LA-13138

MCNP™ Software Quality Assurance Plan

Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

Edited by Patricia W. Mendius, Group CIC-1
Prepared by M. Ann Nagy, Group XTM

An Affirmative Action/Equal Opportunity Employer

LA-13138

UC-705
Issued: April 1996

MCNP™ Software Quality Assurance Plan

Hilary M. Abhold
John S. Hendricks

Los Alamos
NATIONAL LABORATORY
Los Alamos, New Mexico 87545

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

MCNP™ SOFTWARE QUALITY ASSURANCE PLAN

by

Hilary M. Abhold and John S. Hendricks

ABSTRACT

MCNP is a computer code that models the interaction of radiation with matter. MCNP is developed and maintained by the Transport Methods Group (XTM) of the Los Alamos National Laboratory (LANL). This plan describes the Software Quality Assurance (SQA) program applied to the code. The SQA program is consistent with the requirements of IEEE-730.1 and the guiding principles of ISO 9000.

I. PURPOSE

This document is the Software Quality Assurance Plan (SQAP) for modification and maintenance of the Monte Carlo N-Particle (MCNP) radiation transport code. MCNP is developed and maintained by the Transport Methods Group (XTM) of the Los Alamos National Laboratory (LANL). The code models the interaction of radiation with matter.

The particular software covered is the MCNP Fortran and C source code released to the Radiation Shielding Information Center (RSIC) in Oak Ridge, Tennessee. Specifically excluded are non-XTM modified versions or auxiliary codes distributed with MCNP by XTM. Also specifically excluded is the quality assurance of data libraries used by the code.

MCNP is a mature product that is continually being upgraded. This SQAP applies to MCNP upgrades from one major version to another. A major version of MCNP is a version which is released every few years to RSIC for international distribution.

This quality assurance plan covers the entire MCNP software modification life cycle: concept exploration, requirements and design, implementation, test, installation and check-out, operation and maintenance, and retirement. It is consistent with IEEE 730.1[1] and with the guiding principles of ISO 9000.[2]

A. Plan Reviews

The MCNP SQAP is reviewed and approved by the MCNP Board of Directors (BoD) and the Group Office concurrently. The plan and implemented procedures are subject to group office audit at any time.

II. MANAGEMENT

A. Organization

Software quality assurance for MCNP is the responsibility of the Transport Methods Group (XTM) of Los Alamos National Laboratory. XTM is divided into teams, each with a Team Leader who reports to the Group Office. The Monte Carlo team does code development for MCNP. The Group Office is comprised of the Group Leader and Deputy Group Leader. A Board of Directors (BoD) is composed of team members of the Transport Methods Group from the

applications team, data team, and other teams as appropriate. Members of the BoD are assigned by the Group Office.

The Monte Carlo Team is responsible for the maintenance and development of MCNP. The BoD reviews all proposed features of MCNP, reviews their implementation, and has final authority for acceptance of new features. Consensus between the BoD, Group Office, and Monte Carlo Team is required for all changes beyond regular maintenance.

Changes to this Software Quality Assurance Plan are made with the concurrence of the Monte Carlo Team Leader, the MCNP BoD, and the Group Office.

B. Tasks

The MCNP software quality assurance plan applies to all stages of the MCNP software life cycle: concept exploration, requirements and design, implementation, test, installation and checkout, operation and maintenance, and retirement. The following paragraphs describe the iterative tasks involved in the MCNP software lifecycle and identify which tasks help assure software quality.

1. Concept Exploration. A features list is compiled of desired features for future major versions of MCNP. This features list is kept by the Monte Carlo Team. It is documented archivally on the Common File System (CFS). This list includes all features proposed or accepted for the previous major version that did not become part of the previous major version. It also includes proposed new features. At the features review, the MCNP BoD approves or disapproves features for inclusion into MCNP. The features list indicates whether the feature was accepted or rejected. See Fig. 1 for a sample features list.

2. Requirements & Design. Many proposed modifications to MCNP come with the software already developed in the form of patches. These patches range in quality. Some patches are accompanied by good documentation and no further software requirements specifications or software design are necessary.

Date

Desired MCNP 4B Features

(Number Key: H=High Priority, M=Medium, L=Low)
(State Key: N=New, A[U,D]=Accepted [Underway,Done] R=Rejected)

Number	Description	State
Physics		
P-H1	Incorporate data and/or geometric perturbation sensitivity capability. (Schrandt 1/25/82). (Schlumberger request, 10/11/93).	AU(GWM) 10/18/93
P-H2	Incorporate INRAD capabilities (X-6:GPE-91-169) LA-11153-MS (X-6:JSH-92-73) (X-6:JSH-92-257)	AU(JSH)
Variance		
V-H1	Tom Booth's Quasi-deterministic weight window generator	A(JSH)
Internal		
I-H1	Smart PVM workstation cluster capability, particularly load balancing. (Schlumberger, 10/11/93)	A 03/29/94

Fig. 1. Sample Features List

The MCNP BoD decides by consensus which features and/or patches to accept into the next major version of MCNP. The acceptance or rejection of a feature is documented in the features list and the BoD meeting minutes. (See Fig. 2 for an example of the meeting minutes.) Accepted features are prioritized by the BoD and recommendations are made on the software requirements and design (particularly the user interface). For new features not accompanied by a patch, or for substandard patches, new software requirements specifications and software design descriptions are Implementation created by the Monte Carlo Team. These created patches along with their documentation are reviewed and approved as intermediate version patches by the BoD.

3. Implementation. Patches that have been approved for implementation into MCNP are integrated with other patches to form an intermediate version. (See Fig. 3 for an example of a patch.) All lines of code and documentation are independently reviewed by the MCNP code developer who integrates the patch. The developer who integrates the patch may not be the code developer who developed the patch.

4. Test. The Monte Carlo Team tests all new features in accordance with the MCNP Software Verification and Validation Plan in Appendix A.

5. Installation and Checkout. The installation and checkout procedure is described in Appendix C of the MCNP manual.

6. Operation and Maintenance. All new features are independently documented in LANL memos at the intermediate version release by the Monte Carlo Team. (See Fig. 4 for an example of an intermediate version patch release memo.) A bug list is compiled of all claimed bugs or deficiencies. (See Fig. 5 for an example bug list.) Bugs are investigated and corrections are identified. All bug fixes are commented on with references as appropriate in the major patch file converting the last major version of MCNP to the next. A list of bugs is periodically published and circulated (for example, Ref. 4), as well as by the use of an e-mail network for notifying users of new features and bugs found.

7. Retirement. When a new major version of MCNP is released by RSIC, support for the previous version is terminated/discontinued and support for the version before that is terminated. Support for all intermediate versions is dropped and intermediate versions are destroyed. See Fig. 6 for a sample of a letter releasing a major version to RSIC.

Los Alamos
NATIONAL LABORATORY

memorandum

Applied Theoretical & Computational Physics Division
XTM, Transport Methods Group

To/MS: Distribution
From/MS: Monte Carlo team Leader, XTM
Phone/FAX: 5-6789/5-5538
Symbol: XTM-yy-xxx
Date: Month, Day, Year

MCNP™ BoD Meeting Minutes, Date

Date MCNP Board of Directors Minutes

Charter: The MCNP Board of Directors (BoD) is a body of XTM staff with authority over the direction, developments, and software quality assurance (SQA) of MCNP. The BoD reviews and recommends new code features and code modifications, particularly user interface and other issues affecting customers and users, and also considers issues such as data and policy.

Attending the Date meeting: put attending members here.

Resolved Issues

Put appropriate issues here, such as accepted or rejected features, et cetera.

Action Items

Put appropriate action items here.

Agenda for the next meeting

Put appropriate information here.

MCNP is a trademark of the Regents of the University of California, Los Alamos National Laboratory.

Fig. 2. Sample BoD Meeting Minutes

```
c        <<<<< MCNP4XQ makemcnp changes for MCNP4A >>>>> loddat 03/01/96
c********************* FIXES FOR SYSTEM 1 (Cray UNICOS) *****************
c
c     Keep NEWID file for UNICOS
      1 2 23 2 1
      rm -f codef patch

c********************* FIXES FOR SYSTEM 7 (PC   DOS   ) *****************
c
c     Provide changes for X-windows on a PC. Software requirements:
c     MetaWare High C 3.3, DESQview/X 2.0 with MetaWare X11 toolkit.
c     These changes allow for continued support of Lahey graphics.
7 2 25 1 9
if exist mcnpc.c del mcnpc.c
copy mcnpc.id codef
copy patchc patch
prpr
rename compile mcnpc.c
del codef
del patch
del newid
hc386 -f387 -DMSDOS -Hoff=protection -I\dvx\include -c mcnpc.c
7 2 33 1 4
type patchf | find "*define pcdos" | find "xlib"
if errorlevel 1 goto lahey
goto xwin
:lahey
7 2 34 1 5
goto end
:xwin
set lib=\f7713\lib;\hc33\small;\dvx\lib\hc387
386link -nomap -pack mcnp \f7713\lib\hc320 mcnpc -l
hc386,hc387,hcna,x11,sys
:end

0 3 2 4 4
0 3 5 2
*/        <<<<< MCNP4XQ C patch to MCNP4A >>>>>          loddat 03/01/96
*/
*/ ------------------------------------------------------------ mcnpc.id
```

Fig. 3. Sample Patch

```
*ident mc4xq
*/ Remove the DIFFTIME() routine.                          07/03/95 (GWM/GWM)
*d,mc.12,mc.13                                                        <25-26>
*if -def,unix,1
#include <unistd.h>
*/ Make changes for PC MetaWare High C compiler.      02/15/95  (GWM/GWM)
*i,mc.18                                                              <31>
*if def,pcdos,5
/* Define a structure for use with Lahey FORTRAN and MetaWare C.       */
typedef struct
 { char  *text;
   long  leng;
 } Lahey_Char;
*i,mc.50                                                              <63>
XSetWindowAttributes attributes;
*d,mc.73          Replace screen variable for PC X-windows            <86>
      pnt_x, pnt_y, res_x, res_y, scrnnm, width, win_flag, win_shape,
*/
*/ ------------------------------------------------------------- xgopwk
/* Define a structure for use with Lahey FORTRAN and MetaWare C.       */
*d,mc.137                                                            <150>
for(i=0; *font_bas[i] != '\0'; i++)
*d,mc.145,mc.149  Replace screen variable for PC X-windows      <158-162>
scrnnm=DefaultScreen(display);
display_width=DisplayWidth(display,scrnnm);
display_width_mm=DisplayWidthMM(display,scrnnm);
display_height=DisplayHeight(display,scrnnm);
display_height_mm=DisplayHeightMM(display,scrnnm);

c         <<<<< MCNP4XQ Fortran patch to MCNP4A >>>>>    loddat 03/01/96

c    Delete lines 2-10 of PATCHF.
 0 1 2 4 10
c    Add this entire patch to PATCHF.
 0 1 11 2
*/
*/ ------------------------------------------------------------ comdeck zc
```

Fig. 3. Sample Patch (cont.)

```
*ident zc4xq
*d,zc4a.1                                                          <21>
      parameter (kod='mcnp',ver='4xq')
*/ Set mdas in install.fix file to facilitate installation.      03/30/95
*d,zc4a.5                                                          <30>
      parameter (mdas=4000000)
*/
*/ ------------------------------------------------------- block data
*ident bd4xq
*/ Variably dimension ebl, febl, to prevent multigroup crash.
*/ (X-6:ELR-94-362, X-6:JSH-94-468).                  7/25/94 (ELR/JSH)
*d,bd.8  data ebl                                                  <490>
*d,bd4a.13                                                         <575>
     3 hsd/'sequential','direct'/,ibin/'fdusmcet'/,loddat/'03/01/96'/,
*d,cor4-2.24
     3 hdpath/'/usr/local/udata/mcnp'/,
*/
*/ ------------------------------------------------------------- tpefil
*ident tf4xq
*/ Do not flush output, tty buffers: compiler problems.     (JSH) 8/3/94
*d,tf4a.21  go to 162                                              <1299>
      return
*/ Do not flush output, tty buffers: compiler problems.     (JSH) 8/3/94
*d,tf4a.37,tf4a.47  before return                          <1362-1372>
*d,tf.112  call balk                                               <1383>
```

Fig. 3. Sample Patch (cont.)

Los Alamos
NATIONAL LABORATORY
memorandum
Applied Theoretical & Computational Physics Division
XTM, Transport Methods Group

To/MS: Distribution
From/MS: Monte Carlo Team Leader, XTM
Phone/FAX: 5-6789/5-5538
Symbol: XTM-yy-xxx
Date: Month, Day, Year

MCNP4XN

Another intermediate version of MCNP™ is now available: MCNP4XN.

New Features and Corrections in MCNP4XN

The major differences from this MCNP version and the last one, MCNP4XL are:

include major differences here

Minor improvements include:

include minor improvements here.

Availability

Executables for all systems are available on: put info here

Code Review

All MCNP4XN code has been multiply reviewed by members of the Monte Carlo Team.The Patch and Patch Documentation has been reviewed and this memo certifies that:

1) the MCNP4XN patch meets the software requirements specifications documented in memo XTM-yy-xxxx;
2) the patch is properly designed and implemented;

and

3) the patch is properly documented in this memo.

Testing

Put Testing info here.

MCNP is a trademark of the Regents of the University of California, Los Alamos National Laboratory.

Fig. 4. Sample Intermediate Version Release Memo

Date

Possible MCNP 4A Bugs

(H=High Priority, M=Medium, L=Low)

Number	Description	State
H1	Question whether PWT card is used i WWN!:P is used (PWT values in output may be inconsistent.)	N
H2	DXTRAN with flagged-cell tallies - only flagged if parent particle departs the cell	N
H3	•••••	
H4	•••••	
H5	•••••	
H6	Add PCDOS and LAHEY keyword print in print table 98.	D (JSH) 12/19/93
M1	Make F8:N,P,E a warning, not fatal. GWM argues 1/19/94 that F8 and mode N should be fatal.	N

Fig. 5. Sample Bug List

Los Alamos
NATIONAL LABORATORY

Applied Theoretical & Computational Physics Division
XTM, Transport Methods Group
Los Alamos, New Mexico 87545
(505) 667-4189
FAX (505) 665-5538

DATE: Month, Day, Year
IN REPLY REFER TO: XTM-yy-xxx
MAIL STOP: B226
TELEPHONE: (505) 665-6789

Radiation Shielding Information Center
Building 6025, MS6362
Oak Ridge National Laboratory
P.O. Box 2008
Oak Ridge, TN 37831-6362

To Whom It May Concern:

put info about release here.

Software Verification and Validation Report

This letter certifies that MCNP™ version XX meets the requirements of the MCNP Software Verification and Validation Plan.

Very Truly Yours,

John. Q. Team Leader
Monte Carlo Team Leader
Los Alamos National Laboratory

MCNP is a trademark of the Regents of the University of California, Los Alamos National Laboratory

Fig. 6. Sample Letter Releasing a Major Version to RSIC

C. Responsibilities

The Monte Carlo Team is responsible for maintenance and development of MCNP in general, and specifically for:

a) maintaining the features list,
b) developing software requirements specifications and software design for new features and/or substandard patches,
c) integrating patches into MCNP,
d) reviewing the implementation of patches into MCNP;
e) testing new features,
f) documenting new features,
g) maintaining the bug list,
h) investigating bug claims and providing corrections to bugs along with documentation of those bugs,
i) approving, concurrent with the BoD and the Group Office, all changes to MCNP beyond regular maintenance,
j) following SQA procedures defined in this plan.

The Board of Directors has final authority for acceptance of new features into MCNP, and is specifically responsible for:

a) approving additions and deletions of features to MCNP,
b) prioritizing approved items on the features list,
c) making recommendations to the Monte Carlo team on requirements and design for new features and/or substandard patches,
d) approving patches for incorporation into MCNP along with accompanying documentation,
e) approving, concurrent with the Monte Carlo Team and the Group Office, all changes to MCNP beyond regular maintenance,
f) following SQA procedures defined in this plan.

The Group Office is responsible for software quality assurance for MCNP in general, and specifically for:

a) assigning group members to the BoD,
b) approving, concurrent with the Monte Carlo Team and the BoD, all changes to MCNP beyond regular maintenance.

The User Community may:

a) make suggestions for features to be added to the features list,
b) report bugs found to the Monte Carlo Team.

III. DOCUMENTATION

The following documents control the development of MCNP:

a) The MCNP features and bug lists.
b) Chapter 2 of the MCNP Manual.[3] This section of the MCNP manual describes the functions, performance criteria, design constraints, and attributes of MCNP. It also describes the components and subcomponents of the MCNP software design, including data bases and internal interfaces. The external user interface is described in Chapter 3. The MCNP manual is updated at each major version.
c) Patch documentation. All features added since the last publication of the MCNP manual are documented in LANL memoranda and are referenced in the major patch converting the previous version of MCNP to the current version. See Fig. 7 for an example of a LANL memo documenting a patch.
d) The MCNP Software Verification and Validation Plan (SVVP) (see Appendix A).
e) The MCNP Software Verification and Validation Report. This Report is documented in Fig. 6, a sample letter releasing a new major version of the code to RSIC.

f) The MCNP Manual. User documentation is contained in the MCNP manual.

g) The MCNP Software Configuration Management Plan (see Appendix B).

See Appendix C for a cross reference between the MCNP SQA controlling documents and the minimum documentation required by IEEE 730.1.

Los Alamos
NATIONAL LABORATORY
memorandum
Applied Theoretical & Computational Physics Division
XTM, Transport Methods Group

To/MS: Monte Carlo Team Leader, XTM
From/MS: MCNP™ Code Developer, XTM
Phone/FAX: 5/6789/5-5538
Symbol: XTM-yy-xxx
Date: Month, Day, Year

MCNP4XN DOCUMENTATION

This memo describes the required documentation accompanying the MCNP4b patch MNCP4XN.

Software Requirements Specifications

Put the software requirements specification documentation here.

Software Design Description

Put the software design description here.

User Interface

Describe the User Interface Here.

Testing and Validation

Describe the testing and validation of the Patch here.

MCNP is a trademark of the Regents of the University of California, Los Alamos National Laboratory.

Fig. 7. Sample Memo Describing Patch Documentation

IV. STANDARDS, PRACTICES, CONVENTIONS, AND METRICS

This section identifies the standards, practices, conventions, and metrics to be applied to the development of MCNP, and identifies how compliance with these items is to be monitored and assured.

A. Documentation Standards

Patch file commentary must reference the appropriate LANL memoranda and/or other reference, and include the date and initials of the developer. All patch files are reviewed by the Monte Carlo Team and the BoD after implementation into the code. This review is documented by LANL memo.

B. Logic Structure Standards

The ANSI C and Fortran 77 standards are used. All code is compiled with ANSI C and Fortran 77 standard compilers, thus assuring the language standard.

C. Coding Standards

MCNP is programmed in the style of Dr. Thomas N. K. Godfrey, the principal MCNP programmer from 1975-1989. The Fortran used strictly adheres to the ANSI Fortran 77 standard. Variable dimensions for arrays are achieved by massive use of EQUIVALENCE statements and offset indexing. All variables local to a routine are no more than two characters in length, and all COMMON variables are between three and six characters in length. The principal characteristic of Tom Godfrey's style is its terseness. All patches accepted for inclusion in MCNP are put in Godfrey style during the integration and implementation step by MCNP code developers.

D. Commentary Standards

The MCNP patch file has a comment for every change in the code from one major version to another. For absolutely trivial and obvious changes, the comment in the patch file is sufficient. Variable and array names are documented in the MCNP manual.

E. Testing Standards and Practices

MCNP testing is done under the auspices of the SVVP; see Appendix A.

F. Selected Software Quality Assurance Product and Process Metrics

Not applicable.

V. REVIEWS AND AUDITS

A. Purpose

This section defines the technical and managerial reviews and audits to be conducted, how they are to be accomplished, and what further actions are required to see that they are implemented and verified.

B. Feature Reviews

All features and other changes to MCNP must be placed on the MCNP features or bug list, except for trivial code maintenance items (such as deleting references to CTSS and other obsolete systems, removing blank lines, etc.). Approval from the MCNP BoD is required before features can be accepted for integration into MCNP. All memoranda describing the proposed changes and changes to proposed features are circulated through the XTM Group Office for optional Group Office review. All BoD meeting notes and dated features lists are stored archivally.

This review is more of a management than a technical review, and relies on the expertise and experience of the XTM in determining whether a feature is desirable and implementable with the current resources and budget. The outcome of this review is documented as acceptance or rejection of the feature on the features list.

C. Intermediate Version Patch and Patch Documentation Review

All patches to MCNP are reviewed by the Monte Carlo Team and the Board of Directors before and after implementation into intermediate and major versions of the code. The Monte Carlo Team Leader of the Transport Methods Group certifies that intermediate and major version patches to MCNP meet all software requirements specifications and are properly designed and implemented in MCNP. The Team Leader also reviews the documentation of intermediate and major version patches. This patch documentation must include the software requirements specifications, describe the software design, specify the user interface, and describe the testing and validation of the patch.

This review is both technical and managerial, and relies on the expertise of the Monte Carlo Team. It is documented by memoranda; see Fig. 4 for a sample of this review documented on the patch release memo.

D. Coding Reviews and Audits

Coding changes are multiply reviewed. Every line of code is reviewed by both the patch originator and by a separate code developer integrating the patch into an intermediate MCNP version. This review takes place at the same time as patch integration and serves as a functional audit.

When a new intermediate version of MCNP is ready for customer release, every change from the previous version is again reviewed by either the principal MCNP programmer or the Monte Carlo Team leader. This review applies to the intermediate version documentation memo as well and is a physical audit.

The Monte Carlo Team Leader is responsible for ensuring that every line of coding in MCNP is multiply reviewed. The Monte Carlo Team Leader certifies on the memo releasing the intermediate version for customers that these versions have passed these audits. See Fig. 4 for an example of this memo.

E. Documentation Reviews

The MCNP manual is reviewed by all members of the Monte Carlo Team and additional qualified LANL staff who are on the MCNP BoD. The Monte Carlo Team Leader certifies on the memo releasing the MCNP manual that the new version has passed this review. See Fig. 8 for an example of this memo.

F. Cross Reference

See Appendix D for a cross reference between the MCNP SQA reviews and the minimum reviews and audits required by IEEE 730.1.

Los Alamos
NATIONAL LABORATORY

memorandum

Applied Theoretical & Computational Physics Division
XTM, Transport Methods Group

To/MS: Distribution
From/MS: Monte Carlo Team Leader, XTM
Phone/FAX: 5-6789/5-5538
Symbol: XTM-yy-xxx
Date: Month, Date, Year

MCNP4B MANUAL UPDATE

This memo announces the release of the MCNP™ manual version 4B. The manual has been properly reviewed and approved by the Monte Carlo Team and the MCNP Board of Directors. New features described in the manual include:

Put new feature info here

Fig. 8. Sample MCNP Manual Release Memo

VI. TEST

All SQA tests for MCNP are described in the SVVP (see Appendix A).

In addition, a number of benchmark calculations have been performed on MCNP to ensure agreement with physical experiments. The benchmarking program is an ongoing and collaborative effort, and a number of benchmark documents are available from LANL, General Electric, and Westinghouse, who have all done MCNP benchmarking. These benchmarks are:

1) MCNP: Photon Benchmark Problems, Daniel J. Whalen, David E. Hollowell, John S. Hendricks, LA-12196, Los Alamos National Laboratory, Sept. 1991.
2) MCNP: Neutron Benchmark Problems, Daniel J. Whalen, et.al, LA-12212, Los Alamos National Laboratory, Nov. 1991.
3) MCNP: Criticality Safety Benchmark Problems, John C. Wagner, James E. Sisolak, Gregg W. McKinney, LA-12415, Los Alamos National Laboratory, Oct. 1992.
4) Benchmark Analysis of MCNP ENDF/B-VI Iron, John D. Court, Ronald C. Brockhoff, John S. Hendricks, LA-12884, Los Alamos National Laboratory, Dec, 1994.
5) Comparison of First-Principles of MCNP Calculations of NAI and BGO Detector Response Functions to Measurements, Guy P. Estes, John T. Kriese, Robert G. Schrandt, LA-12391, Los Alamos National Laboratory, Sept. 1992.
6) Lawrence Livermore Pulsed Sphere Benchmark Analysis of MCNP ENDF/B-VI, John D. Court, Ronald C. Brockhoff, John S. Hendricks, LA-12885, Los Alamos National Laboratory, Dec. 1994.
7) MCNP ENDF/B-VI Validation: Infinite Media Comparisons of ENDF/B-VI and ENDF/B-V, John D. Court, John S. Hendricks, Stephanie C. Frankle, LA-12887, Los Alamos National Laboratory, Dec. 1994.

The above benchmarks are part of a coordinated MCNP benchmarking effort. In addition, many laboratories throughout the world benchmark MCNP for their own purposes, and the scientific literature includes 10-30 MCNP benchmarks per year by other organizations. These are not part of the MCNP SQAP, but they do provide confidence in the code.

VII. PROBLEM REPORTING AND CORRECTIVE ACTION

All suspected problems with MCNP should be reported to the Monte Carlo Team of the LANL Transport Methods Group. The e-mail address is mcnp@lanl.gov. All reported problems are examined; those that are not user error and are nontrivial are logged into the MCNP bug list and investigated. Trivial bugs (incorrect spelling in the code, etc.) may be added to the patch immediately without further documentation. Cash awards are given as an incentive to get users to report problems. Bug reports are sent to an e-mail distribution list periodically.

Corrective action is logged into the MCNP bug list and implemented in the MCNP patch for the next version of the code. For all but the most obvious corrections, a memo is also prepared describing the problem fix, consequences, and any possible work-arounds; this memo is referenced in the MCNP bug list.

Problems identified in the software development and maintenance process are the responsibility of the Monte Carlo Team Leader, who is also ultimately responsible for software problem reporting and corrective action.

VIII. TOOLS, TECHNIQUES, AND METHODOLOGIES

MCNP patches are written in the 'update' format that can be utilized by commercial software products such as Opcode* Historian or a short preprocessing utility called prpr that is provided with the code.

IX. CODE CONTROL

Major versions of MCNP released to RSIC are controlled by RSIC. Intermediate versions and documentation are controlled by the Monte Carlo Team. They are stored on the Los Alamos Common File System, with password protection where appropriate, and on a restricted access workstation network.

X. MEDIA CONTROL

Major versions of MCNP are delivered to RSIC where they are stored upon RSIC media according to RSIC procedures.

*Opcode, Inc., Austin, Texas

XI. SUPPLIER CONTROL

MCNP is self-contained and does not depend upon supplier software beyond a standard ANSI Fortran 77 compiler and a standard C compiler, graphics libraries, standard Fortran and C math libraries, and PVM software for distributed processor multitasking.

XII. RECORDS COLLECTION, MAINTENANCE, AND RETENTION

All SQA documentation is maintained and retained by the LANL Transport Methods Group.

XIII. TRAINING

No training activities are necessary to meet the needs of the SQAP.

XIV. RISK MANAGEMENT:

Since MCNP is not a commercial code and is not subject to specific cost and schedule constraints, this section is not applicable. Technical quality is, of course, dependent on the expertise of the XTM group, and therefore no special techniques are used to manage the risk of technical inadequacy.

REFERENCES

1. "IEEE Software Engineering Standards Collection," Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017-2394 (April 5, 1991).

2. ISO (International Organization for Standards), "ISO 9000—International Standards for Quality Management," 2nd Edition, ISO Central Secretariat, Case postale 56, CH-1211 Geneve 20, Switzerland (1992).

3. J. F. Briesmeister, Ed., "MCNP—A General Monte Carlo N-Particle Transport Code, Version 4A," Los Alamos National Laboratory report, LA-12625-M (November 1993).

4. Ronald C. Brockhoff, John S. Hendricks, "A New MCNP Test Set," Los Alamos National Laboratory Report LA-12839 (September 1994).

APPENDIX A

MCNP Software Verification and Validation Plan.

I. PURPOSE

The Purpose of the MCNP Software Verification and Validation Plan is to describe the methods used to verify that a) software requirements have been approved by an appropriate authority, b) the requirements have been implemented in the design, and c) the design is implemented correctly by the code. It also describes the methods used to validate that when the code is executed, it complies with the software requirements.

II. VERIFICATION

A. Approval of Requirements

All intermediate and major version patches to MCNP must be accompanied by documentation. (See Fig. 7 for an example.) This documentation must describe the software requirements specifications. The patch, along with its documentation, must be approved by consensus of the Group Office, MCNP BoD, and the Monte Carlo Team. The Monte Carlo Team Leader of the Transport Methods Group shall certify that patches to MCNP meet all software requirements specifications.

B. Approval of Design

All intermediate and major version patches to MCNP must be accompanied by documentation. (See Fig. 7 for an example.) This documentation must describe the software design and show how the design implements the requirements specification. The patch along with its documentation must be approved by consensus of the Group Office, MCNP Board of Directors, and the Monte Carlo Team. The Monte Carlo Team Leader of the Transport Methods Group shall certify that patches to MCNP meet all software requirements specifications and are properly designed.

C. Approval of Implementation

All intermediate and major version patches to MCNP are reviewed by the Monte Carlo Team and the Board of Directors both before and after implementation into intermediate and major versions of the code. Coding changes must be multiply reviewed. Every line of code is reviewed by both the patch originator or a Monte Carlo Team member and by a separate Monte Carlo Team code developer assigned to integrate the patch into an intermediate MCNP version. The Monte Carlo Team Leader of the Transport Methods Group shall certify that patches to MCNP meet all software requirements specifications and are properly designed and implemented in MCNP.

III. VALIDATION

All new features incorporated into MCNP are validated by test. This test set is described in Ref. 4. These features, after being integrated into an intermediate version of MCNP, must:

a) pass the MCNP test set for the previous patch version to ensure that they do not affect unrelated parts of the code, and
b) pass the MCNP test set modified to include the new feature.

To 'pass' the test set means that by running the test problems, it is shown that MCNP incorporates the requirements that the test set is designed to measure, reproducing the results exactly when the same random number seed is used.

Each bug correction shall result in a modification of the MCNP test set so that had the test problems been so modified, they would have caught the bug that was corrected. Thus by design, each bug fix causes all old versions of MCNP to fail the new test set. All changes to the test set will be documented in the test set documentation.

Intermediate versions of MCNP shall be provided to the approximately 200 MCNP users at Los Alamos and some to customers outside Los Alamos, so that MCNP is run on a wide variety of problems many thousands of times before release to RSIC.

APPENDIX B

MCNP Software Configuration Management Plan

I. PURPOSE

The purpose of the MCNP Software Configuration Management Plan is to describe the methods of configuration identification, control, and status accounting that apply to the MCNP software development process.

II. SCOPE

The MCNP software configuration management plan applies only to the MCNP Fortran and C source codes. Auxiliary codes, update memos, data libraries, system libraries, and documentation, etc., are not covered.

III. CONFIGURATION IDENTIFICATION

A major version of MCNP is released through the Radiation Shielding Information Center in Oak Ridge, Tennessee, approximately every 2-3 years. Major versions of MCNP are uniquely identified as "MCNPX" with X incrementing numerically and/or alphabetically. Major versions of MCNP have been approved for release to RSIC by consensus between the Monte Carlo Team, the Board of Directors, and the Group Office. The major versions of MCNP and their release dates have been:

RELEASE DATE	VERSION NAME
8/77	MCNP1A
5/78	MCNP1B
9/79	MCNP2
12/80	MCNP2B
5/82	MCNP2C
2/83	MCNP2D
9/83	MCNP3
9/85	MCNP3A
8/88	MCNP3B
3/91	MCNP4
10/93	MCNP4A.

Intermediate versions of MCNP are those revisions of a major version that include new features but have not been released to RSIC. Intermediate versions of MCNP are uniquely identified with a unique version name like "MCNPXK," with MCNPX the major version, and K incrementing alphabetically. Intermediate versions of MCNP also have a unique load date. Intermediate versions of MCNP have been approved for release to customers by the leader of the Monte Carlo Team.

Developmental versions of MCNP are "work in-progress" versions of the code. They are not to be distributed outside the Monte Carlo Team. They are identified by the MCNP intermediate version number that they modify, but with a different load date. The load date must be printed every time the code is executed.

One of the principal attractions of MCNP is that it can be easily modified by users for their unique needs. Appendix C of the MCNP manual describes how to modify MCNP. These user-modified versions of MCNP are not covered by the MCNP SQAP, nor are they subject to MCNP configuration management.

IV. CONFIGURATION CHANGE CONTROL

All changes to intermediate and major versions of the code are made by implementing approved patches. When the Monte Carlo Team decides that an intermediate version is ready to become a major version of MCNP, approval is received from both the MCNP BoD and the Group Office concurrently. Then the intermediate version is "frozen" and the addition of new features is prohibited. After sustained use by friendly users, the Monte Carlo Team will determine that the code has demonstrated sufficient robustness for international release. Then, with the approval of the MCNP BoD and the Group Office, the last intermediate version is renamed as the new major version, frozen, and sent to RSIC.

A major version of MCNP is archival and is not to be modified. If significant errors are discovered in a major version, either

1) a corrected version will be sent to RSIC for international distribution or
2) a correction patch or install.fix file (see Appendix C of the MCNP manual) will be sent to RSIC and made available electronically so that MCNP may be reinstalled with the correction or
3) a work-around will be sent to customer users and documented for later distribution to all users. (A work-around is a message to users on how to avoid the problem.)

If a corrected version or correction patch is distributed, the MCNP version will be renamed.

All intermediate and major version patches are documented. The documentation must describe the software requirements specifications, describe the software design, specify the user interface, and describe the testing and validation of the patch. This documentation should be appropriate for inclusion in chapters 2 (physics) and 3 (user interface) of the MCNP manual. In the patch, the revision must include a commentary describing the patch, giving the date it was implemented, and referencing the patch documentation.

At each new major version of MCNP, the MCNP manual must be revised accordingly.

V. CONFIGURATION STATUS ACCOUNTING

The configuration status of MCNP is documented in the features list. The features list records whether the feature has been approved, disapproved, or is a new feature and, for approved features, whether work is underway or is finished. If the work is finished, the date and version number are listed. The major versions of MCNP are documented in the MCNP manual.

VI. RESPONSIBILITIES

The Monte Carlo Team is responsible for software configuration management for MCNP.

APPENDIX C

Cross Reference Between MCNP SQA Controlling Documents and the Minimum Documentation Required by IEEE 730.1

I. SOFTWARE REQUIREMENTS SPECIFICATION

The software requirements specification for the current major RSIC-released version of MCNP is Chapter 2 of the MCNP manual. Intermediate and major version patch documentation describes the software requirements specification for each individual patch.

II. SOFTWARE DESIGN DESCRIPTION

The software design description (SDD) for the current major RSIC-released version of MCNP is in Chapter 2 of the MCNP manual. Intermediate and major version patch documentation describe the design of each individual patch.

III. SOFTWARE VERIFICATION AND VALIDATION PLAN (SVVP)

See Appendix A.

IV. SOFTWARE VERIFICATION AND VALIDATION REPORT (SVVR)

The Monte Carlo Team Leader shall certify by LANL memorandum that each major release version of MCNP to RSIC meets the SVVP. Figure 6 (sample letter releasing a new version to RSIC) contains this certification.

V. USER DOCUMENTATION

The MCNP manual specifies and describes the required MCNP data and control inputs, input sequences, options, program limitations, and other activities and items necessary for successful execution of MCNP.

VI. SOFTWARE CONFIGURATION MANAGEMENT PLAN (SCMP)

See Appendix B.

APPENDIX D

Cross Reference Between the MCNP SQA Review and Audits and the Minimum Reviews and Audits Required by IEEE 730.1

I. PURPOSE

This section describes the minimum set of reviews and audits required by IEEE 730.1 and explains how the MCNP SQA reviews and audits fulfill these requirements.

II. SOFTWARE REQUIREMENTS REVIEW

A Software Requirements Review ensures the adequacy of the requirements stated in the software requirements specification. The MCNP review that accomplishes this function is the Patch and Patch Documentation Review.

III. PRELIMINARY DESIGN REVIEW

The Preliminary Design Review evaluates the technical adequacy of the preliminary design of the software as depicted in the preliminary software design description. The MCNP review that accomplishes this function is the Patch and Patch Documentation Review.

IV. CRITICAL DESIGN REVIEW

The Critical Design Review determines the acceptability of the detailed software designs, as depicted in the detailed software design description, in satisfying the requirements of the software requirements specification. The MCNP review that accomplishes this function is the Patch and Patch Documentation Review.

V. SOFTWARE VERIFICATION AND VALIDATION PLAN REVIEW

The Software Verification and Validation Plan Review evaluates the adequacy and completeness of the verification and validation methods defined in the SVVP. Since the MCNP SVVP is Appendix A of the MCNP SQAP, it is reviewed at the same time as the SQAP.

VI. D6 FUNCTIONAL AUDIT

A functional audit is held prior to software delivery to verify that all requirements specified in the software requirements specification have been met. The MCNP review that accomplishes this function is the Coding Review and Audit.

VII. D7 PHYSICAL AUDIT

A Physical Audit is held to verify that the software and its documentation are internally consistent and are ready for delivery. The MCNP Review that accomplish this function is the Coding Review and Audit.

VIII. IN-PROCESS AUDITS

An in-process audit is held to verify consistence of design. The MCNP review that accomplishes this function is the Features review.

IX. MANAGERIAL REVIEWS

Managerial reviews assess the execution of all of the actions and the items identified in the SQAP. The MCNP review that accomplishes this is a Group-Office-directed audit of the software quality assurance program for MCNP.

X. SOFTWARE CONFIGURATION MANAGEMENT PLAN REVIEW

This review evaluates the adequacy and completeness of the configuration management methods defined in the SCMP. Since the SCMP is part of the SQAP, it is reviewed along with the SQAP.

XI. POST MORTEM REVIEW

A Post Mortem Review is held at the conclusion of the project to assess the activities implemented on the project and to provide recommendations of appropriate actions. This review is not appropriate to the MCNP software development process, since it is an iterative, on-going process and is not concluded.

APPENDIX E

Cross Reference Between the MCNP SQAP, IEEE 730.1, and ISO-9000-3

TABLE I: DOCUMENT AND STANDARD CROSS REFERENCE

MCNP SQAP		IEEE 730.1	ISO 9000-3
1.	Purpose	3.1	5.5.1
1.1	Plan Reviews	3.6.2.4,3.6.2.8,3.6.2.9	4.1.1.3
2.	Reference Documents	3.2	not applicable
3.	Management	3.3	4.1.1.2.1,4.1.1.2.3,5.4.2.2
3.1	Organization	3.1	not applicable
3.2	Tasks	3.3.2,3.4.2	5.1,5.4.1,5.4.2.1
3.2.1	Concept Exploration	3.3.2	not applicable
3.2.2	Requirements & Design	3.3.2	5.3.1,5.6.2
3.2.3	Implementation	3.3.2	not applicable
3.2.4	Test	3.3.2	not applicable
3.2.5	Installation & Checkout	3.3.2	not applicable
3.2.6	Operation & Maintenance	3.3.2	5.10.5,5.10.6
3.2.7	Retirement	3.3.2	not applicable
3.3	Responsibilities	3.3.3	not applicable
4.	Documentation	3.4,3.4.1,3.4.2.1,3.4.2.2,3.4.2.5	5.4.4,5.4.5
5.	Standard, Practices, Conventions, and Metrics	3.5,3.5.1	5.6.3
5.1	Documentation standards	3.5.2	not applicable
5.2	Logic structure standards	3.5.2	not applicable
5.3	Coding standards	3.5.2	not applicable
5.4	Commentary standards	3.5.2	not applicable
5.5	Testing standards and practices	3.5.2	not applicable
5.6	Selected software quality assurance product and process metrics	3.5.2	not applicable
6.	Reviews and Audits	3.6	4.3,5.4.3,5.6.4
6.1	Purpose	3.6.1	not applicable
6.2	Features Review	3.6.2.7	not applicable

TABLE I: DOCUMENT AND STANDARD CROSS REFERENCE (cont.)

MCNP SQAP		IEEE 730.1	ISO 9000-3
6.3	Intermediate Version Path and Patch Documentation Review	3.6.2.1,3.6.2.2,3.6.2.3	not applicable
6.4	Coding Reviews and Audits	3.6.2.5,3.6.2.6	not applicable
6.5	Documentation Reviews	3.6.3	not applicable
6.6	Cross Reference	3.6.2	not applicable
7.	Test	3.7	5.7.1
8.	Problem Reporting and Corrective Action	3.8	4.4
9.	Tools, techniques, and methodologies	3.9	5.4.2.3,6.6
10.	Code control	3.10	not applicable
11.	Media control	3.11	not applicable
12.	Supplier control	3.12	not applicable
13.	Records collection, maintenance, and retention	3.13	not applicable
14.	Training	3.14	6.9
15.	Risk Management.	3.15	not applicable
A.	MCNP Software Verification and Validation Plan	3.4.2.3	4.1.1.2.2,5.4.6
B.	MCNP Software Configuration Management Plan	3.4.2.6	6.1.1,6.1.2,6.1.3.1,6.1.3.2,6.1.3.3
C.	Cross Reference between MCNP SQA controlling documents and the minimum documentation required by IEEE 730.1	3.6.3	not applicable
D.	Cross reference between the MCNP SQA reviews and audits and the minimum reviews and audits required by IEEE 730.1	3.6.2	not applicable
E.	Cross reference between the MCNP SQAP, IEEE 730.1, and IEEE 9000-3	not applicable	not applicable
F.	Definitions and Acronyms	not applicable	not applicable
G.	Figures and Tables	not applicable	not applicable

APPENDIX F

Definitions and Acronyms

Benchmark	- The process of reviewing software requirements and/or implementation to ensure that the models which the requirements describe are consistent with the physical reality they are intended to portray.
BoD	- MCNP Board of Directors
CFS	- Common File System in the LANL Integrated Computer Network
Developmental Version	- "work in-progress" version of the code, not released to users
Group Office	- XTM management consisting of the Group Leader and Deputy Group Leader.
IEEE	- Institute of Electrical and Electronics Engineers
Intermediate Version	- those revisions of a major version that include new features but have not been released to RSIC
ISO	- International Organization for Standardization
LANL	- Los Alamos National Laboratory
Major Version	- version released to RSIC for international distribution
MCNP	- Monte Carlo N-Particle radiation transport code
Patch	- a file in 'update' utility format that changes one version of MCNP to another
RSIC	- Radiation Shielding Information Center
SCMP	- Software Configuration Management Plan
SQA	- Software Quality Assurance
SQAP	- Software Quality Assurance Plan
SVVP	- Software Verification and Validation Plan

SVVR - Software Verification and Validation Report

Validation (for software) - The process of evaluating software to ensure compliance with requirements. This process can include the requirements to be as close as possible to physical benchmarks, but it can also include software requirements that have no relation to the physical world (for example, the requirement could be that the software must include PVM multitasking.)

Verification (for software) - The process of evaluating the products of a given phase to ensure correctness and consistency with respect to the products and standards provided as input to that phase

XTM - LANL Transport Methods Group

TABLE II: MCNP TASKS

	Monte Carlo Team	Board of Directors	Group Office	Documentation
Concept Exploration				
• Compile Features List	X			Features list, stored archivally on CFS
• Features Review				
- Approve or disapprove features for inclusion		X		Acceptance or Rejection documented on Features List
- Prioritize features		X		documented on the features list
Requirements & Design				
• Patch Review				
- Check Patch requirements & design to see if they exist		X		documented by LANL memo
- Make recommendations about additional requirements & design		X		documented by LANL memo
- Review patch for correctness; does the design implement the requirements?	X			documented by LANL memo
- Review Patch documentation for completeness	X			documented by LANL memo
• Develop patch requirements & design	X			documented in patch documentation
• Document patch requirements & design	X			by LANL memo
• Approve patches & patch documentation	X	X	X	by LANL memo
Implementation				
• Implement Patches	X			in patch commentary
• Coding Audit/Review	X			by LANL memo
• Review patch commentary	X			by LANL memo

TABLE II: MCNP TASKS (cont.)

	Monte Carlo Team	Board of Directors	Group Office	Documentation
Test				
• Run the MCNP test set on the patch; correct code until the patch patches the test set	X			not necessary
• Modify the test set so that the new feature is tested	X			in test set documentation
Installation & Checkout				
• Install MCNP & run test set	X			by LANL memo to user
Operation & Maintenance				
• Compile & Maintain Bug List	X			bug list, stored archivally on CFS
• Investigate Bug Claims	X			in bug list
• Implement Appropriate Bug Fixes	X			in patch commentary
Retirement				
• Approve major version release	X	X	X	by LANL memo

Los Alamos
NATIONAL LABORATORY
Los Alamos, New Mexico 87545