Requirements for a new nuclear data structure
Part 1: Vision and Goals

Prepared by WPEC Subgroup #38 (subgroup title: *"A modern nuclear database structure beyond the ENDF format"*)

Dated: May 24, 2013

Introduction: This document summarizes the findings of the first meeting of WPEC Subgroup 38 (SG38), which was organized to develop a new evaluated nuclear data structure[1] and then oversee the transition from the current standard (ENDF-6) to the new structure. The first milestone for SG38 is to capture the basic requirements that the new structure must meet in order to address the needs of the current as well as the anticipated future evaluated nuclear data community. By clearly outlining these needs that the new structure must meet, SG38 can better define the work required to develop and deploy the new structure.

ENDF-6 has had a long and fruitful history as the preferred format for storing and exchanging evaluated nuclear data. Together with processing codes, it plays a pivotal role between nuclear physicists and reactor physicists, allowing the exchange of data between different computer codes. Today, however, it is showing signs of age. In particular, the ENDF-6 format places unnecessary limitations on the types of reactions and the precision at which data can be stored. Also, each new generation of nuclear scientists and engineers must overcome a steep learning curve (having nothing to do with physics, only with how data are stored in ENDF) before they are able to use the data. These users are applying nuclear data towards solving a broad range of problems (in medical physics, global security and advanced detector designs among others) that stretch the ENDF format beyond its original design. There is also a strong desire, particularly among the younger generation, to adopt software concepts and technologies that are more modern, more familiar and more broadly utilized than the 1960's-vintage ENDF format.

Although a new structure is needed, the subgroup also recognizes that many decades and much effort have been invested in the ENDF format and the tools that use it. In order to be useful, the new structure must fit into the existing 'infrastructure' for nuclear data, continuing to meet the needs of evaluators, processors and users of the data. Part of the goal of SG38 is to ensure that these needs are met, and to promote a smooth transition (likely lasting a decade) from ENDF-6 to the new structure.

While SG38 was originally formed to develop the structure for storing nuclear data, a few other priorities were identified during the first meeting. These include the importance of defining an Application Programming Interface (API) for reading and writing data in the new structure, and the importance of highlighting similarities with other nuclear data efforts (experimental data, for example) in order to develop common data containers that can be used by all efforts.

[1] In this document we differentiate between a 'structure' and a 'format'. A format includes both a structure (that is, the overall organization of data) plus the details of how it is represented in a file.

At the first SG38 meeting, we looked at nuclear data from several different perspectives, including a history of evaluated data formats and an overview of the software that use them. From each of these perspectives we have collected requirements that capture what different parts of the nuclear data workflow need from a new data structure and from the new software tools that will support that structure. These requirements are also summarized in the final section of this document. A companion document (Part 2), to be developed at the May 21-22, 2013 workshop, will provide a project plan for the subgroup.

History: Nuclear data have been stored in computer-readable format for over 50 years. Before outlining the needs for a new structure, we must understand some of the history of evaluated nuclear data formats, the decisions that led to the widespread adoption of ENDF-6 as the standard way to store data, and some of the most recent developments in the field.

The nuclear data community is responsible for providing evaluated nuclear data. An evaluation, or reaction evaluation, is a reduced set of model parameters that best represent all reactions involving a projectile and a target (MAT number in ENDF). To be useful for simulations, each evaluation must be complete: quantities (e.g. the cross sections) must be defined over a broad range of energies (usually 10^{-11} to 20 MeV for incident neutrons) , even when no experimentally measured data are available. It is the task of the evaluator(s) to assess the most probable value of a quantity at any energy, resolving issues of discrepant measurement, assigning values (by an educated guess or based on model calculations) where no data are available and providing data in computer-readable format.

Historically, several formats were in existence to store the evaluated nuclear data, including KEDAK in Germany, UKNDL in the UK, ENDL at Livermore National Laboratory in the USA, ENDF in the rest of the USA, and SOKRATOR in the Soviet Union. These formats were incompatible and hindered free data exchange, which was agreed upon since the Geneva Summit in 1955. Format translation programs were developed locally, but were often limited to selected evaluations. Some attempts were made to develop flexible processing codes to handle “all” formats. This was usually done by translating original evaluations into a common internal format (e.g., FEDGROUP by P. Vertes), but the codes were generally quite unstable since local data efforts were uncoordinated.

The ENDF/B-IV evaluated nuclear data library became generally available in the eighties. At the time it was the most comprehensive compilation of general-purpose evaluations and soon found broad usage all over the world. In addition, the NJOY-91 data processing system was released, which could handle the data in ENDF format. This fact contributed greatly to the widespread usage of the ENDF/B-IV library.

The ENDF/B-V library was released in the USA in 1979. It was sponsored by industry and for this reason it was not released to the users outside the USA and Canada. This decision indirectly contributed to the establishment of the joint European JEF library evaluation project in Europe and the JENDL project in Japan. Both of these projects adopted a subset of the ENDF format as their bases. The main advantage of the ENDF format was detailed documentation, which provided rules and procedures to uniquely define the cross sections and other quantities at each point, including the prescription of interpolation rules for tabulated data. The European approach was to build on improvements to the ENDF/B-IV library, which resulted in the JEF-1 library and later on the JEF-2.2 library, which was the main workhorse for the industry in Europe until recently. Japan followed a more independent route, relying extensively on evaluations from nuclear reaction models, culminating in the release of the JENDL-2 library. The most recent libraries are ENDF/B-VII.1 in the USA, JEFF-3.1.2 in Europe, JENDL-4.0 in Japan, CENDL-3.1 in China and ROSFOND-2010 in Russia. Other special-purpose libraries also exist and are available from the Nuclear Reaction Data Centres electronically over the Internet. The bulk of the evaluated nuclear data libraries are given in the ENDF format version six, also known as ENDF-6 [1].

The ENDF format thus received worldwide recognition in the USA, Europe, Japan, China, as well as in the former Soviet Union. The main advantage of this choice was the ease of data exchange and the availability of a fairly robust NJOY data processing system, which soon made other formats and data processing systems obsolete. In addition, the AMPX code system from Oak Ridge was poorly maintained for some time. The only real alternative available to general users was the Pre-Pro suite of codes [2] from the IAEA, but their main purpose was data verification, validation and display.

While ENDF-6 and NJOY are now the de-facto standard tools for storing and processing nuclear data, several other developments in formats and processing codes have recently been made. These include the Generalized Nuclear Data (or GND) structure, which has been designed at Lawrence Livermore National Laboratory (LLNL) in the USA and is intended as a modern replacement for both the legacy ENDL format of LLNL and for ENDF-6 [3]. New developments in processing include the code TRENDF, which is being developed at the CEA in France to support the Monte Carlo code TRIPOLI, as well as an updated version of AMPX at Oak Ridge [4], and the Livermore code Fudge which can now process nuclear data in multiple formats including ENDL and the new GND structure, as well as in ENDF-6 by first converting to GND.

These developments are indicative of renewed international interest in using nuclear data to solve emerging problems. They also mean that we have a new opportunity to work on modernizing and simplifying the underlying nuclear data infrastructure all at once. By simultaneously developing new tools for storing data and for processing data, we can design a more consistent set of tools that better suit the needs of all users.

Purpose of the new structure and expected benefits

: The nuclear data community today must support not only the traditional application areas of fission energy, nuclear waste handling, criticality safety, defense, etc., but also emerging areas in non-proliferation, accelerator driven systems, and uncertainty quantification. In the future, other applications will emerge and our tools must be able to adapt in order to profit from these new opportunities. Also, at a time when the field is shrinking due to retirements, the community must be able to attract new talent who prefer and are adept at working with new computational tools. These needs together mean that a modern and user-friendly structure for storing nuclear data is required. The new structure should simplify storing and handling nuclear data, and should present the data in a manner that is more intuitive and useful to a new generation of scientists and engineers. It should also build upon all the nuclear data community has learned and achieved with ENDF-6.

Our focus should therefore be to minimize the effort needed to become proficient in utilizing nuclear data. We can achieve this by using modern concepts to structure data, and standardized tools (e.g. XML or HDF5) to store and exchange the data. We should focus on defining the nuclear data *language*, i.e., the structure used for storage and exchange. As budgets continue to decrease, we need to find better ways to enhance collaborations and reduce the overall cost of improving nuclear data. Taking advantage of the mature tool sets that are already available for handling hierarchical data makes sense and will open the nuclear data world to a broader community.

While learning the ENDF format and writing a parser for it is currently a 'rite of passage' for new entrants into the field, this is a waste of our resources as precious time is spent wrestling with file I/O instead of being spent on physical understanding. Current ENDF-6 processing codes must combine two complicated tasks: the data I/O required in order to access the parameters and the algorithms that actually manipulate and use the data. This results in code that is overly complex and difficult to decipher. This problem only grows worse as we seek to modernize and extend capabilities. While ENDF-6 can be extended to store any conceivable data set, a better practice is to streamline our workflow by abstracting out the underlying data handling and allowing each user to focus directly on the concepts (*physics*) important to their needs.

To provide better support to traditional applications and adapt to emerging needs, we need a flexible, powerful and easy-to-use tool set. At the heart of that tool set is the structure in which the nuclear data are stored. In turn, this structure guides the development of the data containers in the computer programs that interact with reaction data (the 'API' or application programming interface). Good structure design will lead to good tool design and to more robust and powerful applications. Good tool design can also limit the transition costs in the adoption of a new structure and tool set.

In developing a new structure, we must aim to “optimize developer happiness” by giving them a set of tools and a structure that allows them to meet user needs quickly. Along the way, this is an opportunity to relax limitations imposed by the legacy ENDF format (see the History section) and to correct some problems that have crept into the legacy ENDF format (e.g. synchronizing the masses, eliminating redundant data and storing more exclusive reaction channels among other problems).

The structure hierarchy should reflect our understanding of nuclear reactions/decay in a simple, and when stored in an ASCII file, human-readable fashion. It should also utilize nesting to clarify relationships between data and to uniquely specify data. A sample physics-based hierarchy for storing nuclear data is seen in figure 1.

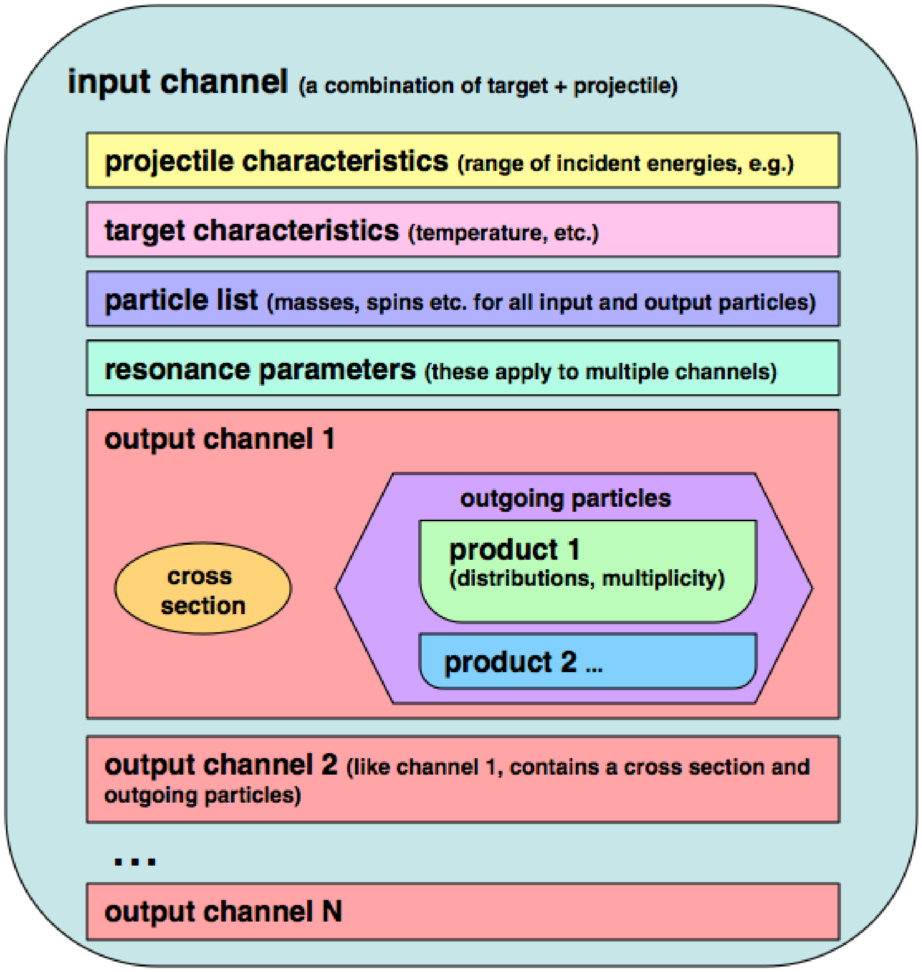

Figure 1: A Venn diagram with a sample hierarchy for organizing nuclear data. The hierarchy starts with a 'reaction', which contains information about the target and projectile. Nested inside the reaction we find all the possible outgoing 'channel' sections, and inside each of those we find a cross section and a list of outgoing products (each with its multiplicity and distributions). This hierarchy can be stored many ways: using a computer file system where each level of the hierarchy corresponds to a new directory, or in meta-languages like XML and HDF5.

As seen in figure 1**,** the data for each outgoing reaction channel are nested together, making the physical process clear (resonance parameters are an exception since they apply to multiple channels). This data nesting is a significant break from the ENDF legacy, reflecting both how we think about nuclear reactions experimentally and theoretically and also how the data are used in particle transport codes. The example uses descriptive strings rather than integer flags to label different parts of the data, making searching simpler. The hierarchy is also portable: it is not bound to any specific computer language, but can be expressed easily in xml, in binary meta-languages like HDF5, or by using files and folders in a computer file-system.

The example above is not the only possible way to store nuclear reaction data. Another possibility would be to group data first by energy range, since application codes often use different data tables and event generators in different energy ranges. For example, in MCNPX

[5], simulations of neutrons colliding with hydrogen can use the ACE "lwtr.10t" data from 1E-5 to 4 eV, the ACE "1001.70c" data from 4 eV to 150 MeV, the CEM event generator from 150 to 800 MeV, and the FLUKA event generator at energies above 800 MeV. A structure that groups data by energy range could help clarify how the data will be used. An example of this type of organization is seen in figure 2.

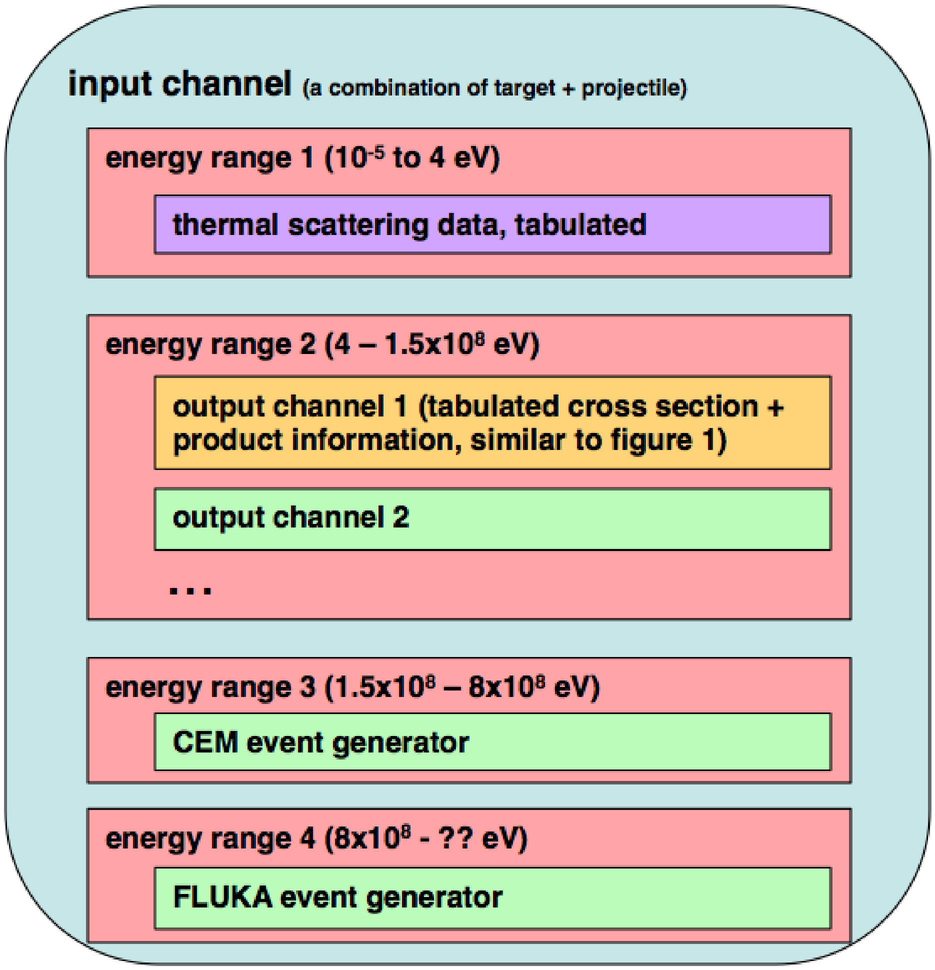

Figure 2 shows another possible hierarchy for organizing nuclear data. Here the data are grouped first by energy range and then by reaction channel, etc. This structure might better reflect how the data are used by simulation codes, where different models are used in different incident energy regimes.

Organizing the hierarchy to reflect nuclear reaction physics should help improve data quality and consistency. For example, when the elastic cross section and angular data are separated in ENDF-6 format into MF3 and MF4, it is easy to update the cross section (MF3) and forget to simultaneously update the angular data (MF4) to be consistent. This may lead to problems, as recent studies have shown the importance of emission distributions, particularly for elastic and inelastic reactions.

Organizing the hierarchy also offers the opportunity to store and use data at a level best suited to a given application. For example, models of the Prompt Fission Neutron Spectrum often can provide estimates of the average neutron multiplicity nubar, and both should be collocated in the same fission reaction hierarchy.

So far we have described the potential advantages of moving to a new structure. One major obstacle to making this move is the sheer number of legacy codes that can only read ENDF-6 formatted nuclear data. Therefore, for the near term (~10 years), an important requirement for the new structure will be to continue handling all data types present in ENDF-6, and for the infrastructure to provide backwards-compatibility with the legacy format, including translating both to and from ENDF-6. This has the added benefit that it enables benchmarking and testing data and tools using the new structures against legacy ENDF-6 results.

Moving nuclear data technology away from ENDF-6 will require substantial time and effort from the community. We expect the move to offer several immediate advantages, however:

- a nested, hierarchical structure making the data clearer and easier to understand,
- defining a "master" data representation and providing a history showing how other data (grouped data, for example) were derived from that master representation,
- improved flexibility in the types of data that can be stored, and
- the ability to attract new talent who wish to use modern tools and who may be put off by the existing set of nuclear data tools.

System Overview: To succeed, the new nuclear data structure must not only be easily integrated into the current system of evaluating, processing and using the data (a multi-step process with many different interfaces where quality control and workflow issues are important), but must also help make that system more robust. An overview of the current system (along with some possible additions) is given in Figure 3. The new structure design should improve the interactions and enhance workflow between experiment/theory, evaluation, application codes and regulators. A central desire is to simplify how data are generated and used by reducing the need for direct data handling and manipulation by the average user.

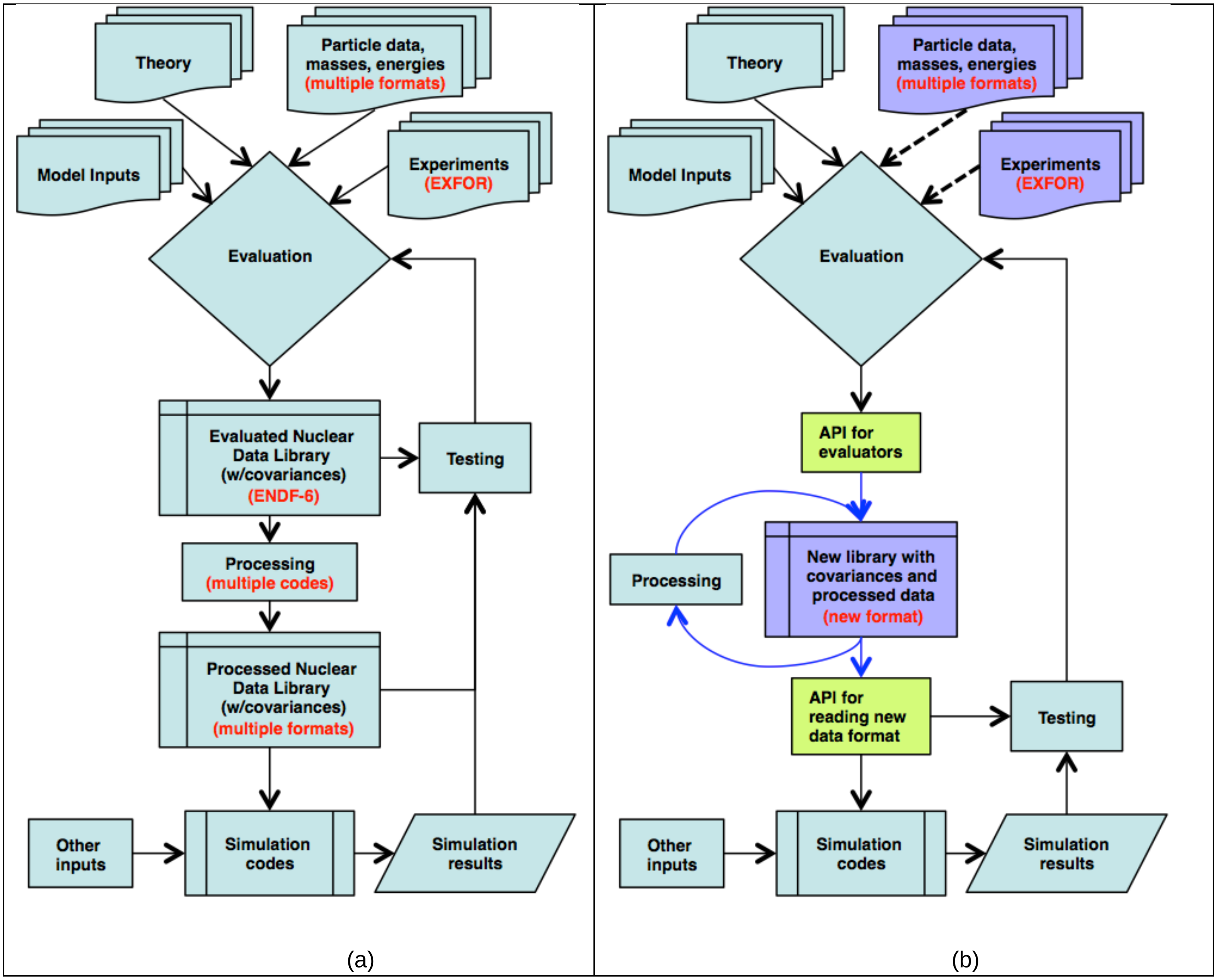

Figure 3: A summary the nuclear data 'workflow', showing how data are generated by evaluators, processed and used for simulations, tested and refined. Figure 3(a) shows the current workflow, and figure 3(b) shows how a new structure should fit into the existing workflow. The new structure must simplify how data are handled but must not disrupt other parts of the workflow. A few other changes are present in figure 3(b), including new APIs for reading and writing nuclear data (in green), and data containers that are used in evaluated, experimental and particle data (all of which are shown in purple). These databases should all use the same structure for basic data containers.

As seen in figure 3, the nuclear data workflow involves many different data storage structures even though the physics contents of each structure are similar. The first step towards simplifying this workflow is to identify common features in different structures (experimental, evaluated and processed data), and design general-purpose containers that can be re-used for storing each type. Examples of general-purpose data containers include lists of x-y pairs, tables and matrices. These are all used in multiple databases, and by standardizing these basic types we can simplify data handling (for more detail please see the section on data containers).

The workflow can also be made simpler and more robust by defining an Application Programming Interface (API) for accessing data rather than requiring each user to implement their own file i/o. The API would provide an abstraction from how the data are stored, and protect users from future changes in data storage. This data abstraction would make the software that uses it more robust since the structure details could be changed without touching the API. The API should be capable of accessing, modifying and writing the new nuclear data structure. In addition, a set of unit tests for the API should be developed.

Another avenue for improving workflow is to merge the structures for 'evaluated' and 'processed' nuclear data. Processed data in particular was separated historically into many competing and often proprietary formats used by the many application codes that need nuclear data. These applications require processing to go from the basic nuclear data evaluation to the application library for a specific code. All these codes have been written in different languages ranging from Fortran to C, C++ or Java. A better solution is to make a single storage structure that is general enough to handle evaluated data as well as several forms of processed data, along with a single API for reading and writing the data. In the near term, application codes would need translators to put data into their proprietary formats, but in the longer term they should move to using the new API.

Once a suitable API for accessing nuclear data is defined, it should be then implemented in multiple languages, likely including Java and C with wrappers for C++ and FORTRAN codes. Wrappers in other languages (e.g. python) could be provided at a later stage or could be contributed by users. The API could also provide specific (low level) operations that allow for the elimination of redundant data such as summing partial data (e.g. cross sections) into total data. Simple unit conversion could also be added.

One more important feature of the existing nuclear data system is redundancy for quality assurance. This means having multiple codes and applications developed independently to help test and fix data. The new structure will need redundant software infrastructure for various purposes including basic visualization, data and physics checking for evaluation QA, and basic processing (resonance reconstruction, Doppler broadening, etc.). The new structure will also need translators for putting data into widely used proprietary formats such as ACE for MCNP(X), so that the output can be compared to the results from other codes such as NJOY.

<u>Benefits and requirements for data evaluation and processing</u>: In the previous section we discussed how a new structure might fit into the existing system for evaluated nuclear data. There are a few more features that the new structure should support in order to be useful for assuring the quality of a nuclear data library. We recognize that the conversion of legacy ENDF-6 evaluations into the new structure cannot adhere to all of the recommendations in this section. Thus, the new structure must allow for backwards compatibility.

One possible solution to this would be to require that all evaluations performed after date X[2] follow these recommendations, while making them optional in evaluations performed before that date.

Multiple representations: For storing a cross section, it is often convenient to use a compact parameterized form, especially in the resonance region. Before plotting, these parameters must be reconstructed into a pointwise representation. The new structure should streamline plotting as well as checking by allowing both the original and derived representations of a quantity to be stored simultaneously. The same principle also applies to other quantities, such as distributions, that may have both parameterized and pointwise forms. In each case, the hierarchical nature of the new structure allows storing multiple forms at the same level.

The same nested structure could also be used to store processed data along side the evaluated data. In this way a single file could provide data that can be easily plotted, checked and interpolated. For example, for a cross section the following forms could be stored simultaneously:

- Resonance parameters with a background cross-section,
- Pointwise data with a given reconstruction precision and interpolation mode,
- Probability tables (pointwise or multigroup with a weighting flux),
- Grouped data (with a weighting flux)

Support for a larger variety of output channels: Another potential advantage of the new structure could be in handling more refined reaction channels. Evaluated data should be given in the greatest detail known by the evaluator. For example, consider the two processes:

H3 + He3 → H1 + (He5 → n + He4)

H3 + He3 → n + (Li5 → H1 + He4)

These reactions give the same final products, and in the current formats they must be lumped together even though they provide different kinematic properties for neutron and proton. In the new system, these reactions should be treated separately.

Accommodating new features: Evaluators may wish to use new forms for storing nuclear data, which have not been adopted by the governing organization. In this case it would be convenient for the API to skip over the unrecognized section rather than crashing. This may not be possible for all types of data, but where possible it will add flexibility.

Require evaluators to provide detailed information needed to reproduce and extend their evaluations: An evaluation should include or link to the experimental data, models and parameters that were used when it was created. These would help future evaluators who might wish to recreate an evaluation in order to revise or extend it, and it would also help users to

[2] Date X will be determined later but may, for example, be one year after the release of the new structure.

assess the quality of the evaluation. This requirement could be particularly important if different parts of one evaluation were handled by different evaluators. Extra data would allow users to understand how both contributions were created, and to be aware of any possible inconsistencies.

This requirement means that the new structure must have containers which support storing input files and links to experimental data.

Extensions for handling covariance data: Covariances make up the fastest-growing part of modern nuclear data libraries, and a new structure is an opportunity to explore new solutions to the longstanding problem of storing and using covariances. Some features were satisfactorily resolved in the past but others still need to be resolved. In particular, there is currently no provision for storing covariances between energy and angular distributions of secondary particles.

An interesting approach that has scarcely been used so far is the ENDF MF 30 format, consisting of model parameter covariances along with sensitivity profiles. No processing for this format has yet been developed, but it offers some interesting possibilities particularly in view of error propagation for basic nuclear parameters.

Coupled energy angular distributions uncertainties (ENDF MF 36) seem to add excessive complexity at present stage. This capability can be circumvented by the separability assumption using angular covariances (MF 34) and secondary energy covariances (MF 35). Further analyses are needed to evaluate the impact of such simplifications.

Eliminating unneeded redundancy: At the first WPEC 38 subgroup meeting there was much agreement to remove redundant data (i.e., data that can be derived from other data) for an evaluation. For example, Q-values are not needed as they can be calculated from masses. In ENDF files, Q-values are often inconsistent with the masses given in the same evaluation. Furthermore, in an ENDF file multiple mass values are sometimes listed for one particle. These inconsistencies lead to many downstream issues. Since many evaluations share common particles (here, the term particle includes composite objects like nuclei), it would be desirable to have a separate dataset that contains particles and their properties (e.g., mass, spin, charge). That is, at least two data structures should be defined. One structure defining particle properties, and another structure defining reactions and their properties for a projectile and a target (i.e., a reaction evaluation). The structure focusing on reactions would link to the particle data used to generate the evaluation, reducing the likelihood of inconsistent values due to typographical errors that creep in when the same data are stored in multiple places. A separate standardized data structure for particle information might also encourage more consistent usage across multiple evaluation efforts for the same reactants.

A particle properties database: Significant benefits could be realized by storing particle properties in a single place that contains all particle properties required for an evaluation. The result would be similar to the RIPL database, except general enough to store particles other than nuclides. This dataset must contain version information so that an evaluation can reference a particular version without the need for copying the particle information into the evaluation. The reaction structure should use links to reference data in external databases, such as particle properties. The structure of a link must be defined so as to be uniquely resolvable.

Reaction designators: Another important concept is to have a consistent model that allows one to interpret the data structure naturally without having to resort to a complicated rule set depending on where one is located in the data tree. The reaction structure should support any type of particle for projectile, target and product. Furthermore, there should be no restriction on the type of allowed reactions. In this vein it is recommended that nesting be used for products which themselves can breakup into other products. For example, nesting should be used to make the decay clear in the following output channel:

- H3 + He3 $\rightarrow$ H1 + (He5 $\rightarrow$ n + He4)

Some additional information may be needed to resolve reactions that contain the same product list. For example, elastic scattering can be divided into shape and compound elastic. Some mechanism needs to be included in the structure so that these reactions are unique (e.g., a search can retrieve the shape elastic reaction but not the compound elastic reaction).

The need to resolve reactions goes beyond the need for clarity. An important asset of ENDF is the sharing and comparing of evaluations. However, ENDF limits detailed comparisons by limiting reaction types and forms for the data. The finer the reactions are resolved, the more checking becomes possible between evaluations. This additional information also applies to interpreting the data for processing. Processing codes must not make any assumptions about how to handle the data. Instead, the assumptions should be specified in the data for clarity and to make the processing and APIs more robust and easier to maintain.

Basic Data containers:

One reason that the ENDF-6 format has lasted for about 5 decades is that it defines a small set of convenient, basic data containers (e.g., TAB1 and TAB2) that are used for storing data. These containers reduce the coding for reading and writing files, and make it easier to check the data, both visually and with codes. It is imperative that the new evaluated nuclear data structure follows the ENDF-6 example by defining basic data containers.

Since the experimental nuclear data format (EXFOR) and the nuclear structural format (ENSDF) are also planning to modernize, it would be beneficial if basic data containers can be defined that are also compatible with EXFOR and ENSDF. A collaborative effort with the EXFOR and ENSDF community is recommended.

Many participants expressed interest in an ASCII representation for the data when expressed in a file. However, it is also realized that a binary file representation may be desired at times. The design of the basic containers should be consistent with both ASCII and binary file representations of the data.

As with other requirements, basic data containers should eliminate the need for codes to speculate on the contents of containers. Basic data containers should include specification of precision, physical units (e.g., SI), interpolation law, and the type of data contained (e.g., integer, real or character) so as to be self-consistent. Also, some containers should allow for absent or <null> values.

The new structure and its basic data containers should not restrict the size of data, although it is recognized that a practical implementation may require limits. If stored in an ASCII file, the number of values in one line can be fixed for convenience but should not be fixed by the structure definition. For example, a square 3x3 symmetric matrix with triangular representation stored on three lines as

```
1.0
2.0   3.0
4.0   5.0   6.0
```

is equivalent to all 6 values stored on one line as

```
1.0   2.0   3.0   4.0   5.0   6.0
```

It is imperative that the representation of the data be compatible with modern computer languages and not contain any idiosyncrasies of some languages. For example, in ASCII form the floating point number ‘1.2345’ can be also be stored as ‘0.12345e1, 1234.5e-3, etc. but should not be stored as ‘1.2345+0’ or as ‘1.2345+D0’ as these forms are idiosyncratic to FORTRAN and cannot be read by other languages without extra coding.

<u>Summary of requirements:</u> At its first meeting, subgroup 38 has identified some basic requirements that a new nuclear data structure and infrastructure must cover in order to meet the needs of the nuclear data community and to begin to replace ENDF-6. The new structure must:

- Be governed by a long-term WPEC subgroup that will define the structure and maintain the documentation.
- Use a hierarchy that reflects our understanding of nuclear reactions and decays, and that clearly and uniquely specifies all data.
- Define APIs for reading and writing data in the structure. Open-source substantiations of these APIs will ideally be available for C, C++, Java and Fortran before adoption.

- Support storing multiple representations of the same quantity simultaneously (e.g. evaluated and processed forms).
    - It will be required that evaluators identify their primary evaluated form if multiple forms are given.
- Support both inclusive and exclusive reaction data (i.e., discrete reaction channels as well as sums over them).
    - Requirements for consistency between inclusive and exclusive evaluated data is to be decided by the data projects.
- Contain provisions for evaluators and data processors to provide records of all the information needed to reproduce and extend their data. Information includes a bibliography, links to the EXFOR data used including descriptions of the corrections applied by the evaluator, a description of codes and input parameters, and comments.
    - Additionally, the name and affiliation of evaluators and processors of new data or data modifications will be required.
- Encourage the elimination of inconsistent nuclear masses, levels, and lifetimes by providing a way to link to an external particle database, specifically RIPL, to specify these quantities globally for each projectile-target combination. A capability to locally override these global values will also be provided.
- Support any particle and any combination of reaction products (and subsequent decay products).
- Require the evaluator or processor to specify the physical units and interpolation of the data. Also, the data structure should contain provisions for identifying the number of significant digits used by the evaluator and during processing.
- Support backwards-compatibility with ENDF-6 as long as possible by providing a capability to translate the new structure into ENDF-6 format., However, in the long term new features will likely be added that cannot be translated back to ENDF-6.

In addition to nuclear reactions, a hierarchical structure could also be used to organize nuclear structure data (as in ENSDF), experimental data (as in EXFOR), and reaction model parameters (as in RIPL). This leads to an additional goal:

- The structure should include reusable low-level data containers that are general enough to be shared between data products (e.g., EXFOR, RIPL and ENSDF)

The goals above apply to the structure for storing data. In order to facilitate the adoption of these goals, it was recognized that supporting infrastructure is required to use the new data structure. This infrastructure should:

- Use open source codes to manipulate, search, plot, process, translate and check the data for quality. For better quality assurance and data checking, it is recognized that at least two independent sets of code infrastructure be available that can be compared.
- Be forgiving, meaning that access routines for the new structure must be able to recover gracefully and continue working if they encounter data containers that are not yet officially recognized as part of the structure.

These goals define the basic capabilities that must be included in the new nuclear data structure. Subgroup 38 has defined these goals as the first step towards actually implementing the new structure. The next step (which should take place during the March ND2013 meeting in New York and/or the May JEFF collaboration meeting in Paris) will be to start working on implementation details.

Bibliography:

[1] "ENDF-6 Formats Manual: Data Formats and Procedures for the Evaluated Nuclear Data Files ENDF/B-VI and ENDF/B-VII", ed. A. Trkov and D. Brown, BNL Report BNL-90365-2009 Rev.2, CSEWG Document ENDF-102 (2011). url: http://www.nndc.bnl. gov/csewg/docs/endf-manual.pdf

[2] D. Cullen, "PREPRO 2010: 2010 ENDF/B Pre-Processing Codes", IAEA Report IAEA-NDS-39, Rev. 14 (2010)

[3] C.M. Mattoon et al., "Generalized Nuclear Data: a New Structure (with Supporting Infrastructure) for Handling Nuclear Data", Nuclear Data Sheets **113**, 2145 (2012)

[4] M.E. Dunn and N.M. Greene, "AMPX-2000: A Cross-Section Processing System for Generating Nuclear Data for Criticality Safety Applications", ANS Transactions (2002)

[5] D. B. Pelowitz, ed., "MCNPX User's Manual, Version 2.7.0", Los Alamos National Laboratory report LA-CP-11-00438 (April 2011)