harnesses not required for support). NDS is generally used in the development and testing of other software or system elements. | 1. NDS may be Bespoke Software or Commercial Software (either CNF or FOSS). 2. NDS may be used to test or exercise other software or hardware as part of that product’s development. 3. Consideration should be given to Commonwealth needs for access to identified NDS over the life cycle. |

* + 1. Level

The SWLIST shall identify the level of the Software product (ie, item, component or unit) in the system hierarchy. Software items may be designated as ‘configuration items’ while the Software architectural design process transforms items into ‘components’ and the Software detailed design process refines components into ‘units’.

* + 1. Language

The SWLIST shall identify the programming language used / to be used to develop each Software product.

* + 1. Software Size Information
       1. General

Software size details in the SWLIST shall be provided in Source Lines of Code (SLOC) (or thousand SLOC (KSLOC)), or an equivalent development-related unit of measure (eg, function points) with the Contractor’s recommended methodology for converting to SLOC.

The SWLIST shall clearly identify whether the Software sizing information provided is an actual value (denoted ‘(A)’) or estimated value (denoted ‘(E)’) (eg, ‘542,341 SLOC (A)’).

Where Software sizing information is an estimated value, the SWLIST shall include the most recent date at which the estimate was considered valid.

Except for the Estimated Total Size, other size estimates may be expressed either as an absolute value, using the same units as for the Estimated Total Size, or as a relative value (ie, a percentage).

* + - 1. Estimated Total Size

For each item of Bespoke Software the SWLIST shall identify the estimated or actual total size of all code in accordance with the requirements of clause 6.2.9.1.

* + - 1. Reused Unmodified Code Required

For each item of Bespoke Software, the SWLIST shall identify the estimated or actual size of the code to be reused without modification in accordance with the requirements of clause 6.2.9.1.

* + - 1. Estimated Modified Code Required

For each item of Bespoke Software, the SWLIST shall identify the estimated or actual size of the codeto be modified (ie, reused with modification) in accordance with the requirements of clause 6.2.9.1.

* + - 1. Estimated New Code Required

For each item of Bespoke Software, the SWLIST shall identify the estimated or actual size of new codeto be developed in accordance with the requirements of clause 6.2.9.1.

* + 1. Development Standard

The SWLIST shall identify the software development standard applied to each extant Software product or that will be applied to Software products during development or modification.

* + 1. Assurance Standard

The SWLIST shall identify the software assurance standard applied to each extant Software product or that will be applied to Software products during development or modification.

* + 1. Software Assurance Level

The SWLIST shall identify the Software assurance level applied to each extant Software product or that will to be applied to Software products during development or modification.

The SWLIST shall define the Software assurance levels where these differ from the assurance levels specified for an assurance standard that was identified in response to clause 6.2.11.

* + 1. Source Code Availability

For each item of Bespoke Software, the SWLIST shall indicate the availability of Source Code.

* + 1. Development Agency

The SWLIST shall identify the development agency for each Software product.

* + 1. Support Agency

The SWLIST shall identify the support agency for each Software product.

* + 1. Target Platform

The SWLIST shall identify the target (computing) platform for each Software product.

* + 1. Target Environment

The SWLIST shall identify the target environment (eg, operating system) for each Software product.

* + 1. Software Support Environment

The SWLIST shall describe the support environment needed for each Bespoke Software product, including any development and/or test environment(s) (eg, compilers, editors, debuggers, computer aided software engineering tools, and special test equipment (eg, simulators and stimulators)).

* + 1. Delivery Information

The SWLIST shall include delivery information, including for each delivery:

if the Software product is delivered separately or as part of a higher level system / hardware component;

if the Software product is delivered separately (which may include maintenance / version updates), the method of delivery (eg, online, media);

the delivery location, recipient, delivery date and milestone to which it relates; and

installation, configuration, adaptation and compatibility information, as applicable.

* + 1. Software Rights

If restrictions (including Intellectual Property rights, Export Approvals or other limitations) apply to Bespoke Software or Commercial Software related to a Key Commercial Item, the SWLIST shall include cross-reference to such provisions as described in the TDSR Schedule for licensing or delivery restrictions, or directly to the applicable agreement (eg, an applicable Technical Assistance Agreement).