Requirements for a top level hierarchy for a next generation nuclear data format

D.A. Brown,[1, *] A. Koning,[2, †] D. Roubtsov,[3, ‡] R. Mills,[4, §] C. Mattoon,[5, ¶] and M. White[6, **]
[1] *NNDC, BNL, Upton, NY, USA*
[2] *NRG Petten, Netherlands*
[3] *AECL, Chalk River Laboratories, Canada*
[4] *NNL, United Kingdom*
[5] *LLNL, Livermore, CA, USA*
[6] *LANL, Los Alamos, NM, USA*

(Dated: December 11, 2013)

This document attempts to compile the requirements for the top-levels of a hierarchical arrangement of nuclear data such as is found in the ENDF format. This set of requirements will be used to guide the development of a new set of formats to replace the legacy ENDF format.

CONTENTS

---

[*] Corresponding author: dbrown@bnl.gov
[†] koning@nrg.eu
[‡] roubtsovd@aecl.ca
[§] robert.w.mills@nnl.co.uk
[¶] mattoon1@llnl.gov
[**] morgan@lanl.gov

I. INTRODUCTION

This document attempts to compile the requirements for the top-levels of a hierarchical arrangement of nuclear data such as is found in the ENDF format. This set of requirements will be used to guide the development of a new set of formats to replace the legacy ENDF format. These formats will all follow the same hierarchy, ensuring that future users and developers of nuclear data will always be able to find and store their data in what ever medium it is presented in the future.

The authors of this requirements document have had many discussions with members of the nuclear data community over the past few months and attempted to capture all the ideas and needs in one document. With a high degree of confidence we can say we did not capture all of them. What follows is our attempt to put together what we heard from the community with the our collective experiences the ENDF and other nuclear data formats.

In this document, requirements are called out and numbered so that they can be clearly referenced in later work. We also tend to use XML in example markup and to denote elements in the hierarchy. This is only a matter of convenience as the data should be serializable in any hierarchical form (e.g. HDF5, ROOT, Python classes). We denote major nodes/elements in the XML-like notation `<element>` and attributes of these nodes/elements without the brackets, e.g. `attribute`.

A. Goals/Main Requirements

In the Nov 2012 WPEC meeting, we laid out the goals of the format.

MAIN REQUIREMENTS

R:1 The hierarchy should reflect our understanding of nuclear reactions and decays, and clearly and uniquely specify all data.

R:2 It should support storing multiple representations of the same quantity simultaneously (e.g. evaluated and processed data).

R:3 Should support both inclusive and exclusive reaction data (i.e., discrete reaction channels as well as sums over those channels).

R:4 It should eliminate redundancy where possible.

R:5 It should make use of the general-purpose data containers designed by the first SG38 project group.

Some of these goals may seem contradictory (allowing multiple representations while at the same time eliminating redundancy, for example), but realize we are talking about what format will support, not the requirements of a specific library project. It is up to each library's project manager(s) to enforce specific requirements (e.g., that only raw, not processed, data be stored in a particular library or that all the cross sections in a library be stored in a particular group structure). We also comment that we will need to strike a balance between a deep or shallow hierarchy since some storage schemes perform better with a flatter hierarchy (e.g. HDF5) while a deep hierarchy may make sense for organizing the data more clearly.

We also add some other requirements that shape our outlook on this hierarchy.

EXTRA REQUIREMENTS

R:1 Grandfather in all ENDF data and ENDF formats.

R:2 Correct, wherever possible, ENDF mistakes and inconsistencies.

R:3 "We don't have the resources to shoot all of our users in the foot. So we'll give the gun to the users so they can shoot their own feet." [1].

B. Scope of data to support

An evaluated nuclear reaction data hierarchy must support an incident particle (projectile) impinging on target material. This target material may be either a single atom or atomic nucleus or a collection of atoms which the projectile on which (in)coherently scatters off. Projectile can be the traditional n, p, d, t, ^{3}He, α, γ, or e^- or any other single (composite) particle e.g. ^{12}C. The projectile list is not inclusive and may be expanded to handle other particles such as muons and pions.

What data is to be stored is a balance between what an evaluator can provide and what a particular application needs. Therefore, it is useful to look at the most common use cases:

- **Particle Transport:** For transport, all cross sections and outgoing energy and angle probabilities for all emitted particles for chosen reactions that are energetically possible over a given range of incident energy E. Also, multiplicities for all emitted particles if not constant. These can be parametric (e.g. RRR or Watt Spectrum) or tabular. Probability distributions must span all energetically allowed E' and angle.
  - **Deterministic transport:** WRITE ME
  - **Monte Carlo transport:** WRITE ME
- **Transmutation/Isotope Burn-Up:** For isotopic accretion/depletion, need cross sections and optionally outgoing spectra for chosen reactions that are energetically possible over a given range of incident energy E. Also must know how (and be able to sample when) all produced particles decay so that a time dependent isotope inventory may be computed.
- **Astrophysical network calculation:** WRITE ME
- **Web Retrieval:** For archival, no there are no completeness requirement as this data will not be used in applications, the data will only most likely be visualized.
- **Uncertainty Quantification:**

Data will be documented, so need facility for documenting data clearly, concisely and as machine and human readable as possible. Along with the documentation, version information for format, documentation, evaluation, codes used in evaluation, etc. need to be stored.

Additionally, we need to support covariances/uncertainty on all tabulated and parametric data or alternatively an ensemble of values/tables

C. Complications

As we consider solutions to these issues, we must strike a balance between the legacy (e.g. ENDF) solution, which we are all familiar with, and what makes most sense physically and what is expedient.

1. *Is it a material property or a reaction property?*

Several kinds of data can be legitimately thought of as reaction or as a property of the target material. A case in point is the gamma branchings from an excited nuclear state of a nucleus. The nucleus can be placed in an excited state with an incident neutron and the nucleus then decays via a gamma cascade through the lower levels of the nucleus with the tabulated gamma branchings.

Given this, we view *all particle/material properties as data that is independent of the excitation mechanism.* This includes (but is not limited to):

- For atomic nuclei:
  - target mass
  - number of neutrons, protons (and maybe even hyperons!)
  - nuclear level schemes (energies, spins, parities, ...)
  - gamma and decay branching ratios from excited states of a level
  - level lifetimes, level widths
  - emission spectra from particle decays
- For elements:
  - target mass
  - isotopic composition
- For atoms/ions:
  - target mass
  - atomic shell properties (binding energies, spins, parities, ...)
  - gamma and decay branching ratios from excited states of a level
  - level lifetimes, level widths
  - emission spectra from particle decays

  - charge state
- For composite materials (as encountered in thermal neutron scattering):
  - target density (at STP)
  - target stoichiometry
  - equation of state

These lists may be amended as needed in the discussion below and a deeper discussion of them will be presented in the requirements for the material properties database. Indeed, it was recognized at the previous WPEC meeting that, in order to ensure consistency of masses, Q values, levels and gammas within an evaluation, an external database is needed to perform this role library-wide. This database addresses main requirement #4.

2. *Different optimal representation in different physical regimes*

There are different optimal representations of data in different physical regimes. For example, at low energies neutron scattering is best described with an R matrix approach (in the resolved resonance region), and tabulated data above the (n, n') threshold. This is depicted in Figure 1. This implies for example

- different physical regimes may change concept of what is a target (e.g. low energy neutrons (in)coherently scattering off atoms in material)
- different macroscopic environments change effective microscopic data (e.g. Doppler broadening)
- different incident energies affect what particles are produced (e.g. pre-equilibrium, multifragmentation, particle production, spallation)

This fact is recognized in the design of the legacy ENDF format and is a reality we too must confront. So, we not only must consider different optimal representations (e.g. resonance parameters) in different physical regimes, but we also must consider

- different alternate representations (e.g. pointwise resolved resonance data);
- the matching (and potentially overlap) between representations;
- a mechanism to "glue" them together, especially in cases where the concept of a target or reaction changes dramatically (e.g. thermal neutron scattering on molecule transitioning to high energy neutron resolving the nuclei in the atoms of the molecule)

3. *Ensuring consistency*

As we design the format(s) and supporting infrastructure, it is important to maintain internal consistency of the data. Within an evaluation, we must ensure at the very least consistency between

- Cross section sum rules
  - Summing to the total cross section
  - All (n, n') cross sections sum to total inelastic (ditto for other similar reaction types); similar to MT=3
- prompt nubar + all delayed nubars = total nubars
- Masses, Q values, thresholds, upper energy bounds on secondary distributions
- Normalization
- Energy and forward momentum balance
- Consistent energy ranges
- gamma branchings
- processed and original data

Between evaluations, we must also ensure consistency between

- Fission product yields and decay data linkage
- Resolved and unresolved resonance regions and the fast reaction region
- Fast reaction region and the particle production region
- Masses, etc. and material properties
- Covariances and mean values between data common to both the Neutron Standards and CIELO projects

Some of this can be handled with a simple hyperlink. Others may require capability within external processing/manipulation infrastructure. In the following discussions, we will point to features of the hierarchy that enable the maintenance of consistency.

Material properties are a special case, especially when dealing with legacy ENDF evaluations: there is no guarantee that data stored in ENDF uses consistently the same masses or level schemes for materials. As such, we may need to override any external material properties database with local versions within an evaluation.

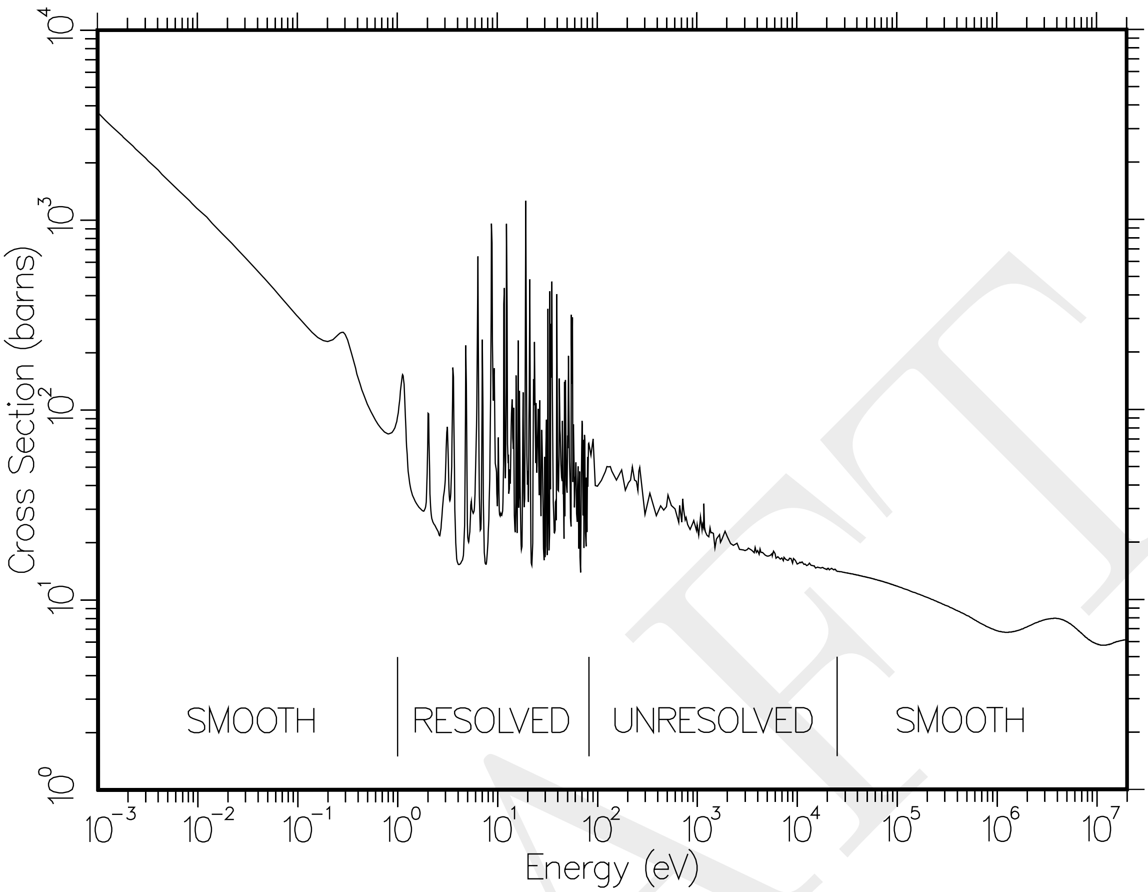


FIG. 1. Cartoon of energy regimes in a neutron induced reaction. The high energy region labeled "Smooth" is usually handled in a very different way than the low energy regions using the R matrix approach.

II. BASIC CAPABILITIES

Here we detail the things we need in order to build up the top level hierarchy. We will not go into the details in many of these as they are detailed in the other requirements discussions. However, any requirements we must impose to make the top levels work better will be called out.

A. Required low-level containers

The low level containers are dealt with in another requirements document (satisfying main requirement #5). Here we list what we need for the top level hierarchy discussion and how we will refer to them.

LOW-LEVEL CONTAINER REQUIREMENTS

R:1 floats (`float`), integers (`int`) and strings (`string`)

R:2 list or vector (`<list>`), must specify type of object in the list

R:3 matrix (`<matrix>`), must specify dimensions. May be banded, symmetric, etc.

R:4 table (`<table>`), like a matrix, but the columns have labels and units and maybe even data-type information

R:5 orthogonal function expansion, Legendre polynomials being the most obvious (`Legendre`)

R:6 `<interp2d>`: interpolation table for univariate data, i.e. x vs. $f(x)$

R:7 `<interp3d>`: interpolation table for bivariate data, i.e. (x, y) vs. $f(x, y)$

R:8 `<interp4d>`: interpolation table for trivariate data, i.e. (x, y, z) vs. $f(x, y, z)$

R:9 `<axis>`: Where appropriate (particularly on interpolated types), we need to specify interpolation details, units, labels, etc.

  R:9.a Specify names of x, y, z, ... axes

  R:9.b Specify normalization (if any)

  R:9.c Specify units in all directions

  R:9.d Specify interpolation scheme(s) or group boundaries

R:9.e If interpolation the table refers to a probability distribution function (PDF), we also must specify whether is Normal or Log-Normal [4].

R:10 `<text>`: marked up text, either in HTML, Markdown, or plain old text. The format must be denoted.

R:11 hyperlinks (`<link>`) (more on this in the next subsection)

DISCUSSION POINT It has been suggested by several members of the nuclear data community to include uncertainty directly into elements such as the `<interp2d>` table. This would make plotting the uncertainty simpler at the expense of introducing an additional data synchronization problem between the mean values and the covariance data. RESOLUTION This idea is still under discussion. If we do this, we must put a `<link>` to the covariance as a `nativeData` for the uncertainty.

In this document, we attempt to keep to these definitions.

B. Links

Links (`<link>`) are an important part of the new format(s) and allow the evaluator to refer to other elements within the file or even to elements in external files or databases. Examples of data which use links include:

- Distributions for one reaction product may be treated as the recoil from another product, requiring a link to the other product.
- Production cross sections may be listed as an energy-dependent multiple of another cross section, requiring a link to the other cross section.
- Covariances are stored in a separate file from the quantities they correlate. Links are necessary to associate the covariance with the correct data.

Because the data is stored hierarchically, the path within a document can be followed straightforwardly. It is useful to think of these paths as similar to paths in a Unix filesystem, but with the top level of a document referred to with a URL.

`<link>` REQUIREMENTS

R:1 The paths may be absolute so that they can refer to external documents or relative so that they can make in-document referrals.

R:2 The URL's of the documents and the schema location (in the case of an XML version of the format) may be placeholders or may refer to actual locations on the internet.

One can easily imagine that one is using a nuclear data library on a computer not directly connected to the internet so external links may not be available. In that case, it would be up to the user of the data to remap the URL's to the actual location of the data files on their own computer system.

C. Material designation

We need to decide on common names and GND/Fudge is evolving. So here we detail what we use for the discussion of the hierarchy here. The actual designation will be discussed in detail in the context of the particle database. A simpler version described in Ref. [3]. Here we detail a simple format that can handle material specification for thermal neutron scattering on polyethylene using same system as a proton on ^{238}U:

- **Aliases:** A limited number/scope of aliases for commonly used particles, such as e for electron or a for alpha or n for neutron. Also to associate a level of an isotope with an isomer.
- **Compounds:** `c_Free_String_Describing_Material` can be used to specify say H in ZrH or the phase of the material. Useful for TSL data
- **Elements:** `Sym0` e.g. `Fe0` or `C0`, useful for atomic data
- **Isotopes:** `SymA` e.g. `Fe56`
- **Levels of an isotope:** `SymA_eN`, e.g. `V51_e1` for the first excited state of ^{51}V or `SymA_c` for continuum.
- **Electronic shells of an atom:** `Sym0_eN`, e.g. `V0_e1` for the first shell of ^{nat}V or `Sym0_c` for continuum.

D. Reaction designation

We need to decide on reaction nomenclature and GND/Fudge is evolving. So we detail what we use in the discussion of the hierarchy. This will be discussed in detail in the context of the low level data containers. We use what is detailed in [3] and examples are shown in Table I.

We note that this scheme is more general than ENDF's MT designator and this scheme does not muddle MF and MT (as what happens in the fission reactions in ENDF). In GND's scheme, the reaction designator is unique and derivable from the reaction products (and their decay products if this is a breakup reaction) However, the user does have the ability to define their own.

Whatever is finally agreed on for reaction designators should follow the following recommendations:

REACTION DESIGNATOR REQUIREMENTS

R:1 Should be shared/agreed upon with EXFOR

GND reaction label	ENDF MT
n + Pu239 → n + Pu239	2
n + Pu239 → n + Pu239 [compound elastic]	
n + Pu239 → n[multiplicity:'2'] + Pu238	16
n + Pu239 → n[multiplicity:'3'] + Pu237_e1	
n + Pu239 → n + Pu239_e1	51
n + Pu239_m1 → n + Pu239_c	91
n + Pu239 → Pu240 + gamma	102
n + Pu239 → Pu240_e1 + gamma	
C12 + Pu239 → C12_e2 + Pu239_e1	
n + Be7 → (Be8 → He4[multiplicity:'2'])	

TABLE I. Example of reaction labels in GND. ENDF MT numbers are listed when possible, but some GND reactions have no MT equivalent. From Ref. [3]

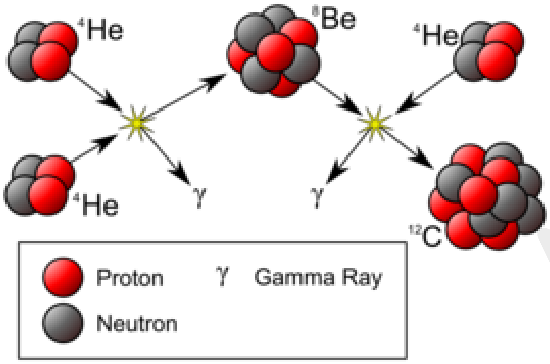


FIG. 2. The famous triple-α reaction.

R:2 Should not be limited to simple targets (we need to denote thermal neutron scattering data)

R:3 Support aliases for things like "`elastic`", "`total_fission`", "`capture`"

R:4 Support need to distinguish input vs. output channels

R:5 Allow n-body processes

R:6 Support variable multiplicity processes

R:7 Support sequential processes (esp. 2-body)

R:8 Support annotation (e.g. "`compound_elastic`" and "`shape_elastic`")

DISCUSSION POINT As an exercise for any reaction designator, see if the famous triple α reaction must be encoded as two separate reactions:

- `He4 + He4 -> Be8 + g`
- `He4 + Be8 -> (C12_Hoyle -> C12 + g)`

E. Derived vs. original data

According to main requirement #2, a mechanism is needed that can specify what data set is "original" and what is derived. To accommodate this, we must allow the storing of the original and derived data at the same level in the hierarchy. Derived data must point back to the original data with a `<link>`. In GND, this `<link>` is denoted with a `nativeData` attribute.

There are many cases where such a capability would be useful:

- Doppler broadened data at a temperature $T > 0^\circ$K should link to the 0°K data.
- Grouped and pointwise data
- Angular distributions converted between pointwise angular tables and Legendre moments
- Any (and all) parameterized data converted to pointwise
- Changes in interpolation schemes (e.g. log-log to lin-lin)
- Resonance data converted to pointwise

There are some cases where multiple `<link>`'s are needed to specify the original data:

- Resonances with smooth backgrounds (this is allowed in ENDF, and we argue below that it should NOT be allowed in the new hierarchy)
- Monte Carlo realizations of a data set should point to the mean value and the associated covariance
- Average energy deposited, average forward momentum deposited and KERMA factors are all derived using the energy balance of all the particles emitted in a reaction

DERIVED DATA FLAG REQUIREMENTS

R:1 Derived data contains a simple attribute with a `<link>` to the original data. Either arrange the hierarchy so that we only need to link to one e.g. `<form>` element or allow multiple links. In GND and in this document, we refer to this flag as the `nativeData` attribute.

F. `<form>`s

In GND, the different versions of the same data are each encapsulated within a `<form>` element. The `form` element is the lowest level of the top-level hierarchy before encountering the actual containers holding the data.

`<form>` REQUIREMENTS

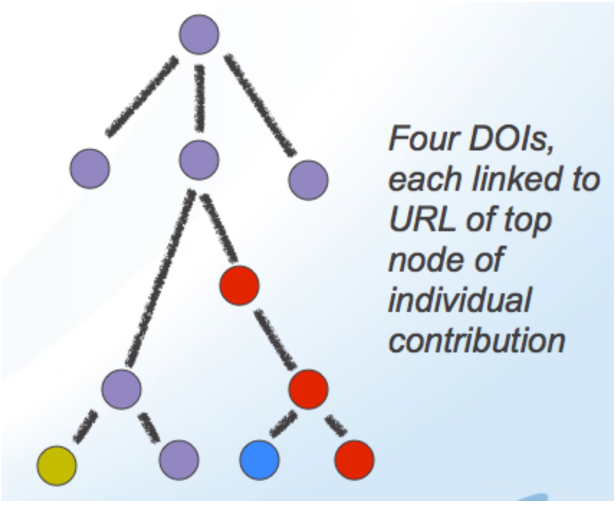


FIG. 3. Cartoon illustrating the construction and documentation of a frankenevaluation. Each colored node is an independent evaluated part and all are assembled together to form the complete evaluation. At the uppermost node of each set of colored nodes, the evaluator should have a `<documentation>` element and corresponding DOI.

R:1 A `<form>` element to contain one specific implementation of a data (e.g. an `<interp2d>` table plus additional attributes.

R:2 Specify reference frame

R:3 `nativeData` attribute.

R:4 Could optionally be a `<link>` to a `<form>` element in another evaluation (for use with `<metaEvaluation>`s).

G. Documentation

The documentation for an evaluation or part of an evaluation is in a way the most essential piece of information. With it, we can tell who performed the evaluation and how they did it. This is essential both for attributing credit (and blame ;)) and for debugging problems in an evaluation.

BACKGROUND DISCUSSION FROM US (OSTI) AND EU REQUIREMENTS

Because each part of a data file may be evaluated separately creating a “frankenevaluation”, we must allow `<documentation>` elements at many different levels in our data hierarchy. Additionally, since each node in the tree representing the data hierarchy can have its own URL, each `<documentation>` element could have its own Digital Object Identifier (DOI). This is illustrated in Figure 3

Each `<documentation>` element should have roughly the same structure as is illustrated in Figure 4. Although it is desirable to have enough detail in the documentation to reconstruct the evaluation exactly as the evaluator has produced it (complete with renromalizations of fitted data, etc.), the variety of different processes that evaluators use to create an evaluation make developing a more detailed documentation specification unworkable.

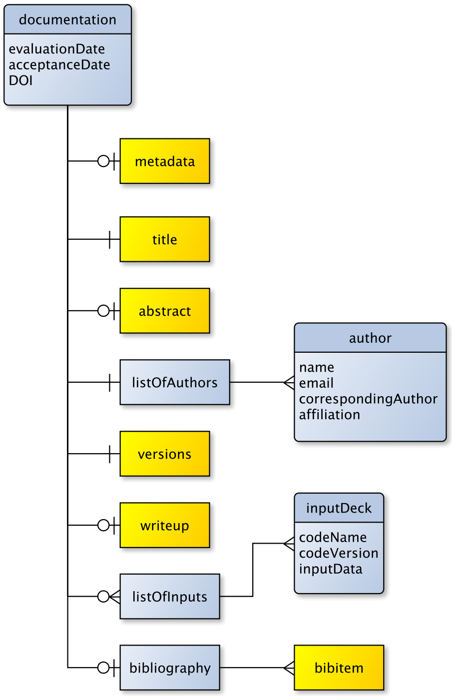


FIG. 4. Basic structure of a `<documentation>` element.

DOCUMENTATION REQUIREMENTS

R:1 Allow metadata (what here? just keywords for search engines, or more?)

R:2 Have markup for the DOI (using a `<link>` element)

R:3 Have markup for title (using a `<text>` element)

R:4 Have markup for the evaluation date (authors make this up)

R:5 Have markup for the library acceptance date (library maintainers make this up)

R:6 Have markup for abstract (optional, using a `<text>` element)

R:7 Have markup for authors (names, affiliation, email, etc.). Who is corresponding author? How should this be structured?

R:8 A mechanism for storing the input decks from codes used by the evaluators to prepare the evaluation (`<listOfInputs>` and `<inputDeck>` elements)

R:9 Have markup for the evaluation version

R:10 Allow free text write up (using a `<text>` element)

R:11 Have markup for the bibliography. How should this be structured? In principal it should be shared with EXFOR.

III. THE TOP LEVEL: ONE EVALUATION

The top level of data files in all major libraries is the "evaluation", consisting of one target material and one projectile and all the data that goes with the reactions between this pair. This arrangement is familiar to the nuclear data community and should be embraced going forward.

Because of the different kinds of evaluations, what happens below the uppermost node in the hierarchy can differ from sublibrary to sublibrary. There are three main classes of sublibraries that concern us:

- **Thermal scattering law data:** neutrons reacting with such low energy that the de Broglie wavelength of the neutron is too large for the neutron to resolve individual nuclei (in principal other particles could do this too)
- **Atomic scattering data:** electron and photon interactions with atoms
- **Nuclear reaction data:** any projectile impinging with enough energy to interact with an atomic nucleus. This collection of data can include resonance data which is arguably different enough from fast reaction data to merit its own discussion.

An ENDF-like decay sublibrary is discussed in the context of a material properties database.

For the purpose of simplifying discussion and focusing on the main structure of an `<evaluation>`, we suppress the derived data elements that are used in specific transport applications. These are discussed in Section XII.

DISCUSSION POINT As the resonances and the fast regions are two distinct physical representations of data in two different energy regimes, it might make sense to require that they NOT be together in the same evaluation and that users use the `<metaEvaluation>` markup to combine them. This simplifies bookkeeping and ensures that users understand that they are different things that they must combine themselves. This is a change from the ENDF mindset. This would complicate the outgoing particle distributions in the resonance region since many ENDF forms don't support computing the angular distributions from the resonance parameters and rely on those tables being given in the fast files. RESOLUTION This idea was generally supported, but is something each library project will need to decide among themselves as the format should support both a legacy arrangement and this proposed arrangement.

DISCUSSION POINT Within an evaluation in a particular sublibrary, one must ask whether to arrange the data per-energy or per-reaction. For data to be per-energy or energy-major, we mean that all data (all reactions, cross sections, distributions, etc.) for one incident energy are collected together in one parent element. For data to be per-reaction or reaction-major, we mean that all data (cross sections, distributions as a function of incident energy) for one reaction are collected together in one parent element. A per-energy arrangement is particularly convenient for Hauser-Feshbach (and other) modeling codes because one normally computes one energy at a time. An energy-major arrangement has certain benefits and drawbacks:

- **Energy-major benefits**
  - Natural output of a reaction model such as EMPIRE or TALYS
  - Energy-major is natural for sampling in Monte Carlo transport
  - One can see at a glance what channels open and compete with one another
  - Helps eliminate background cross sections in the resonance region
- **Energy-major drawbacks**
  - Very difficult to plot say a cross section as a function of incident energy
  - Very difficult to compare to experimental data
  - Requires major refactoring of Monte Carlo codes are needed to get benefits
  - Difficult to diagnose unphysical discontinuities as a function of incident energy
  - Hard for deterministic codes to use
  - Not familiar to users as legacy ENDF data is stored with the reaction-major arrangement

RESOLUTION It is our opinion that the benefits of an energy-major arrangement do not outweigh the drawbacks and so we recommend maintaing the ENDF-style reaction-major arrangement. However, denoting the energy range of validity of an evaluation coupled with the `<metaEvaluation>` concept allow an evaluator to achieve the effect of an energy-major arrangement by having only one incident energy in an evaluation. It was suggested that someone develop a tool to combine these one-energy sized evaluations into a complete reaction-major evaluation. This tool would then be reusable for data generated using any Hauser-Feshbach code.

In all cases, the data can be arranged using a consistent set of rules. Because of the different optimal representations in the fast and resonance regions, the rules however distinguish between data broken out by reaction and tabulated one at a time (using the `<setOfReactions>` element) or whether they are derived from a parameterized form such as in the resonances region (using the `<resonances>` element). This top level arrangement is shown in Figure 5.

`<evaluation>` REQUIREMENTS

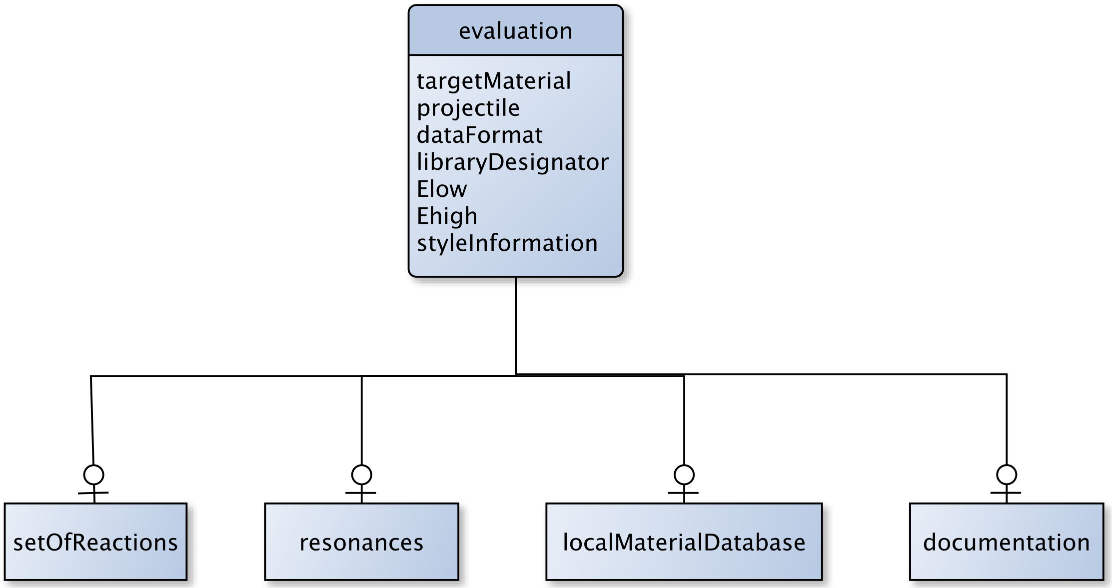


FIG. 5. Top level arrangement of an `<evaluation>` element. The “$\overset{\circ}{+}$” symbol means that there may be at most one of these child elements.

R:1 Require one target material (e.g. “Fe0”)

R:2 Require one projectile (e.g. “n”)

R:3 Require the specification of the data format

R:4 Require the library designator (i.e. a name string that says “ENDF/B” and a version string that says “VI.1”)

R:5 Optionally support other data the library maintainer needs for proper data management, in a `styleInformation` attribute.

R:6 Require a `temperature` attribute: for low enough energy projectiles, this is a crucial piece of information. For neutrons, Doppler broadening is important to get self-shielding corrections. For astrophysical applications, need temperature of plasma so can handle Coulomb screening properly.

R:7 Require an `Elow` and `Ehigh` attributes to specify the range of validity of this evaluation.

R:8 Optionally file-wide `<documentation>`

R:9 Optionally a material database to override defaults with values local to the evaluation (the a `<localMaterialDatabase>` element, not described in this document)

R:10 Optionally a `<setOfReactions>` element (more on `<setOfReactions>` in the first subsection)

R:11 Optionally a `<resonances>` element (more on `<resonances>` in the second subsection)

A. The collection of reactions: `<setOfReactions>`

Below the `<evaluation>` markup, most of the data lives in the `<setOfReactions>` branch. This branch is pictured in Figure 6. Here there are two kinds of reactions: exclusive (`<reaction>`) and inclusive (`<summedReaction>`).
DISCUSSION POINT It may be advantageous to split this element out between the `<reaction>` and an `parameterizedTwoBodyReaction` elements to clearly denote the special channels that are best represented as parameterized $d\sigma(E)/d\Omega$ data. This option is shown in Figure 7. RESOLUTION This option adds an extra layer of complexity and only makes sense if there is no way of having a conditional representation of the reaction data.

B. Inclusive reactions: `<summedReaction>`

In GND, inclusive reactions are encoded in a `<summedReaction>` element. This element includes the cross section itself (and this may be connected to covariance data). Additionally, there is a list of links to act he reactions which are meant to be summed together to match the cross section data in the element.

`<summedReaction>` REQUIREMENTS

R:1 A reaction designator that is an alias, e.g. “total” or “absorption”

R:2 A cross section

R:3 A list of links to the reactions whose cross section

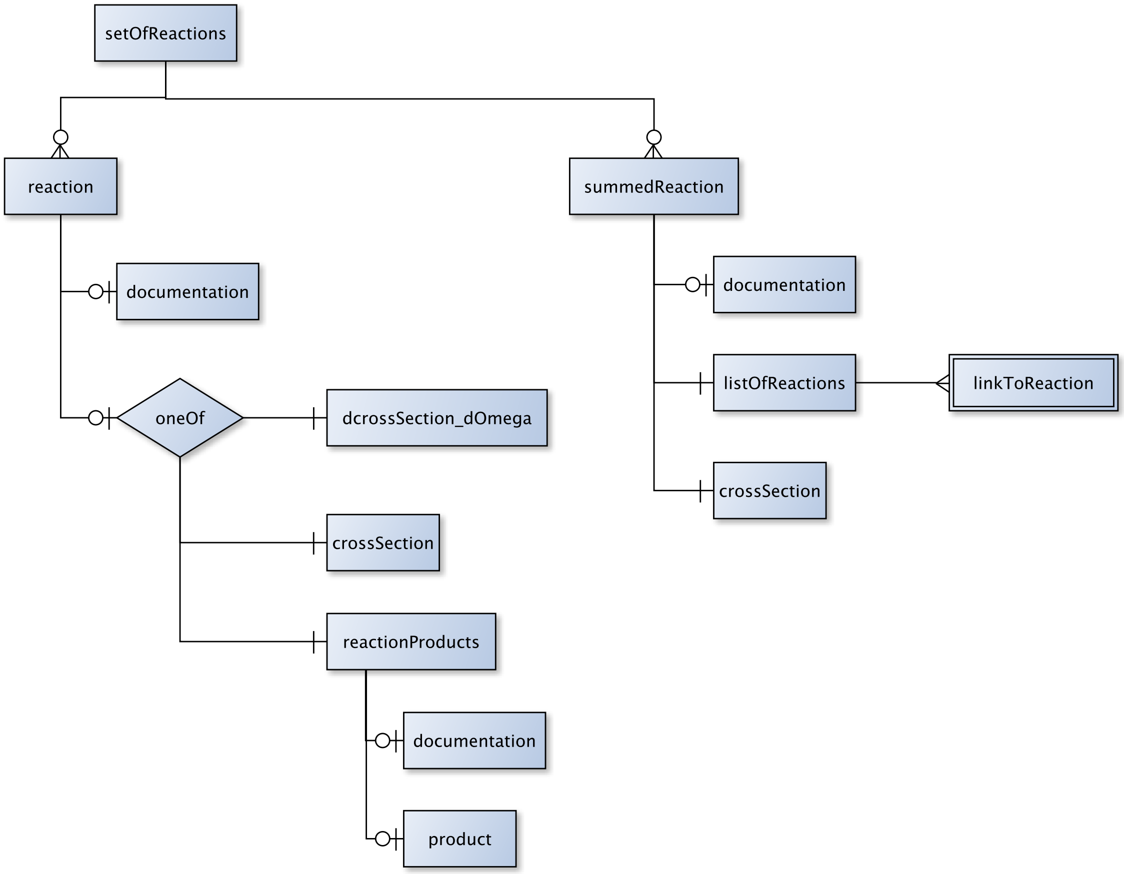


FIG. 6. A possible arrangement of inclusive and exclusive reactions in the `<setOfReactions>` element. The “$\overset{\circ}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\circ}{\wedge}$” symbols means there may be zero or more of these child elements.

is meant to match the cross section tabulated here

C. The `<reaction>` element

With a reaction major arrangement, there is one common motif in the three different sublibraries, which for a lack of a better name, we call the `<reaction>` element. This element denotes one reaction that can be sampled in a Monte Carlo code. In it, we specify the `<channel>`, the reaction `<crossSection>` and the outgoing particle distributions for all emitted particles.

THIS IS WORKHORSE PART OF HIERARCHY. MOST DATA WILL GO THROUGH THESE FORMS BEFORE PLOTTING, PROCESSING FOR TRANSPORT ETC.

NEED TO DESCRIBE THE MENTAL MODEL OF MONTE CARLO CODES AND HOW THAT INFORMS THIS DATA ARRANGEMENT

NEED THE TABLE OF ALLOWED COMPONENTS AND FORMS

Applies to all hadronic projectiles (excludes (anti)electrons) and gammas

No longer distinct resonances; smooth cross sections; “reasonable” number of open channels enumerable by excitation of residual+list of emitted particles. Typically En =100s keV $- >$ 20-30 MeV

DISCUSSION POINT Implementation of breakup and/or multistep reactions? GND scheme complex. Occurrences include

- light element breakup
- (n,gf) reactions
- reactions to unstable residuals where we the decays are short enough they must be accounted for in transport

RESOLUTION This is handled by the `<decayProducts>` element in section III F

`<reaction>` REQUIREMENTS

R:1 An optional `<documentation>`

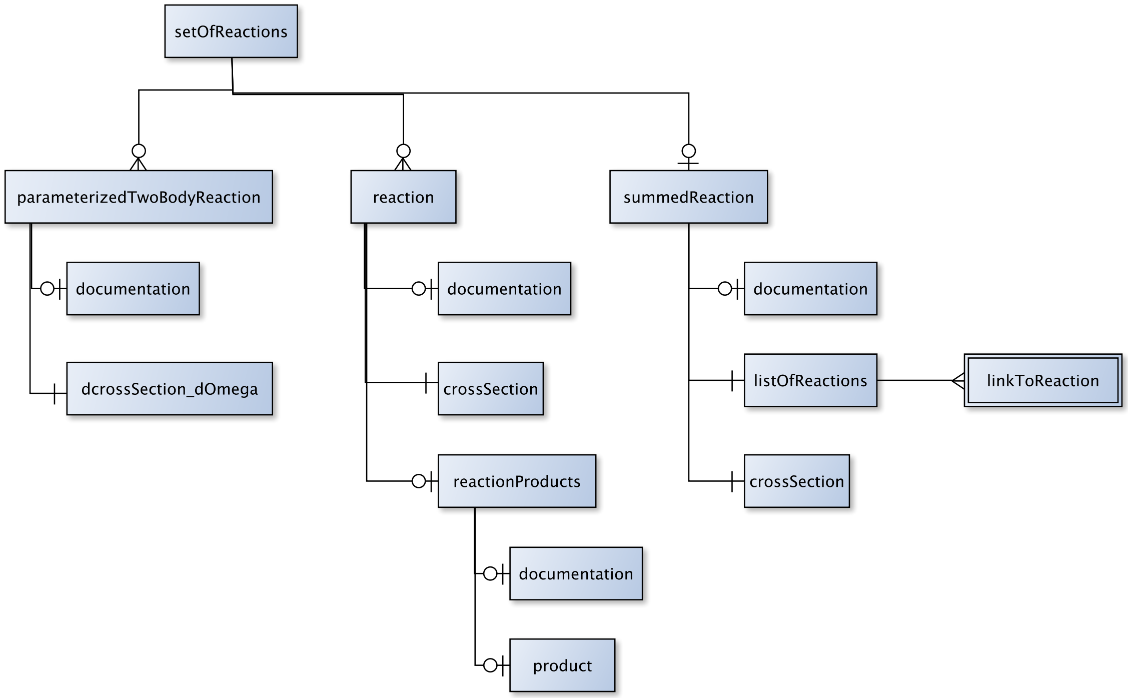


FIG. 7. An alternative way of arranging inclusive and exclusive reactions in the `<setOfReactions>` element which includes the special parameterized two-body data. The "$\overset{\circ}{+}$" symbol means that there may be at most one of these child elements, the "+" symbol means there is exactly one of these child elements and the "$\overset{\circ}{\wedge}$" symbols means there may be zero or more of these child elements.

R:2 The kinematic type (e.g. two-body, n-body, uncorrelated). Elastic reactions and all resonance reactions using the R matrix formalism are two-body reactions. GND refers to these by the name `<genre>`. Table II lists allowed kinematic types.

R:3 Either

  R:3.a A `<crossSection>`

  R:3.b A `<reactionProducts>` element listing the reaction `<product>`s. From the list of products it should be possible to reconstruct the reaction designator in the `<channel>` element.

R:4 Or a `<dcrossSection_dOmega>` element

R:5 Any multiplicities for emitted particles with constant multiplicity. This helps specify the "partition" and the kinematic type.

R:6 The ENDF MT if appropriate (deprecated)

R:7 A flag to denote whether to use relativistic or non-relativistic kinematics when handling this channel

R:8 Optional energy released from processing (may also want this per-product and for `<parameterizeTwoBodyReaction>` data). May want this rather as KERMA.

R:9 Optional forward momentum deposit from processing (may also want this per-product and for `<parameterizeTwoBodyReaction>` data)

R:10 Optional transfer matrix for group-size deterministic calculations

The distributions, etc. (and even the cross section itself) may have sublibrary class specific `<form>`s.

TABLE II. Kinematic types for `<channel>` elements.

kinType	Description
two-body	only two products are emitted per channel, the products are correlated, and only the center-of-mass angular distribution is needed in order to calculate the double-differential distribution
uncorrelated	the products are uncorrelated from each other, and a complete double-differential distribution is required in order to describe each product
activation	no outgoing particle distributions are needed since this is just activation data

DISCUSSION POINT A `<channel>` element could be used by decay data, atomic scattering, thermal neutron scattering and nuclear reaction data to denote the reaction in finer detail than is possible with a simple reaction designator. Within nuclear reaction data it will be used for fast reactions and the resonance region differently. RESOLUTION This adds a extra, unneeded, requirement on the class structure of the code reading this data.

DISCUSSION POINT We can optionally store the Q value and threshold energies, but these are derivable if one knows the identity of the initial and final state particles. Requiring that they be given in a channel potentially introduces an internal consistency error if the values are not kept in sync with any external material property database. RESOLUTION Putting in the Q values in the `<reaction>` is useful, but we shouldn't take them seriously.

D. The `<dcrossSection_dOmega>` element

There are many cases where it is more convenient to write two-body scattering data as $d\sigma(E)/d\Omega$ rather than as a separate cross section $\sigma(E)$ and angular distribution $P(\mu|E)$ where $d\sigma(E)/d\Omega = \sigma(E)P(\mu|E)$. These include:

- Thermal Scattering Law (TSL) data, see Section XI
- Large Angle Coulomb Scattering (LACS) data, see Section IX
- Photo-atomic data described with the Klein-Nishina (KN) formula, see Section VI A

Indeed, in the case of large angle Coulomb scattering (LACS) data, the singularities in the Rutherford cross section prevent us from integrating to find the total cross section $\sigma(E)$. Therefore, we must provide a facility for flagging a reaction as a special parameterized two-body reaction and a facility for storing $d\sigma(E)/d\Omega$.

`<dcrossSection_dOmega>` REQUIREMENTS

R:1 The actual implementation (which depends on the nature of the described data).

R:2 An optional `<documentation>`

E. Cross section: $\sigma(E)$

A `<crossSection>` element would be used by atomic scattering and nuclear reaction data. It is analogous to ENDF's MF=3 or 23 files.

DISCUSSION POINT It was suggested to give cross sections as ratios to e.g. total? This would eliminate sum rule failings. One could then manipulate say the $(n, 2n)$ reaction data without breaking e.g. (n,abs) and (n,tot). RESOLUTION No, this would intentionally introduce synchronization troubles and require rewriting a lot of code to take advantage of.

DISCUSSION POINT Do we allow production cross sections? RESOLUTION No, because the units on a production cross section and a regular cross section are the same so there may be no way to tell if one mis-filed a production cross section, leading to crazy energy balance bugs. However, this should probably be a deprecated derived data for transport requirement.

`<crossSection>` REQUIREMENTS

R:1 A `<crossSection>` element consists of at least one `<form>` containing an `<interp2d>` element with a dependent variable (the cross section itself) given in units of area and independent variable (projectile's incident energy) in units of energy. The first energy point could be (real or effective) threshold or the lowest energy supported by the encapsulating evaluation. A `<crossSection>` is assumed to be zero outside of the specified energy region.

R:2 A specification of the Lorentz frame of the data is not needed, cross sections are Lorentz invariant. That said, the dependent variable of a cross section (the incident energy E could be in the lab or center of mass frame and should be specified).

R:3 A `<crossSection>` element may have multiple `<form>`s.

R:4 All derived `<form>`s have to point to the `nativeData`

R:5 A `<crossSection>` element may have `<documentation>`.

R:6 A `<crossSection>` element may `<link>` to `<covariance>` data. The `<link>` refers to the *original* data, not a derived `<form>`.

R:7 A `<PURR>` table (see section XII)

F. `<reactionProducts>`, `<decayProducts>` and `<product>` elements

A `<reactionProducts>` element lists the reaction `<product>`s. In GND, a `<reactionProducts>` element is referred to as `<outputChannel>`. From the list of products it should be possible to reconstruct the reaction designator in the `<reaction>` element. Similarly a `<decayProducts>` element lists the `<product>`s that another `<product>` may decay into.

`<reactionProducts>` REQUIREMENTS

R:1 List of `<product>` elements

`<decayProducts>` REQUIREMENTS

R:1 List of `<product>` elements

R:2 `kinType` (discussed above)

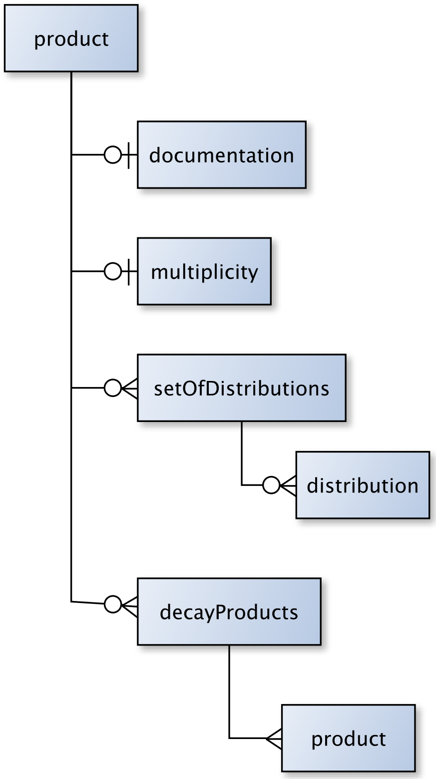


FIG. 8. Overview of a `<product>` element. The “$\overset{\circ}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\circ}{\wedge}$” symbols means there may be zero or more of these child elements.

R:3 `Q`, optional but should sync with material properties

R:4 `lifetime`, optional but handy

The `<product>` structure is given in Figure 8. Each `<product>` should have:

`<product>` REQUIREMENTS

R:1 The particle's identity

R:2 The multiplicity

R:3 An optional ENDF conversion flag

R:4 All outgoing particle distributions for that particle. One should be flagged as the `nativeData`.

G. Multiplicities: $M(E)$

A `<multiplicity>` element would be used by atomic scattering and nuclear reaction data. It may have an alternate name (e.g. `<promptNubar>` for $\bar{\nu}_p$ for prompt fission neutrons). It is analogous to ENDF's MF=12 (for gammas) or MF=1, MT's 452, 455 or 456 (fission $\bar{\nu}$'s).

MULTIPLICITY REQUIREMENTS

R:1 Only use this for non-integer or continuously varying multiplicity.

R:2 Allow common sense element names, e.g. `<promptNubar>`

R:3 A `<multiplicity>` element consists of at least one `<form>` containing an `<interp2d>` with a dependent variable (the multiplicity itself) given in units of number of emitted particles and an independent variable (projectile's incident energy) in units of energy. The first energy point could be (real or effective) threshold or the lowest energy supported by the encapsulating evaluation. The `<multiplicity>` is assumed to be zero outside of the specified energy region. If the `<multiplicity>` is variable and given as a non-integer, it is up to the code using the data to interpret the data correctly

R:4 A specification of the Lorentz frame of the data is not needed, multiplicities are Lorentz invariant. That said, the dependent variable of a multiplicity (the incident energy E could be in the lab or center of mass frame and should be specified).

R:5 A `<multiplicity>` element may have multiple `<form>`s.

R:6 All derived `<form>`s have to point to the `nativeData`

R:7 A `<multiplicity>` element may have `<documentation>`.

R:8 A `<multiplicity>` element may `<link>` to `<covariance>` data. The `<link>` refers to the *original* data, not a derived `<form>`.

H. `<setOfDistributions>` and `<distribution>` elements

The `<setOfDistributions>` contain all of the outgoing probability tables associated with a reaction product. For transport applications this is the $P(\mu, E'|E)$ (and variants). There may be more than one distribution defined (derived vs. original). For uncorrelated data, this will include both angular and the outgoing energy distributions (similar to the combination of MF 4 and 5 in ENDF-6).

In GND, the `<setOfDistributions>` is named `<distributions>` and the `<distribution>` element is named `<component>`.

`<setOfDistributions>` REQUIREMENTS

R:1 List of `<distribution>`s, one of which is the `nativeData` and is flagged by all the others as such. The `<distribution>`s contain the `<form>` elements.

DISCUSSION POINT It was requested that we allow a `<link>` to a distribution rather than a `<distribution>`. This construct would be helpful in storing processed data at various temperatures for Monte Carlo transport where one only heats the cross sections. One could then generate the heated cross sections and store the cross sections in evaluations at different temperatures and connect them with the `metaEvaluation` markup. To reduce the massive redundancy in the outgoing distributions (they never get heated), all the distributions in the heated evaluations could then link back to the zero temperature file's distributions. In fact, it may be more economical to link to the entire `<reactionProducts>` element.

I. Resonances

In this section, we describe resonance data as in ENDF's MF=2, MT=151 files. Our proposed hierarchy is given in Figure 9. This data describes resonances that are observable in neutron cross sections for $E = 0$ eV → 100's keV (or higher for charged particles). In ENDF, this data is only used for neutrons, but should be legal for charged particle reactions and even photonuclear data.

`<resonances>` ELEMENT REQUIREMENTS

R:1 Optional documentation

R:2 A list of the channels referred to in this evaluation. Traditional ENDF SLBW, MLBW and Reich-Moore formats support only capture, elastic, fission, total and a catch-all competitive channel. The R matrix formalism can support *any* two-body final state.

R:3 A resolved resonance region

R:4 Optionally an unresolved resonance region

Both the RRR and the URR share the same master channel list. This aids in reconstruction since the number and kind of channels does not change with energy unless a threshold opens up.

`<channel>` REQUIREMENTS

R:1 The reaction designator; for resonances, this also specifies the "partition" (see Lane and Thomas [5]) II D. It is expected that this designator maps correctly onto one in the `<setOfReactions>` list, otherwise there may be problems when reconstructing resonances.

R:2 If channel not in reaction list, specify particles out, Q, etc.; particles in particle database so have spin, parity, energy, mass, charge, etc, so that the correct `<reaction>` can be added to the `<setOfReactions>` when reconstructing resonances.

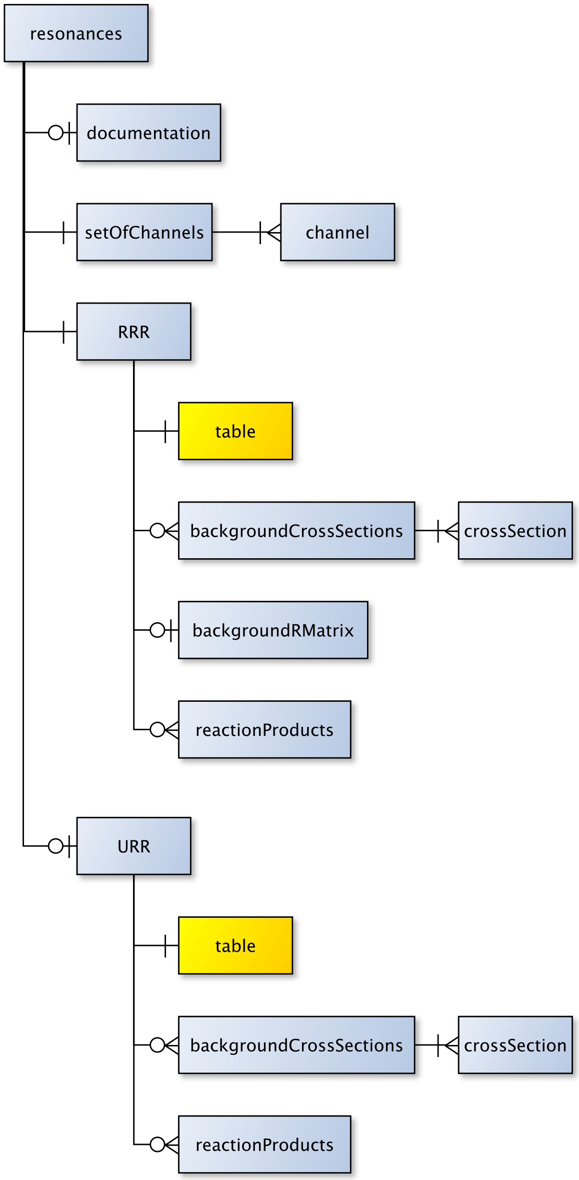


FIG. 9. Our proposed resonance data hierarchy. The "$\overset{\circ}{+}$" symbol means that there may be at most one of these child elements, the "+" symbol means there is exactly one of these child elements and the "$\overset{\circ}{\wedge}$" symbols means there may be zero or more of these child elements.

R:3 All quantum numbers needed to uniquely specify the reaction, this is needed for resonances as well. In particular, the spin s of the channel, the orbital angular momentum l, the total angular momentum J and any other quantum numbers.

R:4 Configurable channels to denote whether corresponds to actual two-particle final state or effective one (as in fission or competitive channels). Only two-body channels can be used to compute angular distributions; need to be able to flag “effective” channels

- R:5 List s of each resonance (resolves and ENDF ambiguity).
- R:6 Boundary parameter B_c
- R:7 Channel radius vs true channel radius
- R:8 Sign of reduced width
- R:9 To override the defaults, optionally specify
  - R:9.a phase $\varphi_c(E)$
  - R:9.b shift $S_c(E)$
  - R:9.c penetrability $P_c(E)$
  - R:9.d hard-sphere radii a_c (with potential dependence on energy). Likely need to be able to break it into multiple regions so that e.g. the RRR can have a constant one while the URR can have an energy dependent one.

DISCUSSION POINT Would configurable ignored or collapsed channels (like γ ones in Reich-Moore approximation)? Or is the Reich-Moore approximations on photons the only one that makes sense in practice?

DISCUSSION POINT Would user-definable (possibly fake) quantum numbers, e.g. fission mode, be useful? Must define whether combine using angular momentum adding rules (for BB) or incoherently. K might be appropriate (esp. for deformed nuclei).

DISCUSSION POINT The channel wish list is very big. That said, a `<channel>` has all of the attributes of the `<reaction>` element. Does it make sense to completely separate the `<channel>` concept of the `<reaction>` element? RESOLUTION While this is still under debate, there appears to be little gain in doing this, but it does confuse the requirements discussion.

To understand the hierarchy of resonance data, it is helpful to understand a little about R matrix theory. In it, we divide the universe into the inside of a box and the outside of the box. Inside the box is the reaction zone, where all the interesting nuclear (or other) reaction business occurs (see Figure 10). We have little chance of modeling what goes on the box correctly without a lot of work. Outside the box we write all incoming and outgoing relative two-body scattering states in a basis of analytic wave functions, usually taken to be free ones. We then match wave functions on the box boundary. This matching is done in a clever way involving Bloch surface operators on the box boundary and from this we arrive at a Green’s function of the projected Bloch-Schödingier equation, also known as the R matrix:

$$R_{cc'} = \sum_{\lambda} \frac{\gamma_{\lambda c}\gamma_{\lambda c'}}{E_\lambda - E} \tag{1}$$

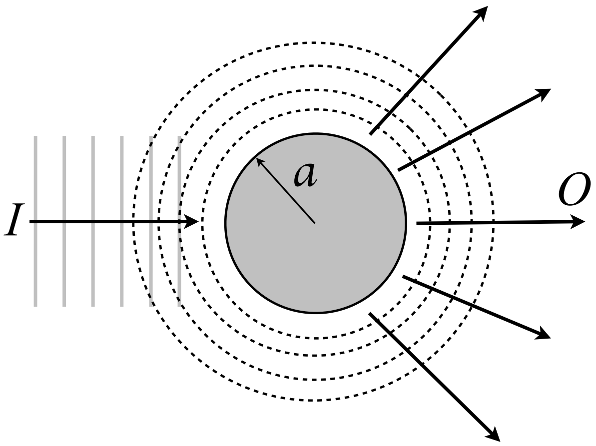


FIG. 10. A cartoon representation of R matrix theory. We first divide the universe into inside a box and out. Inside the box is the reaction zone, that we have little chance of modeling correctly without a lot of work. Outside the box we write all incoming and outgoing relative two-body scattering states in a basis of analytic wave functions, usually taken to be free ones. We then match wave functions on the box boundary.

These factors $\gamma_{\lambda c}$ are the reduced widths of the channel c, E_λ becomes the resonance energy (it is a pole in the Laurent series expansion of the Green’s function) and λ is the resonance (pole) index. The channel index c contains all the quantum numbers needed to describe the outgoing two-particle state and all of those quantum numbers are described in the `<channel>` element markup above.

DISCUSSION POINT Putting the R matrix itself in the format is silly because we’d be replacing a set of resonance parameters with basically a reconstructed version (see Eq. (1)), but packed in an complicated and not very usable fashion. If you want a reconstructed version, use point-wise cross section tables. RESOLUTION Agreed.

DISCUSSION POINT Kapur-Peiers and/or Wigner-Eisenbud? Both approaches use different boundary parameters B_c. They are mathematically equivalent, but the RRR approximations in ENDF all use Wigner-Eisenbud formulation. RESOLUTION No, because in Kapur-Peiers, one sets the boundary constant $B_c = L_c$. This leads to a complex pole E_λ, forcing us to mix data types (`complex` vs. `float`) in the `<table>` element in the `<RRR>` element.

With the R matrix, it is possible to compute exactly the channel-channel scattering matrix $U_{cc'}$:

$$\begin{aligned} U_{cc'} =& e^{-i(\varphi_c+\varphi_{c'})}\sqrt{P_c}\sqrt{P_{c'}} \\ & \times \{[\mathbf{1}-\mathbf{R}(\mathbf{L}-\mathbf{B})]^{-1}[\mathbf{1}-\mathbf{R}(\mathbf{L}^*-\mathbf{B})]\}_{cc'} \end{aligned} \tag{2}$$

where The logarithmic derivatives of the outgoing channel function are

$$L_c \equiv a_c \frac{O'_c(a_c)}{O_c(a_c)} = \left[r_c \frac{\partial \ln O_c}{\partial r_c}\right]_{r_c=a_c} \tag{3}$$

and we write

$$L_c = S_c + iP_c. \tag{4}$$

The penetration factor is $P_c = \Re L_c$ and the shift factor is $S_c = \Im L_c$. Both take their names from their function in the simple complex square well scattering model. φ_c is the phase factor

$$\varphi_c \equiv \arg O_c(a_c) = \arctan \frac{\Im O_c(a_c)}{\Re O_c(a_c)} \tag{5}$$

The constant B_c is the so-called "boundary parameter" which must be specified to correctly compute the scattering matrix, but is not always clearly given.

With the scattering matrix, one can compute the channel cross sections, the total cross section, and all angular distributions. Angle integrated cross section can be written as sum over all entrance channels $c = \{\alpha J \ell s\}$ and exit channels $c' = \{\alpha' J' \ell' s'\}$ that lead from partition α to α':

$$\sigma_{cc'} = \pi \lambdabar_c^2 g_c |\delta_{cc'} - U_{cc'}|^2 \tag{6}$$

So, the total cross section for channel c is

$$\sigma_c \equiv \sum_{c'} \sigma_{cc'} = 2\pi \lambdabar_c^2 (1 - \Re U_{cc}) \tag{7}$$

The factor of g_c is the probability of getting the correct J from the spins of the collision partners (according to Fröhner) and is $g_c = (2J+1)/((2i+1)(2I+1))$.

The Blatt-Beidenharn to construct the $d\sigma_c/d\Omega$ for the (usu.) elastic channel [8]. Is valid for any two-body system in the center-of-momentum ($\equiv$ center of mass usu.). Spin algebra may only be valid in non-relativistic limit (HAVE TO CHECK). Although $d\sigma_c/d\Omega$ can be written as a Lorenz covariance quantity, we will write the outgoing dependence on angle in the pair center of mass frame and the incident energy in the laboratory frame.

For particles with arbitrary spin, we have

$$\frac{d\sigma_{\alpha,\alpha'}(E)}{d\Omega} = \frac{1}{k^2(2i+1)(2I+1)} \sum_{s,s'} \sum_{L=0}^{\infty} B_L(\alpha s, \alpha' s'; E) P_L(\mu) \tag{8}$$

and

$$\begin{aligned} B_L(\alpha s, \alpha' s'; E) =& \frac{(-)^{s-s'}}{4} \sum_{J_1,J_2} \sum_{\ell_1,\ell_2} \sum_{\ell_1',\ell_2'} \bar{Z}(\ell_1 J_1 \ell_2 J_2 s L) \bar{Z}(\ell_1' J_1 \ell_2' J_2 s' L) \\ & \times (\delta_{\alpha\alpha'} \delta_{\ell_1 \ell_1'} \delta_{ss'} - U^{J_1}_{\alpha \ell_1 s, \alpha' \ell_1' s'}(E))^* (\delta_{\alpha\alpha'} \delta_{\ell_2 \ell_2'} \delta_{ss'} - U^{J_2}_{\alpha \ell_2 s, \alpha' \ell_2' s'}(E)) \qquad (9) \\ =& \frac{(-)^{s-s'}}{4} \sum_{c_1=\{\alpha \ell_1 s_1 J_1\}} \sum_{c_1'=\{\alpha' \ell_1' s_1' J_1'\}} \sum_{c_2=\{\alpha \ell_2 s_2 J_2\}} \sum_{c_2'=\{\alpha' \ell_2' s_2' J_2'\}} \bar{Z}(\ell_1 J_1 \ell_2 J_2 s L) \bar{Z}(\ell_1' J_1 \ell_2' J_2 s' L) \\ & \times \delta_{ss_1} \delta_{s's_1'} \delta_{J_1 J_1'} \delta_{ss_2} \delta_{s's_2'} \delta_{J_2 J_2'} (\delta_{c_1 c_1'} - U_{c_1 c_1'}(E))^* (\delta_{c_2 c_2'} - U_{c_2 c_2'}(E)) \qquad (10) \end{aligned}$$

where

$$\bar{Z}(\ell_1 J_1 \ell_2 J_2, sL) = \sqrt{(2\ell_1+1)(2\ell_2+1)(2J_1+1)(2J_2+1)} (\ell_1 \ell_2 00, L0) W(\ell_1 J_1 \ell_2 J_2, sL) \tag{11}$$

and $W(\ell_1 J_1 \ell_2 J_2, sL)$ is a Racah coefficient.

I use the notation $\sum_c = \{\alpha \ell s J\} = \sum_\ell \sum_s \sum_J$. ENDF uses the notation $\sum_c = \sum_\ell \sum_s$, so ENDF needs an extra sum over J_1 and J_2.

This is detailed in several places including [5–8]. Given the mathematical completeness of the theory, it is no surprise that we mostly just view the R matrix parameters as simple fit parameters and then essentially get all of this for free.

DISCUSSION POINT We comment that the R matrix approach works for *any* two-body reaction, relativistic or not, as long as the incoming and outgoing relative states can be clearly defined. In the nuclear data community we often forget this fact and so lose the ability to represent our charged particle data in an R matrix inspired form, reducing the quality and scope of data available to several communities who need it:

- Inertial Confinement Fusion community needs all sorts of charged particle incident data
- Astrophysical community needs the (p, γ) reaction among many others
- For Nuclear Resonance Fluorescence, need to support (γ, γ') data
- Primary gammas are a complete mess. How do we handle these? ENDF approach is a kludge. A multi-step R matrix approach could handle it.

E_λ	$\Gamma_{\lambda tot}$	$\Gamma_{\lambda c0}$	$\Gamma_{\lambda c1}$	$\Gamma_{\lambda c2}$	...
eV	eV	eV	eV	meV	...
1.23	9.433	0	2.33E-03	7.1	...
1.46	4.833	0	2.33E-03	4.6	...
3.45	1.78	1.78	0	0	...
...	...	...	...	...	

FIG. 11. Sample table of resonance parameters.

RESOLUTION Although Lane and Thomas provide a mechanism for doing this [5], GND provides a `decayProduct` markup that fulfills the same need. **DISCUSSION POINT** Is this enough to handle (*,gf) and/or fission reactions through class II states?

1. *Resolved Resonances*

The ENDF format supports several different approximations to the full R matrix theory. It also supports background cross sections to add into the reconstructed resonances and a background R matrix to build in correctly the effects of distant resonances (replacing $R_{cc'} \to R_{cc'} + R_c^{back}\delta_{cc'}$). In ENDF, what is stored is the resonance energy and the resonances widths $\Gamma_{\lambda c} = 2P_c\gamma^2_{\lambda c}$.

DISCUSSION POINT Option to store width amplitudes $\gamma_{\lambda c}$ instead of widths. No sign confusions, they are not energy dependent and they do not vanish at threshold. (Thank you Fröhner for this suggestion). **RESOLUTION** Means we'd need to have an excellent grasp on what the penetrabilities really mean for γ (we know what they are for neutrons and charged particle channels). We'll still need $\Gamma_{\lambda c}$ for fission and competitive channels too since there is no notion of penetrability in those cases. We'd also need to know the relativistic version of the penetrabilities too. One could tabulate effective penetrabilities in the `<channel>` such that $\Gamma_{\lambda c}$ comes out right.

DISCUSSION POINT Channel major arrangement or maintain resonance major arrangement? What I mean is, are the rows in the "table" mean one row/resonance with all the channels as columns as in ENDF? Or do we switch to having a list of channels at the top with a list of resonances associated with each channel? Either way the matrix $\Gamma_{\lambda c}$ is sparse.

RESOLVED RESONANCE REGION REQUIREMENTS

R:1 The actual `<table>` of resonance parameters. The simplest arrangement is column is shown in Figure 11; We may need to also tabulate the `<link>` of the `<table>` column to the `<channel>` element.

R:2 LMax (an NLS-like thing) to specify the maximum ℓ value to sum to so as to get potential scattering correct

R:3 A flag to denote the approximation used in the interpretation of the resonance parameters. In ENDF's LRF=7 format, this is analogous to the KLRF flag. Supported approximations should include:

- R:3.a Pure potential scattering with either hard sphere or tabulated energy and/or ℓ-dependent scattering radius. Allows cross section and angular distribution calculation.
- R:3.b Single Level Breit Wigner (SLBW) approximation with 1 resonance. Allows cross section and angular distribution calculation.
- R:3.c ENDF style SLBW. Allows only cross section calculation.
- R:3.d Multi Level Breit Wigner (MLBW). Allows cross section and angular distribution calculation.
- R:3.e ENDF style MLBW. Allows cross section and angular distribution calculation for elastic reactions.
- R:3.f Reich-Moore. Allows cross section and angular distribution calculation.
- R:3.g ENDF style Reich-Moore. Allows cross section and angular distribution calculation for elastic reactions.
- R:3.h Full R matrix. Allows cross section and angular distribution calculation.

R:4 All background R matrix options `KBK` of ENDF. The ENDF manual lists several approaches:

- R:4.a `KBK=0` Dummy resonances
- R:4.b `KBK=1` Tabulated complex function of energy
- R:4.c `KBK=2` Fröhner's parameterization
- R:4.d `KBK=3` Tabulated phase shifts

Because the ENDF approximations to R matrix theory often result in mis-matches with experimental cross section data, a background cross section is sometimes added to the reconstructed resonance cross section. There are several ways to affect this correction, either with a set of fake resonances (with e.g. $E_\lambda < 0$), energy dependent scattering radii or modified phase factors (the last two can be implemented in the `<channel>` element). If one chooses to tabulate the background directly, use the `<backgroundCrossSection>` markup:

BACKGROUND CROSS SECTIONS REQUIREMENTS

R:1 Not given for (n,tot), but associated with the actual reaction it is getting added.

R:2 They should be kept with the RR, not the high energy file so that the association is explicit.

R:3 Go in the `<crossSection>` element of the `<backgroundCrossSection>` element.

R:4 When do reconstruction, all goes into a smooth background

R:5 May have multiple background regions as a consequence, the original one, associated with RR parameters (the native data) and the reconstructed one

DISCUSSION POINT Flag fake resonances somehow?

DISCUSSION POINT Since several of the ENDF RR approximations DO NOT support angular distributions, the ENDF format provides the ability to store those distributions separately. Should we support this too? RESOLUTION Reluctant agreement.

DISCUSSION POINT Need to clarify rules for the resolved and unresolved region widths for threshold reactions.

DISCUSSION POINT We will need tests to ensure consistency between the `<channel>`s, the `<reaction>`s in the `<setOfReactions>` and between the `<channel>`s and the `<RRR>` and `<URR>` columns.

2. *Unresolved Resonances*

This is an averaged version of R-matrix motivated parameterization. What is stored is not the resonance parameters, but ensemble averages of them: averaged first over ensembles of imagined resonances, then over the width distributions of the resonances. The widths are assumed to be distributed according to a χ^2 distribution with a channel dependent number of degrees of freedom. Additionally, the average inter-resonance spacing $\Delta(E)$ and the numbers of degrees of freedom for the χ^2 distributions are needed. In ENDF, the resonances are assumed to be in the SLBW approximation before averaging leading to the particular parametric form of the cross sections in the ENDF manual.

DISCUSSION POINT ENDF assumes the resonances are SLBW. CALENDF and other codes can use other parameterizations. Should the approximation be a flag too? RESOLUTION Agreed.

UNRESOLVED RESONANCE REGION REQUIREMENTS

R:1 Need number of degrees of freedom associated with each channel in the channel listing

R:2 Need a `<table>` of URR parameters. This table must include columns for incident energy, mean level spacing, average widths for all channels.

R:3 ENDF assumes SLBW, allowing the construction of average cross section and PURR tables. Allow all approximations supported for the RRR.

R:4 An `<axis>` to determine how to interpolate in incident energy among the average parameters.

R:5

DISCUSSION POINT NJOY can compute cross section probability distributions $P(\sigma_x|E)$ for all $x \in [\gamma, el, tot, f, ...]$ with the PURR module. Should we have a spot for it in the hierarchy? RESOLUTION Yes, but PURR tables for $P(\sigma_x|E)$ for all $x \in [\gamma, el, tot, f, ...]$ reactions do not go in the URR table. They go in the reconstructed cross section element in the appropriate `<reaction>`'s `<crossSection>`.

3. *How to use the RRR markup for special circumstances*

For just a scattering radius: there is no special markup, just use the RR scheme, but dont have any resonances and set NLS to 0. Specify the scattering radius.

For Potential Scattering: dont have special markup, just use RR scheme, but dont add resonances or make NLS >> number of specified resonances. Either way, set NLS.

IV. THE OTHER TOP LEVEL: ONE COLLECTION OF COVARIANCES

Users of nuclear data need covariance data to quantify uncertainty on the metrics of importance in their specific application. These metrics (such as k_{eff} in a criticality calculation) may have a deep dependence on the underlying data. Our users actually use the covariances with deterministic group-wise methods (using the “Sandwich formula” below) or with Monte Carlo techniques. We must do what we can to simplify both modes of covariance use.

The legacy ENDF manner in which nuclear data covariance are stored is complex: the ENDF manual [2] take over 80 pages to describe seven distinct types of data. Arguably, there should be one “simple” format to govern them all, after all a covariance matrix is, at its heart, just a matrix.

That said, we must deal with covariances not just within an observable, but across observables and evaluations. These covariances can also be quite large, far exceeding the size of the evaluations to which they refer. Therefore, as in GND, we recommend keeping the covariances in separate files. In Figure 12 we show the structure of the top level of a file containing covariances.

DISCUSSION POINT Keep covariance and underlying data separate, but associated. RESOLUTION Agreed.

`<setOfCovariances>` REQUIREMENTS

R:1 Optional documentation

R:2 One or more covariances, either in `<covariance>` elements or `<weightedSumOfCovariances>` elements

A. Covariance Definitions

When we measure a quantity x_i, we assume that we do not actually get the “true” value given by Nature, but

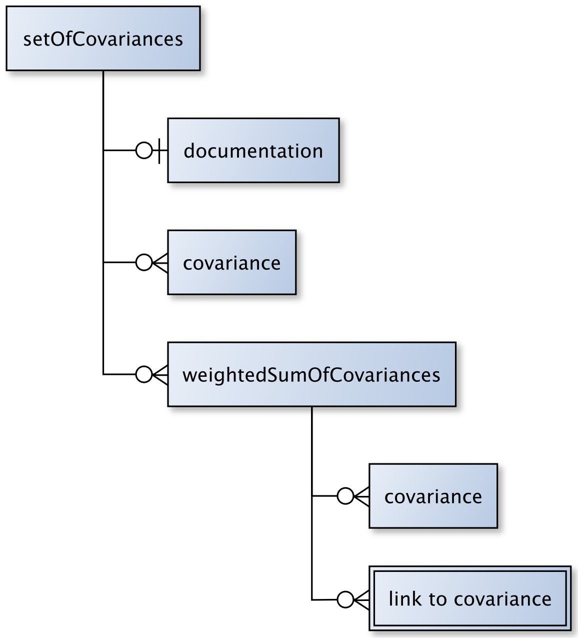


FIG. 12. Top level of a file containing (hopefully related) covariance data. The “$\overset{\circ}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\circ}{\wedge}$” symbols means there may be zero or more of these child elements.

rather one sample from a probability distribution function (PDF). Depending on the nature of the observable, the PDF might be Normal or Log-Normal [4] or something else. For our purposes, we will assume that the PDF is either Normal or Log-Normal since the Central Limit Theorem guarantees that in the limit of large numbers of samples the peak of any PDF can be well approximated by a Normal distribution. We also include Log-Normal as an option since it forces values of an observable to be positive definite but otherwise behaves like a Normal distribution [4].

For our measurement x_i, the PDF has an expectation value of $\langle x_i \rangle = \int dx_i PDF(x_i) x_i$ and this would be stored in the ENDF file. The uncertainty on x_i is Δx_i. We define:

- **covariance**:

$$\begin{aligned}\mathrm{cov}x_{ij} &= (\Delta^2 x)_{ij} \qquad (12)\\ &= \int dx_i dx_j PDF(x_i, x_j)(x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \qquad (13)\\ &= \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle \qquad (14)\end{aligned}$$

- **variance**:

$$\mathrm{var}x_{ij} = \mathrm{cov}x_{ii}\delta_{ij} = (\Delta x_i)^2 \delta_{ij} \tag{15}$$

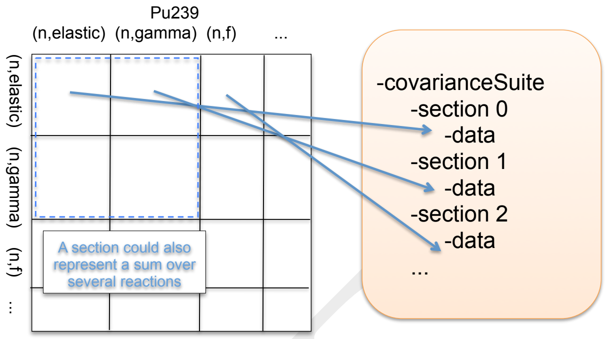


FIG. 13. Block construction of a covariance matrix.

- **uncertainty**:

$$\mathrm{unc}x_i = \sqrt{\mathrm{var}x_{ii}} = \Delta x_i \tag{16}$$

- **correlation**:

$$\begin{aligned}\mathrm{corr}x_{ij} &= \mathrm{cov}x_{ij}/\mathrm{unc}x_i\mathrm{unc}x_j \qquad (17)\\ &= \mathrm{cov}x_{ij}/\Delta x_i \Delta x_j \qquad (18)\end{aligned}$$

- **relative covariance**:

$$\mathrm{rcov}x_{ij} = \mathrm{cov}x_{ij}/\langle x_i \rangle \langle x_j \rangle \tag{19}$$

Here, the covariance is a real, symmetric, positive $N \times N$ matrix. A covariance may be sparse or dense or even (band) diagonal.

B. Packing a covariance matrix

How can allow evaluator to correlate any two things? Well, a covariance is just a matrix so we can build one up using matrix block composition, as illustrated in Figure 13. In other words

$$\text{one block} = (\text{a row data set}) \times (\text{a column data set}) \tag{20}$$

For this composition to make sense then, we must be able to associate a row or a column of a block with the correct underlying data. This is illustrated in Figure 14. Because the data itself is contained in a `<form>` element, the `<link>` must point directly there.

DISCUSSION POINT If covariance links to data and data links to covariance, we will need a link consistency checker. RESOLUTION Agreed.

We are now in a position where we can define the structure of a `<covariance>` element. The structure is shown in Figure 15. In it, we note the row and column markups to ensure linkage to the original data. We also note the presence of an `axis` element. This determines how the underlying data in the original `<form>` element packs into

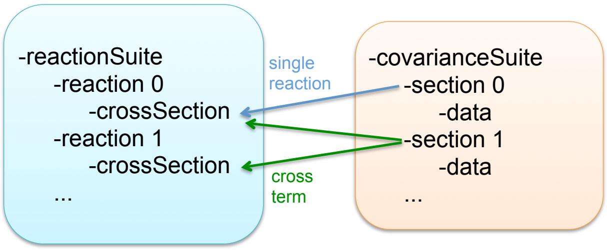


FIG. 14. Coupling between reaction data and the corresponding covariance

the covariance matrix itself. This now is a technical challenge for multivariate (two- and three- dimensional) data and is a discussion topic to be had in the low level container discussion.

To understand the role of the `<axis>` element in this context, it is useful to look at the example of the covariance on a cross section. In ENDF, a cross section $\sigma(E)$'s covariance is given group wise as $\Delta^2\sigma_{ij}$. The group boundaries can be thought of as forming a basis function expansion where the basis functions are window functions:

$$B_i(E) = \begin{cases} 0 & E < E_i \\ 1/(E_{i+1} - E_i) & E_i \le E \le E_{i+1} \\ 0 & E > E_{i+1} \end{cases} \qquad (21)$$

To write the continuous covariance on the cross section $\Delta^2\sigma(E_1, E_2)$, we write

$$\Delta^2\sigma(E_1, E_2) = \sum_{ij} B_i(E_1)\Delta^2\sigma_{ij}B_j(E_2) \qquad (22)$$

Thus, the basis function encodes the interpolation rule (in this case group-wise) and is encoded in the `<axis>` element for the E direction. The `<axis>` element also defines the packing rule for the underlying covariance in that it describes which energy in this case maps to what row index in the covariance matrix $\Delta^2\sigma_{ij}$.

DISCUSSION POINT Who will be responsible for determining the data packing rules for multivariate data? To date in GND, C. Mattoon and D. Brown have iterated a little but the result was very ENDF-like. Also, for multivariate data, more than one `<axis>` are needed. Finally, while an elect packing scheme has been set for 3D data (such as $P(\mu|E)$ in MF=34 and $P(E'|E)$ in MT=35), no packing scheme has been set for 4D data.

DISCUSSION POINT The packing scheme for parametric data (say for resonances) will require a special type of axis that simply lists which parameter is assigned what row/column index in the matrix.

`<covariance>` REQUIREMENTS

R:1 Optional documentation

R:2 A `<row>` element which includes a `<link>` to the original underlying data in a `<form>` element and the `<axis>` element to decide the data to cell in the matrix mapping.

R:3 For on-diagonal blocks of covariance, the `<column>` is the `<row>`. For off-diagonal blocks, the `<column>` must also be specified in the same format as the `<row>` element.

R:4 The `<matrixData>` element containing the matrix itself.

Within the covariance element is the matrix data itself. See Figure 16.

`matrixData` REQUIREMENTS

R:1 Flag to denote whether this covariance is absolute or relative

R:2 Flags denoting any normalization constraints on the matrix (i.e. for covariance on probability distributions, e.g. $P(\mu, E)$).

R:3 The `<matrix>` itself or a `<matrixSandwich>` (see below)

DISCUSSION POINT Should the storing of a correlation matrix and uncertainties separately be allowed? This might make plotting easier at the minor expense of an increased level of bookkeeping. RESOLUTION This is still under discussion.

C. Weighted sums of covariance

How can an evaluator break up the covariance into components, say statistical errors from a fit and systematic errors arising from experimental normalizations? This is easily addressed by adding a `<weightedSumOfCovariances>` element since a covariance matrix is just a matrix and the some of two covariance matrices is still a covariance matrix. Incidentally, this can be used to encode the sum rules of a cross section into the covariance itself.

DISCUSSION POINT Link bookkeeping troubles if allow `<weightedSumOfCovariances>` construction for cross correlations? Have to require `<link>` matching between all parts of covariance.

DISCUSSION POINT Should we allow a `<weightedSumOfCovariances>` within the "Sandwich Formula" below? This would allow for an even more flexible covariance construction, but may be difficult to code in processing codes.

`<weightedSumOfCovariances>` REQUIREMENTS

R:1 Numerical weights (in ENDF, these are just `floats` of a component

R:2 Either the components as `<covariance>`s or `<link>`s to `<covariance>`s

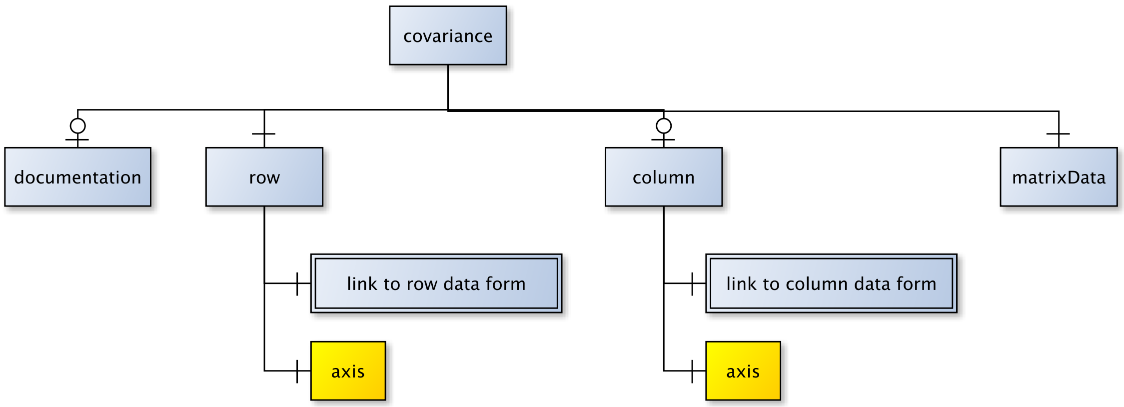


FIG. 15. The “$\overset{\bigcirc}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\bigcirc}{\wedge}$” symbols means there may be zero or more of these child elements.

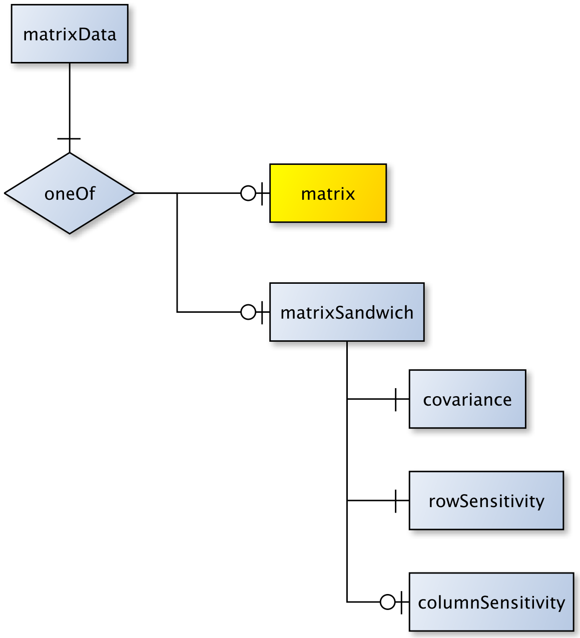


FIG. 16. Arrangement of parts of a `<matrixData>` element. The “$\overset{\bigcirc}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\bigcirc}{\wedge}$” symbols means there may be zero or more of these child elements.

D. The “Sandwich Formula”

Often times we have a parameter f_i that we want the covariance on, but it depends on something else, say $\vec{x}$ and it would be much more efficient to store the covariance on $\vec{x}$ directly. A case in point is the RRR parameters. The reconstructed cross section from tens of resonance parameters may have thousands of energy points to achieve a reasonable accuracy.

If $\vec{f}(\vec{x})$, the $\mathrm{sens}_{ij} = \partial f_i(\langle\vec{x}\rangle)/\partial x_j$ is the sensitivity matrix. Assuming that

$$f_i(\vec{x}) \approx f_i(\langle\vec{x}\rangle) + \sum_j \frac{\partial f_i(\langle\vec{x}\rangle)}{\partial x_j}(x_j - \langle x_j\rangle) \tag{23}$$

is a good approximation to the variation of $\vec{f}(\vec{x})$ around $\langle\vec{x}\rangle$, we can evaluate the covariance of f using the “sandwich formula”:

$$\mathrm{cov} f_{ij} = \sum_{i'j'} \mathrm{sens}_{ii'} \mathrm{cov} x_{i'j'} \mathrm{sens}_{j'j} \tag{24}$$

The “sandwich formula” can be reframed in terms of the relative covariance

$$\mathrm{rcov} f_{ij} = \sum_{i'j'} \mathrm{rsens}_{ii'} \mathrm{rcov} x_{i'j'} \mathrm{rsens}_{j'j} \tag{25}$$

provided

$$\mathrm{rsens}_{ij} = x_j \frac{\partial f_i(\langle\vec{x}\rangle)}{\partial x_j} = \frac{\partial f_i(\langle\vec{x}\rangle)}{\partial(\ln x_j)} \tag{26}$$

The “sandwich formula” provides the scheme for deterministic uncertainty propagation.

DISCUSSION POINT In many cases, the sensitivity on model parameters can precomputed. In this case, we may not need to store the sensitivity matrix itself. Should we allow this? It makes for smaller files, but shifts the burden of computing the sensitivities to the processing codes. RESOLUTION Yes, this is already the case for RRR parameters.

As an aside, the covariance admits an eigendecomposition into N eigenvalues λ_i with eigenvectors $\vec{v}_i$. The covariance can be diagonalized in the eigenbasis by

$$\mathrm{cov} x_{ij} = (\vec{v}_k)_i \lambda_k (\vec{v}_k)_j \tag{27}$$

This is the "sandwich formula" again, but here the eigenvalues play the role of the sandwiched covariance matrix and the eigenvectors play the role of the sensitivity matrix. Often times the effective rank of a matrix N_{eft} is much smaller than the actual rank N because many of the eigenvalues are sufficiently close to zero that they may be neglected. The process of taking the main eigenvalues is called principal component analysis (PCA). Thus the "sandwich formula" storage scheme can be used to efficiently pack covariance matrices even in the absence of underlying parameter dependencies but using PCA.

So, to support the "Sandwich Formula", we must define the structure of a `<matrixSandwich>` and a `<sensitivity>`:

`<matrixSandwich>` REQUIREMENTS

R:1 The underlying parameter `<covariance>`

R:2 A `<sensitivity>` for the rows of the covariance

R:3 If the block of the matrix is off-diagonal, a `<sensitivity>` for the column as well.

`<sensitivity>` REQUIREMENTS

R:1 Optionally a `<documentation>` element

R:2 The `<matrixData>` for the sensitivity matrix

R:3 A `<column>` with a `<link>` pointing to the `<column>` element's `<axis>` of the underlying parameter covariance matrix. This also defines the packing of the sensitivity matrix since we want them to match up for the matrix multiplication.

R:4 A `<row>` that mimics the `row` one would get if we were storing the full covariance on the derived data. Therefore we need an `<axis>` element to determine the packing of the sensitivity matrix and a (possibly fake) link to the `<form>` of the derived data.

E. Monte Carlo Sampling

How can one use a covariance to generate realizations for a Monte Carlo approach to uncertainty quantification? Well, suppose we have some $\vec{x}$ with a Normal PDF $P(\vec{x})$ specified by the mean $\langle\vec{x}\rangle$ and covariance $\text{cov}x_{ij}$. To find the expectation value of a function $f(x)$, we do

$$\langle f\rangle = \int d\vec{x}P(\vec{x})f(\vec{x}) \approx \frac{1}{N}\sum_R f(\vec{x}_R)P(\vec{x}_R) \qquad (28)$$

Where the sum is a sum over realizations of $\vec{x}$ drawn from the PDF. To generate the realizations, we use principal component analysis (PCA) again:

$$\vec{x}_R = \langle\vec{x}\rangle + \sum_i \xi^i_R \vec{v}_i\sqrt{\lambda_i} \qquad (29)$$

Where ξ^i_R is drawn from a (log) normal distribution.

DISCUSSION POINT Should we support ensembles of evaluations or evaluation parts (like TMC or list-mode output)? Would need index of realizations maybe. Could this be handled using the `metaEvaluation` scheme? RESOLUTION One would need reasonable number of samples N_{samp} for each of the i directions. So, need $(N_{samp})^{N_{eff}}$ samples to effectively sample all of $\vec{x}$'s PDF to reliably propagate uncertainty. Not really an effective savings of space. However, with `nativeData` scheme should be able to accommodate variations.

V. GLUING TOGETHER EVALUATIONS

There is a relatively common need to "glue" together evaluations to make new "effective" or "meta" evaluations. This is often used to connect evaluations from different physical regimes or to assemble new reusable materials in input deck specifications. For example:

- In LANL's MCNP code system, the `xsdir` file allows one to connect the thermal neutron scattering data with the neutron nuclear reaction data and even various high energy models such as CEM. See e.g. Figures 18 and 19 .
- The LLNL transport codes AMTRAN and Mercury both allow one to define target macros to describe the material in a zone.
- ORNL's SCALE package contains a pre-built material composition database.
- At AECL, there is another, similar, facility to connect thermal neutron scattering data at different temperatures and even different phases of the target material.

There are other uses for being able to connect evaluations together:

- Defining elemental evaluations
- Grouping data on same target, but heated to different temperatures
- Defining generic fission fragments through a weighted average of fission fragment evaluations
- Putting together the parts of a TSL evaluation at fixed temperature, but including all the scatterers.
- Defining common material definitions. This helps answer the question "Which concrete?"

Ideally, these could be shared but rarely are because of the wildly different formats used by various projects. This need for "gluing" together evaluations is so common that we should seriously consider supporting it.

The idea of a `<metaEvaluation>` is straightforward. One uses a set of `<axis>` elements to define the grid in

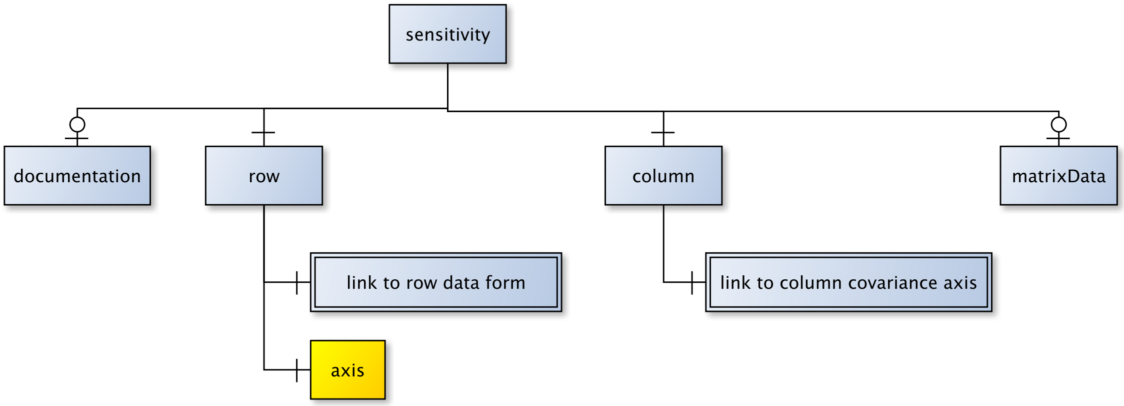


FIG. 17. The “$\overset{\circ}{+}$” symbol means that there may be at most one of these child elements, the “+” symbol means there is exactly one of these child elements and the “$\overset{\circ}{\wedge}$” symbols means there may be zero or more of these child elements.

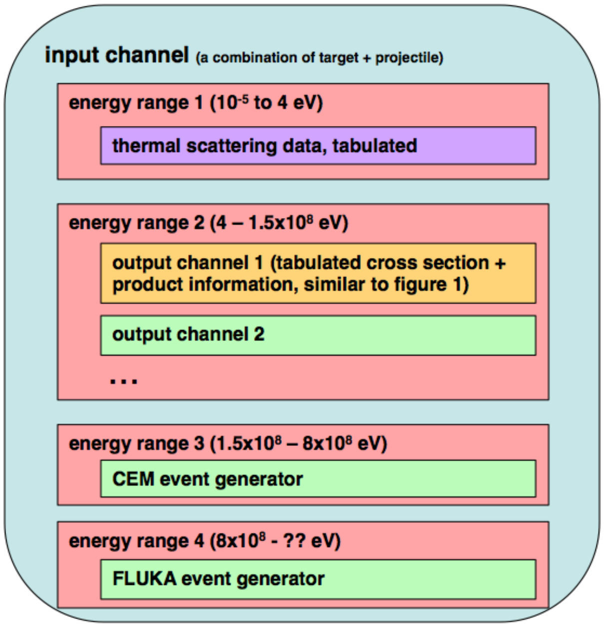


FIG. 18. Gluing together different models from different energy regimes.

some parameter space one wishes to populate with evaluations. The `<axis>`'s could be temperature, incident energy, pressure, etc. The `<axis>` element defines the group boundaries in the parameter space. The `<axis>` elements also define the interpolation scheme to be used in that parameter's direction, but in practice the interpolation information will probably be ignored because each project defines on there own how e.g. they intend to step up in temperature.

`<metaEvaluation>` REQUIREMENTS

R:1 An `projectile` attribute to define what projectile this `<metaEvaluation>` is only valid for (say TSL+fast gluing only for neutrons).

R:2 `<axis>` elements to define the grid in which the evaluations will be inserted

R:3 `<referredEvaluation>` which links to an `<evaluation>` or another `<metaEvaluation>`. This allows one to reuse definitions (so the natural hydrogen `<metaEvaluation>` can be used in the assembly if many different TSL+fast `<metaEvaluation>`s). `<referredEvaluation>` has the following additional attributes:

R:3.a `stoichiometricFraction` tag lets one specify chemical or isotopic make-up if multiple `<referredEvaluation>`s are allowed

R:3.b `stoichiometricFraction` better add up to 1!

R:3.c `axisCoords` to specify where in the grid an evaluation sits.

R:4 Outside of parameter ranges in `axis` tags, the `<metaEvaluation>` does not exist

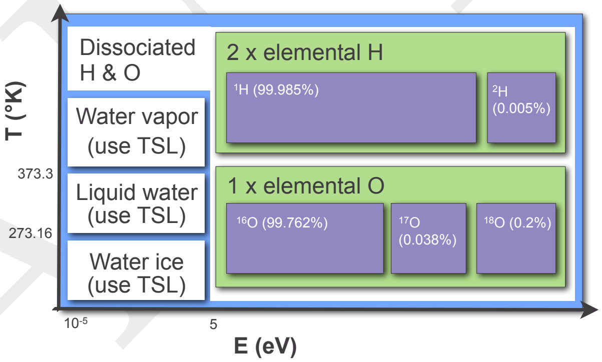


FIG. 19. Gluing together thermal neutron scattering with the higher energy nuclear reaction data.

R:5 `<metaEvaluation>` only valid for listed projectile

R:6 Need tests to make sure every region in `<axes>` covered by a `<referredEvaluation>`.

These are illustrated in Figure 20.

DISCUSSION POINT Is it possible to use say atomic weights instead of `stoichiometricFraction` to specify fractional composition of a material? This would simplify use in several transport code input decks. RESOLUTION This requires a material database shared by all or there will be mistakes. Additionally, testing that the sum of fractional compositions sum to one will be difficult.

DISCUSSION POINT Should the the `<referredEvaluation>` also contain a `nativeData` attribute to handle Doppler broadened data better? RESOLUTION No, this should be done at the `<evaluation>` level so the `nativeData` information is associated with the evaluation file itself and not somewhere else.

DISCUSSION POINT Is there a need for a separate `<metaTarget>` concept to handle arbitrary projectiles so we needn't maintain 7-8 different (but nearly identical) element specifications? RESOLUTION Good point. Maybe allowing `any` or `*` as a projectile would serve this purpose. Alternatively, we could make the `projectile` attribute optional and if it is not present than the `<metaEvaluation>` is valid for all projectiles. Either way the links to the actual evaluation become meaningless. This requires some thought. Perhaps the resolution is to pre-make the elemental `<metaEvaluation>`s for the standard targets with fake URLs. Then users can swap-n-replace them with the correct URLs for their own needs.

VI. SPECIAL REACTION CASE: ATOMIC SCATTERING DATA

Includes only electromagnetic (electrons and gammas) projectiles interacting with the electronic orbitals of an atom. Very similar to nuclear reaction data, but a lot simpler

Analogous to MT=500-599, MF=23, 26, 27, 28

This data is given in a standard `<reaction>` element with the following additional requirements:

ATOMIC REACTION REQUIREMENTS

R:1 outgoing particles are photons or electrons or residual atom

R:2 subshell binding energy, a float with units of energy (do we need this? wouldn't it go in a materials database?)

R:3 fluorescence yield, a float with units eV/photoionization

R:4 outgoing photons may optionally use form factors for coherent and incoherent scattering (see MF=27), see next subsubsection; if this is given, but there are no electrons emitted, then one can skip the cross section since it is given below. Can this be mix and matched with outgoing electron data? need correct kinType for channel

R:5 usual outgoing distributions, with

- R:5.a should include multiplicity (yields)
- R:5.b outgoing electrons or photons may use form equivalent to LAW=1 (continuum, used for bremsstrahlung and ionization) (same as MF=6, LAW=1), or
- R:5.c outgoing electrons or photons may use form equivalent to LAW=2 (two-body elastic) (same as MF=6, LAW=2), or
- R:5.d outgoing electrons or photons may use form equivalent to LAW=8 (energy transfer for excitation, used for excitation and bremsstrahlung), described in MF=26; if so use `<interp2d>` to tabulate the energy transfer $E_T(E)$ for LAW=8

R:6 documentation optional

R:7 covariance links

A. Atomic form-factors for photon scattering

The ENDF system for neutron and photon production data allows two alternatives for storing angular distribution data. One is by probability per unit $\cos(\theta)$ vs. $\cos(\theta)$, and the other is by Legendre coefficients. Actually, neither of these is a "natural" method for photons. The natural method would be atomic form factors or incoherent scattering functions. These are discussed briefly below.

1. *Incoherent Scattering*

The cross section for incoherent scattering is given by:

$$\frac{d\sigma_{\rm incoh}(E,E',\mu)}{d\mu} = S(q;Z)\frac{d\sigma_{KN}(E,E',\mu)}{d\mu}\ , \qquad (30)$$

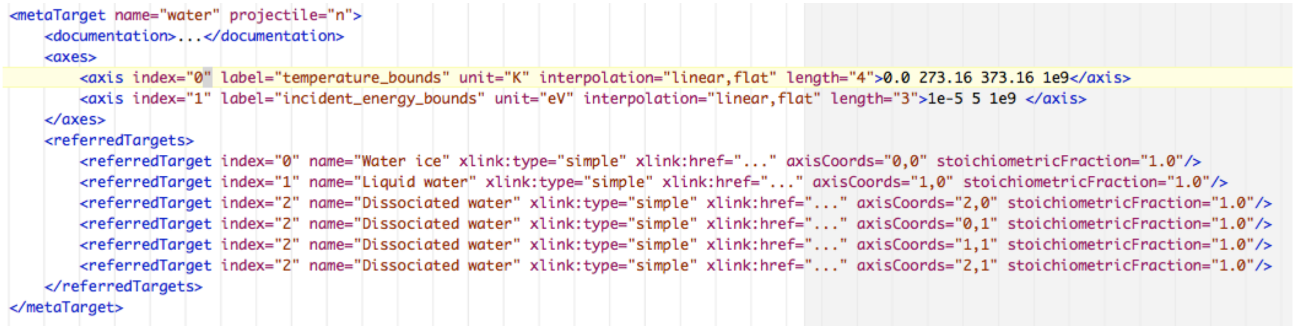
where:

$d\sigma_i/d\mu$ the Klein-Nishina cross section [13] which can be written in a closed form.

$S(q;Z)$ the incoherent scattering function. At high momentum transfer (q), S approaches Z. In the other limit, $S(0,Z)=0$.

q the momentum of the recoil electron (in inverse angstroms).

$$q = \alpha\left[1+\left(\frac{\alpha'}{\alpha}\right)^2 - 2\mu\left(\frac{\alpha'}{\alpha}\right)\right]^{1/2} \qquad (31)$$



```
<metaTarget name="water" projectile="n">
    <documentation>...</documentation>
    <axes>
        <axis index="0" label="temperature_bounds" unit="K" interpolation="linear,flat" length="4">0.0 273.16 373.16 1e9</axis>
        <axis index="1" label="incident_energy_bounds" unit="eV" interpolation="linear,flat" length="3">1e-5 5 1e9 </axis>
    </axes>
    <referredTargets>
        <referredTarget index="0" name="Water ice" xlink:type="simple" xlink:href="..." axisCoords="0,0" stoichiometricFraction="1.0"/>
        <referredTarget index="1" name="Liquid water" xlink:type="simple" xlink:href="..." axisCoords="1,0" stoichiometricFraction="1.0"/>
        <referredTarget index="2" name="Dissociated water" xlink:type="simple" xlink:href="..." axisCoords="2,0" stoichiometricFraction="1.0"/>
        <referredTarget index="2" name="Dissociated water" xlink:type="simple" xlink:href="..." axisCoords="0,1" stoichiometricFraction="1.0"/>
        <referredTarget index="2" name="Dissociated water" xlink:type="simple" xlink:href="..." axisCoords="1,1" stoichiometricFraction="1.0"/>
        <referredTarget index="2" name="Dissociated water" xlink:type="simple" xlink:href="..." axisCoords="2,1" stoichiometricFraction="1.0"/>
    </referredTargets>
</metaTarget>
```

FIG. 20. Sample `<metaEvaluation>` specification, in this case for water. This files requires another `<metaEvaluation>` to specify the composition of dissociated water into the elements hydrogen and oxygen. These then require other `<metaEvaluation>`s to specify the elemental composition of `HO` and `OO` in terms of their isotopics.

$\alpha = E_\gamma / m_0 c^2$,

E'_γ = scattered photon energy,

$\mu = \cos\theta$.

The angular distribution can then easily be calculated. Values of $S(q;Z)$ are tabulated as a function of q in File 27. The user presumably will have subroutines available for calculating q for energies and angles of interest and for calculating Klein-Nishina cross sections. The user will then generate the cross sections for the appropriate cases by calculating q's, looking up the appropriate values of S, and substituting them in the above formula.

2. *Coherent Scattering*

The coherent scattering cross section is given by:

$$\frac{d\sigma_{\mathrm{coh}}(E,E',\mu)}{d\mu} = \pi r_0^2 \left(1+\mu^2\right) \left\{ \left[F(q;Z)+F'(E)\right]^2 + F''(E)^2 \right\} \tag{32}$$

where:

$q = \alpha\left[2(1-\mu)\right]^{1/2}$, the recoil momentum of the atom (in inverse angstroms),

$r_0 = e^2/m_0c^2$, the classical radius of the electron.

$F'(E)$ the real part of the anomalous scattering factor.

$F''(E)$ the imaginary part of the anomalous scattering factor.

The quantity $F(q;Z)$ is a form factor, which can be easily tabulated. At high momentum transfer (q), F approaches zero. In the other limit $F(0;Z)$ tends to Z. The anomalous scattering factors are assumed to be isotropic. In addition, they smoothly approach zero at 1.0 MeV and can be assumed to be zero at higher energies.

An alternative way of presenting the photon scattering data would be to tabulate incoherent scattering functions and form factors. Users could then provide processing codes to generate the cross sections from this information. The calculation is quite straightforward and allows the user to generate all his scattering data from a relatively small table of numbers. The incoherent and coherent scattering data should always be presented as scattering functions and form factors, respectively, whether or not data are included in File 6.

ADDITIONAL PHOTO-ATOMIC REQUIREMENTS

R:1 just `<interp2d>` elements for incoherent scattering function, coherent scattering function, anomalous form factor (both real and imaginary parts)

VII. SPECIAL REACTION CASE: FISSION

In many ways fission is just a regular channel, but physically it is a continuum of channels all lumped together for practicality. Thus, while it fits neatly in our top level hierarchy, at the lowest levels (the components and forms), there are many data types we would like to include. We list them here.

FISSION REQUIREMENTS

R:1 Allow reaction aliases "`total_fission`", "`1st_chance_fission`", "`2nd_chance_fission`", etc. Probably reaction designator annotations can help with this.

R:2 Allow fission to be broken out by chance, but ensure sum rules obeyed.

R:3 Allow FPY data

R:4 Allow prompt, delayed and total $\bar{\nu}$. Ensure sum rules obeyed.

R:5 Allow PFNS using tables or Madland-Nix model

R:6 Break out delayed data by time group and put each group's delayed $\bar{\nu}$ with the groups DFNS and time constant

R:7 Allow $P(\nu)$ data

R:8 Allow the emission of neutrons, gammas, fission fragments (FF), electrons, neutrinos

R:9 Allow all ejected particles to have variable multiplicities and energy-angle spectra

R:10 Allow for semi-derived data such as energy release broken out into components

VIII. SPECIAL COMPONENT CASE: FISSION PRODUCT YIELDS

Fission Product Yields (FPY) are currently stored in their own sub library in the major evaluated data libraries (e.g. ENDF/B-VII.1), but conceptually they really belong in the description of emitted particles from the fission reaction. Because there are many different ways to induce fission, FPYs rightfully belong in a discussion of mid-level data structures.

A. Introduction

In the 2012 Working Party on Evaluation Cooperation (WPEC) meeting, two new subgroups were created: SG-37 to investigate Fission Product Yields (FPYs) and SG-38 to define a possible replacement for the ENDF nuclear data format. The Generalized Nuclear Data (GND) format is the main candidate for replacing the ENDF format and is under active development under auspices of WPEC/SG-38, lead by D. McNabb. GND is an outgrowth of earlier LLNL (US) project to replace LLNLs own internal ENDL format and the initial focus of the GND project was to develop formats and tools for handing neutron and charged particle transport data. SG-38 is now looking toward other ENDF formats and data, in particular, fission product yield (FPY) formats.

In the May 2013 SG-37 meeting, many new theoretical and experimental results were presented and new evaluations and evaluation techniques were presented. The new evaluations provide extensive covariance data which cannot be accommodated the ENDF format. However, users require this covariance data for performing uncertainty quantification in many applications. The concurrent development of the GND format allow us to address many shortcomings of the ENDF format and define a new format that can meet future needs of members of the SG-37 group.

Let us now discuss what data SG-37 intends to store in GND. The Independent Fission Product Yields (IFPY) are the fragments immediately after fission and de-excitation from prompt neutron and gamma emission while the Cumulative Fission Product Yields (CFPY) are the fragments after they are allowed to undergo all (beta and other) decays. The two yields are connected by the Q-matrix:

$$CFPY_i(E) = \sum_{ij} Q_{ij} IFPY_j(E) \tag{33}$$

This implies that, in practice, only one of IFPY and CFPY may be needed. The Q-matrix is a sparse matrix derivable from knowledge of the fission fragment decays and both A. Sonzogni and R. Mills have codes that can compute this matrix from an ENDF-formatted decay sublibrary. Although the Q-matrix is a derived quantity, it is derived from data potentially not associated with the FPYs tabulated (i.e. JEFF yields could in principal use ENDF/B decay data) so should be associated with the IFPY and CFPY.

During the SG-37 meeting, yields deuteron-, alpha-, photonuclear- and other particle induced yields in addition to the traditional neutron- and spontaneous yields were reported. The ENDF format has provisions for all of these.

In the process of evaluating yields, one often derives covariance data relating the yield of an isotope/isomer as a function of incident energy and covariance data relating yields from different isotopes/isomers. In addition, as the Q-matrix is derived from decay data which also has uncertainties on branching ratios, the Q-matrix may also have covariance. The branching ratios enter into the Q-matrix linearly so the covariance calculation is straightforward. The uncertainties on half-lives is typically not so important except in the few cases of a long lived product whose half-life exceeds the integration time used to compute the Q-matrix. In this case, uncertainty propagation is very non-trivial because here since the half-life dependence is strongly nonlinear.

B. Existing ENDF format

The ENDF format make provisions for storing the IFPY in MT=454 and CFPY in MT=459. Both FPYs use the same ENDF format and this format stores tables of (I, YI, dYI), with I denoting the isotope/isomer in question, YI the corresponding yield and dYI the uncertainty on the yield. The yields are given for several incident energies E with a rule for interpolating from one energy to the next.

In practice, the interpolation rule is poorly enforced. For neutron induced fission yields, four energies are typically given which correspond to group boundaries for "thermal", "fission spectrum", and "14 MeV" neutrons. In practice, the yields change slowly with incident energy so this has proven to be a problem only in a few applications.

The ENDF format does not provide a way to store fission yield covariances nor does it provide a way to store the Q-matrix.

C. Detailed FPY format requirements for GND

During the WPEC/SG-37 meeting, D. Brown presented some ideas on possible formats and began a dialog with members of WPEC/SG-37. As a result of subsequent conversations, D. Brown developed a list of requirements for a new FPY format. We expect this list to evolve somewhat as discussions continue.

FPY REQUIREMENTS

R:1 Clear rules for interpolation rather than a few vaguely defined groups (e.g. "thermal", "fission spectrum", "14 MeV"). Do not implicitly include spectrum averages in values.

R:2 Clearly defined range of validity of evaluation that can be matched to other reaction data

R:3 Clear location in the GND reaction hierarchy

R:4 Any incident particle (or none)

R:5 Per isotope/isomer yield ($Y_i(E)$), identical format for IFPY and CFPY

R:6 Per isotope/isomer yield uncertainty ($dY_i(E)$), identical format for IFPY and CFPY

R:7 Facility to store per isotope/isomer covariance on yield ($\Delta^2 Y_i(E, E')$), identical format for IFPY and CFPY.

R:8 Facility to store cross-isotope/isomer covariance ($\Delta^2 Y_{ii'}(E_i, E_{i'}; E_{i'}, E_{i''})$), identical format for IFPY and CFPY. Only IFPYs may be correlated with IFPYs and CFPYs with CFPYs, the Q-matrix couples the IFPY and CFPY.

R:9 Facility to optionally store the Q-matrix which connect the IFPY and CFPY

R:10 Facility to denote which (if any) of IFPY and CFPY is a derived quantity

DISCUSSION POINT Q-matrix can be computed from the decay library. Is Q-matrix something we want to store? It can be a very stringent requirement but if we computed CFPY using the Q-matrices computed from the decay data of the same library, we could store only IFPY data (and related uncertainties and correlations). In this sense CFPY can be considered as a sort of "reconstructed" FPY data as well as cross sections in the resolved resonance region are reconstructed from the resonance parameters. Obviously, this procedure would rely on a complete and consistent decay library and related uncertainties. **RESOLUTION** We want to allow storing Q-matrix as an option, not a requirement. Similarly, we were not requiring the evaluator to provide both the CFPY and the IFPY. However, we did want the evaluator to have the option to store either the CFPY or the IFPY and then the Q-matrix. Then the user can reconstruct what they need for their application. In the event that the evaluator has some fancy pants Bayesian scheme ;) that requires a simultaneous fit of some IFPY and some other CFPY, then that evaluator would have to store everything for the sake of internal consistency.

DISCUSSION POINT Additionally, we would like to investigate the possibility of storing the covariance of the Q-matrix.

D. Discussion of possible implementations

During the WPEC/SG-37 meeting, one "strawman" format was proposed and in discussion with C. Mattoon and B. Beck others were discussed. Here we summarize these discussion and provide pros and cons for each format. We expect that the format will go through many iterations as we attempt to meet the above requirements while maintaining a coherent and (hopefully easy to understand) structured data format.

Figure 21 shows an example of where fission product yields could fit in the current GND reaction hierarchy. As fission products describe the emitted particles of a fission event, it is logical to place them in the fission `<reaction>`s `<outgoingChannel>` of the corresponding `<reactionSuite>`. The collection of all fission product yield data is collected in a `<fissionProductYields>` section. The FPY section has an optional nativeData attribute that specifies which of the IFPY and CFPY is the original source distribution. It is unclear where to put a `<fissionProductYields>` section for spontaneous fission because GND does not yet define the top-level tags when there is no target and projectile.

Within the `<fissionProductYields>` section, we imagine an `<independentFissionProductYields>` section for IFPY, a `<cumulativeFissionProductYields>` section for CFPY and possibly a `<fissionYieldConversionMatrix>` section to store the Q-matrix. We expect the markup for IFPY and CFPY be identical, as in the ENDF format. Figures 2 and 3 show two different possible arrangements for data in the IFPY and CFPY sections.

Figure 22 shows one option. Here the yield tables use a modified version of the GND `<linear>` markup. The `<linear>` markup is attractive for several reasons:

- The interpolation rule specification is well developed.
- Fudge, the main tool for manipulating GND data, has strong data structures for storing X-Y data, including linearization, plotting, etc.
- All data for one nuclide is collected together in a simple, readable way.

The current `<linear>` markup is a general markup used to data consisting of X-Y pairs. In our case, we would like to add dYs as well. The current `<linear>` markup

```
<?xml version="1.0" encoding="UTF-8"?>
<reactionSuite projectile="n" target="Pu239" format="gnd version 1.2" temperature="0 K" xmlns:xlink="http://www.w3.org/1999/xlink">
  <styles> ⋯ </styles>
  <documentations> ⋯ </documentations>
  <particles> ⋯ </particles>
  <reaction label="45" outputChannel="n[multiplicity:'energyDependent', emissionMode:'prompt'] + n[emissionMode:'6 delayed'] + gamma
[total fission]" date="2006-09-01" ENDF_MT="18" fissionGenre="total">
    <crossSection nativeData="resonancesWithBackground"> ⋯ </crossSection>
    <outputChannel genre="NBody" Q="1.98902e8 eV">
      <fissionEnergyReleased nativeData="polynomial"> ⋯ </fissionEnergyReleased>
      <fissionProductYields nativeData="independentFissionProductYields" conversionMatrixGiven="true">
        <independentFissionProductYields numNuclides="780"> ⋯ </independentFissionProductYields>
        <cumulativeFissionProductYields numNuclides="920"> ⋯ </cumulativeFissionProductYields>
        <fissionYieldConversionMatrix> ⋯ </fissionYieldConversionMatrix>
      </fissionProductYields>
      <product name="n" label="n" multiplicity="energyDependent" emissionMode="prompt"> ⋯ </product>
      <product name="n" label="n__a" multiplicity="energyDependent" decayRate="0.013271 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="n" label="n__b" multiplicity="energyDependent" decayRate="0.030881 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="n" label="n__c" multiplicity="energyDependent" decayRate="0.11337 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="n" label="n__d" multiplicity="energyDependent" decayRate="0.2925 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="n" label="n__e" multiplicity="energyDependent" decayRate="0.85749 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="n" label="n__f" multiplicity="energyDependent" decayRate="2.7297 1/s" emissionMode="delayed"> ⋯ </product>
      <product name="gamma" label="gamma" multiplicity="energyDependent"> ⋯ </product></outputChannel></reaction>
</reactionSuite>
```

FIG. 21. A sample GND `<reactionSuite>` demonstrating where the fission product yields could reside within a fission `<reaction>` section in the current GND format.

also allows for only one `<data>` tag whereas we imagine one per nuclide.

DISCUSSION POINT On this option, we have to keep in mind that, in general, there are files with about 1000 FPY data for about 4 incident (neutron) energies. I would prefer option of Fig. 2. To imagine thousands of elements in a horizontal array as described in the option of Fig. 3 is a little bit impractical. RESOLUTION A 1000 x 4 table may be silly and unworkable. However that arrangement is the most ENDF-like, so we put it in as an option.

Figure 23 show another option for storing FPY. Here all data is stored in the GND `<table>` markup. This markup is quite general and compact. It can accommodate any number of isotopes simply by adding another column (or pair of columns if dY is included). We would need to add a provision for specifying an interpolation rule in energy as this is not already provided by the current `<table>` markup. With this, we would need to add quite a bit of coding to Fudge in order to generate plots and manipulate the yield data.

DISCUSSION POINT About the format for FPY covariance data, it was thought that ENDF compact format developed and used to store large covariance matrices would be suitable for this problem. However, there is no option like this proposed in this requirements document. RESOLUTION In GND and the new format there is agreement that there will be only one covariance matrix format and it will much clearer than what is in ENDF. For each dataset that has covariance data, there will be a link (w/ URL) to its own covariance and any (and all) cross covariances with other datasets. It is hoped that this arrangement can be made practical for FPY's so we don't have 1000 mini-FPY tables, each with 1000 URL's pointing to 1000 mini-covariance matrices.

The Q-matrix should be stored in its own section, here called `<fissionYieldConversionMatrix>`. GND already provides a `<matrix>` markup and it is natural to store the Q-matrix itself here. However we need to know how each row/column maps to a yield table. To solve this, in this example we provide the URL to the data for each row/column in the IFPY and CFPY tables. It is unclear at this time if this is the optimal way of referencing column and row elements and it depends on the way FPYs are stored in their corresponding data sections.

IX. SPECIAL REACTION CASE: LARGE ANGLE COULOMB SCATTERING (LACS)

ENDF LACS DISCUSSION, SIMPLIFIED HERE?

Quantum mechanically, charged particle elastic scattering is a sum of Coulomb and Nuclear amplitudes:

$$A = A_{\text{Coulomb}} + A_{\text{nuclear}} \tag{34}$$

The Coulomb piece is analytic and well known. The nuclear piece must be evaluated. The cross section for elastic scattering is of course the square of the amplitude so the differential cross section has three terms:

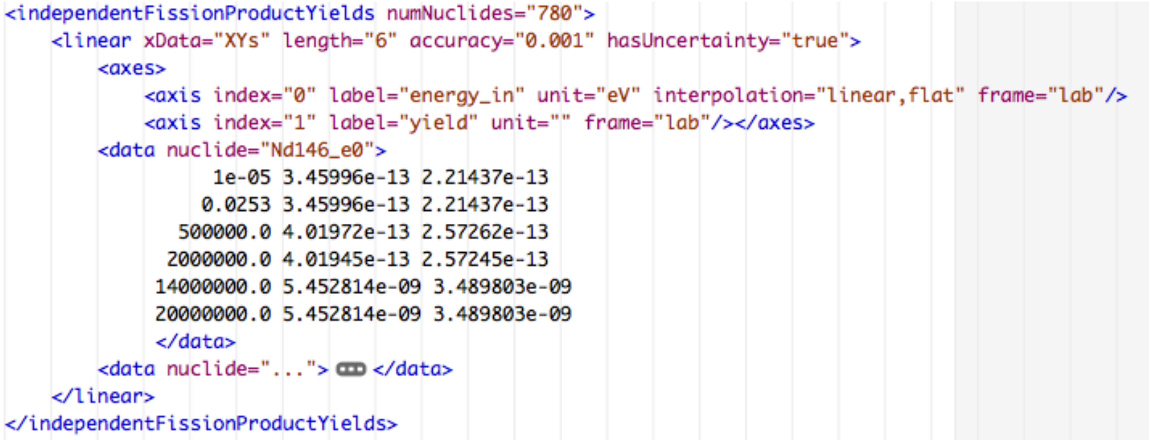
$$\frac{d\sigma_{el}(E)}{d\Omega dE'} = \frac{d\sigma_{\text{Coulomb}}(E)}{d\Omega dE'} + \frac{d\sigma_{int}(E)}{d\Omega dE'} + \frac{d\sigma_{nucl}(E)}{d\Omega dE'} \tag{35}$$

The last two terms in this equation are traditionally lumped together in a "nuclear+interference" term.

Whether the target and the projectile are identical or not, the Coulomb term is very singular:

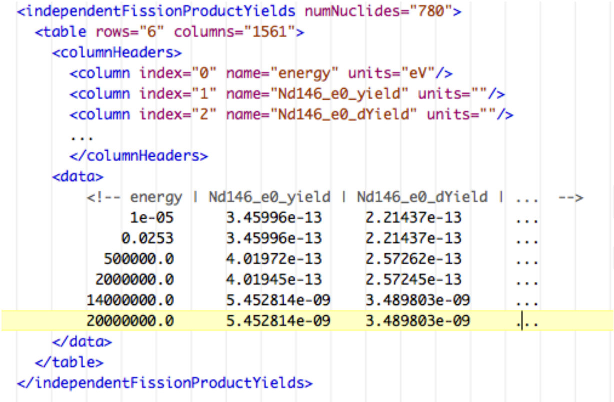
$$\frac{d\sigma_{\text{Coulomb}}(E)}{d\Omega dE'} \propto \frac{\eta^2}{k^2(1-\mu)^2} \tag{36}$$

Therefore, the elastic cross section diverges at small incident E and small angles ($\mu \to 1$). One might think that,



```
<independentFissionProductYields numNuclides="780">
    <linear xData="XYs" length="6" accuracy="0.001" hasUncertainty="true">
        <axes>
            <axis index="0" label="energy_in" unit="eV" interpolation="linear,flat" frame="lab"/>
            <axis index="1" label="yield" unit="" frame="lab"/></axes>
        <data nuclide="Nd146_e0">
                    1e-05 3.45996e-13 2.21437e-13
                   0.0253 3.45996e-13 2.21437e-13
                 500000.0 4.01972e-13 2.57262e-13
                2000000.0 4.01945e-13 2.57245e-13
               14000000.0 5.452814e-09 3.489803e-09
               20000000.0 5.452814e-09 3.489803e-09
               </data>
        <data nuclide="..."> ⋯ </data>
    </linear>
</independentFissionProductYields>
```

FIG. 22. One option for storing FPY. In this variant, the yields from each isotope are given their own `<linear>` section, but with a common statement of interpolation rules.



```
<independentFissionProductYields numNuclides="780">
  <table rows="6" columns="1561">
    <columnHeaders>
      <column index="0" name="energy" units="eV"/>
      <column index="1" name="Nd146_e0_yield" units=""/>
      <column index="2" name="Nd146_e0_dYield" units=""/>
      ...
      </columnHeaders>
    <data>
        <!-- energy | Nd146_e0_yield | Nd146_e0_dYield | ...   -->
             1e-05       3.45996e-13     2.21437e-13      ...
            0.0253       3.45996e-13     2.21437e-13      ...
          500000.0       4.01972e-13     2.57262e-13      ...
         2000000.0       4.01945e-13     2.57245e-13      ...
        14000000.0       5.452814e-09    3.489803e-09     ...
        20000000.0       5.452814e-09    3.489803e-09     ...
    </data>
  </table>
</independentFissionProductYields>
```

FIG. 23. Another option for storing FPY. In this variant, all yields from all isotope are collected together in a `<table>` section. This is more compact than the other variant.

since this is analytic, we don't have to store it and there is no problem. The problem is that since the Coulomb amplitude carries the square-root of these divergences, the interference term in the total elastic differential cross section also carries divergencies.

The traditional workaround is twofold:

- Start the "nuclear+interference" data tables at some finite incident energy where nuclear effects become noticeable. This eliminates the incident energy divergence in the tabulated data.
- Cut-off the "nuclear+interference" term at small angles. At small angles, Coulomb scattering dominates and must be handled in particle transport separately with techniques such as condensed history. ENDF data uses 10° as a cut-off (if I remember correctly), but it is not documented anywhere I can find.

DISCUSSION POINT ENDF puts this data in MF=3 and MF=6, LAW=5. This leads to confusion since what is in MF=3 is not a real cross section, but rather a kludge to get around the divergence. RESOLUTION We recommend putting this data in a special LACS `<dcrossSection_dOmega>`.

LACS REQUIREMENTS

R:1 `<parameterizedTwoBodyReaction>` for LACS data

R:2 `<dcrossSection_dOmega>` for LACS data

R:3 `<form>` for "nuclear+interference" data

R:4 a spot to denote the cut-off angle (since it may not be ENDF's default 10°)

X. SPECIAL REACTION CASE: PARTICLE PRODUCTION

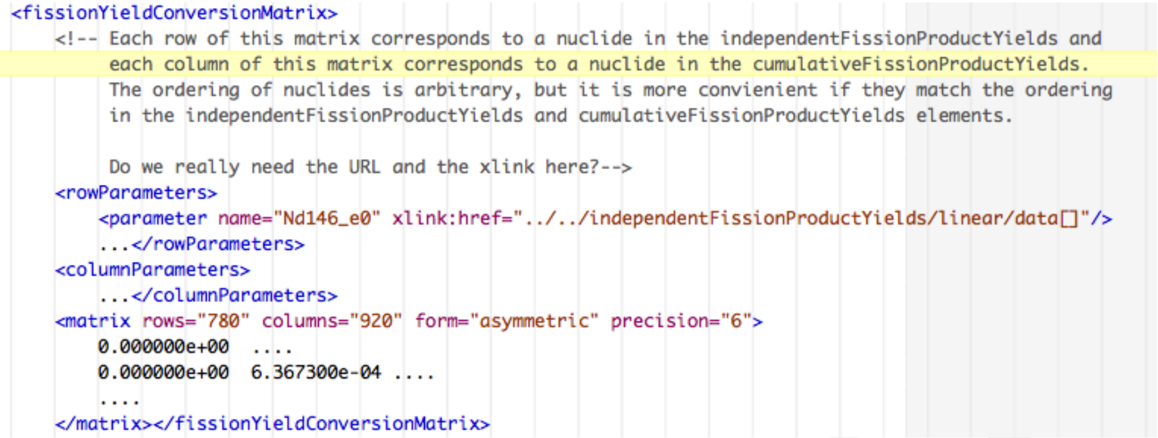
This section details the particle production region, which typically corresponds to nucleon induced reactions with $E > 20 - 30$ MeV. However, it can apply to all hadronic and leptonic projectile and gammas. This data is analogous to ENDF's MT=5 and is not too different from fast region, except that all produced particles have energy dependent multiplicities because so many channels are now open that it gets silly breaking them out individually.

Our existing hierarchy, as presented, works just fine for particle production region, so the requirements list is very short. We comment that it may be advantageous to just use external model (e.g. one of many in GEANT4) and use the `<metaEvaluation>` scheme to match on to the ENDF data.

SPALLATION REQUIREMENTS

R:1 Need reaction alias "spallation"

R:2 For each channel: Need cross section, list of particles considered, and for each particle a variable multiplicity/yield and $P(\mu, E|E)$



```
<fissionYieldConversionMatrix>
    <!-- Each row of this matrix corresponds to a nuclide in the independentFissionProductYields and
         each column of this matrix corresponds to a nuclide in the cumulativeFissionProductYields.
         The ordering of nuclides is arbitrary, but it is more convienient if they match the ordering
         in the independentFissionProductYields and cumulativeFissionProductYields elements.

         Do we really need the URL and the xlink here?-->
    <rowParameters>
        <parameter name="Nd146_e0" xlink:href="../../independentFissionProductYields/linear/data[]"/>
        ...</rowParameters>
    <columnParameters>
        ...</columnParameters>
    <matrix rows="780" columns="920" form="asymmetric" precision="6">
        0.000000e+00  ....
        0.000000e+00  6.367300e-04 ....
        ....
    </matrix></fissionYieldConversionMatrix>
```

FIG. 24. An option for storing the Q-matrix. The matrix itself is stored in a `<matrix>` section which could be sparse or dense. The identities of the rows and columns are denoted in the `<rowParameters>` and `<columnParameters>` sections.

XI. SPECIAL REACTION CASE: THERMAL SCATTERING LAW

Thermal neutron scattering law (TSL) data describe the situation where the de Broglie wavelength of an incident neutron is so large that the neutron wave function can not resolve individual nuclei but rather only see the macroscopic material. The incident neutron cannot be absorbed by the material and may only (in)elastically scatter off of it.

TSL data are given in sub-library 12 (NSUB = 12) in the ENDF6 format (using MF = 7, MT = 2 and MF = 7, MT = 4 data structures). In essence, this sublibrary provides dimensionless scattering kernels on a grid of dimensionless momentum and energy transfer to describe thermal neutron scattering by a number of materials important in applications of nuclear science and technology. The effects of chemical binding of nuclides, dynamics and structure of materials that are important to describe the peculiarities of neutron scattering at low incident neutron energies ($E < 1-10$ eV).

The sublibrary is organized by a nuclide (scatterer) in a given material. For example, in the ENDF/BV-II.1 TSL sublibrary, we have data for Be in beryllium oxide, O in beryllium oxide, C in Graphite, etc. In some cases, only the most important scatterer in a material has the evaluation. For example, we have `c_HinH2O`, or hydrogen in the light water, but there is no evaluation for `c_OinH2O` that implies that usage of the free gas model for thermal neutron scattering by oxygen in the light water is an acceptable approximation. Some evaluations have the data at one particular temperature: for example, data for thermal neutron scattering by H in liquid parahydrogen (`H2`, I = 0) are given at $T = 20.0°$K only. However, many evaluations are given for a number of temperatures T_i (called temperature nodes of TSL data). For example, $S(\alpha, \beta; T_i)$ data for `c_UinUO2` (U in uranium dioxide) are given at eight different temperatures (T_1=296.0°K, ..., T_8=1200.0°K).

It is expected that nuclear data postprocessing codes can read $S(\alpha, \beta; T_i)$ data in the ENDF6 format (MF=7 data structures in particular) and generate the scattering kernels, $d^2\sigma_s(E,T)/dE'd\Omega$, as well as the integral data (such as, scattering cross sections $\sigma_s(E,T)$, average scattering cosine $\bar{\mu}(E,T)$, average E', etc.) in proper physical units (barn per eV per sr, barn, eV, etc.) for incident neutron energies E and neutron outscattering with the energy E' and scattering cosine μ.

The TSL data given in NSUB = 12 are the result of evaluation: the kernels are calculated using theoretical models based on non-relativistic quantum mechanics to describe the interaction of a neutron with a macroscopic number of scatterers (nuclides contained in a given medium at a given temperature). The dynamics (described in terms of vibrational eigenmodes or phonon-type spectra) and structure (e.g., a certain order or correlations in the positions of scatterers in space) of the medium of interest are assumed to be known: all the necessary material data can be calculated using models and codes developed in Condensed Matter Theory or can be taken from available experimental data. The knowledge and a proper representation (parameterization) of dynamics and structure of the media of interest is an important component in building an adequate model of thermal neutron scattering, which in turn will result in the evaluated $S(\alpha, \beta; T_i)$ data in the ENDF-6 format. Then the parameters used in the TSL theoretical models can be adjusted (optimized) to achieve a better agreement of the resultant (double) differential cross-sections or derived integral data (such as $\sigma_{tot}(E)$ or $\bar{\mu}$) with available experimental results.

For example, for crystals (polycrystalline materials), the information about the crystal structure is expressed in terms of the so-called Bragg edges (a discrete set of energies $E_j \sim 1$ meV - 1 eV) and a set of crystallographic structure factors s_j associated with E_j and a neutron scatterer in a crystal unit cell. In addition, one has to estimate the temperature dependent Debye-Waller coef-

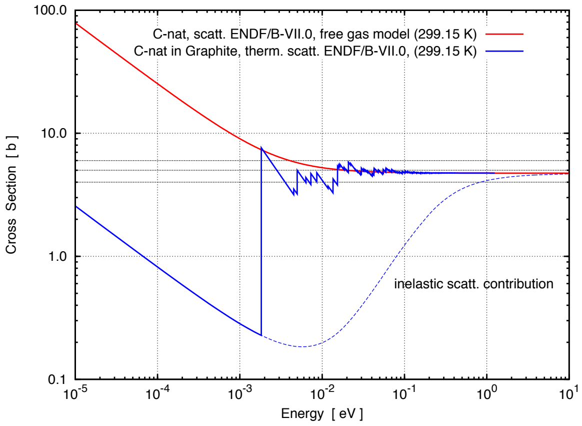


FIG. 25. Elastic scattering cross-sections of carbon at room temperature (free gas model) vs. thermal scattering cross-sections of carbon in graphite at room temperature.

ficient W' (in the units of eV^{-1}). Then it is possible to generate the data structure of MF=7, MT = 2 (i.e., a function $S(E;T_i)$) that in turn can be used to generate the contribution of coherent elastic neutron scattering into the thermal neutron scattering kernel, scattering cross sections, etc., for a given scatterer in the polycrystal.

Figure 25 compares the different elastic scattering prescriptions for two different forms of carbon.

To generate the ENDF MF=7, MT = 4 data structure for polycrystals (i.e., $S(\alpha,\beta;T_i)$ for incoherent inelastic neutron scattering), the phonon density of states (DOS) should be known, $\rho_{ph}(E;T_i)$. If two or more different atoms (ions) are present in a crystal cell (e.g., U and O in the cubic cell of UO2 and both nuclides have the evaluated TNS), then the so-called partial phonon density of state has to be known for each scatterer. (The partial phonon DOS is determined as a contribution from the given atom in a unit cell to the total phonon DOS.) The energy transfer grid (the β-mesh and the value of β_{max}) has to be chosen to describe accurately the specific features of the inelastic scattering of a neutron in a given material. In Figure 25, we show the thermal scattering cross-sections for graphite at room temperature and compare them with the elastic scattering cross-sections of natural carbon (C-nat) obtained within the free gas approximation, using the ENDF/B-VII.0 evaluations.

For liquids, there is no coherent elastic contribution and so only the ENDF MF=7, MT = 4 data blocks are given in evaluations. For the neutron scattering by 1H in a liquid (which is the important scatterer from the standpoint of neutron slowing-down), one can disregard intermolecular coherence effects (σ_{coh}(1H)<< σ_{incoh}(1H)) and apply the incoherent approximation to model the thermal neutron scattering,

$$S(\alpha,\beta;T_i) = S_{incoh}(\alpha,\beta;T_i). \tag{37}$$

Then, the knowledge of dynamics of 1H in a liquid of interest is necessary to build $S(\alpha,\beta;T_i)$ in the ENDF6 format. In particular, one should know the generalized vibrational spectrum of 1H in the liquid. For molecular liquids, the vibrational spectrum can be subdivided into intramolecular, intermolecular (hindered rotations and hindered translations) and low energy diffusion (translational) parts. Each part contributes into $S_{incoh}(\alpha,\beta)$, but can be treated differently in modeling and so can be parameterized differently. For example, the intramolecular part can be approximated as a weighted sum of delta-functions while, for the inter-molecular part, one can use a continuous function, $\rho(E;T_i)$, similar to the phonon DOS in a crystalline solid ($\rho(E;T_i) \propto E_2$ as $E \to 0$). The incoherent inelastic approximation and partitioning of the vibrational spectrum of a liquid turn out to work well to obtain an accurate evaluation of $S(\alpha,\beta)$ for 1H in the light water [9, 10].

Unlike the scattering by 1H, for the thermal neutron scattering by 2H in deuterated liquids, the incoherent approximation is, strictly speaking, not applicable (σ_{coh}(2H)$\sim$ σ_{incoh}(2H)), and one has to build the coherent inelastic part of $S(\alpha,\beta;T_i)$,

$$S(\alpha,\beta;T_i) = (1-f)\times S_{incoh}(\alpha,\beta;T_i) + f\times S_{coh}(\alpha,\beta;T_i) \tag{38}$$

with

$$f = \sigma_{coh}/(\sigma_{coh}+\sigma_{incoh}). \tag{39}$$

There are some models of $S_{coh}(\alpha,\beta)$ that require only the knowledge of the so-called static structure factor(s) of a liquid, $S(q)$, and $S_{incoh}(\alpha,\beta)$. (Then, $\hbar q$ is the neutron momentum transfer, and $\alpha \propto (\hbar q)2/kT$). For a molecular liquid such as heavy water (D2O), the structure factors are $S^{DD}(q;T)$, $S^{DO}(q;T)$, and $S^{OO}(q;T)$, and the deuterium-deuterium one (S^{DD}) is the most important factor to take into account in modeling of $S(\alpha,\beta)$ of 2H in the heavy water. In Figure 26, we show how accurately one can model the total cross-sections of heavy water using the recent ENDF/B TNS evaluations for 2HinD2O. (Elastic scattering by oxygen (16O) was obtained within the free gas approximation to build σ_{tot}(D2O) vs. E_{in}.) Obviously, the model was improved in the ENDF/B-VII.0 evaluation in comparison with the ENDF/B-VI one [9], but further improvements in modeling and development of a new TSL evaluation for 16OinD2O are desirable to reduce the discrepancy with the experimental results for heavy water [10].

For many evaluations included in the ENDF/B-VII (releases 0 and 1) TSL sub-libraries, the generation of evaluations in the ENDF-6 format was done using LEAPR module of NJOY99 nuclear data processing code [11, 12]. In the LEAPR input, evaluators supply the thermal scattering cross-section, mass, and a number of the principal scatter (for a proper normalization), and structuralized data describing the dynamics of the principal scatterer in the material of interest and information (or data) related to the material structure (if required for modeling), as

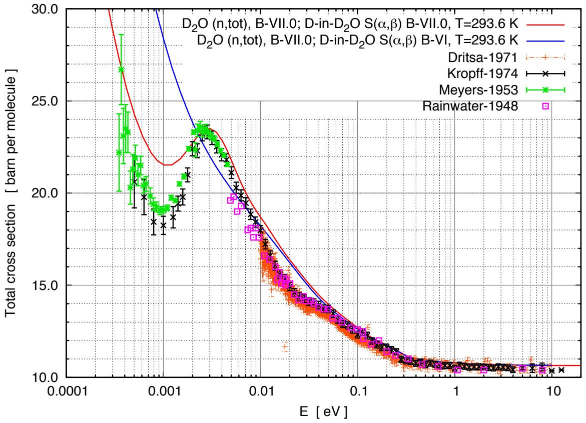


FIG. 26. Total cross-sections of D2O (free gas model for oxygen) using 2H-in-D2O from ENDF/B-VII.0 and ENDF/B-VI.0 TSL evaluations vs. experimental data at room temperature.

well as some comments to be written in the MF = 1, MT = 451 data block of the TSL evaluation.

As we move to a new hierarchy, we seek to maintain the physics encoded in the ENDF format while extending the data to enable new applications.

A. Coherent Elastic

Neutrons can only elastically scatter coherently off of regular substances such as crystals. The differential cross section for such scattering can be written [2]

$$\frac{d^2\sigma}{dE'\,d\Omega}(E,T) = \frac{1}{2\pi E}\sum_{i=1}^{E_i<E} s_i(T)\;\delta(\mu-\mu_i)\;\delta(E-E') \tag{40}$$

where:

$$\mu_i = 1 - \frac{2E_i}{E} \tag{41}$$

The quantity actually given in the file is $S(E,T)$ which the ENDF manual states is conveniently represented as a stairstep function with breaks at the Bragg edges E_i using histogram interpolation.

This data is given in a standard `<parameterizedTwoBodyReaction>` element with the following additional requirements:

COHERENT ELASTIC TSL REQUIREMENTS

R:1 An elastic channel reaction designator that includes the annotation `coherent`

R:2 No `<crossSection>` element in the `<reaction>` element, it belongs in a `<parameterizedTwoBodyReaction>` element.

R:3 An list of `<interp3d>` elements containing the structure factor $S(E,T)$ in the `<distribution>` element. The ENDF manual requires the interpolation in E to be a histogram and it is unclear whether there is a need to relax this requirement. With this requirement, $S(E,T) = S(E_i,T) \equiv s_i(T)/E$

R:4 Optional `<link>` to `<covariance>` data on the structure factor.

R:5 Only one `<form>` of this data is currently possible.

B. Incoherent Elastic

For partially ordered systems, the incoherent approximation to elastic scattering is given by

$$\frac{d^2\sigma}{dE'\,d\Omega}(E,T) = \frac{\sigma_b}{4\pi}\,e^{-2EW'(T)(1-\mu)}\;\;\delta(E-E') \tag{42}$$

where:

σ_b is the characteristic bound cross section (barns),

W' is the DebyeWaller integral divided by the atomic mass (eV^{-1}),

and all the other symbols have their previous meanings. The integrated cross section is easily obtained:

$$\sigma(E) = \frac{\sigma_b}{2}\left(\frac{1-e^{-4EW'}}{2EW'}\right) \tag{43}$$

Note that the limit of σ for small E is σ_b.

This data is given in a standard `<parameterizedTwoBodyReaction>` element with the following additional requirements:

INCOHERENT ELASTIC TSL REQUIREMENTS

R:1 An elastic channel reaction designator that includes the annotation `incoherent`

R:2 No `<crossSection>` element in the `<reaction>` element, it belongs in a `<parameterizedTwoBodyReaction>` element.

R:3 An `<interp2d>` element collecting W' the Debye-Waller integral divided by the atomic mass as a function of temperature.

R:4 The bound cross section σ_b, with units.

R:5 Optional `<link>` to `<covariance>` data on W'.

R:6 Optional `<link>` to `<covariance>` data of σ_b. This is a 1×1 matrix, but could be correlated with W''s covariance.

R:7 Only one `<form>` of this data is currently possible.

C. Incoherent Inelastic

Inelastic scattering is represented by the thermal neutron scattering law, $S(\alpha,\beta,T)$, and is defined for a moderating molecule or crystal by:

$$\frac{d^2\sigma}{d\Omega\, dE'}(E\to E',\mu,T)=\sum_{n=0}^{\mathrm{NS}}\frac{M_n\sigma_{bn}}{4\pi kT}\sqrt{\frac{E'}{E}}e^{-\beta/2}S_n(\alpha,\beta,T) \tag{44}$$

The definitions of the parameters in this equation are given in the ENDF manual.

DISCUSSION POINT As we said above, only the data for the principal scatterer goes in the ENDF file. We can expand this.

This data is given in a standard `<parameterizedTwoBodyReaction>` element with the following additional requirements:

INCOHERENT INELASTIC TSL REQUIREMENTS

R:1 reaction designator that clearly denotes that this reaction is incoherent inelastic data

R:2 No `<crossSection>` element in the `<reaction>` element, it belongs in a `<parameterizedTwoBodyReaction>` element.

R:3 `<interp3d>` table to store $S(\alpha,\beta,T)$ vs. (α,β) at the fixed T of the file for each type of atom in the material.

R:4 covariance on $S(\alpha,\beta,T)$ at fixed T; will be of similar size to $P(\mu,E'|E)$ covariance

DISCUSSION POINT New experiments from NCSU/RPI/ORNL collaboration will directly measure the $d\sigma(E)/dE'd\Omega$. What about storing the covariance directly on the double differential cross section?

DISCUSSION POINT Does not make sense to put covariance on $S(\alpha,\beta;T_i)$, it's too darned big and it is derived by LEAPR anyway. It would make more sense to put covariance on the phonon density of states $\rho(\omega)$ and on the structure factor $S(q)$ but then the logic in LEAPR would need to be captured here. If we do this, we'll need spots in the hierarchy for f_{coh}, $\rho(\omega)$ and $S(q)$. The data files would be much smaller than the corresponding ENDF files.

XII. ADDITIONAL ELEMENTS FOR DERIVED TRANSPORT DATA

This section details any derived data we felt might be useful at this stage to include in the nuclear data hierarchy. In this section, we list the types and suggest where in the hierarchy they may reside.

BRET TO WRITE THIS!!!!

A. Transfer matrix

B. Average energy deposit per particle x, $\langle E'_x\rangle$

C. Average forward momentum deposit per particle x, $\langle p_{zx}\rangle$

D. Fission energy release

E. KERMA

F. $\bar{\mu}(E)$

This is the average forward scattering angle in the lab frame.

DISCUSSION POINT Is there not only a spot in the hierarchy for $\bar{\mu}$, but is there one for its covariance?

G. PURR

H. `<link>`s and `metaEvaluations`

I. Damage cross sections

J. Production cross sections (deprecated)

K. Bonderenko self-shielding corrections

XIII. CASE STUDY: PACKING ENDF/B-VII.1

DISCUSSION POINT DAVE, WRITE ME IF THERE IS TIME

DISCUSSION POINT Do we need a `<library>` and/or `<sublibrary>` markup to index a library?

XIV. CASE STUDY: RECONSTRUCTING RESONANCES

DISCUSSION POINT BRET, PLEASE WRITE ME IF THERE IS TIME

XV. CASE STUDY: PROCESSING AN EVALUATION FOR TRANSPORT APPLICATIONS

DISCUSSION POINT BRET, PLEASE WRITE ME IF THERE IS TIME

ACKNOWLEDGEMENTS

We wish to acknowledge valuable review and feedback by M. Herman, A. Sogzogni, T. Johnson, D. McNabb,

M. Pigni, L. Leal, D. Wiarda, and J. Conlin. This work was performed under the auspices of Department of Energy contract Nos. DE-AC52-07NA27344 (Lawrence Livermore National Laboratory) and DE-AC02-98CH10886 (Brookhaven National Laboratory). The project was partly funded through the Nuclear Data Program Initiative of the American Recovery and Reinvestment Act (ARRA).

Appendix A: List of `<reaction>` requirements

Appendix B: List of `<resonances>` requirements

Appendix C: List of `<dcrossSection_dOmega>`s

KN
TSL
LACS

Appendix D: List of `<distribution>` requirements

1. Angular distributions: $P(\mu|E)$

used by atomic scattering and nuclear reaction data as a table vs. Legendre moments

2. Energy spectra: $P(E'|E)$

used by decay data, atomic scattering and nuclear reaction data

3. Energy-Angle distributions: Variations on $P(\mu, E'|E)$

as a table vs. Legendre moments

4. Angle-Energy distributions: Variations on $P(E', \mu|E)$

5. Uncorrelated Energy-Angle distributions: Variations on $P(\mu|E)P(E'|E)$

TABLE III. Partial list of distribution components and forms currently allowed in GND.

distribution	Form	ENDF equivalent
angular	pointwise	MF4 LTT2
	Legendre	MF4 LTT1
	isotropic	MF4 LTT0
	recoil	MF6 LAW4
energy	pointwise	MF5 LF1
	Evaporation Spectrum	MF5 LF9
	Maxwellian	MF5 LF7
	Watt Spectrum	MF5 LF11
	Madland-Nix	MF5 LF12
	N-Body Phase Space	MF6 LAW6
energy-angular	Kalbach-Mann	MF6 LAW1 LANG2
angular-energy	pointwise	MF6 LAW7
uncorrelated	combination of 'angular' and 'energy'	MF4 and MF5

---

[1] B. Beck, private communication.

[2] Ed. by A. Trkov, M. Herman, D.A. Brown, ENDF/B-VII.1 Manual, *ENDF-6 Formats Manual: Data Formats and Procedures for the Evaluated Nuclear Data Files ENDF/B-VI and ENDF/B-VII,* CSEWG Document ENDF-102, BNL Report BNL-90365-2009 Rev.2 (2011).

[3] C.M. Mattoon, B.R. Beck, N.R. Patel, N.C. Summers, G.W. Hedstrom, D.A. Brown, *Generalized Nuclear Data: A New Structure (with Supporting Infrastructure) for Handling Nuclear Data,* Nuclear Data Sheets, Volume 113, Issue 12, December 2012, Pages 3145-3171, ISSN 0090-3752, `http://dx.doi.org/10.1016/j.nds.2012.11.008`. (`http://www.sciencedirect.com/science/article/pii/S0090375212000944`)

[4] G. Žerovnik, A. Trkov, D.L. Smith, R. Capote, Nuclear Instruments and Methods in Physics Research A 727, pp 33-39, 2013.

[5] A.M. Lane, R.G. Thomas, Rev. Mod. Phys., **30**, pp 257-353 (1958).

[6] P. Descouvemont, D. Baye, Rep. Prog. Phys. **73**, 036301 (2010).

[7] F.H. Fröhner, "Evaluation and Analysis of Nuclear Resonance Data," JEFF Report 18 (2000).

[8] J.M. Blatt, L.C. Biedenharn, Rev. Mod. Phys. **30**, 258 (1958).

[9] M. Mattes, J. Keinert, *Thermal Neutron Scattering Data for the Moderator Materials H_2O, D_2O and ZrH_x in ENDF-6 Format and as ACE Library for MCNP(X) Codes*, Report INDC(NDS)-0470, IAEA, Vienna, 2005.

[10] J.I. Márquez Damián, D.C. Malaspina, J.R. Granada, *Vibrational spectra of light and heavy water with application to neutron cross section calculations*, J. Chem. Phys., Vol. 139, p.024504, 2013.

[11] R.E. MacFarlane, D.W. Muir, *The NJOY Nuclear Data Processing System, Version91*, LA-12740-M, LANL, USA, 1994.

[12] R.E. MacFarlane, A.C. Kahler, *Methods of Processing ENDF/B-VII with NJOY*, Nuclear Data Sheets, Vol.111, pp. 2739-2890, 2010.

[13] O. Klein and Y. Nishina, Z. Phys. 52, 853 (1929)