SOFTWARE REQUIREMENTS SPECIFICATION
FOR
RADIATION EFFECTS SOFTWARE

DOCUMENT CONTROL NUMBER: A-18-079U

VERSION 1.0.3

12/21/2018

PREPARED BY:
PERATON INC.
12975 WORLDGATE DRIVE
HERNDON, VA 2017

RECORD OF CHANGE

Date	Version	Author	Change Description
03/15/2018	1.0	K. Wegner	Initial Publication
03/23/2018	1.0.1	E. Sherman	Updates, corrections, and added Scaling Radiation Dose Maps
05/04/2018	1.0.2	E. Sherman	Completed GENFIT sections
12/21/2018	1.0.3	K. Wegner	Adding FLYTHRU capabilities

Table of Contents

List of Figures

Figure **Title** **Page**

No table of figures entries found.

List of Tables

Table **Title** **Page**

1.0 SCOPE

This section identifies the system and software for which requirements are provided, includes an overview of the system, as well as an overview of document contents.

1.1 Identification

1.2 System Overview

1.3 Document Overview

This document is organized into the following sections:

- Section 1 defines the purpose of the SRS, a summary of its contents, and provides a brief overview of the system
- Section 2 lists all documents referenced by this SRS
- Section 3 identifies all of the requirements for the CSCI(s) identified in Section 1.1 of this document
- Section 4 identifies the specific qualification methods used to verify the software requirements in this document
- Section 5 discusses the traceability of software requirements in this document to parent requirements
- Section 6 contains a list of acronyms used in this document and their definitions

2.0 REFERENCED DOCUMENTS

Table 2-1 contains a list of documents that are referenced within this document.

Table 2-1 Referenced Documents

Document Number	Document Title	Revision	Date

3.0 REQUIREMENTS

3.1 Required States and Modes

3.2 CSCI Capability Requirements

The following paragraphs identify the software requirements (or user stories), grouped by capability (or user function).

Table 3-1 describes the system-level requirements, epics or user stories from which the software requirements are sourced.

Table 3-1 Source Materials and System Requirements

Rqmt ID	Description	Acceptance Criteria	Target Release

3.2.1 Geometry

3.2.1.1 Points in Space

The cartesian representation for points in space ***shall*** be relative to an origin (0,0,0) point on the surface of the earth. The X-Y plane containing (0,0,0) is tangent to the earth. The Z coordinate is normal to the earth sphere at the origin. There is no special meaning to the directions of the X and Y axes except as references in placing objects relative to each other. In the Radiation Effects Software models, this coordinate system is sufficient because all objects will be relatively close to each other and to the origin.

3.2.1.2 Sequence of points

A *sequence of points* is a set of points in space associated with an object. Discrete points in the trajectory of a reentry vehicle is an example of a *sequence of points*. However, a *sequence of points* is not necessarily related to time. The points in a circle separated by 10 degrees form a circular *sequence of points*. RES ***shall*** be capable of creating arbitrary *sequence of points* so that the analyst can study radiation effects as the geometry between burst and receiver is varied.

RES ***shall*** be capable of creating a *linear sequence of points* in 2-space or 3-space by specifying the start and end points and number of equidistance points in between.

RES ***shall*** be capable of creating a *circular sequence of points* in 2-space or 3-space by specifying the radius of a circle, start and end angles and number of equidistance angles in between.

RES ***shall*** be capable of creating an *arbitrary sequence of points* in 2-space or 3-space by reading data points from a file or an internal data array. For example, the points might represent the parabolic trajectory of a reentry vehicle.

3.2.1.3 Relative Geometry of Two Points

Given two points in space RES ***shall*** calculate the height above sea level of each point and the distance between the two points. This calculation is required when determining air density.

In 2-space, given:

the coordinates of the first point (x_1, z_1)
the coordinates of the second point (x_2, z_2)

RES ***shall*** compute the altitude of each point above sea level:

$$h = \sqrt{x^2 + (z + R_E)^2} - R_E$$

where:

R_E is the mean radius of the earth

RES ***shall*** compute the distance between the two points:

$$r = \sqrt{(x_1 - x_2)^2 + (z_1 - z_2)^2}$$

In 3-space, given:

the coordinates of the first point (x_1, y_1, z_1)
the coordinates of the second point (x_2, y_2, z_2)

RES ***shall*** compute the altitude of each point above sea level:

$$h = \sqrt{x^2 + y^2 + (z + R_E)^2} - R_E$$

where:

R_E is the mean radius of the earth

RES ***shall*** compute the distance between the two points:

$$r = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

3.2.2 Air Depth

3.2.2.1 Air Depth Data from Database

RES ***shall*** be able to read from an SQLite database table AtmosphereProfile to get the integrated density and density by altitude.

3.2.2.2 Air Density at Altitude, $\boldsymbol{\rho}$

RES ***shall*** calculate the air density, ρ, based upon altitude and a pre-determined table of air density values. The table is indexed in increasing order of height above sea level. The value for ρ is a non-linear interpolation between two values of the table. For a given height h, RES ***shall*** find adjacent table values such that $h_k \leq h < h_{k+1}$.

RES ***shall*** calculate air density ρ (g/cm^3) at altitude h (cm) as follows:

$$\rho = \rho_k * e^{\ln\left(\frac{\rho_{k+1}}{\rho_k}\right)*\left(\frac{h-h_k}{h_{k+1}-h_k}\right)}$$

Where h_k, h_{k+1} are adjacent heights (cm) and ρ_k, ρ_{k+1} are adjacent air densities (g/cm^3) in the table.

3.2.2.3 Integrated Air Density at Altitude, $\overline{\boldsymbol{\rho}}$

RES ***shall*** calculate the integrated air density, $\overline{\rho}$, based upon altitude and a pre-determined table of integrated air density values. The table is indexed in increasing order of height above sea level. The value for $\overline{\rho}$ is a linear interpolation between two values of the table. For a given height h, RES ***shall*** find adjacent table values such that $h_k \leq h < h_{k+1}$.

RES ***shall*** calculate the integrated air density $\overline{\rho}$ (g/cm^2) at altitude h (cm) as follows:

$$\overline{\rho} = (\,\overline{\rho}_{k+1} - \overline{\rho}_k) * \left(\frac{h - h_k}{h_{k+1} - h_k}\right) + \overline{\rho}_k$$

where h_k, h_{k+1} are adjacent heights (cm) and $\overline{\rho}_k, \overline{\rho}_{k+1}$ are adjacent integrated air densities (g/cm^2) in the table.

3.2.2.4 Effective Air Density Between Two Altitudes, $\hat{\rho}$

RES ***shall*** calculate the *effective air density*, $\hat{\rho}$, over a height interval based upon 2 points in space. The heights of the 2 points are defined to be "*close to each other*" if any of these conditions hold:

- The height of the first point equals the height of the second point
- The integrated air density of the first point equals the integrated air density of the second point
- The horizontal range between the points exceeds 10 times the difference in the heights

If the heights of the 2 points are *close to each other*, then RES ***shall*** compute the *effective air density* $\hat{\rho}$ (g/cm^3) over the height interval as follows:

$$\hat{\rho} = (\rho_1 + \rho_2) / 2$$

where:

ρ_1 and ρ_2 are the air densities (g/cm^3) at the heights of each point

If the heights of the 2 points are *not close to each other*, then RES ***shall*** compute the *effective air density* $\hat{\rho}$ (g/cm^3) over the height interval as follows:

$$\hat{\rho} = \left(\overline{\rho}_1 - \overline{\rho}_2\right)/(h_1 - h_2)$$

where:

$\overline{\rho}_1$ and $\overline{\rho}_2$ are the integrated air densities (g/cm^2) at the heights of each point

h_1 and h_2 are the heights (cm) of each point

3.2.2.5 Air Depth Between Two Points in Space, d

For each type of radiation, there exists a pre-calculated table of radiation effect values. The index into these tables is an air depth value, d.

RES ***shall*** compute the air depth:

$$d = r * \hat{\rho} \quad \text{(g/cm}^2\text{)}$$

where:

$\hat{\rho}$ is the *effective air density* based upon burst and receiver altitudes

r is the slant range between burst and receiver

3.2.3 Devices

3.2.3.1 Devices from Database

RES ***shall*** be able to read from an SQLite database table Devices to get the device names and associated scaling and x-raydata. The scaling data are XData1, XData2, XData3, XData4, XData5, XData6, XData7, and EXA. The x-ray data are FRACA1, FRACA2, FRACA3, FRACA4, TPULSEA1, TPULSEA2, TPULSEA3, TPULSEA4, TSDA, NFRACA, BBTA1, BBTA2, BBTA3, BBTA4.

3.2.3.2 Primary and Secondary Devices

RES ***shall*** be capable of defining a source as either a primary device and yield or a primary device and yield and a secondary device and yield.

3.2.3.3 X-ray Pulse Width

RES ***shall*** be capable of calculating the pulse width for x-ray data. The pulse width, t_w, depends on device and is calculated as follows:

For devices 1, 5, 6, 8, 9, 17, 18, 19, 29, 30, 31:

$$t_w = ta_i w^{Td_1} * 1.0\text{E-9}$$

For device 20:

For i=1,2,3:

$$t_w = ta_i w^{Td_1} * 1.0\text{E-9}$$

For i=4:

$$t_w = \frac{Td_2}{{Bbta_4}^{Td_3}} ta_i w^{Td_1} * 1.0\text{E-9}$$

For device 2:

For i=0:

$$t_w = Td_4 \exp(Td_5 Td_{16}) + Td_6 \exp(Td_7 Td_{16})$$

For i=1:

$$t_w = Td_8 \exp(Td_9 Td_{16}) + Td_{10} \exp(Td_{11} Td_{16})$$

For i=2:

$$t_w = Td_{12} \exp(Td_{13} Td_{16}) + Td_{14} \exp(Td_{15} Td_{16})$$

For i=3:

$$t_w = ta_i w^{Td_1} * 1.0\text{E-9}$$

For device 3:

For i=0:

$$t_w = Td_4 \exp(Td_5 Td_{17}) + Td_6 \exp(Td_7 Td_{17})$$

For i=1:

$$t_w = Td_8 \exp(Td_9 Td_{17}) + Td_{10} \exp(Td_{11} Td_{17})$$

For i=2:

$$t_w = Td_{12} \exp(Td_{13} Td_{17}) + Td_{14} \exp(Td_{15} Td_{17})$$

For i=3:

$$t_w = ta_i w^{Td_1} * 1.0\text{E-9}$$

For device 4:

For i=0,1:

$$t_w = ta_i w^{Td_{17}} * 1.0\text{E-9}$$

For i=2,3:

$$t_w = 1.0$$

For device 24:

For i=0:

$$t_w = Td_4 \exp(Td_5 Td_{18}) + Td_6 \exp(Td_7 Td_{18})$$

For i=1:

$$t_w = Td_8 \exp(Td_9 Td_{18}) + Td_{10} \exp(Td_{11} Td_{18})$$

For i=2:

$$t_w = Td_{12} \exp(Td_{13} Td_{18}) + Td_{14} \exp(Td_{15} Td_{18})$$

For i=3:

$$t_w = ta_i w^{Td_1} * 1.0\text{E-9}$$

For devices 15, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42:

$$t_w = ta_1$$

For devices: 56, 57, 58, 59, 60, 62, 63, 64, 65, 66, 67, 68, 70, 71, 72, 73, 74, 75, 76:

$$t_w = ta_1 * 1.0\text{E-9}$$

For device 11:

$$t_w = ta_i w * 1.0\text{E-9}$$

For devices 21, 22, 23:

$$t_w = ta_i * 1.0\text{E-}9$$

For devices: 44, 45, 47, 53 ,54, 55:

$$t_w = ta_i w^{Tsda} * 1.0\text{E-}8$$

For device 48:

$$t_w = ta_i \exp(w * Tsda) * 1.0\text{E-}8$$

For device 49:

$$t_w = (ta_i \log(w) - Tsda) * 1.0\text{E-}8$$

For devices 50, 51, 69:

$$t_w = \exp(ta_i) w^{Tsda} * 1.0\text{E-}8$$

For device 52:

$$t_w = \frac{ta_i}{Tsda} * 1.0\text{E-}8$$

For all other devices:

$$t_w = 1.0$$

where i is the fraction index, ta_i is TPULSEA i, w is the yield, Td_j is from the database table TDATA at ID j, $Bbta_i$ is BBTA i, and $Tsda$ is TSDA.

3.2.4 Radiation Dose Values

At each point along a *sequence of points*, RES ***shall*** be capable of calculating a set of radiation dose values. These dose values differ from point to point along the path because the air depth to the burst differs at each point.

3.2.4.1 Types of Radiation Dose Values

RES ***shall*** be capable of determining dose values for these types of radiation effects:

Table 3-2 Required Environments

Dose From	Radiation Effect	Units
Neutron, Gamma, Secondary Gamma	Total Number Fluence	particles/cm^2
Neutron, Gamma, Secondary Gamma	Total Energy Fluence	MeV/cm^2
Neutron	1 MeV Equivalent Neutron Fluence	neutrons/cm^2
Neutron, Gamma, Secondary Gamma, X-ray	Ionizing Dose	Rad(Si)
Neutron, Gamma	Biological Tissue Dose Rems-Tis	Rems-Tis
Neutron, Gamma	Biological Tissue Dose Rads-Tis	Rads-Tis
Neutron	Epithermal Neutron Fluence	neutrons/cm^2
Neutron	ASTM 1 MeV Equivalent Dose	neutrons/cm^2
Gamma, X-ray	Ionizing Dose Rate	Rad(Si)/s
Gamma, X-ray	Ionizing Dose Rate Slope	Rad(Si)/s^2
Neutron	Fission Heating	Fissions/g
X-Ray	Forward/Total Energy	cal/cm^2
X-Ray	Forward/Total Energy Rate	cal/cm^2/s

3.2.4.2 Radiation Dose Scaled By Surface Area of Sphere

RES ***shall*** be capable of scaling a radiation dose by the surface area of the sphere containing the receiver and centered at the burst.

$$\chi_{scaled} = \chi * 4\pi r^2$$

where:

χ is a radiation dose

r is the slant range between burst and receiver

3.2.4.3 Radiation Dose Scaled By Balanced Ratio

RES ***shall*** be capable of scaling a radiation dose by a fixed balanced ratio value:

$$\chi_{scaled} = \chi * b$$

where:

χ is a radiation dose

b is a fixed scalar

3.2.4.4 Radiation Dose Scaled By Balanced Environments

RES ***shall*** be capable of scaling a radiation dose by a balanced environment:

$$\chi_{scaled} = \frac{\chi}{\chi_b}$$

where:

χ is a radiation dose

χ_b is a balanced environment

3.2.4.5 Radiation Dose Maps

For each type of radiation effect there is a normalized radiation dose map. The dose map can be scaled based on device parameters including yield. The index into a radiation dose map is an air depth. An interpolation is performed between the discrete values in the map. The dose maps are monotonically decreasing - the larger the air depth, the smaller the effective dose. For this reason there exists an inverse map from dose to air depth. This inverse map is useful when trying to determine the conditions necessary to achieve a specific constraint dose.

3.2.4.5.1 Dose Map Data

RES ***shall*** read the database table Environment to determine rhor and dose data based on DeviceId, DoseKindId, KindEnvId, ShldId, and ParticleID.

For x-ray particles, the DeviceId is NOT the Id of the device, but is instead determined as follows as there are N_{frac} sets of data read where N_{frac} is NFRACA in the Devices table for the specified device. For each N_{frac} the corresponding value of BBTA (BBTA1, BBTA2, BBTA3, or BBTA4) is matched to an XARRAY value in the XARRAY table in the database. The XARRAY index that corresponds to the BBTA value is then the DeviceId for looking up the data.

The data that is read is then scaled by:

$$\chi_{scaled} = \chi * fraca * t_w$$

where fraca is the corresponding FRACA value for the device (FRACA1, FRACA2, FRACA3, or FRACA4) and t_w is the pulse width described in Section 3.2.3.3. The scaled N_{frac} sets of data are then summed.

For x-ray dose rate slope cases the scaling is as follows:

$$\chi_{scaled} = \chi * fraca * t_w * d$$

where *d* is the scaling factor described in Section 3.2.4.6.1.

3.2.4.5.2 Interpolation of Dose Maps

Given an air depth value, d, RES ***shall*** be capable of determining the radiation dose, χ, for each of the radiation effects listed in Section 0. RES ***shall*** determine the value for χ as an interpolation between two values of the relevant dose map. There are 5 possible interpolation methods, depending upon the nature of the dose map. These interpolation methods are:

- Linear X, Linear Y
- Linear X, Natural Log Y
- Linear X, Common Log Y
- Natural Log X, Natural Log Y
- Common Log X, Common Log Y

3.2.4.5.3 Interpolation Linear X, Linear Y

For a given air depth d, RES ***shall*** find adjacent table values such that $d_k \leq d < d_{k+1}$
RES ***shall*** calculate the radiation dose as follows:

$$\chi = (\chi_{k+1} - \chi_k) * \left(\frac{d - d_k}{d_{k+1} - d_k}\right) + \chi_k$$

where d_k, d_{k+1} are adjacent surface densities (g/cm^2) and χ_k, χ_{k+1} are adjacent radiation doses in the table.

3.2.4.5.4 Interpolation Linear X, Ln Y

For a given areal density d, RES ***shall*** find adjacent table values such that $d_k \leq d < d_{k+1}$
RES ***shall*** calculate the radiation dose as follows:

$$exponent = (\ln \chi_{k+1} - \ln \chi_k) * \left(\frac{d - d_k}{d_{k+1} - d_k}\right) + \ln \chi_k$$

$$\chi = e^{exponent}$$

where d_k, d_{k+1} are adjacent surface densities (g/cm^2) and χ_k, χ_{k+1} are adjacent radiation doses in the table.

3.2.4.5.5 Interpolation Linear X, Log$_{10}$ Y

For a given air depth d, RES ***shall*** find adjacent table values such that $d_k \le d < d_{k+1}$
RES ***shall*** calculate the radiation dose as follows:

$$exponent = (\log_{10}\chi_{k+1} - \log_{10}\chi_k) * \left(\frac{d - d_k}{d_{k+1} - d_k}\right) + \log_{10}\chi_k$$

$$\chi = 10^{exponent}$$

where d_k, d_{k+1} are adjacent surface densities (g/cm^2) and χ_k, χ_{k+1} are adjacent radiation doses in the table.

3.2.4.5.6 Interpolation Ln X, Ln Y

For a given air depth d, RES ***shall*** find adjacent table values such that $d_k \le d < d_{k+1}$
RES ***shall*** calculate the radiation dose as follows:

$$exponent = (\ln\chi_{k+1} - \ln\chi_k) * \left(\frac{\ln d - \ln d_k}{\ln d_{k+1} - \ln d_k}\right) + \ln\chi_k$$

$$\chi = e^{exponent}$$

where d_k, d_{k+1} are adjacent surface densities (g/cm^2) and χ_k, χ_{k+1} are adjacent radiation doses in the table.

3.2.4.5.7 Interpolation Log$_{10}$ X, Log$_{10}$ Y

For a given air depth d, RES ***shall*** find adjacent table values such that $d_k \le d < d_{k+1}$
RES ***shall*** calculate the radiation dose as follows:

$$exponent = (\log_{10}\chi_{k+1} - \log_{10}\chi_k) * \left(\frac{\log_{10} d - \log_{10} d_k}{\log_{10} d_{k+1} - \log_{10} d_k}\right) + \log_{10}\chi_k$$

$$\chi = 10^{exponent}$$

where d_k, d_{k+1} are adjacent surface densities (g/cm^2) and χ_k, χ_{k+1} are adjacent radiation doses in the table.

3.2.4.6 Scaling Radiation Dose Maps

RES ***shall*** be able to calculate a scaling factor, s, that scales the radiation dose maps based on yield as well as scaling data stored in the database, such that

$$\chi_{scaled} = \chi * s$$

The scaling factor depends on the particle type as described below.

For neutrons and seconday gammas:

$$s = 6.022E23x_1 w$$

where x_1 is the scaling data XData1 for the specified device in the Devices table in the database, and *w* is the yield.

For prompt gamma dose:

$$s = 2.618E25\frac{x_2}{x_7}w^{x_3+1}$$

where x_2, x_3 and x_7 are the scaling data XData2, XData3, and XData7, respectively, and *w* is the yield.

For prompt gamma dose rate or dose rate slope:

$$s = \frac{x_4}{x_7}w^{x_5}$$

where x_4, x_5 and x_7 are the scaling data XData4, XData5, and XData7, respectively, and *w* is the yield.

For x-rays:

$$s = 1.0E12e_{xa}w$$

where *w* is the yield and e_{xa} is EXA for the specified device in the Devices table in the database except for the following cases. For device 5:

$$e_{xa} = \frac{Asd_1 * w + Asd_2 * w^{Asd_3}}{w}$$

For device 8 and *w* > Asd_4:

$$e_{xa} = \exp\left(\frac{\log\left(\frac{Asd_5}{w}\right)}{Asd_6}Asd_7 - Asd_8\right)$$

For device 9 and *w* > Asd_9:

$$e_{xa} = \exp\left(\frac{\log\left(\frac{Asd_{10}}{w}\right)}{Asd_{11}} Asd_7 - Asd_8\right)$$

where *Asd* is from the ASD table in the database.

3.2.4.6.1 Scaling Dose Rate Slope

For dose rate slope cases RES ***shall*** be able to calculate an additional scaling factor, d, that scales the radiation dose maps based on yield as well as scaling data stored in the database, such that

$$\chi_{scaled} = \chi * s * d$$

For Gamma Dose Rate Slope, RES ***shall*** calculate d as follows:

$$d = 1.0E9 * q * r$$

where

$$q = \frac{\alpha - bl}{\beta^2 b} - \frac{1}{b}$$

$$r = \frac{tpk}{b} \exp\left(\frac{-0.5}{\beta^2}(bl - \alpha)^2 + \frac{0.5}{\beta^2}(\log(tpk) - \alpha)^2\right)$$

where

$$\alpha = \beta^2 + log(tpk)$$

$$bl = \alpha - 1.5\beta^2 - 0.5\beta\sqrt{\beta^2 + 4}$$

$$b = \exp(bl)$$

$$a = \frac{\frac{Ddt_1}{x_4 w^{x_5}} w x_2 w^{x_3}}{Ddt_2 tpk}$$

$$tpk = Apk_{deviceId-1} w^{Ddt_0}$$

where is β is solved from:

$$0 = \beta - a exp(-0.5\beta^2)$$

In the above, *w* is the yield, *Apk* and *Ddt* are from the tables APK and DDT in the database and x_2, x_3, x_4 and x_5 are the scaling data XData2, XData3, XData4, and XData5 from the Devices table in the database.

Note this is not valid for devices IDs greater than 30.

For X-ray Dose Rate Slope, RES ***shall*** calculate *d* as follows:

$$d = 1.0E9 * q * r$$

where

$$q = \frac{\alpha - bl}{\beta^2 b} - \frac{1}{b}$$

$$r = \frac{tpk}{b} \exp\left(\frac{-0.5}{\beta^2}(bl - \alpha)^2 + \frac{0.5}{\beta^2}(\log(tpk) - \alpha)^2\right)$$

where

$$\alpha = \beta^2 + log(tpk)$$

$$bl = \alpha - 1.5\beta^2 - 0.5\beta\sqrt{\beta^2 + 4}$$

$$b = \exp(bl)$$

$$a = \frac{1.0E9 * acon_2}{Ddt_2 * acon_1}$$

$$tpk = acon_1 w^{Ddt_0}$$

$$acon_1 = \begin{cases} 0.5 \; for \; DeviceId = 20 \\ 1.0 \qquad otherwise \end{cases}$$

$$acon_2 = \frac{tpulse}{w^{Ddt_0}}$$

where is β is solved from:

$$0 = \beta - aexp(-0.5\beta^2)$$

and *tpulse* is the pulse width described in section 3.2.3.3.

In the above, *w* is the yield, *Ddt* is from the DDT table in the database and x_2, x_3, x_4 and x_5 are the scaling data XData2, XData3, XData4, and XData5 from the Devices table in the database.

Note this is not valid for devices IDs: 4, 7, 10, 11, 12, 13, 14,15, 16, 21, 22, 23.

3.2.4.7 Combining Radiation Dose Maps

3.2.4.7.1 Combining Particle Types

For radiation effects that have dose from multiple particle types, RES ***shall*** be capable of calculating the dose for either a single particle type or any combination of the available particle types. For a combination of particle types, RES ***shall*** sum the dose for the selected particle types.

3.2.4.7.2 Combining Primary and Secondary Devices

When both a primary and seconday device are defined, RES ***shall*** be capable of combining the doses from the primary and secondary device. For dose cases, RES ***shall*** sum the dose from the primary and secondary devices. For dose rate or dose rate slope cases, RES ***shall*** take the maximum of the primary and secondary values.

3.2.5 GENFIT Capability

A *run* is configured by a set of input parameters including how an independent variable is to behave. The result of a *run* is a table of output values. The rows of the table differ because of the changes to the independent value. Depending on the type of *run*, the required input parameters may include:

- Dispositions of burst and receiver
- Characteristics of the device, primary and secondary yields
- How to vary the independent variable
- Which environmental dose values are to be determined
- Constraint value
- Is it a balanced environment? If so, what is the balanced ratio?

3.2.5.1 Independent Variable

For a particular *run*, RES ***shall*** be capable of iterating over exactly one of the following independent variables:

- altitude of the burst above sea level, h_1
- slant range from burst to receiver, r
- the elevation angle from burst to receiver relative to the tangent plane of the burst, θ
- the horizontal range from burst to receiver in the tangent plane of the burst, δx
- air depth (rhor) from burst to receiver, d
- yield of the device

3.2.5.2 Output Values

RES ***shall*** be capable of calculating a subset of these output parameters based on the type of *run*:

- altitude of the burst above sea level, h_1
- altitude of the receiver above sea level, h_2
- the elevation angle from burst to receiver relative to the tangent plane of the burst, θ
- the horizontal range from burst to receiver in the tangent plane of the burst, δx

- the vertical range from burst to receiver normal to the tangent plane of the burst, δz
- air depth (rhor) from burst to receiver, d
- radiation dose, χ
- radiation dose scaled by a balanced ratio
- radiation dose scaled by $4\pi r^2$
- slant range from burst to receiver, *r*
- yield, *w*

3.2.5.3 Run Kinds

3.2.5.3.1 Run Kind - ENVRNG

This run varies the slant range from burst to receiver in order to determine radiation dose values along a linear sequence of points.

These parameters are fixed at each point in the sequence:

- height of burst, h_1
- elevation angle, θ
- device yield, *w*

This parameter will vary as a linear sequence of points:

- slant range from burst to receiver, r

As r varies, these parameters will also vary at each point in the sequence:

- horizontal range, δx
- vertical range, δz
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined at each point in the sequence:

- radiation dose, χ

RES ***shall*** output the following for this run:

- slant range from burst to receiver, *r*
- radiation environment, χ
- receiver altitude, h_2
- air depth (rhor) from burst to receiver, d
- $4\pi r^2$ radiation environment

3.2.5.3.2 Run Kind - BALRNG

This run is identical to ENVRNG except the radiation dose χ is scaled by a fixed balanced ratio.

3.2.5.3.3 Run Kind - XENVRNG

This run varies the horizontal range from burst to receiver in order to determine radiation dose values along a linear sequence of points.

These parameters are fixed at each point in the sequence:

- height of burst, h_1
- vertical range, δz
- device yield, *w*

This parameter will vary as a linear sequence of points:

- horizontal range, δx

As δx varies, these parameters will also vary at each point in the sequence:

- slant range from burst to receiver, r
- elevation angle, θ
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined at each point in the sequence:

- radiation dose, χ

RES ***shall*** output the following for this run:

- horizontal range from burst to receiver, δx
- radiation environment, χ
- receiver altitude, h_2
- air depth (rhor) from burst to receiver, d
- $4\pi r^2$ radiation environment
- slant range from burst to receiver, *r*
- elevation angle from burst to receiver, θ

3.2.5.3.4 Run Kind - XBALRNG

This run is identical to XENVRNG except the radiation dose χ is scaled by a fixed balanced ratio.

3.2.5.3.5 Run Kind - PI4RHOR

For a given *Dose Map* this run shows the mapping from certain air depth (rhor) values to the corresponding interpolated radiation dose values from the map.

The fixed inputs to this run:

- yield for the relevant *Dose Map, w*

This parameter will vary as a non-linear sequence of points:

- air depth (rhor), d

The values for rhor are selected in a non-linear manner at each power of 10. As an example of the algorithm, suppose that:

- the minumum rhor value is 0.001
- the maximum rhor value is 10,000
- number of steps between minumum and maximum is 9

Then there will be 64 rhor values:

```
0.001,  0.002,  0.003,  0.004,  0.005,  0.006,  0.007,  0.008,  0.009,
  0.1,    0.2,    0.3,    0.4,    0.5,    0.6,    0.7,    0.8,    0.9,
 0.01,   0.02,   0.03,   0.04,   0.05,   0.06,   0.07,   0.08,   0.09,
    1,      2,      3,      4,      5,      6,      7,      8,      9,
   10,     20,     30,     40,     50,     60,     70,     80,     90,
  100,    200,    300,    400,    500,    600,    700,    800,    900,
 1000,   2000,   3000,   4000,   5000,   6000,   7000,   8000,   9000,
10000
```

RES ***shall*** output the following for this run:

- air depth (rhor) from burst to receiver, d
- radiation environment

3.2.5.3.6 Run Kind - BALRHOR

This run is identical to PI4RHOR except the radiation dose χ is scaled by a fixed balanced ratio and if more than one environment is defined by the first environment:

$$\chi_{scaled} = \frac{\chi * b}{\chi_1}$$

where *b* is the fixed balanced ratio and χ_1 is the first defined environment's dose.

3.2.5.3.7 Run Kind - FITRNG

The radiation dose values depend upon the yield of the device. For a given slant range between burst and receiver, it is useful to know what yield of the device results in a fixed, constraint radiation dose value. For example, this helps to determine how far away from the burst is a "safe" distance for a certain device type and radiation type.

This run varies the slant range from burst to receiver, along a linear sequence of points, in order to determine the device yield that achieves a fixed, constraint radiation dose value from a selected *Dose Map*.

These parameters are fixed at each point in the sequence:

- height of burst, h_1
- elevation angle, θ
- the constraint radiation dose value to be achieved, χ

This parameter will vary as a linear sequence of points:

- slant range from burst to receiver, r

As r varies, these parameters will also vary at each point in the sequence:

- horizontal range, δx
- vertical range, δz
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined at each point in the sequence:

- device yield, *w*

RES ***shall*** perform the following algorithm to determine the yield:

- The air depth (rhor) at the current slant range, d, is determined from the geometry
- The minimum yield is set to 0.1
- The maximum yield is set to 2000.0
- A binary search is performed on the yield
- During the binary search, the *Dose Map* for the relevant radiation type is repeatedly rescaled by the yield value
- The binary search ends when the *Dose Map*, indexed by d, results in the constraint dose value

RES ***shall*** output the following for this run:

- yield, *w*
- secondary yield, *w*

- primary yield, *w*
- receiver altitude, h_2
- air depth (rhor) from burst to receiver, d
- horizontal range from burst to receiver, δx
- slant range from burst to receiver, *r*
- $4\pi r^2$ radiation environment
- radiation environment, χ

3.2.5.3.8 Run Kind - FITCON

This run varies the elevation angle from burst to receiver. At each angle, RES ***shall*** determine the slant range that achieves a fixed, constraint radiation dose value from a selected *Dose Map*.

These parameters are fixed at each point in the sequence:

- height of burst, h_1
- the constraint radiation dose value to be achieved, χ
- device yield, *w*

This parameter will vary, creating a sequence of points:

- elevation angle, θ

As θ varies, these parameters will also vary at each point in the sequence:

- horizontal range, δx
- vertical range, δz
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined at each point in the sequence:

- slant range from burst to receiver, r

RES ***shall*** perform the following algorithm to determine the resultant slant range:

- The minimum range is set to 1 meter
- The maximum range is taken from the maximum rhor value in the *Dose Map*
- A binary search is performed on the range
- During the binary search, the dose value is extracted from the *Dose Map* based on the rhor (which is based on the range since air depth is fixed by the elevation angle)
- The binary search for a slant range ends when the extracted dose is approximately equal to the constraint dose value

This run varies the slant range from burst to receiver in order to determine radiation dose values along a linear sequence of points.

RES ***shall*** output the following for this run:

- elevation angle from burst to receiver, θ
- air depth (rhor) from burst to receiver, d
- receiver altitude, h_2
- horizontal range from burst to receiver, δx
- slant range from burst to receiver, *r*
- $4\pi r^2$ radiation environment

3.2.5.3.9 Run Kind - BALCON

This run performs a FITCON calculation for the first environment defined. It then calculates the dose for subsequent environments at the slant range calculated for the first environment.

3.2.5.3.10 Run Kind - FITYLD

This run varies the yield for a device. At each yield value, RES ***shall*** determine the slant range that achieves a fixed, constraint radiation dose value from a selected *Dose Map*.

These parameters are fixed at each point in the sequence:

- height of burst, h_1
- elevation angle, θ
- the constraint radiation dose value to be achieved, χ

This parameter will vary:

- device yield, *w*

As the resultant slant range r varies, these parameters will also vary:

- horizontal range, δx
- vertical range, δz
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined for each yield value:

- slant range from burst to receiver, r

RES ***shall*** iterate over the yield values to determine the resultant slant range for each yield:

- The *Dose Map* for the relevant radiation type is rescaled by the yield value

- The minimum range is set to 1 meter
- The maximum range is taken from the maximum rhor value in the *Dose Map*
- A binary search is performed on the range
- During the binary search, the dose value is extracted from the *Dose Map* based on the rhor (which is based on the range since air depth is fixed by the elevation angle)
- The binary search for a slant range ends when the extracted dose is approximately equal to the constraint dose value

RES ***shall*** output the following for this run:

- yield, *w*
- primary yield, *w*
- secondary yield, *w*
- air depth (rhor) from burst to receiver, d
- receiver altitude, h_2
- horizontal range from burst to receiver, δx
- slant range from burst to receiver, *r*
- $4\pi r^2$ radiation environment

3.2.5.3.11 Run Kind - FITHOB

This run varies the height of the burst. At each height, RES ***shall*** determine the slant range that achieves a fixed, constraint radiation dose value from a selected *Dose Map*.

These parameters are fixed at each point in the sequence:

- elevation angle, θ
- the constraint radiation dose value to be achieved, χ
- device yield, *w*

This parameter will vary:

- height of burst, h_1

As the resultant slant range r varies, these parameters will also vary:

- horizontal range, δx
- vertical range, δz
- height of receiver, h_2
- air depth (rhor) from burst to receiver, d

This is the result to be determined at each height of burst:

- slant range from burst to receiver, r

RES ***shall*** perform the following algorithm to determine the resultant slant range:

- The minimum range is set to 1 meter
- The maximum range is taken from the maximum rhor value in the *Dose Map*
- A binary search is performed on the range
- During the binary search, the dose value is extracted from the *Dose Map* based on the rhor (which is based on the range since air depth is fixed by the elevation angle)
- The binary search for a slant range ends when the extracted dose is approximately equal to the constraint dose value

RES ***shall*** output the following for this run:

- height of burst, h_1
- air depth (rhor) from burst to receiver, d
- receiver altitude, h_2
- horizontal range from burst to receiver, δx
- slant range from burst to receiver, *r*
- $4\pi r^2$ radiation environment

3.2.5.3.12 Run Kind - BALHOB

This run performs a FITHOB calculation for the first environment defined. It then calculates the dose for subsequent environments at the slant range calculated for the first environment.

3.2.5.3.13 Run Kind - ENVHOB and ENVHOB2

RES ***shall*** perform FITHOB on every type of radiation *Dose Map* for a fixed device yield. At each height of burst, RES ***shall*** determine the largest range among the radiation types.

For example, ENVHOB can be used to determine a "safe" distance between burst and receiver considering all possible radiation effects.

RES ***shall*** output the following for this run:

- height of burst, h_1
- environment name for the largest range
- environment units for the largest range
- horizontal range from burst to receiver for the largest range, δx
- slant range from burst to receiver for the largest range, *r*
- air depth (rhor) from burst to receiver for the largest range, d
- $4\pi r^2$ radiation environment for the largest range
- constraint level for the largest range
- balanced radiation environments

3.2.6 FLYTHRU Capability

FLYTHRU models the radiation effects related to the re-entry of a leader and follower RB and a hostile or friendly burst event. The follower RB is the detector or receiver of the radiation.

3.2.6.1 Trajectory Model

FLYTHRU ***shall*** compute the Cartesian position in 3-space of an object at any time along a trajectory model. The trajectory model specifies a terminal position, velocity and angle at impact. The leader and follower RB may use the same trajectory model or there may be an offset between the 2 trajectories.

3.2.6.2 Prompt Dose

Given the Cartesian 3D position of a burst event, FLYTHRU ***shall*** be capable of computing prompt nuclear radiation environments at any point along the trajectory of the receiver RB (see *3.2.1 Geometry*). These are static positions of the receiver RB, as if the burst occurred at the time the RB arrived at this point. The prompt environments are those of the GENFIT Capability and are specified in *Table 3-2 Required Environments*. FLYTHRU ***shall*** calculate the prompt nuclear environments from a threat weapon to a receiver in "**Environment versus Altitude**" (ICON(1) = 5) run mode. The scenario is to model environments at a follower RB relative to a threat weapon intercepting a leading RB. The follower RB is on the same trajectory as the leading RB with a location offset defined by a time spacing, cross-range offset and up/downrange offset.

The user specifies:

- Trajectory of the receiver RB
- Environments to be calculated
- Positions along the trajectory of the receiver RB, for example, starting and ending altitudes, starting and ending times relative to RB atmospheric reentry
- Position of the burst event

FLYTHRU ***shall*** compute:

- Prompt doses for the specified environments at each point
- Horizontal and slant ranges (burst to receiver) at each point
- Angle between burst and receiver at each point
- Height of burst and receiver at each point

3.2.6.3 Accumulating Dose

FLYTHRU ***shall*** calculate the delayed nuclear environments from a threat weapon to a receiver in "Dose Rate Integral" (ICON(1)=0) mode. The follower RB is on the same trajectory as the leading RB with a location offset defined by a time spacing, cross-range offset and up/downrange offset. The scenario is to model environments at a follower RB relative to a threat weapon intercepting a leading RB.

The user specifies:

- Trajectory of the receiver RB
- Environments to be calculated
- Positions along the trajectory of the receiver RB, for example, starting and ending altitudes, starting and ending times relative to RB atmospheric reentry
- Position of the burst event
- Constraint values for x-ray fluence level and accumulated dose

FLYTHRU ***shall*** calculate:

- Delayed dose rate along a follower RB trajectory relative to the threat weapon
- Delayed cumulative dose as a function of time from the hostile burst
- Time spacing between the leading and follower RBs at atmospheric re-entry, such that a specific x-ray fluence level is achieved
- The up/down range offset and/or cross range offset where the cumulative delayed dose is equal to a user input value

FLYTHRU ***shall*** use existing RANC methodology to determine the delayed and accumulated dose values.

3.2.7 OVRDRV Capability

3.2.8 FTPRNT Capability

3.2.9 MDET Capability

3.2.10 Graphical User Interface Capability

3.3 CSCI External Interface Requirements

3.3.1 XML Files

3.3.1.1 XML Input File

RES ***shall*** read an input xml file of the following format:


```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<simulations xmlns="http://tempuri.org/XMLSchema.xsd">
  <simulation>
    <title>Title</title>
    <classification>UNCLASSIFIED</classification>
    <notes></notes>
    <weapon>
      <primarydevice>testweapon1</primarydevice>
      <primaryyield units="kt">0.1</primaryyield>
      <secondarydevice></secondarydevice>
      <secondaryyield units="kt">0.0</secondaryyield>
    </weapon>
    <geometry>
      <runkind>FITYLD</runkind>
      <yield>
        <value>10000.0</value>
        <minvalue units="kt">10.0</minvalue>
        <maxvalue units="kt">1000.0</maxvalue>
        <numsteps>10</numsteps>
      </yield>
      <heightofburst>
        <value units="ft">10000.0</value>
        <minvalue units="ft">0.0</minvalue>
        <maxvalue units="ft">10000.0</maxvalue>
        <numsteps>10</numsteps>
      </heightofburst>
      <angle>
        <value units="degrees">0.0</value>
        <minvalue units="degrees ">0.0</minvalue>
        <maxvalue units="degrees ">360.0</maxvalue>
        <numsteps>36</numsteps>
      </angle>
      <slantrange>
        <value units="ft">1000.0</value>
        <minvalue units="ft">100.0</minvalue>
        <maxvalue units="ft">10000.0</maxvalue>
        <numsteps>10</numsteps>
      </slantrange>
```

```
      <rhor>
        <value units="g/cm^2">100.0</value>
        <minvalue units="g/cm^2">10.0</minvalue>
        <maxvalue units="g/cm^2">1000.0</maxvalue>
        <numsteps>10</numsteps>
      </rhor>
    </geometry>
    <environments>
      <balanced>false</balanced>
      <environment>
        <name>environment 1</name>
        <dosekind>Dose</dosekind>
        <kindenv>Total Number Fluence</kindenv>
        <ishld>NONE</ishld>
        <particles>
          <neutron>true</neutron>
          <gamma>false</gamma>
          <secondarygamma>false</secondarygamma>
          <xray>false</xray>
        </particles>
        <constraint>1.0</constraint>
        <balancedratio>1.0</balancedratio>
      </environment>
    </environments>
    <atmosphere>standard</atmosphere>
  </simulation>
</simulations>
```

3.3.1.2 XML Output File

RES ***shall*** write an output xml file that repeats the input file with the addition of a results section of the following format. The results columns depend on the outputs for the Run Kind as described in Section 3.2.5.3.

```
  <results>
   <result>
    <name>environment 1</name>
    <columns>
     <column>
      <name>yield</name>
      <units>kt</units>
     </column>
     <column>
      <name> primary yield</name>
      <units>kt</units>
```

```
      </column>
      <column>
        <name>secondary yield</name>
        <units>kt</units>
      </column>
      <column>
        <name>rhor</name>
        <units>g/cm2</units>
      </column>
      <column>
        <name>altitude</name>
        <units>ft</units>
      </column>
      <column>
        <name>horizontal range</name>
        <units>ft</units>
      </column>
      <column>
        <name>slant range</name>
        <units>ft</units>
      </column>
      <column>
        <name>pi4renv</name>
        <units>particles/cm2</units>
      </column>
    </columns>
    <data>
      <row>10, 0, 10, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>109, 0, 109, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>208, 0, 208, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>307, 0, 307, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>406, 0, 406, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>505, 0, 505, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>604, 0, 604, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>703, 0, 703, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>802, 0, 802, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>901, 0, 901, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
      <row>1000, 0, 1000, 500, 10000, 18641.4, 18641.4, 9.42625e+13</row>
    </data>
  </result>
</results>
```

3.4 CSCI Internal Interface Requirements

3.5 CSCI Internal Data Requirements

3.6 Adaptation Requirements

3.7 Safety Requirements

3.8 Security and Privacy Requirements

3.9 CSCI Environment Requirements

3.10 Computer Resource Requirements

3.10.1 Computer Hardware Requirements

3.10.2 Computer Hardware Resource Utilization Requirements

3.10.3 Computer Software Requirements

3.10.4 Computer Communications Requirements

3.11 Software Quality Factors

3.12 Design and Implementation Constraints

3.13 Personnel-Related Requirements

3.14 Training-Related Requirements

3.15 Logistics-Related Requirements

3.16 Other Requirements

3.17 Packaging Requirements

3.18 Precedence and Criticality of Requirements

4.0 QUALIFICATION PROVISIONS

5.0 REQUIREMENTS TRACEABILITY

6.0 ACRONYMS

Acronym	Definition
CSCI	Computer Software Configuration Item
DID	Data Item Description
IRS	Interface Requirements Specification
SRS	Software Requirements Specification

7.0 NOTES BY WALLACE WESTLAKE

7.1 Calculation Modes

GENFIT ***shall*** calculate the prompt nuclear environments from a threat weapon to a receiver in one of fifteen GENFIT run modes. These modes describe a single point calculation or a 2-D curve by a parameter sweep through the independent variable.

GENFIT ***shall*** :

1. Calculate the *environment* at requested slant range(s).
2. Calculate the *balanced environment ratios* at requested slant ranges(s).
3. Calculate the *environment* as a function of range times $4\pi r^2$.
4. Calculate the *balanced environment ratios* of environment times $4\pi r^2$.
5. Calculate the *yield* required to achieve a specified environment as a function of range.
6. Calculate the *slant range* for a specified environment as a function of yield.
7. Calculate the *slant range* to an environment as a function of threat weapon height of burst.
8. Calculate the *balanced environment ratio* as a function of threat weapon height of burst.
9. Calculate the *environment* as a function of air density.
10. Calculate the *most restrictive range* for a set of environments as a function of threat weapon height of burst.
11. Calculate the *environment at specific angles* from the burst in the altitude-range plane.
12. Calculate the *balanced environment ratio at specific angles* from the burst in the altitude-range plane.
13. Calculate the *slant range* for a set of balanced environments as a function of yield.
14. Calculate the *environment* along a line.
15. Calculate the *balanced environment* along a line.

7.2 Environments

GENFIT ***shall*** calculate the radiation environments at the receiver weapon based on the selection of a threat weapon, threat weapon yield, dose type, and response function that are defined in GENFIT.

GENFIT ***shall*** :

1. Calculate *dose* from prompt radiation from any combination of neutrons, secondary gammas, and gammas. Dose calculation:
    a. Calculate neutron fluence. (neutrons/cm^2)
    b. Calculate energy fluence (MeV/cm^2)
    c. Calculate 1 MeV equivalent fluence (neutrons/cm^2)
2. Calculate *dose from X-rays*. Calculate *forward energy* or *total energy* (cal/cm^2)
3. Calculate *dose rate* from prompt radiation. Calculate from any combination of neutrons, secondary gammas, and gammas.
    a. Calculate *neutron fluence rate* (neutrons/cm^2/s)
    b. Calculate *energy fluence rate* (MeV/cm^2/s)
    c. Calculate 1 MeV equivalent *fluence rate* (neutrons/cm^2/s)
4. Calculate *dose rate* from X-rays. Calculate forward energy or total energy (cal/cm^2/s)
5. Calculate *ionizing dose* in silicon from prompt radiation. Calculate from any combination of neutrons, secondary gammas, and gammas. (Rad-Si)
6. Calculate *ionizing dose* in silicon from X-rays. (Rad-Si)
7. Calculate *ionizing dose* in silicon from X-rays and all prompt radiation (neutrons, secondary gamma, and gamma). (Rad-Si)
8. Calculate *ionizing dose* in silicon from prompt radiation. Calculate from primary gamma. (Rad-Si/s)
9. Calculate *ionizing dose rate* in silicon from X-rays. (Rad-Si/s)
10. Calculate *ionizing dose rate* in silicon from X-rays and prompt radiation (primary gamma). (Rad-Si/s)
11. Calculate *ionizing dose rate slope* (ionizing dose double dot) from prompt radiation. Calculate from primary gamma. (Rad-Si/s^2)
12. Calculate *ionizing dose rate slope* (ionizing dose double dot) from X-rays. (Rad-Si/s^2)
13. Calculate *ionizing dose rate slope* (ionizing dose double dot) from X-rays and prompt radiation (primary gamma). (Rad-Si/s^2)
14. Calculate the peak neutron flux (neutrons/cm^2/s)
15. Calculate the peak 1-MeV equivalent neutron flux (neutrons/cm^2/s)

7.3 Normalized Environment Data

GENFIT ***shall*** perform all calculations for threat weapons defined in GENFIT by using the normalized environment data stored in the dose.dat, fdose.dat, xdose.dat and xddose.dat files. GENFIT ***shall*** :

1. Calculate *environments* for *prompt environments* from neutrons, secondary gammas, and prompt gammas for pre-calculated weapons in dose.dat.
2. Calculate *environments* for *fission heating* for calculated weapons in fdose.dat.
3. Calculate *environments* for *total X-ray dose* for calculated weapons in xdose.dat.
4. Calculate *environments* for *total X-ray dose rate* for calculated weapons in xddose.dat

7.4 X-ray Shielding

GENFIT shall calculate *X-ray environments* considering X-ray shielding from a database of materials.