DOE/NV/25946–2548

XSim: A Radiological Data Simulation Architecture

Eric Wagner
July 31, 2015

Remote Sensing Laboratory
Operated by National Security Technologies, LLC

This manuscript has been authored by National Security Technologies, LLC, under Contract No. DE-AC52-06NA25946 with the U.S. Department of Energy

DISCLAIMER

Contents

Chapter 1

Components

The XSim architecture consists of several components. The first is a freely released XML file format in the form of an XML Schema Definition (XSD) file attached in Appendix A. Simulation files compliant with this format shall reproduce coherent values on any platform supporting the XSim standard. XSimCore is C#/.NET code for a reference library released under an open source license capable of interpreting the XML simulation files given proper authentication and producing measurement values for given location and times.

The XML file consists of a series of required objects and many optional objects. The basic structure of the file securely provides for three levels of privileges and administratively for four levels. These are:

1. Field level: Shall provide an end-user with instrument values when an interpreting code is given hardware generated location and time.
2. Oracle level: Shall report instrument values for any user provided location and time.
3. Observer level: Shall provide read-only access to all the scenario data including meta-data not required for generating specific instrument values.
4. Author level: Shall provide read-write access to all the scenario data including the capability to issue updates trusted by field units.

1.1 Field Application

The file format is designed to minimize the effort required for a field application in producing data. When writing software for this purpose the steps required to be followed include:

1. Obtain the current scenario file based on autoupdate settings and location
2. Validate and apply any supplemental scenario files
3. Authenticate a user with field data access rights or better
4. Decrypt the field data if required

5. At least every minute check if the current location and time corresponds to an exercise inject. If so present the inject to the user.
6. In order to calculate a scalar instrument's value:
   (a) Iterate over the radiation source terms
      i. Determine the source activity at the current location
      ii. Apply the instrument's efficiency factor for that source term
      iii. Apply any time varying adjustments for the source term
      iv. Check for any current weather events and apply a shielding factor for water attenuation
   (b) Adjust the value for dead time if the parameter is present
   (c) Adjust the value for percent bias if the parameter is present
   (d) Adjust the value for gaussian error if the parameter is present
   (e) Adjust the value for linear error if the parameter is present
   (f) Adjust the value for any temporal values if the parameter is present
   (g) Limit the measurement to the instrument's maximum or minimum value
   (h) If auto ranging is turned on, adjust to provide a value between 1 and 1000 and apply a suitable metric prefix to the units
   (i) Append the units to the value
   (j) Report the value with units to the user
7. Check for scenario updates at the frequency suggested by the current parameter

1.2 XSim Base XML

In order to complete the specified steps several data objects are stored in an XML file specified by the XSD listed in Appendix A. Data for the secured portions are encrypted with the Advanced Encryption Standard (AES) cipher using a 128 bit key that can be deciphered by users with the appropriate credentials. These credentials may consist of a shared password, user specific password, user specific certificates or none at all. Files with no authentication can be built with no actual encryption although this would allow players to view the entire scenario at any time, in effect giving author level privileges to everyone. The XSim standard includes two main file types, a base file and a delta file. If a simulation needs to be updated from the base file an author can have a delta file distributed and applied to it. This delta file will only include the XML objects that have been changed, removed or inserted.

All strings that have the potential to be displayed to the user can have localized versions stored in the file simultaneously. The procedures specified in RFC 4647[7] will be followed in matching the current locale to the localized text stored in the file. When no match is available the first listed entry shall be utilized.

All encrypted blocks of XML data shall redeclare utilized namespaces. Existing namespaces are not inherited from the parent file.

What follows is a description of the XSim XML objects:

1.2.1 Simulation File (SimFile)

The base file. This can contain a base file, multiple delta files and a digital signature for each element.

1.2.2 Base File (SimFileCore)

The core file that forms the basis for every simulation. It includes in an unencrypted form at least:

- File Format Version - The version of the base XSD file utilized to generate this file
- File Minimum Version - The minimum version of the XSD that can meaningfully parse the contents of this file
- Scenario Title - Short title of the scenario
- Description - Short description of the scenario
- Author - Name of the primary author
- Version - The current version of this file covering a specific event. Should monotonically increase as modifications are made, all successive version shall have a greater version number than the prior version
- Universally Unique IDentifier (UUID) - A unique serial number[4] that designates the event covered
- Start Time - No values, data, nor injects are presented before this time
- Stop Time - No values, data, nor injects are presented after this time
- Replacement Only With Trusted - Boolean value indicating that successive versions of the file shall have the same UUID, a higher version number, and a valid digital signature
- Require Revocation Checks - Boolean value indicating the current validity of a certificate must be confirmed before accepting it as valid
- Root Certificates - Certificates that the signature on all signed delta files must be traced, to be considered valid. The certificate is not required to be a certificate authority
- Intermediate Certificates - List of certificates utilized in trust chaining
- Splash Screen - A localized image or multimedia file to present when a user loads this scenario
- Online Updates - A Uniform Resource Locator (URL) pointing to the location of the current version of the simulation file. The latest file will be at URL/current. A delta file between a version number and the current file will be at URL/versionNumber. The frequency of automatic polling of the URL is also specified.
- File Encryption - A boolean flag indicating encryption is utilized within the file.
- User List - List of individuals and their associated permission levels for accessing the remainder of the file

- Field Data - An XML object representing the data required to generate measurement values in the field. May be encrypted
- Observer Data - An XML object representing meta data incorporated into the scenario by the designers. **Note:** This data block can be stripped with no effect on the field data production. May be encrypted
- Author Data - An encrypted private key corresponding to the first of the root certificates and intermediate password keys to allow changing of encrypted data without changing user passwords **Note:** This data block can be stripped with no effect on the field data production.
- Embedded Files - Files, such as Geographic Information System (GIS), multimedia, etc that are embedded into the scenario

1.2.3 Delta File (SimFileDelta)

A supplemental file to the base file that can be utilized to issue changes without requiring users to download the entire simulation file again.

- UUID - The unique id for the file this delta applies
- Prior Version - The version of the file this delta will be applied to for changes
- New Version - The resultant version the file will be after this delta is applied
- Deleted Objects - A list of references to XML objects to delete
- Unencrypted Content - A list of new and replacement nodes for unencrypted sections of the simulation
- Encrypted Field Content - A list of new and replacement nodes for encrypted sections of the field data. Content is encrypted to the prior version's field key
- Encrypted Author Content - A list of new and replacement nodes for encrypted sections of observer data. Content is encrypted to the prior version's author key

1.3 XSim Field XML

The data included in this section is usually ZLIB compressed[3] (with the required two header bytes), and encrypted with AES-128 Cipher Block Chaining (CBC). The randomized Initial Vector (IV) has the cipher text concatenated on to it and the byte stream is stored in the CipherText element.

1.3.1 Field Data (FieldDataType)

This object contains all the data required for producing data points at any latitude and longitude. It consists of a choice of either:

- PlainText - The unencrypted Field Data Contents object specified below
- CipherText - The AES-128 CBC encrypted data stream holding the Field Data Contents object as the root object. The key is generated from data in the user list specified in the Base File.

1.3.2 Field Data (FieldDataContents)

This object contains all the required data to produce quantitative measurements for instruments in a scenario. Each field can be repeated multiple times so the standard supports multiple instruments, radiation sources, etc.

- Map Layer - GIS layers that would be available to field units. For example a simulated evacuation zone. **Note:** Client interpreters are not required to parse this field so exercise critical information should not be stored in these maps. Consider to be supplemental information only
- Specific Instrument - Represents a list of specific instruments forming pointing to a generic instrument that can generate data for it
- Instrument - Represents a list of instrument types
- Radiation Source - Representing a list of radiation source types to be detected
- Field Team - Represents a list of field teams consisting of multiple people and instruments
- Inject - A list of injects to event participants. Injects are location and time sensitive
- Weather - A list of weather events
- Field Files - A collection of embedded files only accessible to users with the field permission or better

1.3.3 Map Layer (mapType)

For the prior objects that included GIS layers this specifies the values incorporated.

- Visible Name - A list of localized strings tagged with their Internet Engineering Task Force (IETF) language tag
- Layer Order - A positive integer indicating the layer order of the maps. One is the base layer and higher numbers are layered above it
- Local File Ref - A reference to a file embedded in the scenario file
- Web Address - A URL that indicates a location where a layer file is stored
- Linear μ - A polynomial equation that indicates the photon attenuation coefficient in m^{-1} over the energy domain in keV. Solid regions on the map between the current location and a source have this shielding coefficient applied. Implementing this feature is optional as of this version of the standard.

1.3.4 Specific Instrument (instrumentLink)

This object links unique instruments to instruments templates or models.

- Instrument Type Ref - A reference to an Instrument Type object
- Serial Number - A string representing the serial number
- GUID - A Globally Unique Identifier (GUID) reference to the specified instrument
- Performance - A time dependent floating point value that scales the instrument's response from that of the generic instrument

1.3.5 Generic Instrument Type (instrumentType)

This object includes the specification of an instrument utilized in the scenario. There are two types: spectral or scalar. Both types have the following in common:

- Visible Name - Localized name of the instrument type
- Manufacturer - A string representing the manufacturer
- Model - A string representing the model of the instrument
- Percent Error - An error term to be added to the true measurement. The error term's value is randomly selected from a flat distribution between the negative of the original value and the original value, both multiplied by the specified percent error
- Percent Bias - An error term to be added to the true measurement of a fixed percentage of the original value
- Gaussian Error σ - Specifies a polynomial equation to determine the value of σ from the measurement value. The error term's value is randomly selected from the gaussian normal distribution with the calculated σ and centered on zero
- Dead Time - A polynomial equation ranging from 0 to 1 for the dead time over the count rate domain
- Graphical User Interface (GUI) Display - A reference to an Instrument Skin object which is an embedded file containing a zipped archive of a HyperText Markup Language version 5 (HTML5) application driven by JavaScript. The host application will set the HTML5 environment's JavaScript variables: *value* (the scalar measurement value), *latitude* (the current latitude), *longitude* (the current longitude), *measTime* (the Global Positioning System (GPS) time of measurement) and *spectra* (an array of channel values)

The Scalar Instrument has the following additional properties:

- Max Value - A floating point maximum
- Min Value - A floating point minimum
- Units - A localized string representing the base units

- Auto Scale - A boolean that if true requires the field app to place the value between 1 and 1000 with the appropriate metric prefix

The Spectral Instrument has the following properties:

- Energy Calibration - A polynomial equation ranging over the energy in keV and the channel domain
- Minimum Count Time - The minimum number of seconds to collect a measurement
- Maximum Count Time - The maximum number of seconds to collect a measurement
- Rollover - A boolean indicating if channels should roll over if the maximum count level is exceeded
- Max Counts - The maximum number of counts in a channel
- Number of Channels

1.3.6 Instrument Efficiencies (InstrumentEfficiency)

This object defines an instrument's response to a given source term as a function of distance. For point sources the distance is that between the point source and the simulated detector. For other sources it is the distance between the ground and the simulated detector.

- Instrument Reference - A pointer to the instrument this efficiency object corresponds
- Efficiency Curves - Either a polynomial equation or a list of interpolated values. This standard only requires support for linear interpolation (order=1). Positive values of the order represent polynomial curve fits and negative values represent spline fits for interpolation.

1.3.7 Radiation Source (radSrcType)

The radiation source object represents the source driving all the data presented in the simulation

- Airborne - A boolean value indicating if this material is airborne
- Area of Effect - A region outside of which this source in undetectable. It can be a cuboid, spherical, conic or polynomial region.
- Spatial Distribution - The source can be one of four sub-types: point, moving point, grid, or position independent
- Resuspension - The resupsension parameters for this source based on constant, National Council on Radiation Protection (NCRP)[6], Anspaugh[1], modified Anspaugh[2], or Maxwell-Anspaugh[5] models
- Instrument Response - An array of efficiency objects for each instrument that can respond to this source

The sub-types of sources have the following additional parameters.

- The point source has the following properties:
  - Location - The Latitude, Longitude, and Altitude in World Geodetic System 1984 (WGS84) coordinates as a function of time representing where the source is located. This value can optionally be derived from an external URL. The contents obtained from the URL shall be Unicode Transformation Format, 8 bit blocks (UTF-8) encoded comma delimited text of the latitude, longitude and elevation with respect to the WGS84 reference frame
  - Activity - A time indexed array of the source activity in Bq
- The grid source consists of a rectilinear grid of locations and the value at those locations:
  - Bounds - A cuboid region bounding the grid
  - UTMZone - A positive integer representing the Universal Transverse Mercator (UTM) zone the coordinates are specified. If zero the coordinates are in latitude and longitude in the WGS84 projection. UTM coordinates in the southern hemisphere are represented with a negative zone number
  - xCoordinates - The coordinates must cover the entire region specified in the bounds
  - yCoordinates - The coordinates must cover the entire region specified in the bounds
  - interpolation order - The fit to be performed to calculate values between grid points. Positive values indicate the order of the fitting polynomial, negative values indicate the order of a spline. Currently only the value 1 is required to be supported.
  - Distribution - A list of time indexed 2-D arrays of activity levels in Bq/m^2. Each array can incorporate sub grid source(s) for higher resolution of a particular area
- The position independent source is for an overlaying level of radioactivity that has no dependence on the geographical location. Primarily used for background. It can be a constant value, randomized over a fixed range with a linear or normal probability distribution.
  - Type - The type of activity variation of the background level. It can be: Constant, Randomized Linear, and Randomized Gaussian.
  - ArealActivity - The time dependent activity in units of Bq/m^2.
  - Randomization Magnitude - For linear distributions values will be linearly distributed $\pm$ this percentage. For Gaussian distributions the values will be normally distributed from the areal activity with σ defined as this percentage of the value.

1.3.8 Inject (InjectType)

Injects provide for the opportunity for controllers to provide supplemental data to steer the course of an exercise. This specification allows for injects to be specified by location and/or temporal bounds.

- Title - Localized text to display for the subject of the inject
- Inject Time - The time at which an inject should be presented to an exercise player

- Late Injection - A boolean value. If true and a new inject is delivered as a delta update covering the location but occurs in the past the inject shall be presented immediately. Otherwise it is discarded.
- Override Level - If multiple injects occur at the same time and location only the one with the highest override level expressed as a positive integer are displayed.
- Authorization Level - Allows the author to reserve some injects for controllers only
- Autoplay Multimedia - Boolean if true the associated multimedia shall be presented immediately to the exercise participant
- File - Optional reference to a localized embedded file
- Source Person - Reference to a simulated source person
- Location - A region where an inject will be delivered

1.3.9 Field Team (FieldTeamType)

This object represents a simulated field team for generating controller injected data.

- Name - Name of the field team
- Team Leader - Reference to a simulated person object
- Team Members - References to several simulated person objects
- Instruments - References to several instrument objects

1.3.10 Weather Events (WeatherEvents)

This object represents weather events that may have an effect on the scenario, namely rain.

- Type - Localized string describing the event
- Region - Where the weather occurs
- Override Level - If multiple weather events occur at the same time and location only the one with the highest override level expressed as a positive integer are displayed
- Start Time - When the weather event starts
- Stop Time - When the weather event ends
- Precipitation Rate - Float value representing the cm/hr of rain

1.3.11 Simulated Person (simPerson)

- First Name
- Last Name
- Phone Number
- Email Address
- User Photo - Reference to an embedded file

1.3.12 Author Data (authorDataContents)

An object containing planning notes, meta data, and/or methodologies utilized in the scenario design that are not disclosed to participants in an event.

- Notes - Collection of random notes on the scenario
- Map Layers - GIS layers for use by the authors. For example, contours of the various Protective Action Guides (PAG) levels.
- Instrument Meta Data
- Source Meta Data
- Field Team Meta Data
- Author File - A collection of files only accessible to accounts with the author or observer level

1.3.13 Instrument Skin (instrumentDisplaySkin)

A customized instrument display represented by a ZIP compressed HTML5 archive animated by JavaScript. The host application will set the environment's following JavaScript variables once a second: *value* (the scalar measurement value), *latitude* (the current latitude), *longitude* (the current longitude), *measTime* (the GMT time of measurement), and *spectra* (an array of channels).

1.3.14 Embedded File (embeddedFile)

This object contains a file for use by the simulation embedded into it.

- File Name
- Multipurpose Internet Mail Extensions (MIME) type
- data

1.3.15 Resuspension

The current version supports five types of resuspension

Constant Represents a fixed value for resuspension based on the measurement location's climate

- Dry - floating point value in m^{-1} for arid conditions
- Wet - floating point value in m^{-1} for wet conditions

Anspaugh Based on the formulas originally presented by Anspaugh [1] and shown in Equation 1.1 with t measured in days from deposition.

- Start Time - The GPS time at which the original deposition was made

$$S_f(t) = 10^{-4}e^{-0.15\sqrt{t}} + 10^{-9} \tag{1.1}$$

Modified Anspaugh Based on modified formulas presented by Anspaugh [2] shown in Equation 1.2 with t measured in days from deposition.

- Start Time - The GPS time at which the original deposition was made

$$S_f(t) = (10^{-5}e^{-0.07t} + 6 \times 10^{-9}e^{-0.003t} + 10^{-9}) \times 10^{\pm 1} \tag{1.2}$$

Maxwell-Anspaugh Based on modified formula presented by Maxwell [5] shown in Equation 1.3 with t measured in days.

- Start Time - The GPS time at which the original deposition was made

$$S_f(t) = (10^{-5}e^{-0.07t} + 7 \times 10^{-9}e^{-0.002t} + 10^{-9}) \times 4.2^{\pm 1} \tag{1.3}$$

NCRP Based on formula recommended by the U.S. NCRP [6] shown in Equation 1.4 with t measured in days.

- Start Time - The GPS time at which the original deposition was made

$$S_f(t) = \begin{cases} 10^{-6} & t < 1 \text{ d} \\ \frac{10^{-6}}{t} & 1 \text{ d} < t < 1,000 \text{ d} \\ 10^{-9} & t > 1,000 \text{ d} \end{cases} \tag{1.4}$$

1.3.16 Regions

Several objects include references to geographic regions. There are four region types: Cuboid, Poly, Sphere and Conic. They contain combinations of location pairs and triplets; both include latitude and longitude on the WGS84 projection. The triplets also include altitude in meters above the WGS84 ellipsoid.

Cuboid Region

- Bottom South West Corner - A location triplet
- Top North East Corner - A location triplet

Poly Region

- Altitude Minimum - Minimum height above the WGS84 ellipsoid
- Altitude Maximum - Maximum height above the WGS84 ellipsoid
- List of Points - Each point is a location pair and the points are specified in counter clockwise orientation. A minimum of 3 points is required.

Sphere

- Center - A location triplet
- Radius - In meters

Conic Region This is a cone with the apex at the center of the Earth. The axis is along a line from the vertex to the axis location pair. FIXME need another parameters

- Axis - A location pair defining the axis of the cone
- Altitude Minimum - Minimum height above the WGS84 ellipsoid
- Altitude Maximum - Maximum height above the WGS84 ellipsoid

1.3.17 Equations

Several equation types are defined and repeatedly referenced.

Polynomial Equation Consists of the sum of a list Terms, each consisting of

- Coefficient - Floating point value the raised argument is multiplied by
- Exponent - Floating point value the argument is raised

Time Dependent Float An equation to produce a scalar floating point value varying with time. The time dependent float can either be specified as a list of floating point values tagged with times (TemporalInterpolatedFloat) where the result is a linear interpolation between neighboring points, or calculated from the polynomial equation tagged with the closest time before the current time (equation domain is in seconds from the tagged time). For times prior to the first time or after the last time point the produced value is undefined.

Time Dependent Float Array An equation to produce a vector of floating point values that vary with time. The time dependent float can either be specified as a list of floating point arrays tagged with times (TemporalInterpolatedFloatArray) where the result is a linear interpolation between neighboring points, or calculated from the array of polynomial equations tagged with the closest time before the current time (equation domain is in seconds from the tagged time). For times prior to the first time or after the last time point the produced value is undefined.

1.4 Oracle objects

Objects at the Oracle level are in a separate encrypted block and only include data required to build the parameters for the field data. None of these objects are required in order to generate field data.

1.4.1 Observer Data Objects

- Author Notes - Notes on the creation of the scenario that are not disclosed to participants in the event but as a reference on the methodologies utilized in the scenario design
- Map Layers - Map objects to assist in the scenario creation but not disclosed to participants
- Instrument Metadata - Instrument metadata objects that correspond to the previously defined instruments
- Radiation Source Metadata - Radiation source metadata objects that correspond to the previously defined radiation sources
- Field Team Metadata - Field team metadata objects that correspond to the previously defined field teams
- Author Files - Miscellaneous files

1.4.2 Instrument Meta Type (InstrumentMetaType)

The contents of this object are only visible to those with Observer or Author access to the scenario. It incorporates data for generating efficiency models for the various source terms. An instrument can be sensitive to multiple types of radiation simultaneously. All energy values are in units of keV.

- Field Instrument - Reference to an instrument object
- Alpha Efficiency - A polynomial equation covering the energy domain of alpha particles
- Alpha Minimum Energy - Lower bound on α energies to apply the efficiency curve
- Alpha Max Energy - Upper bound on α energies to apply the efficiency curve
- Beta Efficiency - A polynomial equation covering the energy domain of electrons
- Beta Minimum Energy - Lower bound on β energies to apply the efficiency curve
- Beta Maximum Energy - Upper bound on β energies to apply the efficiency curve
- Gamma Efficiency - A polynomial equation covering the energy domain of photons
- Gamma Minimum Energy - Lower bound on γ energies to apply the efficiency curve
- Gamma Maximum Energy - Upper bound on γ energies to apply the efficiency curve
- Exposure Meter - A boolean indicating its an exposure rate meter. Utilizing the Berger calculation to generate an exposure rate measurement in addition to the other specified efficiencies

1.4.3 Spectral Instrument Meta Type (SpectralInstrumentMetaType)

The contents of this object are only visible to those with Observer or Author access to the scenario. It incorporates data for generating model spectra. All energy values are in units of keV.

- Peak LET Tail - A polynomial equation describing the low energy tail of a photo-peak
- Peak Full Width Half Maximum (FWHM) - A polynomial equation describing the resolution of a photo-peak
- Generate Compton - A boolean flag indicating if a Compton continuum should be generated as part of the spectra
- Peak to Compton Ratio - A floating point value relating the height of the peak to the height of the continuum as specified in IEEE 325
- Cs-137 Ratio Type - Utilize a Cs-137 ratio vice Co-60 for the Peak to Compton Ratio

1.4.4 Field Team Meta Data (fieldTeamMetaType)

The contents of this object are only visible to those with Observer or Author access to the scenario.

- Field Team - Reference to the field team this object applies
- Notes

1.4.5 Inject Metadata (injectMetaType)

The contents of this object are only visible to those with Observer or Author access to the scenario.

- Inject - Reference to the parent Inject object
- Notes

1.4.6 Radiation Source Metadata (radSrcMetaType)

The contents of this object are only visible to those with Observer or Author access to the scenario. It incorporates data for generating radiation spectra for calculating efficiency models.

- Visible Name - A localized name to present to users
- Comments - General comments
- Field Source - A reference to an object of type Radiation Source
- Mix - A reference to the radionuclide mix utilized by this source.

1.4.7 Radionuclide Mix (radionuclideMix)

The contents of this object are only visible to those with Observer or Author access to the scenario. It incorporates a listing of all the isotopes that form a mix

- Visible Name - A localized name to present to users
- Isotope - A list of objects of the type Isotope

1.4.8 Isotope (isotopeType)

The contents of this object are only visible to those with Observer or Author access to the scenario. It incorporates a listing of all the isotopes available for radionuclide mixes.

- A - The atomic mass
- Z - The atomic number
- m - Boolean indicating if its metastable
- abundance - Floating point number indicating the relative abundance of the isotope within a mix base on activity. The sum is not required to be normalized

1.5 Author Object

The author object is solely for the purpose of allowing a cryptographically secure modification of an existing scenario. It contains:

- Author Certificate Private Key - The private key corresponding to the first trustOnlyRoot certificate.
- User Password Keys - A list of keys for those users utilizing passwords. Allows for changing of the encryption keys for the field data without knowledge of user passwords

Chapter 2

XSimCore C# Library Specifications

A reference library based on the .NET libraries has been created with the interface document provided in Appendix B. It consists of nine functions to call and the client must provide eight callback functions. This library only concerns itself with the field data and makes no attempt to access the observer or author data.

2.1 Library Functions

2.1.1 LoadScenario

Prototype:

Error LoadScenario(System.IO.Stream baseFile, String ietfLanguage, XSimClient callbacks)

Description: This function should be called prior to all other functions. The reference library does not support incremental files at this time.

Parameters:

- baseFile: A stream currently pointing to the start of an XML text data stream containing the file
- ietfLanguage: The IETF language code for localization efforts. For example "en-us" for US English
- callbacks: An object that contains all the required function callbacks

Returns: The function returns an error code of *noErr* upon success.

2.1.2 AddSupplementalFile

Prototype:

Error AddSupplementalFile(System.IO.Stream supplementalFile)

Description: Future placeholder to allow for supplemental file support. Currently returns an *unimplemented* error code.

Parameters:

- supplementalFile - A stream currently pointing to the start of an XML text data stream containing the supplemental contents

Returns: The function returns an error code of *noErr* upon success.

2.1.3 GetUserNames

Prototype:

String [] GetUserNames()

Description: Call this function to retrieve a list of user names that are currently stored in the scenario file as authorized users.

Returns: The function returns an array of username strings.

2.1.4 SetCurrentUser

Prototype:

Error SetCurrentUser(String aUser)

Description: Call this function to trigger the decryption of the file contents. This can only be called once between loading a scenario file and closing the library.

Parameters:

- aUser - Provide a username equivalent to one returned from the *GetUserNames* function call

Returns: The function returns an error code of *noErr* upon success.

2.1.5 HeartBeat

Prototype:

IDictionary<Instrument, Object> HeartBeat(**double** latitude, **double** longitude, **double** altitude, **double** height, DateTime gpsTime, **bool** generateDictionary)

Description: Call this function at least once a minute, preferably once a second. It will prompt the library to calculate instrument readings and check for injects. Given the resolution of the simulation file is on a per second basis, calling it more than once a second will only alter the noise values.

- latitude - The WGS84 based latitude
- longitude - The WGS84 based longitude
- altitude - The WGS84 based altitude
- height - The height in meters from the ground
- gpsTime - The GPS time provided for the Greenwich Mean Time (GMT) time zone. Note this is currently 17 seconds off of Coordinated Universal Time (UTC)
- generateDictionary - Set this boolean flag to prompt the library to return a dictionary object of all instruments interest has been registered with and their values

Returns: The function returns either *null* or a dictionary object containing instruments and their values.

2.1.6 RegisterInstrumentInterest

Prototype:

Error RegisterInstrumentInterest(Instrument anInst, **bool** interested)

Description: Given that some instruments may take a significant effort to calculate values on mobile devices, values are only calculated for instruments that the client has expressed a current interest. This function can set the interest level for specified instruments.

- anInst - An instrument object generated by the library
- interested - The boolean flag specifying if data should be calculated for that instrument during the *HeartBeat* calls

Returns: The function returns an error code of *noErr* upon success.

2.1.7 GetUniqueInstruments

Prototype:

Instrument[] GetUniqueInstruments()

Description: Unique instruments are those that have serial numbers, possibly customized efficiency values, and GUIDs. Each one is based on a generic instrument.

Returns: The function returns all the current unique instruments in the scenario.

2.1.8 GetGenericInstruments

Prototype:

Instrument[] GetGenericInstruments()

Description: Generic Instruments are instrument models specified in the scenario. For example and ADM-300 would be a generic instrument whereas an ADM-300 with S/N 1234 is a unique instrument.

Returns: The function returns all the current generic instruments in the scenario.

2.1.9 CloseLibrary

Prototype:

void CloseLibrary()

Description: Call this function to release all the resources currently utilized by the library. Calling this from inside a call back currently has undefined behavior.

2.2 Callbacks

The library's primary feedback mechanism is via callbacks. The client program must implement the callbacks for full functionality.

2.2.1 ScalarMeasurement

Prototype:

void ScalarMeasurement(Instrument anInst, **double** latitude, **double** longitude, **double** altitude, **double** height, System.DateTime aTime, **float** aValue, String units)

Description: This function will be called when a scalar measurement is available.

- anInst - The instrument the scalar value was generated from for the current heartbeat call.
- latitude - The WGS84 based latitude
- longitude - The WGS84 based longitude
- altitude - The WGS84 based altitude
- height - The height in meters from the ground
- aTime - The GPS time provided for the GMT time zone. Note this is currently 17 seconds off of UTC

- aValue - The scalar value to present to the user
- units - A string for the units to present to the user

2.2.2 SpectralMeasurement

Prototype:

void SpectralMeasurement(Instrument anInst, **double** latitude, **double** longitude, **double** altitude, **double** height, DateTime aTime, UInt32[] spectra)

Description: Provides the client with a spectrum for the current second in time.

- anInst - The instrument the scalar value was generated from for the current heartbeat call.
- latitude - The WGS84 based latitude
- longitude - The WGS84 based longitude
- altitude - The WGS84 based altitude
- height - The height in meters from the ground
- aTime - The GPS time provided for the GMT time zone. Note this is currently 17 seconds off of UTC
- spectra - An array of channel values to present to the user

2.2.3 InjectNotification

Prototype:

void InjectNotification(String title , **bool** autoPlay, String fileName, String mimeType, System.IO.Stream injectData)

Description: This function is called when an inject for the given time and location is detected

- title - A short string representing the title of the inject
- autoPlay - A flag indicating if the associated media should automatically play (true) or if it should play after user acknowledgement
- fileName - A file name for the media, not necessarily user friendly, may be empty
- mimeType - The MIME type of the data
- injectData - A stream pointing to the beginning of the data and terminating at the end

2.2.4 WeatherUpdate

Prototype:

void WeatherUpdate(**double** latitude, **double** longitude, **double** altitude, String weatherType)

Description:

- latitude - The WGS84 based latitude
- longitude - The WGS84 based longitude
- altitude - The WGS84 based altitude
- weatherType - A user friendly string describing a weather condition that is currently applied to the generated data. For example, heavy rain, dry and dusty, etc.

2.2.5 PasswordRequest

Prototype:

String PasswordRequest(String userName)

Description: This call is made when a user is selected that has a password set.

- userName - The string representing the user for which a password is required

Returns: A string containing the password belonging to this user.

2.2.6 DecryptKeyRequest

Prototype:

Byte[] DecryptKeyRequest(String certIssuer, String certSerialNumber, Byte[] encryptedKey)

Description: This call is made when a user with a client certificate authorization is selected. The decrypted key shall be an AES 128 bit key.

- certIssuer - A string representing the X.500 issuer of the certificate.
- certSerialNumber - A string representing the serial number of the certificate.
- encryptedKey - The byte sequence to be decrypted with the certificate's private key

Returns: A byte array of the decrypted key.

2.2.7 ErrorHandler

Prototype:

Error ErrorHandler(Error errorCode, String errorMessage, **object** parameter)

Description: This callback is initially called when an error is encountered.

- errorCode - The initial error value.
- errorMessage - An unlocalized string describing the encountered error.
- parameter - An optional object of relevance to the error.

Returns: The client shall return *noErr* if it wishes to attempt to ignore the error within the library. The state of the library may be undefined after such a response. If any other value is returned the library shall terminate the thread of execution.

2.2.8 UpdateFile

Prototype:

void UpdateFile(Uri replacementURI, Uri supplementalURI, **bool** defaultUpdate)

Description: When a heartbeat is called after a time interval specified in the scenario to check for an updated scenario file this function will be called.

- replacementURI - The URL for the location of the current version of the scenario
- supplementalURI - The URL for the location of a supplemental file to apply to the current file
- defaultUpdate - A flag indicating that the current simulation file should be updated without user input

For the current implementation given the lack of support for *AddSupplementalFile* the client should close the library and load in the new scenario file.

Appendices

Appendix A

XSim XML Schema Document

Listing A.1: XSim XML Schema Document

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns="http://www.nnss.gov/rsl/sim" xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#" version="1"
      targetNamespace="http://www.nnss.gov/rsl/sim" xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
      xmlns:xenc11="http://www.w3.org/2009/xmlenc11#" xmlns:xsim="http://www.nnss.gov/rsl/sim"
      elementFormDefault="qualified" id="rsl">
  <xs:element name="SimFile">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="BaseFile" type="SimFileCore" minOccurs="0" />
        <xs:element name="SupplementalFile" type="SimFileDelta" minOccurs="0" maxOccurs="unbounded" />
        <xs:element ref="ds:Signature" minOccurs="0" maxOccurs="unbounded" >
          <xs:annotation>
            <xs:documentation>This signature shall apply to either the baseFile or supplementalFile.
                X509 certificates in the signature block will be available as intermediate
                certificates for that signature and may be utilized for future
                signatures.</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="SimFileCore">
    <xs:sequence>
      <xs:element name="FileFormatVersion" type="xs:positiveInteger">
        <xs:annotation>
          <xs:documentation>The version of this xsd utilized for the simulation
              file</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="FileMinVersion" type="xs:positiveInteger">
        <xs:annotation>
          <xs:documentation>The minimum version of this xsd that can be used to process this
              simulation file without error, although some functionality may be missing. Adding xml
              elements to the end of an existing sequence shall not cause parsing
              errors.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Title" maxOccurs="unbounded">
        <xs:complexType>
          <xs:simpleContent>
            <xs:annotation>
              <xs:documentation>The title of the event this simulation file covers. Multiple
                  languages can be specified. If no language match is found the selected language
                  element is undefined.</xs:documentation>
            </xs:annotation>
            <xs:extension base="xs:string" >
              <xs:attribute name="id" type="xs:ID" use="required"/>
              <xs:attribute name="ietfTag" type="xs:string"  default="en-us" />
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="Description" maxOccurs="unbounded">
        <xs:complexType>
          <xs:simpleContent>
            <xs:annotation>
              <xs:documentation>A short description of the event viewable to all</xs:documentation>
```

```
47              </xs:annotation>
48              <xs:extension base="xs:string">
49                <xs:attribute name="id" type="xs:ID" use="required"/>
50                <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
51              </xs:extension>
52            </xs:simpleContent>
53          </xs:complexType>
54        </xs:element>
55        <xs:element name="Author">
56          <xs:complexType>
57            <xs:simpleContent>
58              <xs:annotation>
59                <xs:documentation>The name of the primary author of the simulation
                      event</xs:documentation>
60              </xs:annotation>
61              <xs:extension base="xs:string">
62                <xs:attribute name="id" type="xs:ID" use="required"/>
63              </xs:extension>
64            </xs:simpleContent>
65          </xs:complexType>
66        </xs:element>
67        <xs:element name="SimVersion" type="xs:positiveInteger">
68          <xs:annotation>
69            <xs:documentation>The version number of the files contents. Should monotonically increase
                  with every save of a modified file of the same name.</xs:documentation>
70          </xs:annotation>
71        </xs:element>
72        <xs:element name="SimFileUUID">
73          <xs:annotation>
74            <xs:documentation>A serial number for this scenario. Remains constant throughout version
                  changes. New events should reset this number. Letters must be upper
                  case.</xs:documentation>
75          </xs:annotation>
76          <xs:simpleType>
77            <xs:restriction base="xs:string">
78              <xs:pattern value="[A-F0-9]{8}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{12}"/>
79            </xs:restriction>
80          </xs:simpleType>
81        </xs:element>
82        <xs:element name="ReplaceOnlyWithTrusted">
83          <xs:complexType>
84            <xs:simpleContent>
85              <xs:annotation>
86                <xs:documentation>Only replace this document with one with the same simFileUUID, a
                      greater SimVersion value, and the simFile object is trusted by either the reading
                      application or trustOnly element's certificate.</xs:documentation>
87              </xs:annotation>
88              <xs:extension base="xs:boolean">
89                <xs:attribute name="id" type="xs:ID" use="required" />
90              </xs:extension>
91            </xs:simpleContent>
92          </xs:complexType>
93        </xs:element>
94        <xs:element name="RequireCertificateRevocationChecks">
95          <xs:complexType>
96            <xs:simpleContent>
97              <xs:annotation>
98                <xs:documentation>If set to true a certificate is not considered valid until its
                      revocation status is checked.</xs:documentation>
99              </xs:annotation>
100
101             <xs:extension base="xs:boolean">
102               <xs:attribute name="id" type="xs:ID" use="required" />
103             </xs:extension>
104           </xs:simpleContent>
105         </xs:complexType>
106       </xs:element>
107       <xs:element name="StartTime" type="MarkedTime" >
108         <xs:annotation>
109           <xs:documentation>The starting date and time of the simulated events. No information shall
                  be displayed to a user before this time.</xs:documentation>
110         </xs:annotation>
111       </xs:element>
112       <xs:element name="StopTime" type="MarkedTime" >
113         <xs:annotation>
114           <xs:documentation>The end date and time of the simulated events. No information shall be
                  displayed to a user after this time.</xs:documentation>
115         </xs:annotation>
116       </xs:element>
117       <xs:element name="TrustOnlyRoot" type="MarkedCert" minOccurs="0" maxOccurs="unbounded">
118         <xs:annotation>
119           <xs:documentation>If replaceOnlyWithTrusted is specified the trusted values are
                  implementation specific unless trustOnlyRoot is specified. Then only signatures
                  generated by certificates that can be chained back to one of the trustOnlyRoot
                  certificate authorities are considered valid. These certificates are NOT required to
                  be certificate authorities.</xs:documentation>
120         </xs:annotation>
121       </xs:element>
122       <xs:element name="IntermediateCerts" type="MarkedCert" minOccurs="0" maxOccurs="unbounded">
123         <xs:annotation>
124           <xs:documentation>Intermediate certificates utilized in establishing certificate
                  chains.</xs:documentation>
```

```
125         </xs:annotation>
126       </xs:element>
127       <xs:element name="SplashScreen" minOccurs="0" maxOccurs="unbounded">
128         <xs:annotation>
129           <xs:documentation>An image or multimedia file to present when a user loads this
                scenario.</xs:documentation>
130         </xs:annotation>
131         <xs:complexType>
132           <xs:sequence>
133             <xs:element name="SplashImage" type="xs:IDREF"  />
134           </xs:sequence>
135           <xs:attribute name="id" type="xs:ID" use="required" />
136           <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
137         </xs:complexType>
138       </xs:element>
139       <xs:element name="OnlineUpdates" minOccurs="0">
140         <xs:annotation>
141           <xs:documentation>A location where the current version of the file is located and if
                updates should automatically be installed. The location should specify the location of
                a current full version. The library will first attempt to download a file at the
                location/existingVersion which should be a supplemental file between the
                existingVersion and the current version.</xs:documentation>
142         </xs:annotation>
143         <xs:complexType>
144           <xs:sequence>
145             <xs:element name="Location" type="xs:anyURI" />
146             <xs:element name="AutoUpdate" type="xs:boolean" />
147             <xs:element name="Frequency" type="xs:positiveInteger">
148               <xs:annotation>
149                 <xs:documentation>The number of minutes between attempts to check for an updated
                      file.</xs:documentation>
150               </xs:annotation>
151             </xs:element>
152           </xs:sequence>
153           <xs:attribute name="id" type="xs:ID" use="required" />
154         </xs:complexType>
155       </xs:element>
156       <xs:element name="FileEncryption" type="xs:boolean" minOccurs="0">
157         <xs:annotation>
158           <xs:documentation>If encryption is present indicates the file contents are encrypted. If
                this is the case then the noAuth authentication method is disallowed. Encryption shall
                be AES 128.</xs:documentation>
159         </xs:annotation>
160       </xs:element>
161       <xs:element name="Authorization"  maxOccurs="unbounded">
162         <xs:annotation>
163           <xs:documentation>Listing of users with access to the file contents and keys to decrypt the
                data accessible to them.</xs:documentation>
164         </xs:annotation>
165         <xs:complexType>
166           <xs:sequence>
167             <xs:element name="UserName" type="xs:string" />
168             <xs:element name="Auth" type="Authentication" />
169             <xs:element name="Access" type="AuthLevel" />
170           </xs:sequence>
171           <xs:attribute name="id" type="xs:ID" use="required"/>
172         </xs:complexType>
173       </xs:element>
174       <xs:element name="FieldData" type="FieldDataType" />
175       <xs:element name="ObserverData" type="ObserverDataType" minOccurs="0" />
176       <xs:element name="AuthorData" type="AuthorDataType" minOccurs="0">
177         <xs:annotation>
178           <xs:documentation>This element contains the private key corresponding to the first
                trustOnlyRoot certificate and the keys for password based
                authorizations.</xs:documentation>
179         </xs:annotation>
180       </xs:element>
181       <xs:element name="AttachedFiles" type="EmbeddedFile" minOccurs="0" maxOccurs="unbounded" />
182     </xs:sequence>
183     <xs:attribute name="id" type="xs:ID" use="required"/>
184   </xs:complexType>
185
186   <xs:complexType name="SimFileDelta">
187     <xs:annotation>
188       <xs:documentation>This object represents a delta modification to an earlier sim file. New
            unencrypted data can only replace or added to originally unencrypted data. Encrypted field
            content can only replace or added to originally encrypted field content. Encrypted author
            data can only replace or added to originally encrypted author content.</xs:documentation>
189     </xs:annotation>
190     <xs:sequence>
191       <xs:element name="PriorFileUUID">
192         <xs:simpleType>
193           <xs:restriction base="xs:string">
194             <xs:pattern value="[A-F0-9]{8}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{12}"/>
195           </xs:restriction>
196         </xs:simpleType>
197       </xs:element>
198       <xs:element name="PriorVersion" type="xs:positiveInteger"/>
199       <xs:element name="NewVersion" type="xs:positiveInteger" />
200       <xs:element name="NodeToDelete" type="xs:IDREF" minOccurs="0" maxOccurs="unbounded"/>
201       <xs:element name="UnencryptedContent" type="NewContent" minOccurs="0"/>
202       <xs:element name="EncryptedFieldContent" type="xs:base64Binary" minOccurs="0">
```

```
203         <xs:annotation>
204           <xs:documentation>This element is zlib compressed and encrypted to the pre-existing field
                  key. The decrypted text shall be an unencrypted FieldDataContents
                  node.</xs:documentation>
205         </xs:annotation>
206       </xs:element>
207       <xs:element name="EncryptedAuthorContent" type="xs:base64Binary" minOccurs="0">
208         <xs:annotation>
209           <xs:documentation>This element is zlib compressed and encrypted to the pre-exisiting author
                  key. The decrypted text shall be an unecrypted AuthorDataType node.</xs:documentation>
210         </xs:annotation>
211       </xs:element>
212     </xs:sequence>
213     <xs:attribute name="id" type="xs:ID" use="required"/>
214   </xs:complexType>
215
216   <xs:complexType name="NewContent" >
217     <xs:annotation>
218       <xs:documentation>If encryption is enabled this data block requires the namespaces to be
              respecified. The namespace can't be assumed to be inherited from the parent xml
              file.</xs:documentation>
219     </xs:annotation>
220     <xs:sequence>
221       <xs:element name="NodeToReplace" type="xs:anyType" minOccurs="0" maxOccurs="unbounded">
222         <xs:annotation>
223           <xs:documentation>The replaced node must have the same id value as the one it is
                  replacing</xs:documentation>
224         </xs:annotation>
225       </xs:element>
226       <xs:element name="NodesToInsert" minOccurs="0" maxOccurs="unbounded">
227         <xs:complexType>
228           <xs:sequence>
229             <xs:element name="ParentNode" type="xs:IDREF" />
230             <xs:any processContents="strict" maxOccurs="unbounded">
231               <xs:annotation>
232                 <xs:documentation>A series of nodes to add under the designated
                        parentNode.</xs:documentation>
233               </xs:annotation>
234             </xs:any>
235           </xs:sequence>
236         </xs:complexType>
237       </xs:element>
238     </xs:sequence>
239   </xs:complexType>
240
241   <xs:complexType name="FieldDataContents" >
242     <xs:annotation>
243       <xs:documentation>If encryption is enabled this data block requires the namespaces to be
              respecified. The namespace can't be assumed to be inherited from the parent xml
              file.</xs:documentation>
244     </xs:annotation>
245     <xs:sequence >
246       <xs:element name="MapLayer" type="MapType" minOccurs="0" maxOccurs="unbounded"/>
247       <xs:element name="SpecificInstrument" type="InstrumentLink" minOccurs="0" maxOccurs="unbounded"
              />
248       <xs:element name="InstrumentType" type="InstrumentType" minOccurs="0" maxOccurs="unbounded"/>
249       <xs:element name="RadiationSource" type="RadSrcType" minOccurs="0" maxOccurs="unbounded" />
250       <xs:element name="FieldTeam" type="FieldTeamType" minOccurs="0" maxOccurs="unbounded"/>
251       <xs:element name="Inject" type="InjectType" minOccurs="0" maxOccurs="unbounded"/>
252       <xs:element name="Weather" type="WeatherEvents"  minOccurs="0" maxOccurs="unbounded"/>
253       <xs:element name="FieldFile" type="EmbeddedFile"  minOccurs="0" maxOccurs="unbounded"/>
254     </xs:sequence>
255     <xs:attribute name="id" type="xs:ID" use="required"/>
256   </xs:complexType>
257
258   <xs:complexType name ="FieldDataType">
259     <xs:annotation>
260       <xs:documentation>If CipherText is chosen a fieldDataContents object is compressed with zlib
              and encrypted with AES128 CBC where the first 16 bytes here represent the
              IV.</xs:documentation>
261     </xs:annotation>
262     <xs:choice>
263       <xs:element name="PlainText" type="FieldDataContents" />
264       <xs:element name="CipherText" type="xs:base64Binary" />
265     </xs:choice>
266   </xs:complexType>
267
268   <xs:complexType name="ObserverDataType">
269     <xs:annotation>
270       <xs:documentation>If CipherText is chosen an ObserverDataContents object is compressed with
              zlib and encrypted with AES128 CBC where the first 16 bytes here represent the
              IV.</xs:documentation>
271     </xs:annotation>
272     <xs:choice>
273       <xs:element name="PlainText" type="ObserverDataContents" />
274       <xs:element name="CipherText" type="xs:base64Binary" />
275     </xs:choice>
276   </xs:complexType>
277
278   <xs:complexType name="ObserverDataContents">
279     <xs:sequence>
280       <xs:element name="AuthorNotes" minOccurs="0" maxOccurs="unbounded">
```

```
281           <xs:complexType>
282             <xs:simpleContent>
283               <xs:annotation>
284                 <xs:documentation>Notes on the creation of the scenario that are not disclosed to
                        participants in the event but as a reference on the methodologies utilized in the
                        scenario design.</xs:documentation>
285               </xs:annotation>
286               <xs:extension base="xs:string">
287                 <xs:attribute name="id" type="xs:ID" use="required"/>
288               </xs:extension>
289             </xs:simpleContent>
290           </xs:complexType>
291         </xs:element>
292         <xs:element name="MapLayer" type="MapType" minOccurs="0" maxOccurs="unbounded">
293           <xs:annotation>
294             <xs:documentation>Maps to assist in the scenario creation but not disclosed to
                    participants.</xs:documentation>
295           </xs:annotation>
296         </xs:element>
297         <xs:element name="Instrument" type="InstrumentMetaType" minOccurs="0" maxOccurs="unbounded" />
298         <xs:element name="Source" type="RadSrcMetaType" minOccurs="0" maxOccurs="unbounded" />
299         <xs:element name="FieldTeam" type="FieldTeamMetaType" minOccurs="0" maxOccurs="unbounded" />
300         <xs:element name="AuthorFiles" type="EmbeddedFile" minOccurs="0" maxOccurs="unbounded" />
301         <xs:element name="VersionNotes" maxOccurs="unbounded">
302           <xs:complexType>
303             <xs:sequence>
304               <xs:element name="versionNumber" type="xs:positiveInteger" />
305               <xs:element name="changeNotes" type="xs:string" />
306             </xs:sequence>
307           </xs:complexType>
308         </xs:element>
309       </xs:sequence>
310     </xs:complexType>
311
312     <xs:complexType name="Authentication" abstract="true">
313       <xs:attribute name="id" type="xs:ID" use="required"/>
314     </xs:complexType>
315
316     <xs:complexType name="PasswordHash">
317       <xs:complexContent>
318         <xs:extension base="Authentication">
319           <xs:annotation>
320             <xs:documentation>The salt is concatenated with the user supplied password and hashed with
                    the algorithm. No file encryption is supported.</xs:documentation>
321           </xs:annotation>
322           <xs:sequence>
323             <xs:element name="PasswordSalt" type="xs:string" />
324             <xs:element name="Hash" type="xs:base64Binary" />
325             <xs:element name="Algorithm" type="xs:anyURI"
                      default="http://www.w3.org/2001/04/xmlenc#sha256"/>
326           </xs:sequence>
327         </xs:extension>
328       </xs:complexContent>
329     </xs:complexType>
330
331     <xs:complexType name="NoAuth">
332       <xs:complexContent>
333         <xs:extension base="Authentication">
334           <xs:annotation>
335             <xs:documentation>This method does not require any information from the user in order to
                    access the contents of the file.</xs:documentation>
336           </xs:annotation>
337           <xs:sequence>
338             <xs:element name="FieldKey" type="xs:base64Binary" />
339             <xs:element name="ObserverKey" type="xs:base64Binary" minOccurs="0" />
340             <xs:element name="AuthorKey" type="xs:base64Binary" minOccurs="0" />
341           </xs:sequence>
342         </xs:extension>
343       </xs:complexContent>
344     </xs:complexType>
345
346     <xs:complexType name="IndividualCert">
347       <xs:complexContent>
348         <xs:extension base="Authentication">
349           <xs:annotation>
350             <xs:documentation>As of this version only RSA v1.5 is supported as the
                    EncryptionMethod.</xs:documentation>
351           </xs:annotation>
352           <xs:sequence>
353             <xs:element name="CertName" type="ds:X509IssuerSerialType" />
354             <xs:element name="FieldKey" type="xenc:EncryptedKeyType" />
355             <xs:element name="ObserverKey" type="xenc:EncryptedKeyType" minOccurs="0"/>
356             <xs:element name="AuthorKey" type="xenc:EncryptedKeyType" minOccurs="0"/>
357           </xs:sequence>
358         </xs:extension>
359       </xs:complexContent>
360     </xs:complexType>
361
362     <xs:complexType name="PasswordAuth">
363       <xs:complexContent>
364         <xs:extension base="Authentication">
365           <xs:sequence>
```

```
366             <xs:element name="PasswordKey" type="xenc11:DerivedKeyType">
367               <xs:annotation>
368                 <xs:documentation>This is the key generated by a user supplied password. The Algorithm
                        SHALL be pkcs-5#pbkdf2 in this version.</xs:documentation>
369               </xs:annotation>
370             </xs:element>
371             <xs:element name="FieldKey" type="xenc:EncryptedKeyType" >
372               <xs:annotation>
373                 <xs:documentation>This is a key encrypted by the passwordKey. When decrypted it can be
                        used to decrypt the field contents. The Algorithm SHALL be kw-aes128 in this
                        version.</xs:documentation>
374               </xs:annotation>
375             </xs:element>
376             <xs:element name="ObserverKey" type="xenc:EncryptedKeyType" minOccurs="0">
377               <xs:annotation>
378                 <xs:documentation>This is a key encrypted by the passwordKey. When decrypted it can be
                        used to decrypt the authorship values. The algorithm SHALL be kw-aes128 in this
                        version.</xs:documentation>
379               </xs:annotation>
380             </xs:element>
381             <xs:element name="AuthorKey" type="xenc:EncryptedKeyType" minOccurs="0">
382               <xs:annotation>
383                 <xs:documentation>This is a key encrypted by the passwordKey. When decrypted it can be
                        used to decrypt the certificate to digitally sign modifications and access
                        passwords. The algorithm SHALL be kw-aes128 in this version.</xs:documentation>
384               </xs:annotation>
385             </xs:element>
386           </xs:sequence>
387         </xs:extension>
388       </xs:complexContent>
389     </xs:complexType>
390
391     <xs:complexType name="MarkedTime">
392       <xs:simpleContent>
393         <xs:extension base="FormattedDateTime">
394           <xs:attribute name="id" type="xs:ID" use="required"/>
395         </xs:extension>
396       </xs:simpleContent>
397     </xs:complexType>
398
399     <xs:complexType name="MarkedCert">
400       <xs:annotation>
401         <xs:documentation>Base 64 encoded data of a DER formatted X509 Certificate.</xs:documentation>
402       </xs:annotation>
403       <xs:simpleContent>
404         <xs:extension base="xs:base64Binary" >
405           <xs:attribute name="id" type="xs:ID" use="required"/>
406         </xs:extension>
407       </xs:simpleContent>
408     </xs:complexType>
409
410     <xs:complexType name="AuthLevel">
411       <xs:simpleContent>
412         <xs:extension base="UntaggedAuthLevel">
413           <xs:attribute name="id" type="xs:ID" use="required"/>
414         </xs:extension>
415       </xs:simpleContent>
416     </xs:complexType>
417
418     <xs:simpleType name="UntaggedAuthLevel">
419       <xs:restriction base="xs:string">
420         <xs:enumeration value="Field" >
421           <xs:annotation>
422             <xs:documentation>This authorization level will only report data based on hardware
                    generated location and times.</xs:documentation>
423           </xs:annotation>
424         </xs:enumeration>
425         <xs:enumeration value="Oracle" >
426           <xs:annotation>
427             <xs:documentation>This authorization level will only report data for an selected location
                    and time.</xs:documentation>
428           </xs:annotation>
429         </xs:enumeration>
430         <xs:enumeration value="Observer" >
431           <xs:annotation>
432             <xs:documentation>This authorization level will provide full access to all data in the
                    scenario in a read only format.</xs:documentation>
433           </xs:annotation>
434         </xs:enumeration>
435         <xs:enumeration value="Author" >
436           <xs:annotation>
437             <xs:documentation>This authorization level will provide full access to all data and the
                    capability to modify the values.</xs:documentation>
438           </xs:annotation>
439         </xs:enumeration>
440       </xs:restriction>
441     </xs:simpleType>
442
443     <xs:complexType name="MapType">
444       <xs:sequence>
445         <xs:element name="LayerOrder" type="xs:positiveInteger">
446           <xs:annotation>
```

```
447           <xs:documentation>Indicates the layer order of the maps. One is the base layer and higher
                  numbers are layered above it</xs:documentation>
448         </xs:annotation>
449       </xs:element>
450       <xs:element name="VisibleName" maxOccurs="unbounded">
451         <xs:complexType>
452           <xs:simpleContent>
453             <xs:extension base="xs:string">
454               <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
455             </xs:extension>
456           </xs:simpleContent>
457         </xs:complexType>
458       </xs:element>
459       <xs:element name="LocalFile" type="xs:IDREF" minOccurs="0" maxOccurs="unbounded"/>
460       <xs:element name="WebAddress" type="xs:anyURI" minOccurs="0"/>
461       <xs:element name="LinearMu" type="PolynomialEquation" minOccurs="0">
462         <xs:annotation>
463           <xs:documentation>Indicates the attenuation coefficient for photons.</xs:documentation>
464         </xs:annotation>
465       </xs:element>
466     </xs:sequence>
467     <xs:attribute name="id" type="xs:ID" use="required"/>
468   </xs:complexType>
469
470   <xs:complexType name="InstrumentLink">
471     <xs:sequence>
472       <xs:element name="InstrumentRef" type="xs:IDREF" />
473       <xs:element name="SerialNumber" type="xs:string" />
474       <xs:element name="GUIDLink">
475         <xs:simpleType>
476           <xs:restriction base="xs:string">
477             <xs:pattern value="[A-F0-9]{8}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{4}-[A-F0-9]{12}"/>
478           </xs:restriction>
479         </xs:simpleType>
480       </xs:element>
481       <xs:element name="PerformanceEff" type="TimeDependentFloat" minOccurs="0" maxOccurs="unbounded"
              />
482     </xs:sequence>
483   </xs:complexType>
484   <xs:complexType name="InstrumentType" abstract="true">
485     <xs:sequence>
486       <xs:element name="Manufacturer" type="xs:string" />
487       <xs:element name="Model" type="xs:string" />
488       <xs:element name="VisibleName" maxOccurs="unbounded">
489         <xs:complexType>
490           <xs:simpleContent>
491             <xs:extension base="xs:string">
492               <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
493             </xs:extension>
494           </xs:simpleContent>
495         </xs:complexType>
496       </xs:element>
497       <xs:element name="PercentError" type="xs:float" minOccurs="0"  maxOccurs="1" default="0.0"/>
498       <xs:element name="PercentBias" type="xs:float" minOccurs="0"  maxOccurs="1" default="0.0"/>
499       <xs:element name="GaussianErrorSigma" type="PolynomialEquation" minOccurs="0" maxOccurs="1">
500         <xs:annotation>
501           <xs:documentation>The polynomial to calculate sigma is based on the measured value before
                  any noise terms are added.</xs:documentation>
502         </xs:annotation>
503       </xs:element>
504       <xs:element name="DeadTime" type="PolynomialEquation" minOccurs="0" maxOccurs="1">
505         <xs:annotation>
506           <xs:documentation>The calculated value in the parameter and the function provides the dead
                  time from 0 to 1.</xs:documentation>
507         </xs:annotation>
508       </xs:element>
509       <xs:element name="GUIDisplay" type="xs:IDREF" minOccurs="0" maxOccurs="1" >
510         <xs:annotation>
511           <xs:documentation>An instrument display will be a zip archive of a html5 application driven
                  by javascript. The host application will set the environment's javascript variables:
                  value (the scalar measurement value), latitude (the current latitude), longitude (the
                  current longitude), measTime (the GMT time of measurement), and spectra (an array of
                  channels).</xs:documentation>
512         </xs:annotation>
513       </xs:element>
514     </xs:sequence>
515     <xs:attribute name="id" type="xs:ID" use="required"/>
516   </xs:complexType>
517
518   <xs:complexType name="InstrumentMetaType">
519     <xs:sequence>
520       <xs:element name="FieldInstrument" type="xs:IDREF" />
521       <xs:element name="ExposureMeter" type="xs:boolean" />
522       <xs:element name="AlphaEff" type="PolynomialEquation" />
523       <xs:element name="AlphaMinEnergy" type="xs:double" />
524       <xs:element name="AlphaMaxEnergy" type="xs:double" />
525       <xs:element name="BetaEff" type="PolynomialEquation" />
526       <xs:element name="BetaMinEnergy" type="xs:double" />
527       <xs:element name="BetaMaxEnergy" type="xs:double" />
528       <xs:element name="GammaEff" type="PolynomialEquation" />
529       <xs:element name="GammaMinEnergy" type="xs:double" />
530       <xs:element name="GammaMaxEnergy" type="xs:double" />
```

```
531     </xs:sequence>
532     <xs:attribute name="id" type="xs:ID" use="required"/>
533   </xs:complexType>
534
535   <xs:complexType name="SpectralInstrumentMetaType">
536     <xs:complexContent>
537       <xs:extension base="InstrumentMetaType">
538         <xs:sequence>
539           <xs:element name="PeakLETail" type="PolynomialEquation" />
540           <xs:element name="PeakFWHM" type="PolynomialEquation" />
541           <xs:element name="GenerateCompton" type="xs:boolean" />
542           <xs:element name="GeneratePeakTails" type="xs:boolean" />
543           <xs:element name="Peak2ComptonRatio" type="xs:double" />
544           <xs:element name="Cs137RatioType" type="xs:boolean" />
545         </xs:sequence>
546       </xs:extension>
547     </xs:complexContent>
548   </xs:complexType>
549
550   <xs:complexType name="InstrumentEfficiency">
551     <xs:sequence>
552       <xs:element name="InstrumentRef" type="xs:IDREF" />
553       <xs:element name="EffCurves" type="DistanceEfficiency" maxOccurs="unbounded" />
554     </xs:sequence>
555   </xs:complexType>
556
557   <xs:complexType name="DistanceEfficiency" abstract="true">
558     <xs:sequence>
559       <xs:element name="Channel" type="xs:positiveInteger" />
560     </xs:sequence>
561   </xs:complexType>
562
563   <xs:complexType name="AnalyticalEfficiency">
564     <xs:complexContent>
565       <xs:extension base="DistanceEfficiency">
566         <xs:sequence>
567           <xs:element name="EffCurve" type="PolynomialEquation" >
568             <xs:annotation>
569               <xs:documentation>Efficiency for the given energy as distance in meters from the
                      source. 0 kev for scalar instruments.</xs:documentation>
570             </xs:annotation>
571           </xs:element>
572         </xs:sequence>
573       </xs:extension>
574     </xs:complexContent>
575   </xs:complexType>
576
577   <xs:complexType name="InterpolatedEfficiency">
578     <xs:complexContent>
579       <xs:extension base="DistanceEfficiency">
580         <xs:sequence>
581           <xs:element name="Distances" type="DoubleList" />
582           <xs:element name="Efficencies" type="DoubleList" />
583           <xs:element name="Order" type="xs:integer" default="1" />
584         </xs:sequence>
585       </xs:extension>
586     </xs:complexContent>
587   </xs:complexType>
588
589   <xs:complexType name="Resuspension" abstract="true">
590     <xs:attribute name="id" type="xs:ID" use="required" />
591   </xs:complexType>
592
593   <xs:complexType name="ConstantResuspension">
594     <xs:complexContent>
595       <xs:extension base="Resuspension">
596         <xs:sequence>
597           <xs:element name="Dry" type="xs:float" />
598           <xs:element name="Wet" type="xs:float" />
599         </xs:sequence>
600       </xs:extension>
601     </xs:complexContent>
602   </xs:complexType>
603
604   <xs:complexType name="AnspaughResuspension">
605     <xs:complexContent>
606       <xs:extension base="Resuspension">
607         <xs:sequence>
608           <xs:element name="StartTime" type="FormattedDateTime" />
609         </xs:sequence>
610       </xs:extension>
611     </xs:complexContent>
612   </xs:complexType>
613
614   <xs:complexType name="ModifiedAnspaughResuspension">
615     <xs:complexContent>
616       <xs:extension base="Resuspension">
617         <xs:sequence>
618           <xs:element name="StartTime" type="FormattedDateTime" />
619         </xs:sequence>
620       </xs:extension>
621     </xs:complexContent>
```

```
622   </xs:complexType>
623
624   <xs:complexType name="MaxwellAnspaughResuspension">
625     <xs:complexContent>
626       <xs:extension base="Resuspension">
627         <xs:sequence>
628           <xs:element name="StartTime" type="FormattedDateTime" />
629         </xs:sequence>
630       </xs:extension>
631     </xs:complexContent>
632   </xs:complexType>
633
634   <xs:complexType name="NcrpResuspension">
635     <xs:complexContent>
636       <xs:extension base="Resuspension">
637         <xs:sequence>
638           <xs:element name="StartTime" type="FormattedDateTime" />
639         </xs:sequence>
640       </xs:extension>
641     </xs:complexContent>
642   </xs:complexType>
643
644   <xs:complexType name="RadSrcType">
645     <xs:sequence>
646       <xs:element name="Airborne" type="xs:boolean" />
647       <xs:element name="AreaOfEffect" type="GeoRegion" />
648       <xs:element name="SpatialDistribution" type="SourceDistribution" />
649       <xs:element name="Resuspension" type="Resuspension"/>
650     </xs:sequence>
651     <xs:attribute name="id" type="xs:ID" use="required"/>
652   </xs:complexType>
653
654   <xs:complexType name="SourceDistribution" abstract="true" >
655     <xs:sequence>
656       <xs:element name="InstrumentResponse" type="InstrumentEfficiency" maxOccurs="unbounded"/>
657     </xs:sequence>
658     <xs:attribute name="id" type="xs:ID" use="required"/>
659   </xs:complexType>
660
661   <xs:complexType name="RadSrcMetaType">
662     <xs:sequence>
663       <xs:element name="VisibleName" type="xs:string" />
664       <xs:element name="Comments" type="xs:string" />
665       <xs:element name="FieldSource" type="xs:IDREF" />
666       <xs:element name="Mix" type="xs:IDREF" />
667     </xs:sequence>
668     <xs:attribute name="id" type="xs:ID" use="required"/>
669   </xs:complexType>
670
671   <xs:complexType name="TimeDependentMix">
672     <xs:sequence>
673       <xs:element name="TimePoint" type="FormattedDateTime"/>
674       <xs:element name="Mix" type="RadionuclideMix" />
675     </xs:sequence>
676   </xs:complexType>
677
678   <xs:complexType name="RadionuclideMix">
679     <xs:sequence>
680       <xs:element name="MixName" type="xs:string" />
681       <xs:element name="Isotope" type="IsotopeType" maxOccurs="unbounded" />
682     </xs:sequence>
683     <xs:attribute name="id" type="xs:ID" use="required"/>
684   </xs:complexType>
685
686   <xs:complexType name="IsotopeType">
687     <xs:sequence>
688       <xs:element name="A" type="xs:positiveInteger" />
689       <xs:element name="Z" type="xs:positiveInteger" />
690       <xs:element name="m" type="xs:boolean" />
691       <xs:element name="Abundance" type="xs:float" />
692     </xs:sequence>
693     <xs:attribute name="id" type="xs:ID" use="required"/>
694   </xs:complexType>
695
696   <xs:complexType name="PointSource">
697     <xs:complexContent>
698       <xs:extension base="SourceDistribution">
699         <xs:sequence>
700           <xs:element name="Location" type="Location" minOccurs="1" maxOccurs="unbounded">
701             <xs:annotation>
702               <xs:documentation>If only one location is given it is assumed to be constant throughout
                      the event. </xs:documentation>
703             </xs:annotation>
704           </xs:element>
705           <xs:element name="Activity" type="TimeDependentFloat" maxOccurs="unbounded">
706             <xs:annotation>
707               <xs:documentation>Units shall be in Bq. If only one value is given it is assumed to be
                      constant throughout the event.</xs:documentation>
708             </xs:annotation>
709           </xs:element>
710         </xs:sequence>
711       </xs:extension>
```

```
712       </xs:complexContent>
713     </xs:complexType>
714
715     <xs:complexType name="Location" abstract="true">
716       <xs:sequence>
717         <xs:element name="TimePoint" type="FormattedDateTime" />
718       </xs:sequence>
719     </xs:complexType>
720
721     <xs:complexType name="FixedLocation">
722       <xs:complexContent>
723         <xs:extension base="Location">
724           <xs:sequence>
725             <xs:element name="LocationPoint" type="LocationTriplet"/>
726           </xs:sequence>
727         </xs:extension>
728       </xs:complexContent>
729     </xs:complexType>
730
731     <xs:complexType name="HardwareRef">
732       <xs:complexContent>
733         <xs:extension base="Location">
734           <xs:sequence>
735             <xs:element name="Reference" type="xs:string">
736               <xs:annotation>
737                 <xs:documentation>A reference specifing a URL in order to access the current location
                        of a source. The contents of the response shall consist of comma delimeted text
                        of: latitude, longitude, elevation. The latitude and longitude shall be in decimal
                        degrees and the elevation shall be in meters. The reference frames shall be
                        WGS84.</xs:documentation>
738               </xs:annotation>
739             </xs:element>
740           </xs:sequence>
741         </xs:extension>
742       </xs:complexContent>
743     </xs:complexType>
744
745     <xs:complexType name="TimeDependentAnalyticalLocation">
746       <xs:complexContent>
747         <xs:extension base="Location">
748           <xs:annotation>
749             <xs:documentation>The range of the polynomial equations correspond to the name of the
                    element. The domain is seconds from this object's current time
                    point.</xs:documentation>
750           </xs:annotation>
751           <xs:sequence>
752             <xs:element name="Latitude" type="PolynomialEquation" />
753             <xs:element name="Longitude" type="PolynomialEquation" />
754             <xs:element name="Altitude" type="PolynomialEquation" />
755           </xs:sequence>
756         </xs:extension>
757       </xs:complexContent>
758     </xs:complexType>
759
760     <xs:complexType name="GridSource">
761       <xs:complexContent>
762         <xs:extension base="SourceDistribution">
763           <xs:sequence>
764             <xs:element name="Bounds" type="CuboidRegion" />
765             <xs:element name="UTMZone" type="xs:nonNegativeInteger" minOccurs="0" default="0">
766               <xs:annotation>
767                 <xs:documentation>If X and Y Coordinates are in UTM then use this field to specify the
                        zone in which the grid is located. A value of zero indicates the coordinates are
                        in WGS-84 decimal degrees.</xs:documentation>
768               </xs:annotation>
769             </xs:element>
770             <xs:element name="XCoordinates" type="DoubleList" >
771               <xs:annotation>
772                 <xs:documentation>Coordinate values should be WGS-84 longitude values in decimal
                        degrees. Values shall start at the lowest value and continuously increase in
                        value. The grid coordinates shall be inclusive of the entire region specified by
                        the bounds.</xs:documentation>
773               </xs:annotation>
774             </xs:element>
775             <xs:element name="YCoordinates" type="DoubleList" >
776               <xs:annotation>
777                 <xs:documentation>Coordinate values should be WGS-84 longitude values in decimal
                        degrees. Values shall start at the lowest value and continuously increase in
                        value. The grid coordinates shall be inclusive of the entire region specified by
                        the bounds.</xs:documentation>
778               </xs:annotation>
779             </xs:element>
780             <xs:element name="InterpolationOrder" type="xs:positiveInteger" default="1">
781               <xs:annotation>
782                 <xs:documentation>A polynomial will be fit between data points for an interpolated
                        value in both directions. The order must be less than the least number of
                        coordinates in either direction.</xs:documentation>
783               </xs:annotation>
784             </xs:element>
785             <xs:element name="Distribution" type="GridValues" maxOccurs="unbounded">
786               <xs:annotation>
787                 <xs:documentation>If only one time value is given it is presumed to be constant
```

```
                    throughout the event. Otherwise values are linearly interpolated over
                    time.</xs:documentation>
788             </xs:annotation>
789           </xs:element>
790
791         </xs:sequence>
792       </xs:extension>
793     </xs:complexContent>
794   </xs:complexType>
795
796   <xs:complexType name="GridValues">
797     <xs:sequence>
798       <xs:element name="Values" type="FloatList" maxOccurs="unbounded" >
799         <xs:annotation>
800           <xs:documentation>There should be one values list element for every yCoordinate. The values
                  list in each element will correspond to the xCoordinates. Units shall be in
                  Bq/m2.</xs:documentation>
801         </xs:annotation>
802       </xs:element>
803       <xs:element name="TimePoint" type="FormattedDateTime" minOccurs="0"/>
804       <xs:element name="SubGrid" type="GridSource" maxOccurs="unbounded" minOccurs="0" />
805     </xs:sequence>
806   </xs:complexType>
807
808   <xs:complexType name="BackgroundSource" >
809     <xs:complexContent>
810       <xs:extension base="SourceDistribution">
811         <xs:sequence>
812           <xs:element name="Type">
813             <xs:simpleType>
814               <xs:restriction base="xs:string">
815                 <xs:enumeration value="Constant" />
816                 <xs:enumeration value="RandomizedLinear" />
817                 <xs:enumeration value="RandomizedGaussian" />
818               </xs:restriction>
819             </xs:simpleType>
820           </xs:element>
821           <xs:element name="ArealActivity" type="TimeDependentFloat" maxOccurs="unbounded"/>
822           <xs:element name="RandomizationMagnitude" type="xs:float" default="10.0" minOccurs="0">
823             <xs:annotation>
824               <xs:documentation>Values will be distributed plus or minus this percentage. If the type
                      is RadomizedLinear the values will have a linear distribution. If the type is
                      RadomizedGaussian the percentage of the value will form the standard deviation of
                      the distribution.</xs:documentation>
825             </xs:annotation>
826           </xs:element>
827         </xs:sequence>
828       </xs:extension>
829     </xs:complexContent>
830   </xs:complexType>
831
832   <xs:complexType name="GeoRegion" abstract="true">
833     <xs:attribute name="id" type="xs:ID" use="required"/>
834   </xs:complexType>
835
836   <xs:complexType name="CuboidRegion">
837     <xs:complexContent>
838       <xs:extension base="GeoRegion">
839         <xs:sequence>
840           <xs:element name="BottomSouthWestCorner" type="LocationTriplet" />
841           <xs:element name="TopNorthEastCorner" type="LocationTriplet" />
842         </xs:sequence>
843       </xs:extension>
844     </xs:complexContent>
845   </xs:complexType>
846
847   <xs:complexType name="PolyRegion" >
848     <xs:complexContent>
849       <xs:extension base="GeoRegion">
850         <xs:sequence>
851           <xs:element name="AltitudeMinimum" type="xs:double" />
852           <xs:element name="AltitudeMaximum" type="xs:double" />
853           <xs:element name="Point" type="LocationPair" minOccurs="3" maxOccurs="unbounded" >
854             <xs:annotation>
855               <xs:documentation>All points shall be specified in a counter clockwise
                      orientation</xs:documentation>
856             </xs:annotation>
857           </xs:element>
858         </xs:sequence>
859       </xs:extension>
860     </xs:complexContent>
861   </xs:complexType>
862
863   <xs:complexType name="SphereRegion" >
864     <xs:complexContent>
865       <xs:extension base="GeoRegion">
866         <xs:sequence>
867           <xs:element name="Center" type="LocationTriplet" />
868           <xs:element name="Radius" type="xs:double" />
869         </xs:sequence>
870       </xs:extension>
871     </xs:complexContent>
```

```
872   </xs:complexType>
873
874   <xs:complexType name="ConicRegion" >
875     <xs:complexContent>
876       <xs:extension base="GeoRegion">
877         <xs:sequence>
878           <xs:element name="Axis" type="LocationPair" />
879           <xs:element name="ApexAngleCos" type="xs:double">
880             <xs:annotation>
881               <xs:documentation>The cosine of the full angle at the center of the
                      earth.</xs:documentation>
882             </xs:annotation>
883           </xs:element>
884           <xs:element name="AltitudeMinimum" type="xs:double" />
885           <xs:element name="AltitudeMaximum" type="xs:double" />
886         </xs:sequence>
887       </xs:extension>
888     </xs:complexContent>
889   </xs:complexType>
890
891   <xs:complexType name="InjectType">
892     <xs:sequence>
893       <xs:element name="Title" maxOccurs="unbounded">
894         <xs:complexType>
895           <xs:simpleContent>
896             <xs:annotation>
897               <xs:documentation>The title of this inject. Multiple languages can be specified. If no
                      language match is found the selected language element is
                      undefined.</xs:documentation>
898             </xs:annotation>
899             <xs:extension base="xs:string" >
900               <xs:attribute name="id" type="xs:ID" use="required"/>
901               <xs:attribute name="ietfTag" type="xs:string"  default="en-us" />
902             </xs:extension>
903           </xs:simpleContent>
904         </xs:complexType>
905       </xs:element>
906       <xs:element name="InjectTime" type="FormattedDateTime"/>
907       <xs:element name="LateInjection" type="xs:boolean">
908         <xs:annotation>
909           <xs:documentation>If true and a new inject is delivered as a delta update covering the
                  location the inject shall be presented immediately.</xs:documentation>
910         </xs:annotation>
911       </xs:element>
912       <xs:element name="AuthorizationLevel" type="AuthLevel" default="Field"/>
913       <xs:element name="OverrideLevel" type="xs:positiveInteger" default="1">
914         <xs:annotation>
915           <xs:documentation>If multiple injects occur at the same time and location only the ones
                  with the highest override level are displayed.</xs:documentation>
916         </xs:annotation>
917       </xs:element>
918       <xs:element name="AutoplayMultimedia" type="xs:boolean" />
919       <xs:element name="File" minOccurs="0" maxOccurs="unbounded" >
920         <xs:complexType>
921           <xs:simpleContent>
922             <xs:extension base="xs:IDREF" >
923               <xs:attribute name="id" type="xs:ID" use="required"/>
924               <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
925             </xs:extension>
926           </xs:simpleContent>
927         </xs:complexType>
928       </xs:element>
929       <xs:element name="SourcePerson" type="xs:IDREF" minOccurs="0" />
930       <xs:element name="Location" type="GeoRegion"/>
931     </xs:sequence>
932     <xs:attribute name="id" type="xs:ID" />
933   </xs:complexType>
934
935   <xs:complexType name="InjectMetaType">
936     <xs:sequence>
937       <xs:element name="Inject" type="xs:IDREF" />
938       <xs:element name="Notes" type="xs:string" />
939     </xs:sequence>
940     <xs:attribute name="id" type="xs:ID" use="required"/>
941   </xs:complexType>
942
943   <xs:complexType name="TimeDependentFloatArray" abstract="true">
944     <xs:sequence>
945       <xs:element name="TimePoint" type="FormattedDateTime" maxOccurs="unbounded"/>
946     </xs:sequence>
947   </xs:complexType>
948
949   <xs:complexType name="TemporalInterpolatedFloatArray">
950     <xs:complexContent>
951       <xs:extension base="TimeDependentFloatArray">
952         <xs:sequence>
953           <xs:element name="FloatArray" type="DoubleList"/>
954         </xs:sequence>
955       </xs:extension>
956     </xs:complexContent>
957   </xs:complexType>
958
```

```
959   <xs:complexType name="TemporalAnalyticalFloatArray">
960     <xs:complexContent>
961       <xs:extension base="TimeDependentFloatArray">
962         <xs:sequence>
963           <xs:element name="ChannelEq" type="PolynomialEquation" maxOccurs="unbounded" />
964         </xs:sequence>
965       </xs:extension>
966     </xs:complexContent>
967   </xs:complexType>
968
969   <xs:complexType name="TimeDependentFloat" abstract="true">
970     <xs:sequence>
971       <xs:element name="TimePoint" type="FormattedDateTime" />
972     </xs:sequence>
973     <xs:attribute name="id" type="xs:ID" use="required"/>
974   </xs:complexType>
975
976   <xs:complexType name="TemporalInterpolatedFloat">
977     <xs:complexContent>
978       <xs:extension base="TimeDependentFloat">
979         <xs:sequence>
980           <xs:element name="Value" type="xs:double" />
981         </xs:sequence>
982       </xs:extension>
983     </xs:complexContent>
984   </xs:complexType>
985
986   <xs:complexType name="TemporalAnalyticalFloat">
987     <xs:complexContent>
988       <xs:extension base="TimeDependentFloat">
989         <xs:sequence>
990           <xs:element name="Value" type="PolynomialEquation" />
991         </xs:sequence>
992       </xs:extension>
993     </xs:complexContent>
994   </xs:complexType>
995
996   <xs:simpleType name="FormattedDateTime">
997     <xs:restriction base="xs:dateTime" >
998       <xs:pattern value="[0-9]{4}-[01][0-9]-[0-3][0-9]T[0-2][0-9]:[0-5][0-9]:[0-5][0-9]Z" />
999     </xs:restriction>
1000  </xs:simpleType>
1001
1002  <xs:complexType name="ScalarInst">
1003    <xs:complexContent>
1004      <xs:extension base="InstrumentType">
1005        <xs:sequence>
1006          <xs:element name="MaxValue" type="xs:float"/>
1007          <xs:element name="MinValue" type="xs:float"/>
1008          <xs:element name="Units" maxOccurs="unbounded">
1009            <xs:complexType>
1010              <xs:simpleContent>
1011                <xs:extension base="xs:string">
1012                  <xs:attribute name="ietfTag" type="xs:string" default="en-us" />
1013                </xs:extension>
1014              </xs:simpleContent>
1015            </xs:complexType>
1016          </xs:element>
1017          <xs:element name="AutoScale" type="xs:boolean" />
1018        </xs:sequence>
1019      </xs:extension>
1020    </xs:complexContent>
1021  </xs:complexType>
1022
1023  <xs:complexType name="SpectralInst">
1024    <xs:complexContent>
1025      <xs:extension base="InstrumentType">
1026        <xs:sequence>
1027          <xs:element name="MinMeasurementTime" type="xs:positiveInteger" default="1">
1028            <xs:annotation>
1029              <xs:documentation>The time required to collect a measurement in
                      seconds.</xs:documentation>
1030            </xs:annotation>
1031          </xs:element>
1032          <xs:element name="MaxMeasurementTime" type="xs:positiveInteger" default="600" minOccurs="0"
                  />
1033          <xs:element name="EnergyCalibration" type="PolynomialEquation" >
1034            <xs:annotation>
1035              <xs:documentation>The domain is the channel number the range is the energy in
                      keV</xs:documentation>
1036            </xs:annotation>
1037          </xs:element>
1038          <xs:element name="Rollover" type="xs:boolean" >
1039            <xs:annotation>
1040              <xs:documentation>Indicates if channels should roll over if maxCounts is
                      exceeded</xs:documentation>
1041            </xs:annotation>
1042          </xs:element>
1043          <xs:element name="MaxCounts" type="xs:positiveInteger" />
1044          <xs:element name="NumberChannels" type="xs:positiveInteger" />
1045        </xs:sequence>
1046      </xs:extension>
```

```
1047      </xs:complexContent>
1048    </xs:complexType>
1049
1050    <xs:simpleType name="IntegerList">
1051      <xs:list itemType="xs:positiveInteger" />
1052    </xs:simpleType>
1053
1054    <xs:simpleType name="FloatList">
1055      <xs:list itemType="xs:float" />
1056    </xs:simpleType>
1057
1058    <xs:simpleType name="DoubleList">
1059      <xs:list itemType="xs:double" />
1060    </xs:simpleType>
1061
1062    <xs:complexType name="PolynomialEquation">
1063      <xs:sequence>
1064        <xs:element name="Term" type="PolynomialPair" maxOccurs="unbounded" />
1065      </xs:sequence>
1066    </xs:complexType>
1067
1068    <xs:complexType name="PolynomialPair">
1069      <xs:sequence>
1070        <xs:element name="Coefficient" type="xs:double" />
1071        <xs:element name="Exponent" type="xs:double" default="0" />
1072      </xs:sequence>
1073    </xs:complexType>
1074
1075    <xs:complexType name="LocationPair">
1076      <xs:sequence>
1077        <xs:element name="Latitude" type="xs:double" />
1078        <xs:element name="Longitude" type="xs:double" />
1079      </xs:sequence>
1080    </xs:complexType>
1081
1082    <xs:complexType name="LocationTriplet">
1083      <xs:sequence>
1084        <xs:element name="Latitude" type="xs:double"/>
1085        <xs:element name="Longitude" type="xs:double"/>
1086        <xs:element name="Altitude" type="xs:double" default="0" minOccurs="0"/>
1087      </xs:sequence>
1088    </xs:complexType>
1089
1090    <xs:complexType name="EmbeddedFile">
1091      <xs:sequence>
1092        <xs:element name="FileName" type="xs:string" />
1093        <xs:element name="MimeType">
1094          <xs:simpleType>
1095            <xs:restriction base="xs:string">
1096              <xs:pattern value="[a-z0-9]+\/[a-z0-9\.\-]+([\+][a-z0-9]+)?"/>
1097            </xs:restriction>
1098          </xs:simpleType>
1099        </xs:element>
1100        <xs:element name="Data" type="xs:base64Binary" />
1101      </xs:sequence>
1102      <xs:attribute name="id" type="xs:ID" use="required"/>
1103    </xs:complexType>
1104
1105    <xs:complexType name="SimPerson">
1106      <xs:sequence>
1107        <xs:element name="FirstName" type="xs:string" />
1108        <xs:element name="LastName" type="xs:string" />
1109        <xs:element name="PhoneNumber" type="xs:string" minOccurs="0" />
1110        <xs:element name="EMailAddress" type="xs:string" minOccurs="0" />
1111        <xs:element name="UserPhoto" type="xs:IDREF" minOccurs="0" />
1112      </xs:sequence>
1113      <xs:attribute name="id" type="xs:ID" use="required"/>
1114    </xs:complexType>
1115
1116    <xs:complexType name="FieldTeamType">
1117      <xs:sequence>
1118        <xs:element name="Name" type="xs:string" />
1119        <xs:element name="TeamLeader" type="xs:IDREF" />
1120        <xs:element name="TeamMember" type="xs:IDREF" maxOccurs="unbounded" minOccurs="0"/>
1121        <xs:element name="Instrument" type="xs:IDREF" maxOccurs="unbounded" />
1122      </xs:sequence>
1123      <xs:attribute name="id" type="xs:ID" use="required"/>
1124    </xs:complexType>
1125
1126    <xs:complexType name="FieldTeamMetaType">
1127      <xs:sequence>
1128        <xs:element name="Notes" type="xs:string" />
1129        <xs:element name="FieldTeam" type="xs:IDREF" />
1130      </xs:sequence>
1131      <xs:attribute name="id" type="xs:ID" use="required"/>
1132    </xs:complexType>
1133
1134    <xs:complexType name="WeatherEvents">
1135      <xs:sequence>
1136        <xs:element name="Type" maxOccurs="unbounded">
1137          <xs:complexType>
1138            <xs:simpleContent>
```

```
1139              <xs:extension base="xs:string" >
1140                <xs:attribute name="id" type="xs:ID" use="required"/>
1141                <xs:attribute name="ietfTag" type="xs:string"  default="en-us" />
1142              </xs:extension>
1143            </xs:simpleContent>
1144          </xs:complexType>
1145        </xs:element>
1146
1147        <xs:element name="Region" type="GeoRegion" />
1148        <xs:element name="OverrideLevel" type="xs:positiveInteger" default="1" />
1149        <xs:element name="Start" type="MarkedTime" />
1150        <xs:element name="End" type="MarkedTime" />
1151        <xs:element name="PrecipitationRate" type="xs:float" >
1152          <xs:annotation>
1153            <xs:documentation>In units of cm/hr.</xs:documentation>
1154          </xs:annotation>
1155        </xs:element>
1156      </xs:sequence>
1157      <xs:attribute name="id" type="xs:ID" use="required" />
1158    </xs:complexType>
1159
1160    <xs:complexType name="AuthorDataType">
1161      <xs:sequence>
1162        <xs:element name="AuthorCertKey" type="xs:base64Binary" minOccurs="0">
1163          <xs:annotation>
1164            <xs:documentation>The private key for the first trustOnlyRoot
                  certificate.</xs:documentation>
1165          </xs:annotation>
1166        </xs:element>
1167        <xs:element name="UserPasswordKeys" type="PasswordKey" maxOccurs="unbounded" minOccurs="0" />
1168      </xs:sequence>
1169    </xs:complexType>
1170
1171    <xs:complexType name="PasswordKey">
1172      <xs:sequence>
1173        <xs:element name="PasswordAuthRef" type="xs:IDREF" />
1174        <xs:element name="UserKey" type="xs:base64Binary" />
1175      </xs:sequence>
1176    </xs:complexType>
1177  </xs:schema>
```

Appendix B

XSim C# Client Interface Document

Listing B.1: XSim C# Client Interface

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Security.Cryptography;
6  using XSim = gov.nnss.rsl.xsim;
7
8
9  namespace gov.nnss.rsl.xsim
10 {
11     public interface XSimClient
12     {
13         void ScalarMeasurement(Instrument anInst, double latitude, double longitude, double altitude,
               double height, System.DateTime aTime, float aValue, String units);
14         void SpectralMeasurement(Instrument anInst, double latitude, double longitude, double
               altitude, double height, DateTime aTime, UInt32[] spectra);
15         void InjectNotification(String title, bool autoPlay, String fileName, String mimeType,
               System.IO.Stream injectData);
16         void WeatherUpdate(double latitude, double longitude, double altitude, String weatherType);
17         String PasswordRequest(String userName);
18         Byte[] DecryptKeyRequest(String certIssuer, String certSerialNumber, Byte[] encryptedKey);
19         Error ErrorHandler(Error errorCode, String errorMessage, object parameter);
20         void UpdateFile(Uri replacementURI, Uri supplementalURI, bool defaultUpdate);
21     }
22
23     public enum Error
24     {
25         noErr = 0,
26         memErr, // memory allocation failure in library
27         ioError, // standard io error
28         noUserSpecified, // attempted to call a function which requires a current user
29         malformedFileFormat, // XML source file isn't to specification
30         noSuchUser, // attempted to specify a user that doesn't exist
31         badPassword, // password provided isn't valid
32         parseError, // Basic XML parsing syntax error
33         instrumentNotFound, // instrument specified not found
34         earlyTime, // requesting data from a time before the simulation starts
35         lateTime, // requesting data from a time after the simulation ends
36         unimplemented, // this function or option is not implemented yet
37         userAlreadySet, // A user has already been successfully set
38         unexpectedException, // An exception was thrown that was not expected
39     }
40
41     public interface XSimFieldLibrary
42     {
43         Error LoadScenario(System.IO.Stream baseFile, String ietfLanguage, XSimClient callbacks);
44         Error AddSupplementalFile(System.IO.Stream supplementalFile);
45         String[] GetUserNames();
46         Error SetCurrentUser(String aUser);
47         // returns a dictionary of the instruments and the Double or UInt32 array object if
               generateDictionary is true, otherwise null
48         IDictionary<Instrument, Object> HeartBeat(double latitude, double longitude, double altitude,
               double height, DateTime gpsTime, bool generateDictionary);
49         Error RegisterInstrumentInterest(Instrument anInst, bool interested);
```

```
50          Instrument[] GetUniqueInstruments();
51          Instrument[] GetGenericInstruments();
52          void CloseLibrary();
53      }
54
55      public interface Instrument
56      {
57          System.IO.Stream GetSkin();
58          String GetName();
59          String GetSerialNumber();
60          Guid GetGUID();
61          Boolean IsGeneric();
62      }
63  }
```

Appendix C

Acronyms

AES Advanced Encryption Standard

CBC Cipher Block Chaining

FWHM Full Width Half Maximum

GIS Geographic Information System

GMT Greenwich Mean Time

GPS Global Positioning System

GUI Graphical User Interface

GUID Globally Unique Identifier

HTML5 HyperText Markup Language version 5

IETF Internet Engineering Task Force

IV Initial Vector

MIME Multipurpose Internet Mail Extensions

NCRP National Council on Radiation Protection

PAG Protective Action Guides

URL Uniform Resource Locator

UTC Coordinated Universal Time

UTF-8 Unicode Transformation Format, 8 bit blocks

UTM Universal Transverse Mercator

UUID Universally Unique IDentifier

WGS84 World Geodetic System 1984

XML Extensible Markup Language

XSD XML Schema Definition

Appendix D

Bibliography

[1] L. R. Anspaugh, J. H. Shinn, P. L. Phelps, and C. N. Kennedy. Resuspension and redistribution of plutonium in soils. *Health Physics*, 29(4):571–582, October 1975.

[2] Lynn R. Anspaugh, Steven L. Simon, Konstantin I. Gordeev, Ilya A. Likhtarev, Reed M. Maxwell, and Sergei M. Shinkarev. Movement of radionuclides in terrestrial ecosystems by physical processes. *Health Physics*, 82(5):669–679, May 2002.

[3] P. Deutsch. Zlib compressed data format specification. Request for Comments 1950, Network Working Group, May 1996.

[4] P. Leach. A universally unique identifier (UUID) URN namespace. Request for Comments 4122, Network Working Group, July 2005.

[5] R. M. Maxwell and L. R. Anspaugh. An improved model for prediction of resuspension. *Health Physics*, 101(6):722–730, December 2011.

[6] National Council on Radiation Protection and Measurements. Recommended screening limits for contaminated surface soil and review of factors relevant to site-specific studies. Report 129, NCRP, 1999.

[7] A. Phillips and M. Davis. Matching of language tags. Request for Comments 4647, Network Working Group, September 2006.