

CHAPTER 4 - SOFTWARE MANAGEMENT

1. PURPOSE. This policy establishes the principles and requirements that govern the planning, acquisition, development, maintenance and use of Agency software resources. This policy also defines the roles and responsibilities for implementing these principles and requirements.
2. SCOPE AND APPLICABILITY. This policy applies to all EPA organizations and their employees. It also applies to the personnel of agents (including State agencies, contractors and grantees) of EPA who are involved in the design, development, acquisition, operation and maintenance of Agency software, data and information systems. The requirements of this policy apply to existing as well as new or modified/ enhanced software systems.
3. BACKGROUND.
 - a. Directly or indirectly, most EPA managers are involved with automated information systems or the information resources management process. This involvement can be with the information itself and related resources, e.g., personnel, equipment, funds, systems and technology (hardware and software). As agencies become increasingly dependent on information technology to accomplish their basic missions, it is essential that these technologies be acquired and used in a rational way.
 - b. The EPA software management program is needed to manage and protect EPA information as a valuable national resource; promote cross-media analysis and information interchange for environmental results; reduce costs while maximizing benefits for program management and improve the quality, uniformity and maintenance of software products.
 - c. The objectives of EPA's software management program include the following:
 - (1) Secure EPA's investment in information collection, processing, dissemination, use, storage and disposition.

(a) Much of EPA's software investment is custom software (i.e., developed by in-house or contractor staff), as opposed to software commercially marketed or developed by other government agencies.

(b) It is important that systems development, operation and maintenance be managed to ensure that this investment yields software products which are sound, maintainable and not subject to disruption.

(2) Improve the quality, uniformity and maintenance of software systems.

(a) Decisions regarding the selection of such items as computer environment, programming languages, processing techniques, ergonomic screen design, terminal key functions and documentation products have been left up to the individual project officer, contractor or in-house developers.

(b) This has resulted in some successful systems, while others have difficulties attributed to the lack of an effective software management program.

(3) Improve the cost-effective acquisition, development, maintenance and ongoing operation of software systems.

(a) EPA spends a significant amount of its information resource dollars on custom software development, maintenance and ongoing operation of information systems.

(b) Improving the cost-effectiveness of these efforts can be achieved by standardizing techniques, methods, products and tools for systems engineering for all phases of the information systems life cycle and by the acquisition and use of commercial software where appropriate.

(4) Promote inter-agency cooperation and sharing of software and data.

(5) Improve the end-user computing environment and access to EPA's information resources.

(a) EPA is increasingly relying on end-user computing. The key to end-user computing is the availability of easy-to-use software tools and ready-to-go" applications software.

(b) This can be achieved through several measures, including standardizing and supporting software tools for the end-user computing environment; providing training, software revisions and user support; expanding the "information center" approach to support the end-user computing environment; promoting access by Agency staff to information systems and resources; and developing and disseminating systems engineering standards and guidelines for all software life cycle phases of end-user developed applications.

(6) Develop plans for future software investments in areas with high payoff for the Agency's mission.

(a) While tools such as fourth generation languages have measurable benefits and significant productivity gains, there are future areas of software investment which promise even greater benefits and gains.

(b) These include greater reliance on generic, off-the-shelf software applications, as opposed to developing custom software; office automation software with greater levels of integration of functions, features and capabilities; expert systems or artificial intelligence applications for EPA mission and program goals; geographic information systems for environmental analysis; and the development and enforcement of software engineering standards to gain a greater degree of discipline and rigor in the software process.

d. The policies described in the remainder of this chapter a framework for establishing this software management program.

4-3

IRM POLICY MANUAL

2100
7/21/7

4. AUTHORITIES.

- a. OMB Circular No. A-130, Management of Federal Information Resources, December 12, 1985.
- b. NBS FIPS PUB 38, Guidelines for the Documentation of Compute. Programs and Automated Data Systems, February 15, 1976.
- c. NBS FIPS PUB 64, Guidelines for Documentation of Computer Programs and Automated Data Systems for the Initiation Phase, August 1, 1979.
- d. NBS FIPS PUB 105, Guidelines for Software Documentation Management, June 6, 1984.
- e. NBS FIPS PUB 106, Guidelines on Software Maintenance.
- f. NBS FIPS PUB 101, Guidelines for Lifecycle Validation, Verification and Testing of Computer Software.
- g. EPA Office Systems Feasibility Study, Implementation and Operational Guidelines, January 1985 (OIRM).
- h. EPA ADABAS Application Development Procedures Manual, October 17, 1984 (revised December 2, 1985), NDPD.

5. POLICY. It is EPA policy to enhance the management of software throughout its life cycle. It is also EPA policy that software developed by or acquired for the Agency will use EPA standard software tools and adhere to EPA standards and guidelines.

- a. The use of existing government and commercially available and tested software application packages is required wherever technically and economically feasible.
- b. Whenever custom programming is required, maximum use of automated tools for software design, development, testing and maintenance will be made.
- c. EPA offices and staff will jointly acquire and share software resources wherever possible. This applies to the acquisition of proprietary software products and development of software under contract or with in-house resources. Software that has the potential for being shared will be developed or acquired after an evaluation of the general requirements of interested offices.

4-4

IRM POLICY MANUAL

2100

7/21/87

- d. Copyright laws and other measures designed to protect legitimate proprietary interests in software and data must be rigidly enforced. Classified and unclassified data and software must be protected from improper access, use, alteration, manipulation or unauthorized disclosure as a result of criminal, fraudulent or other improper actions.
- e. In the absence of overriding efficiency considerations, all software resources must: satisfy functional requirements; provide interfaces consistent with users' needs and skill levels; meet users' availability needs; provide data integrity; provide response times acceptable to users under routine and unusual conditions (i.e., peak workloads, equipment failure); and meet users' security requirements.
- f. EPA program officials will adhere to Federal Information Processing Standards (FIPS) and guidelines as published or adapted for the Agency in developing, documenting, maintaining and using software applications.
- g. EPA program officials managing the development or ongoing operation of software applications are responsible for the management of life cycle costs, conformance to software standards and data base administration procedures, training, operations maintenance and user support and evaluation.
- h. The development of all application systems will conform to the Agency's system development life cycle methodology.

i. The use of fourth generation or other non-procedural languages and tools is recommended in lieu of third generation, procedural language-based custom development efforts. Customized third generation or procedural languages and tools may be required to meet functional requirements for reasons of security, portability and efficiency. The use of assembler languages is restricted to exceptional situations, such as when modifying an existing program written in assembler language, writing a program for an operating system and an application requiring the use of assembler language.

j. All EPA applications systems development efforts must use the Agency's standard application programming languages.

4-5

IRM POLICY MANUAL

2100

7/21/87

k. Applications should be designed to require the least possible amount of computer operator and programmer support for execution.

l. EPA program officials will periodically review all software resources to determine and prevent obsolescence of software. Indicators of obsolescence include: dependence on obsolete peripherals; running in an emulation mode; inadequate operating system or documentation and more than 5 years since the last substantial redesign.

m. Information technology provided to EPA employees and their agents is to be used for official business only. EPA managers and supervisors are responsible for ensuring appropriate use of this technology by their employees.

6. RESPONSIBILITIES.

a. The Office of Information Resources Management (OIRM) is responsible for:

(1) Managing information resources, functions and activities within EPA, in accordance with the Paperwork Reduction Act of 1980 (P.L. 96-511), Federal Information Processing Standards (FIPS), OMB Circular No. A-130 (Management of Federal Information Resources) and other

Federal regulations.

(2) Defining EPA software management/engineering policies, standards and guidelines in the interests of standardization, productivity and effective management of software and information resources.

(3) Review and approval of technical specifications for software requested by OARM, ORD and the program offices.

(4) Publishing plans and guidance for administrative, program and research/laboratory systems.

(5) Conducting compliance reviews.

4-6

IRM POLICY MANUAL

2100

7/21/87

b. The Assistant Administrators, Associate Administrators, Regional Administrators, Laboratory Directors, Headquarters Staff Directors, General Counsel and Inspector General are responsible for:

(1) Ensuring compliance with software management policies, standards and guidelines.

(2) Managing the software life cycle, process and products within their program(s).

c. The Senior IRM Officials are responsible for:

(1) Approving microcomputer proprietary software.

(2) Initially approving requisitions for acquisitions of information technology prior to their review by NDPD and/or OIRM.

d. The Director, National Data Processing Division, is responsible for:

(1) Acquiring all general purpose, non-application specific software such as operating systems, data base management systems, etc.

(2) Approving system-oriented proprietary software.

e. The Procurement and Contracts Management Division and the Grants Administration Division are responsible for:

(1) Ensuring that all policy, standards and guidelines specified by OIRM are incorporated in Requests for Proposals (RFPs), Interagency Agreements (IAGs), Cooperative Agreements, Grants, Contracts and Sub-Contracts.

f. Each EPA Manager, Supervisor, or Project Officer engaged in information resources management activities is responsible for:

(1) Conforming to the software management/engineering program policies, methods, standards, guidelines and techniques contained in this and related documents.

4-7

IRM POLICY MANUAL

2100
7/21/87

g. Each EPA employee, contractor and grantee engaged in information resources management activities is responsible for:

(1) Conforming to Agency software management/engineering program policies, methods, standards, guidelines and techniques.

7. DEFINITIONS

a. "Application Software" means software specifically produced for the functional use of a computer system, e.g., payroll, inventory control, environmental monitoring and scientific modeling.

b. "Artificial Intelligence, Expert, or Knowledge-based Systems refers to a class of systems that employ decision

rules developed through human experience and from human knowledge to solve problems that require a high degree of human expertise.

c. Data Base Management System (DBMS)" is the software product that provides data structure containing unrelated data stored, so as to optimize accessibility, control redundancy and offer multiple views of the data to multiple application programs.

d. "Documentation" refers to information to support the effective design, management, operation, maintenance and transferability of ADP resources, and to facilitate the interchange of information. Documentation includes analysis, technical documents and specifications which are produced in the software life cycle (e.g., project request, feasibility study, cost/benefit, functional requirements, data requirements, system/subsystem specifications, program specifications, data base specifications, test plan, user's manual, operations manual, test reports and maintenance procedures).

e. Fourth Generation (4GL) Programming Language refers to modern programming languages (e.g., INFO, FOCUS) designed for end-users or to increase programmer productivity, which have a number of tools such as English language

4-8

IRM POLICY MANUAL

2100

7/21/87

syntax, dictionaries, screen builders and reference to data by name. These languages tend to be dependent on specific computer architectures and are not usually transportable. They usually imply a proprietary Data Base Management System (DBMS) or Data Management System (DMS).

f. "Geographic Information System (GIS)n is a system that combines geographic and/or cartographic analysis capabilities with a computer data base system that can support data entry, data management, data manipulation and data display.

g. "Non-procedural Language" see definition for Fourth

Generation (4GL) Programming Language under "e".

h. "Procedural or High Order Language" see definition for Third Generation Language (3GL) under "o".

i. "Software means computer programs, procedures, rules and possibly associated documentation and data pertaining to the operation of a computer system.

j. "Software Engineering" refers to the discipline of applying software tools, techniques and methodologies to promote software quality and productivity.

k. "Software Life Cycle" is the period of time beginning when a software product is conceived and ending when the product no longer performs the functions for which it was designed. The software life cycle is typically broken into phases, such as requirements, design, programming and testing, installation and operation and maintenance.

l. "Software Maintenance" means the performance of those activities required to keep a software system operational and responsive after it is accepted and placed into operation. It is the set of activities which result in changes to the originally accepted (baseline) product. These changes consist of modifications required to: (1) insert, delete, extend and enhance the baseline system (perfective maintenance); (2) adapt the system to changes in the processing environment (adaptive maintenance) and (3) fix errors (corrective maintenance).

4-9

IRM POLICY MANUAL

2100

7/21/87

m. "Software Tools refers to packaged, often commercial, computer program(s) used to help develop, test, analyze or maintain computer programs, data and information systems. Examples include statistical software such as SAS, SPSS, sort systems, etc.

n. "Testing" refers to examining the behavior of a program by executing the program on sample data sets.

o. Third Generation (3GL) Programming Language is a programming language that usually includes features such as nested expressions and parameter passing, that can run on a variety of different computer systems and are independent of machine architecture (e.g., COBOL, BASIC, FORTRAN, PL/I). It is a problem oriented language that facilitates the expression of a procedure as an explicit algorithm. In contrast to fourth generation programming language, third generation programming language is usually independent of a data base management system and is transportable between different computer architectures.

8. PROCEDURES AND GUIDELINES. Procedures and guidelines for the Agency's software management program will be issued under separate cover.