DE LA RECHERCHE À L'INDUSTRIE

SG43 Status WPEC 2020

Code Infrastructure to support a general nuclear database structure (GNDS)

Goals

- To define an interface (API) for reading/writing GNDS
- To define checks to “validate” new evaluations

Stretch Goals

- To develop and share implementations of:
  - Reading/writing tools for evaluation manipulations
  - Visualization tools
  - Tools to assist with uncertainty quantification

- To develop and share implementations of
  - Checking tools

API

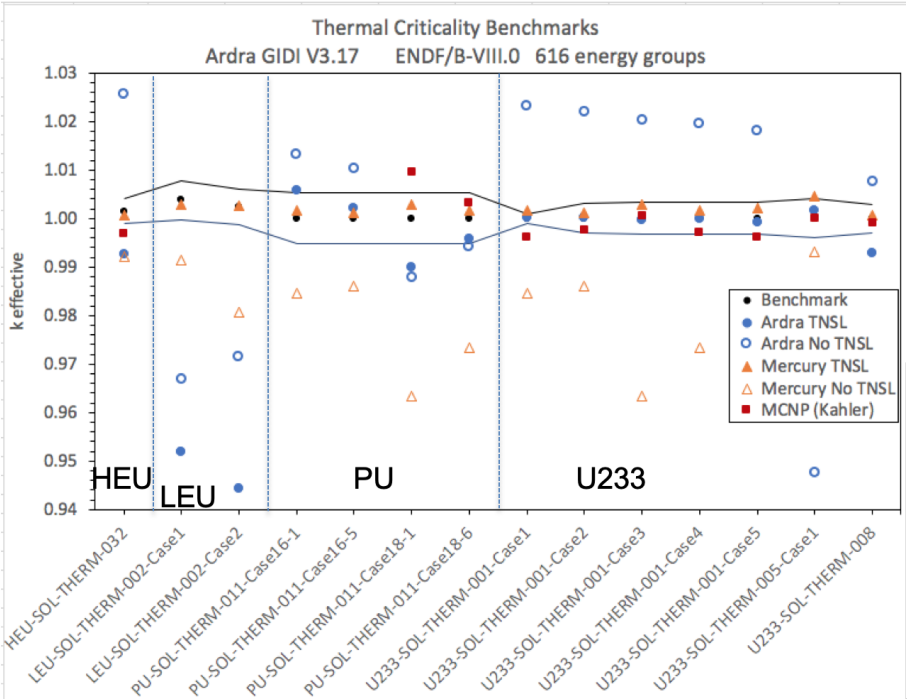
- Discuss together
  - NEA SG-43
- Implement separately (situation 2019)
  - LLNL has two implementations of read/write API, compliant with GNDS 1.9 (ENDF/B-VIII.0)
    - Fudge (Python) https://github.com/LLNL/FUDGE
    - GIDIplus (C++) https://github.com/LLNL/GIDIplus
  - ORNL has a partial implementation in AMPX (C++)
  - LANL is writing a specification document for NJOY (C++)
  - CEA will be starting its implementation soon in GALILEE (C++)
  - JAEA is planning to use LLNL implementations, when available

GNDS reading/writing implementations

Code Name	Institution	Status	Availability when / where
FUDGE	LLNL	Full Implementation	Now GitHub
GIDI+	LLNL	Full Implementation	Now GitHub
AMPX	ORNL	Partial Implementation (1D XS + Cov)	Soon ORNL site
NJOY	LANL	In progress	2020 GitHub
GALILEE	CEA	In progress	2021? NEA
FRENDY	JAEA	Planned	2023? NEA

Demonstration of capability

- LLNL transport codes have been updated to run problems using GNDS data via GIDI API



Checking

- Four documents have been collected from LANL, BNL and CEA with lists of checks locally implemented

WHAT IS NEXT ?

- Close SG-43 with a summary report on the status of different implementations
- Continue implementations as needed, by each institution, in their processing codes
- **Keep exchanges going** between institutions and other stakeholders
  - As an item in EG-GNDS meetings?
- Restart a new SG in a few years if need is felt

THANKS FOR YOUR ATTENTION