INDC(NDS)-0659
Distr. J

IAEA
International Atomic Energy Agency

INDC International Nuclear Data Committee

ENDF/X: An Extended ENDF Format

(Evolution, not Revolution)

Dermott E. Cullen
National Nuclear Data Center, Brookhaven National Laboratory (alumnus)
Nuclear Data Section, IAEA, Vienna (alumnus)
University of California (retired)

April 2014

IAEA Nuclear Data Section
Vienna International Centre, P.O. Box 100, 1400 Vienna, Austria

Selected INDC documents may be downloaded in electronic form from
http://www-nds.iaea.org/publications
or sent as an e-mail attachment.
Requests for hardcopy or e-mail transmittal should be directed to
NDS.Contact-Point@iaea.org
or to:
Nuclear Data Section
International Atomic Energy Agency
Vienna International Centre
PO Box 100
1400 Vienna
Austria

Printed by the IAEA in Austria

April 2014

INDC(NDS)-0659
Distr. J

ENDF/X: An Extended ENDF Format

(Evolution, not Revolution)

Dermott E. Cullen
National Nuclear Data Center, Brookhaven National Laboratory (alumnus)
Nuclear Data Section, IAEA, Vienna (alumnus)
University of California (retired)

1466 Hudson Way
Livermore, CA 94550
Tel. +1 925 443 1911
E-mail: redcullen1@comcast.net
Website: http://home.comcast.net/~redcullen1

April 2014

TABLE OF CONTENTS

Introduction

Recently there has been yet another round of complaints about the ENDF format not being modern and general enough to handle today's nuclear data. This has led to suggestions to abandon the current ENDF and move on to a new format. The complaints I hear I fear are based upon not understanding the primary purpose of ENDF and a lack of experience in using the ENDF format and not being flexible enough to deal with the current format. Personally I don't think that any changes to the ENDF format are NECESSARY.

But here I address the complaints that I have recently heard about the limitations of the ENDF format, and I suggest minor changes that will completely handle these complaints. In turn I would ask those who are complaining and feel that extensions are needed PLEASE give us some examples where these extensions are NEEDED. Personally I am not aware of any such data, but I am keeping an open mind and I would love to see examples that really REQUIRE extensions.

I propose a few fairly simple extensions to the current ENDF/B format; what I call ENDF/X. Compared to other suggested **revolutionary** changes, my **evolutionary** approach has the advantage that it maintains compatibility with the existing ENDF/B format that we have used successfully for almost fifty years, and still allows the format to be extended for use with other types of data.

In addition to my suggested changes to the ENDF/B format I also include a brief history of ENDF/B, in the hope that the experience we have gained over the last almost 50 years will be of help to the present generation of nuclear data developers and users. PLEASE let's not make the mistake of learning nothing from history.

Lastly I finish by identifying what I see as the weak point in the current infrastructure that we use to handle evaluated nuclear data; to me it is not the format of the data.

Overview

The ENDF/B format is now almost 50 years old. Periodically during almost its entire lifetime there have been claims that this format is inadequate and MUST be replaced by something more modern, more specifically optimized for one computer or another. Fortunately, to date the ENDF/B format has survived in more or less its original form. I say fortunately, because even with all of its claimed limitations it has been a great success at meeting the needs it was originally designed for.

My personal feeling is that many of the complaints about the ENDF/B format are because today people have forgotten, or never knew, the purpose ENDF/B was initially designed to serve. So let me tell you: **ENDF/B was designed to allow nuclear data to be exchanged between data users in a simple, computer readable, but computer independent form** (it could be read on any computer). What was most important was reliability and simplicity, so that virtually anyone (experimentalist, evaluator, data processor, and end users) could have easy access to the data, to create it or use it. Efficiency was never a major concern; then and even today ENDF/B is a rather small database (the entire ENDF/B-VII.1 library is less than 1 gigabyte) [5]. This is so small that I can easily fit the entire library onto my iPhone.

Unfortunately over the years we have lost sight of this purpose and we have moved away from it, and put progressively more pressure on ENDF/B to be extended for purposes that it was never intended to serve. Here I propose some simple what I call evolutionary extensions of the ENDF/B format to both preserve the infrastructure that we have built up over the past almost 50 years, and to still try to address the complaints that I have heard about the limitations of the current ENDF/B format.

My Proposal: Evolution, not Revolution

I propose that we introduce changes to ENDF/B in small, evolutionary, increments, rather than throwing away the existing proven format, and making revolutionary changes. What I propose is:

1) **Maintain the current format** that we use for neutron and photon data; as well as electron and charged particles. This is the system it has taken us almost 50 years to put together, with the investment of many millions of dollars, and hundreds, if not thousands, of man-years of efforts by the entire nuclear data community. In my humble opinion it would be **crazy** to even consider replacing this tried and true system with something totally new.

2) **Extend the definition of MAT/MF/MT**, to allow more values for each of the fields. But in order to accommodate my point 1) I propose that we can do this without changing the columns used by each of these.

   I proposed that we extend the definitions of MAT/MF/MT from their current **decimal** definitions (10 values) to **alphanumeric** (36 values). I propose that initially we use this extension only for the lead column of each field. Even this simple extension will increase the number of possible values by a factor of 3.6: MAT = 9999 to 36000, MF = 99 to 360, MT = 999 to 3600. Later if even more possible values are needed this can be extended to all columns, which would increase the number of available values to more than would be accomplished by adding another column to each of these fields, as some people have suggested.

   I further propose that these extensions are not necessary for use with our current data, so that we can maintain compatibility with our existing data files.

3) **Extend the precision of the data fields** from six per line (6 X 11) to three per line (3 X 22) to allow for increased precision required for new data.

   I propose that in addition to our current record types TAB1 and LIST, each of which uses up to six data fields per line (6 X 11) we introduce new record types TAB1X and LISTX, each of which would use up to three data fields per line (3 X 22). This need only affect the data tables, not the header lines for each record type.

   I will merely mention that 22 columns would allow for a full 16 digits of precision available using 64 bit arithmetic, plus a sign (+ or -), decimal, and four columns for an exponent (E+12) = 22 columns.

I further propose that these extensions are not necessary for use with our current data, so that we can maintain compatibility with our existing data fields. Although here we may want to make an exception to handle correlation data, which may be easier to handle using these extended formats than the current somewhat awkward procedures and formats.

New types of data may be defined in the ENDF/B Formats and Procedures Manual, ENDF-102 using either the current formats or these new extended precision record types.

Those are all of the changes that I propose; so simple that they can be completely defined in little more than one page of text. And most important these changes need not have any impact on our current data processing codes**. By definition any new types of data that are added to the ENDF format using my extensions currently have no support codes, and should not expect existing codes to be necessarily be extended to handle new data, i.e., if you want different types of data you should expect to supply the infrastructure to support it or the funding to have somebody else do it for you.**

Background Information

I feel that these are the only changes needed to address the "problems" that people have recently expressed concerning the current ENDF/B format. With that as an introduction I will now supply more background concerning ENDF/B.

Development of ENDF began in the early 1960s by Henry Honeck at Brookhaven National Laboratory. At the time in the United States each laboratory had its own internal systems for handling nuclear data. Each of these systems was designed to meet the needs of each individual laboratory, but they were generally incompatible, to the point where nuclear data could not be easily exchanged between laboratories for use or comparison.

At this time the British already had a nuclear data system, and Honeck used this as a guideline to try and develop a common system for use in the United States. His idea was not simply a computer readable format, but also of equal if not even greater importance, a strict set of rules defining the data; at the time it was surprising how much in-house jargon had been adopted at many laboratories using the same terminology for completely different physical data. For almost 50 years these rules and formats have been documented in ENDF-102 [1]. These rules were designed to be used – and obeyed – by the entire nuclear data community: experimental measurers, evaluators, data processors, and data users. Over the years ENDF-102 has also served as the repository for clarification of rules based upon their actual use in all areas of application.

You might think that this idea of a common nuclear data format for use by everyone in the United States would be applauded and immediately adopted by everyone. Unfortunately, such was not the case: many people thought it was a good idea, but such is human nature that each of them thought that their in-house system was the obvious choice for everyone to use. So initially there was a "turf war" between competing systems to see which, if any would be adopted. I almost hate to write this, but the reality of why ENDF/B beat out all of these other systems, some of which were much older, and had more experience, and were arguably better than ENDF/B, is that Henry Honeck moved from Brookhaven to Washington, gained control of budget money and dictated that ENDF/B would be used by everyone – that is everyone who wanted funding. Needless to say this didn't make Henry very popular, but it did get ENDF/B up and running and led to the system used today throughout the world.

My point in telling you how ENDF/B came to be accepted is merely to stress that it didn't win a beauty contest, nor was it judged to be the best available format. What was most important was to halt the "turf war" between competing formats and get everyone to adopt the same format, even if they were initially forced to. That is what Henry Honeck managed to achieve. Today there is no such pressure on ENDF/B users to move on to a completely different format. By now for almost 50 years users throughout the world have used the simple text-based ENDF format that can be read on any type of computer by anyone with something as simple as a text reader. Why should they give this up for some data based system, such as TML, which may or may not still be here 10 or 20 years from today; whereas, I think we can assume simple text will be here, since it is the fastest growing mode of communications between billions of kids today.

A big step toward the success of the ENDF/B system was the creation of the National Nuclear Data Center (NNDC), at Brookhaven National Laboratory, demonstrating the need for a national effort, and the formation of the Cross Section Evaluation Working Group (CSEWG), with members from most major United States laboratories that created and/or used nuclear data. This body periodically convenes at the NNDC to oversee adoption of all rules and procedures relating to ENDF/B and to set evaluation priorities. This helped to bring the laboratories closer and make everyone an insider who controlled the direction that ENDF/B was to take in the future.

Why it is called ENDF/B? Originally Honeck proposed two data files: **ENDF/A**, which was to store partial evaluations in any format, that could be contributed by any evaluator and in turn used by any evaluator as a portion of their evaluations, and **ENDF/B**, which was to store complete evaluations that could actually be used in applications. As such ENDF/A was an idea rather than a fixed format. It soon became obvious that the ENDF/A idea was not very practical; the problem was that anybody could dump anything they wanted into it using any definitions and any format that they wanted, which made it impractical for anyone to get anything back out of such a system, that they could reliably understand and use. By 1970, while working at the NNDC, I proposed that we essentially abandon the ENDF/A idea, and we insist that if evaluators wanted to contribute or even exchange data it must be done using the ENDF/B format, with a simple flag at the beginning of the file indicating it is an incomplete evaluation. From this point on at least to my knowledge the idea ENDF/A idea was abandoned.

A few years after the ENDF effort commenced NNDC also started developing a new computer based system to handle the storage and exchange of experimentally measured data. Initially it was proposed that this be in a computer data base structure format appropriate for use with the new computer language PL/1, which was to be "the wave of the future". Needless to say this was a wave that never broke and fortunately we decided not to follow this route. Based on the early success of ENDF/B and using it as a guideline I designed the **EXFOR** system as the **EX**change **FOR**mat for experimentally measured data. Similar to ENDF/B this was an 80 column per line format, with 66 columns assigned to data, and 14 columns for identifying information (the equivalent of ENDF/B, MAT/MF/MT). It was obvious that there were far more experimental measurements than neutron evaluations, so that even back then I visualized that the EXFOR accession and sub-accession field could be extended from decimal and alphanumeric, exactly as I am now proposing for ENDF/X.

Because of its name: **EXFOR** = the **EX**change **FOR**mat people have not forgotten what this format was designed to accomplish: exchange experimental data between data creators and users. Largely because of this there have been few changes to the EXFOR system I designed over 40 years ago. In contrast I fear too many people have forgotten that this is exactly what ENDF/B was designed to do: exchange data, and because of this almost throughout its entire lifetime there has been continuous pressure to change ENDF/B and somehow make it more general.

By the mid-1970s, ENDF/B had become a great success, with most of the major United States laboratories using it, and participating in the Cross Section Evaluation Working Group (CSEWG), the body that oversees ENDF/B rules and content. One major reason for the success of ENDF/B was the development and release of many computer codes that used the ENDF/B format.

Still up to this time many laboratories used ENDF/B strictly for the purpose it was defined for: to exchange data – period. As a first example I will mention that Henry Honeck eventually moved from Washington to Savannah River where he developed the Joshua system. He understood the purpose of ENDF/B well enough to use it to send and receive evaluated data, but it was not used within his in-house Joshua system. As a second example, where I worked at Lawrence Livermore National Laboratory, when we received ENDF/B formatted data we almost immediately translated it into our internal ENDL format, and all of our subsequent processing was done using the ENDL format. Other laboratories, such as Los Alamos, Oak Ridge, Argonne, Savannah River, etc., had similar in-house systems. At least in the case of Livermore there was never any question of ENDL replacing ENDF/B outside of Livermore. For example, the ENDL only uses simple tables of data, there is no thermal scattering, no resolved or unresolved resonances, etc., which made it totally inappropriate for use in say reactor calculations.

But slowly as funding was decreased more and more, users and laboratories abandoned their in-house formats and started to more and more move toward using the ENDF/B format and to rely on the many codes that could process ENDF/B data. This began to put more pressure on ENDF/B to be more general – not more general to achieve what it was originally designed to do, only exchange data, but rather to meet more and more diverse needs of individual data users and laboratories. But this also had the more subtle effect of reducing the number of different systems that could process ENDF/B data.

ENDF/C

By the mid-1970s ENDF/B was a great success, but even then there were complaints that ENDF/B was inadequate and should be replaced by a new, more up-to-date advanced system named ENDF/C. This was a major effort involving many people, much time and effort, but I knew it was bound to fail. I knew this because even then, only roughly a decade after the start of ENDF/B, so much development had been done using the ENDF/B format that few people if anybody would be willing to abandon ENDF/B and start all over with a great new ENDF/C system. But at the time I felt like Cassandra, because nobody wanted to hear that their effort on ENDF/C were bound to fail, and sure enough just like Cassandra when this effort failed and was abandoned nobody remembered my warning.

Even with that as background I will again play Cassandra today and predict that efforts toward abandoning the ENDF/B format for yet another great new format are bound to fail. It will fail basically for the same reason that the ENDF/C effort failed: there has been too much investment in

ENDF/B to abandon it. Today there is the additional problem there compared to during ENDF/B's formative years, today there are far fewer available resources to try and replace the existing ENDF/B infrastructure. I will end this paper on a positive note by encouraging work and telling you where I think our efforts should be concentrated.

Compatibility

An obvious question to ask is: If the new format is 100% compatible with the old format, why do we need a new format? What do we gain?

Having worked with various nuclear data formats over the last 45 years, I have a lot of experience in using different formats and addressing compatibility issues. I worked at NNDC from 1967 until 1972; I then moved to LLNL. So I have used ENDF/B almost from its start, and for the last 40 years I have also used LLNL's ENDL format. During this time I never had any compatibility problems as far as going back and forth between the ENDF/B and ENDL formats; a one-to-one correspondence was always fairly easy to define, and both formats were flexible enough to extend to meet our application needs. As an example, I will mention that the ENDF/B-VII library uses all of my photon, electron, atomic data libraries [6, 7, 8, 9], which I developed in the ENDL format and Bob MacFarlane, LANL, translated to the ENDF/B format. This required that we extend both ENDL and ENDF/B to handle electron and atomic parameters and also to extend both formats to handle atomic sub-shell parameters for photon data; we had no problem doing these extensions.

But changing ENDF to address the problems of more values of MAT/MF/MT and/or more precision than can be accommodated in the present ENDF format **OBVIOUSLY MAKES BACKWARD COMPATABILITY IMPOSSIBLE**. For example, if a new format has a 5 digit MAT or 3 digit MF or 4 digit MT, these obviously cannot be translated from the new format back into the current ENDF format. Similarly if the new format requires data to 16 digit precision (i.e., full 64 bit precision), this cannot be translated from the new format back into the current ENDF 11 column data format. Conversely if data is backward compatible isn't it obvious that no changes in the ENDF format were needed. So it sounds like Catch-22: If new data requires a new format because doesn't fit in the existing ENDF format, by definition obviously it cannot be made backward compatible.

From the point of view of V&V (Verification & Validation), and QA (Quality Assurance), the previously used versions of data in the ENDF format need to be available and useable for comparison purposes. So any new format would need to address the issue of backward compatibility. Obviously we are faced with the fact that addressing the major complaints that I have heard about the ENDF format will mean giving up backward compatibility. So think long and hard before making the leap to a new format, because once you commit yourself there will be no going back.

The Importance of Code Comparison

One important point to consider is that even with the strict formats and conventions defined by ENDF-102, it has not been that easy to verify that different data processing codes produce the same results even for the simplest quantities of interest.

Roughly two decades after ENDF/B was adopted I began a study that I initially assumed would demonstrate the great strides we had made in the accuracy of our nuclear data since the adoption of ENDF/B. Many nuclear data processing codes from throughout the world were asked to start from nuclear data coded in the ENDF/B format and to process it into unshielded, multi-group cross section [2, 3, 4]. Surely this is the simplest possible quantity that we could derive from our nuclear data files, and we would find excellent agreement.

The results shocked me, and everyone else that saw them. The results showed that no two processing codes produced results that agreed with each other. The results were differences from a few per-cent, to factors of two (100%), to orders of magnitude differences. Let me repeat this because you may think this was a typo: THEY WERE ALL WRONG!!! Every single code had one or more glaring errors.

The one important lesson that we learned from this study is the importance of code comparison. Our codes are far too complicated to allow us to assume that good intentions and hard work will result in accurate results. But once we performed comparisons we were able to recognize problem areas in codes and localize the possible sources of our errors that allowed us to improve our codes.

Hopefully this will serve as a WARNING to anyone developing new nuclear data processing codes, or even improving existing codes, to avoid the mistake of assuming you are perfect; PLEASE compare your results to those produced by other codes. Our experience has been that investing the time to verify code results, actually saves development time and effort, and most important, greatly improves reliability.

Based on this initial study started thirty years ago we have continued our code comparison efforts even up to today. But with the passage of time most of the processing codes that participated in our comparisons that started in the 1980's have by now been abandoned, and today there are only a few nuclear data processing codes that we can compare. To me this is a very scary trend which in the future will make it progressively more difficult, if not impossible, to verify the accuracy of our processed data. This leads me to my conclusions as far as where we should put our efforts in the future.

Where do we go from here

Today once again there are complaints about the limitations of the ENDF/B format, and even effort to develop a new format to replace ENDF/B. As I have also repeatedly mentioned I think that these efforts are in the wrong direction.

Today I feel that the weakest point in the ENDF/B infrastructure is not the format of the data, but rather the age of the code developers and the codes that we are using to process data. Above I have attempted to define how I feel the ENDF format can be easily extended to handle new types of data, so I do not feel that the existing ENDFB format is our biggest problem. What I see as our biggest problems today are that too many of our processing codes were originally written thirty to forty years ago by people who are now retired or approaching retirement. And the codes are written in FORTRAN, the most popular scientific computer language way back then. But today hardly anyone graduating from school is taught FORTRAN; today everyone is taught C and C++.

To me I think it would be crazy to try to develop a new format and a new set of processing codes to handle this new format BOTH at the same time. I feel that it would be better to keep the existing ENDF/B format, and concentrating on developing a new generation of computer codes written in C and C++ to process data in the ENDF/B format. The first goal of these new codes is to verify that they can reproduce the same answers as our current FORTRAN codes. Only after this goal has been met should we address possible further extensions of the current ENDF/B format.

Recently there have been some interesting suggestions to use more modern database management systems to store all of our nuclear data. I think we should applaud these efforts and encourage more of this activity for in-house use, where individual users may want to optimize their methods to access and use nuclear data. But I think it is inappropriate to use these methods as a replacement for ENDF. Again, I ask everyone to concentrate of the primary purpose of ENDF: **To exchange data between laboratories in a computer independent, but computer readable form**. For the exchange of data reliability and accuracy are paramount; efficiency should only be a minor consideration. My prediction is that ten or twenty years from now XML will have gone the way of PL/I, and the many other short lived computer languages, but simple text files will always be with us, so that's what we have used for almost fifty years and it is what we should continue to use in the future.

All nuclear data creators and users should concentrate on the fact that if a new format is adopted **NONE OF THE CODE YOU CURRENTLY USE WILL WORK**. So please carefully weigh any advantages you may see in a new format against the amount of work you will have to do to use it.

I have tried to document here that based on our experience with ENDF/B almost 50 years ago, it will be no small task to develop a new generation of verified codes. But I do encourage people to take on this task, and don't get discouraged. Remember Rome wasn't built in a day – and neither was ENDF/B.

References

[1] A. Trkov, M. Herman and D.A. Brown "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, Report BNL-90365-2009 Rev. 2, December 2011, Brookhaven National Laboratory.

[2] D.E. Cullen, W.L. Zijp and R.E. MacFarlane "Verification of Nuclear Cross Section Processing Codes", Proc. of the Topical Meeting on Advances in Reactor Physics and Core Thermal Hydraulics, 1069, Kiamesha Lake, New York, September 1982.

[3] D.E. Cullen, "Report on the IAEA Cross Section Processing Code Verification Project", IAEA, Vienna, INDC(NDS)-170, May 1985.

[4] D.E. Cullen, "The Accuracy of Processed Nuclear Data", Nuc. Sci. and Eng., 99, 172-181.

[5] D.E. Cullen, "POINT 2012: ENDF/B-VII.1 **Final** Temperature Dependent Cross Section Library", LLNL-TR-534938, January 2012.

[6] D.E. Cullen, "ENDL Type Formats for the LLNL Evaluated Atomic Data Library (EADL), Evaluated Electron Data Library (EEDL), and Evaluated Photon Data Library (EPDL)", Lawrence Livermore National Laboratory, UCRL-ID-117796, Rev. 1, May 2002.

[7] D.E. Cullen, et al. "Tables and Graphs of Photon-Interaction Cross Sections from 10 eV to 100 GeV Derived from the LLNL Evaluated Photon Data Library (EPDL), UCRL-50400, Vol. 6, Rev. 4, Part A: Z = 1 to 50, October 1989.

[8] D.E. Cullen, S.T. Perkins and S.M. Seltzer, "Tables and Graphs of Electron Interaction Cross 10 eV to 100 GeV Derived from the LLNL Evaluated Electron Data Library (EEDL), Z = 1 - 100", Lawrence Livermore National Laboratory, UCRL-50400, Vol. 31, November 1991.

[9] D.E. Cullen, et al. "Tables and Graphs of Atomic Subshell and Relaxation Data Derived from the LLNL Evaluated Atomic Data Library (EADL), Z = 1 - 100", Lawrence Livermore National Laboratory, UCRL-50400, Vol. 30, October 1991.

Appendix: ENDF/B Precision

This is a brief summary of the precision to which we could represent floating point numbers in the ENDF/B format over the years. This may sound strange because the ENDF/B format has not changed much over the years, so how could the precision change? Since each floating point field has always been 11 columns wide, what changed?

Initially the ENDF/B data was written using a FORTRAN format statement E11.4, which on some computers gave 5 digits of accuracy, e.g.," 1.2345E+04", and on other computers gave 4 digits of accuracy, e.g. "0.1234E+05". This was obviously inadequate, since we couldn't reliably output even simple quantities such as ZA = 92238 (needs 5 digits).

The next step still used FORTRAN format statements such as 1PE11.4 to at least force 5 digits of accuracy on all computers, but on some computers this could lead to overflow for negative numbers. In addition, this still wasn't enough accuracy to represent the energy dependent shape of resonances.

In the next step after testing on every type of computer I could find, I dropped the "E", so that we could cram 6 digits of accuracy into 11 columns, e.g., ' 1.23456+04". This was quickly followed by realizing that in most cases the exponent is only one digit, so we could have 7 digits, e.g., " 1.234567+4".

Finally I realized that all energies between .1 eV and 1 GeV could be written in F format allowing for 9 digits, e.g., " 12345.6789". Outside this range there is little structure in the data, so we could accurately use fewer digits, but never fewer than 7, e.g., " 1.234567+4", or in very rare cases 6, " 1.23456+14".

My final change came some years ago when Morgan White (thank you Morgan) mentioned to me that output without an "E" could not be directly read by C and C++ codes. After I determined that C and C++ codes could produce computer dependent results, I decided to include the "E" in my output. This does not affect any numbers between .1 and 10^9; only the few numbers outside this range could 6 digits, e.g., " 1 .23456E-4", or in very rare cases 5 digits, e.g., ' 1.2345E+14"; for positive numbers an additional digit can be included by using the first column (the so-called sign column), e.g, "1.234567E-4", or "1.23456E+14".

I should mention that these improvements in precision were accomplished with minimum changes to my codes. Almost from the beginning of ENDF my codes read and write the data as characters and converted them inside the computer. Above I have mentioned many changes from one precision to another. Each of these changes was accomplished by my changing only the one subroutine that I have used for decades to convert any floating point number into a string of 11 characters that I can then output to the ENDF/B format.

In summary, based upon extensive use of ENDF over many years we have managed to go from the original four digits of accuracy to the nine or even ten digits we have today, without making any changes to the ENDF 11 column format for data. We accomplished this by focusing on the need for accurate data, and being flexible enough to deal with the current format; again, ENDF is far from perfect, but it has met our needs for many years.

We can only hope that the next generation of nuclear data creators and users are as flexible, and are willing to take advantage of the experience I describe here.

Nuclear Data Section
International Atomic Energy Agency
Vienna International Centre, P.O. box 100
A-1400 Vienna, Austria

E-mail: nds.contact-point@iaea.org
Fax: (43-1) 26007
Telephone: (43-1) 2600 21725
Web: http://www-nds.iaea.org