



## System Development and Life-Cycle Management (SDLCM) Methodology

<b>Subject</b> Software Engineering Notebook	<b>Type</b>	Standard
	<b>Identifier</b>	S-3091
	<b>Effective Date</b>	February 2002
	<b>Revision No.</b>	2

Approval

CISSCO Program Director

### A. PURPOSE

This standard establishes the minimum requirements for software engineering notebooks (SENs).

### B. APPLICABILITY

This standard applies to all software elements (for example, modules, units, data files, database subschema), whether new, adapted, or converted, composing a software system under development or in maintenance. This standard does not necessarily apply to unchanged transported software elements.

### C. REFERENCE PUBLICATIONS

The following publications contain related information:

- *SDLCM Methodology Handbook*
- SDLCM Methodology Procedure P-3041, Software Engineering Notebook Processing

### D. STANDARD

The SEN is an implementation workbook that consolidates the information pertinent to a software element (or set of software elements). It also ensures ready access to complete and up-to-date information for modification and auditing purposes.

Only the most current information about software development and testing is maintained in each SEN. To reduce duplication, SENs do not contain information provided in other documents or data sets. Instead the SEN provides a pointer reference to the relevant document(s). Whenever possible, the SENs should provide pointer references to on-line libraries and files rather than maintaining hard copy material.

For each project, identify a standard medium for the SENs. Suggested options include one or a combination of the following:

- On-line (either on personal computers or on the development system, if resources are available)
- Single folder or notebook for each software element

<b>Subject</b> Software Engineering Notebook	<b>Type</b>	Standard
	<b>Identifier</b>	S-3091
	<b>Effective Date</b>	February 2002
	<b>Revision No.</b>	2

- Single folder or notebook containing all software elements for a single system, subsystem, executable image, package, or module

Include at least the following items in the SEN:

- Complete current listings/source code, including prolog, program design language (PDL), unit test plan, and compiled source code
- All applicable certification records (for example, unit design, code, test, and modification)
- For module-level notebooks, include architecture diagrams, module test plans, and associated certification records

SENs will be completed for each developed software element and updated for each modification to a software element. Software developers will maintain control of the SENs until completion of the software development life cycle (begin acceptance testing), at which time the Configuration Management Office will maintain the SEN library to ensure an accurate record of checked out and returned SENs.