

Table 1. Responses from 113 Agencies Surveyed

	Very Common	Fairly Common	Not Very Common	Very Rare	Never Occurs	Don't Know
Software development has dollar overrun	21.2	29.2	25.7	9.7	6.2	8.0
Software development has calendar overrun	30.1	31.9	25.7	8.0	1.8	2.7
The delivered software must be corrected or modified by in-house programmers before it is usable	8.8	34.5	35.4	13.3	6.2	1.8
The software is paid for but never used	-	3.6	16.1	57.1	20.5	2.7
The delivered software is difficult to modify	5.3	37.2	38.1	11.5	4.4	3.5
The contractor's programming practices are such that the software is easily understood by agency programs	14.2	62.8	15.0	6.2	-	1.8

Peter Coad, Jr.
Project Concepts, Inc.
15771 Exeter Street
Westminster CA 92683
(714) 839-8900

Far too often people assume that DoD-STD-2167, *Defense System Software Development*, requires a top-down design approach. Yet this is not true. Some exact quotes are in order. But first, some acronyms and an appendix name are needed:

CSCI Computer Software Configuration Item
SAM Software Acquisition Manager
SDP Software Development Plan
SSPM Software and Standards Procedures Manual
TLCSC Top-Level Computer Software Component

Appendix D Guidelines for Tailoring This Standard

And now a quote from DoD-STD-2167, 4 June 1985:

4.8 *Development methodologies.* The contractor shall use a top-down approach to design, code, integrate, and test all CSCIs, unless specific alternate methodologies have been proposed in either the SSPM or SDP (see Appendix D) and received contracting agency approval (see 6.2).

5.3.1.3 The contractor shall refine all TLCSCs using a top-down design approach, unless specific alternate methodologies have been proposed in either the SSPM or SDP (see Appendix D) and have received contracting agency approval.

5.3.1.4 The contractor may depart from a top-down approach to: (1) address critical lower-level elements or (2) incorporate commercially available, reusable, and Government furnished software. The contractor shall describe the criteria for determining critical lower-level elements in either the SSPM or SDP (see Appendix D). Examples of criteria for determining criticality are software performance, cost, and schedule.

Hence, the default approach is top-down. Top-down with bottom-up is acceptable (with appropriate criteria in place). And alternative methodologies may be utilized (upon contracting agency approval). DoD-STD-2167 is meant to be tailored.

The draft DoD-HDBK-287, *Defense System Software Development Handbook*, 8 August 1986, clarifies the matter further:

4.6.9 (4.8) *Development Methodologies.* There are many development methodologies employed in the software development field. Most, including the top-down approach, work better with certain types of applications and software languages. Therefore, each of the various design methodologies has differing advantages for a contractor. The contractor may propose an alternative development methodology to top-down, which is then approved by the contracting agency. The contractor should demonstrate before the SAM approves its use that the proposed methodology is an effective, uniform, disciplined approach to software development and is easily supported and maintained.

The draft DoD-STD-2167A, 15 August 1986, slashes the 2167 text down to size:

4.7 *Developmental methodologies.* The contractor shall use a top-down approach to design, code, integrate, and test all CSCIs, unless specific alternate methodologies have been proposed in the SSPM or SDP. The SSPM and SDP are subject to contractor agency disapproval.

5.3.1.3 The contractor shall refine all TLCSCs in accordance with the design methodology described in the SSPM or SDP.

The default design approach is top-down. Other methodologies may be utilized (upon contracting agency approval).

DoD-STD-2167 has many options. Defaults are provided. For cost-effective software development, DoD-STD-2167 must be tailored for specific projects.

Table 2. Causes of Case Study Problems

Cause	Case Number								
	1	2	3	4	5	6	7	8	9
Agency overestimated its own state of progress when it let the contract	x	x				x		x	
Incorrect agency management action, such as using inappropriate contract			x	x		x		x	
Agency failed to specify requirements adequately	x	x		x		x	x	x	
Agency overcommitted itself							x		x
Agency failed to manage during execution, including excessive changes	x	x				x	x		x
Agency failed to adequately inspect and test	x	x	x			x		x	