# *Texas A&M University*
# *US Nuclear Data Program*

## *TAMU NSDD CENTER*

## *Miscellanea of an ENSDF evaluator (evaluation issues)*

*N. Nica*

# *Miscellanea of an ENSDF evaluator*

**There are numerous evaluation problems that need attention in every day evaluators' life. Some of them are more important while some are not so much, which means that some may trigger the policy level and some maybe not. *However good governance needs good policies*. Here we present some of these:**

- ***A Suggestion for BrIcc***
- ***NuDat v.s ENSDF***
- ***0 vs. 0.0***
- ***ENSDF Analysis and Utility Programs – Version***
- ***Keeping track of Analysis and Utility Programs***

# 1. Suggestion for BrIcc

**Standard BrIcc comment:**

**147GD  CG MR$IF NO VALUE GIVEN IT WAS ASSUMED MR=1.00 FOR E2/M1,**

**147GD2CG MR=1.00 FOR E3/M2 AND MR=0.10 FOR THE OTHER MULTIPOLARITIES**

- **However, in many situations there is none of transitions is M2 or E3 or higher multipolarity and the comment is not applicable. For most of these situations I almost always forget to delete these comments even at the editorial level of review (resulting from running BrIcc up to the post-review correction stage).**

***PROPOSAL:*** *Could BrIcc detect automatically when this comment is applicable, and insert it only in such cases?*

# 2. NuDat vs. ENSDF

- **Frequent reviewer's comment:**

***…90% of users get data (only) through NuDat will not see any comment…***

- **The evaluator is advised to tune the particular situation for the bulk of data users using NuDat (NNDC) or LiveChart (IAEA) that are "comment blind"**
- **However in many evaluator's decisions comment is irreplaceable!**

***PROPOSAL:***

1. *Advise NuDat and LiveChart users on the opening page to go to ENSDF for a more thorough view of the piece of data they need*
2. *Try to educate data users about the architecture of data evaluation: that ENSDF is the primary site (built from particular datasets making the adopted, where things are explained); then data is taken by the "data only" environments*

# 3. 0 vs. 0.0

- **G.s. is usually adopted as 0.0**
- **I do not remember a policy item in this respect (if it does exist, I might simply missed it)**
- ***Tom Burrows recommended me to adopt ALWAYS 0.0 for g.s.***
- **However I found many situations in ENSDF when the g.s. is given as 0 (even 0.)**
- **It might be that for some evaluators there is no difference in between 0.0 and 0 for g.s.**
- **However it might be that for others the use of 0 is for *reaction datasets with no gammas*, where level energies usually have larger unc's – so perhaps "0" might signify that while the lowest found state, this could be not the g.s., which is 0.0**
- **But … is that true?**
- ***PROPOSAL:***

***Make a general rule for the g.s. 0.0 vs. 0***

# 4. ENSDF Analysis and Utility Programs – Version

- **Distributed by the "NSDD GitHub Repository" page – NEW!**
  **(https://www-nds.iaea.org/public/ensdf_pgm/)**
- **The more familiar distribution page still available**
  **(https://www-nds.iaea.org/public/ensdf_pgm/index_old.htm)**
- **The old page gives some details about the version of the codes (not all)**
- **Both pages have some information about the last version/date of the code but it looks like more work should be done**
- **Many times, e.g. when asking the programmer of the code about it, or simply checking that I use the latest version, we need the exact version of a code, e.g. "BrIcc v2.3e (17-Jun-2020)" – as it appears when running the core**
- ***PROPOSAL:***

- *Can be good that Version/Date should be posted in full by the distribution pages: not only "2020/08" date, or "Version 2" instead of "Version 2.3e"*
- *Alternatively this info can be given in the name of the .exe code, like e.g.: BrIcc_v2.3e_17-Jun-2020.exe (Jun already put the date in the name)*

# 5. Keeping track of Analysis and Utility Programs

- **An evaluation can stay many years in the evaluation pipe before being published and entering ENSDF.**
- **I typically spent several years on a mass chain in the evaluation pipe during which the codes can change.**
- **More generally we do not keep track of the versions and sub-versions of our evaluation codes and after several years it is difficult to see what code was used in case some problems emerged.**
- **There are codes (Ruler, Gabs) that were used for many years with some bugs and after the initial programmer has changed the program was re-written of refurbished.**
- **However keeping track of the codes remains unclear**
- ***PROPOSAL:***

- *One can keep track of the codes by making them place a "d" comment in the dataset about version & date each time we run the code.*
- *This is a way to prompt an evaluator to check for the currency of the codes for each mass chain.*