SG43 Reporting May 2019

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

- 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

Demonstration of capability

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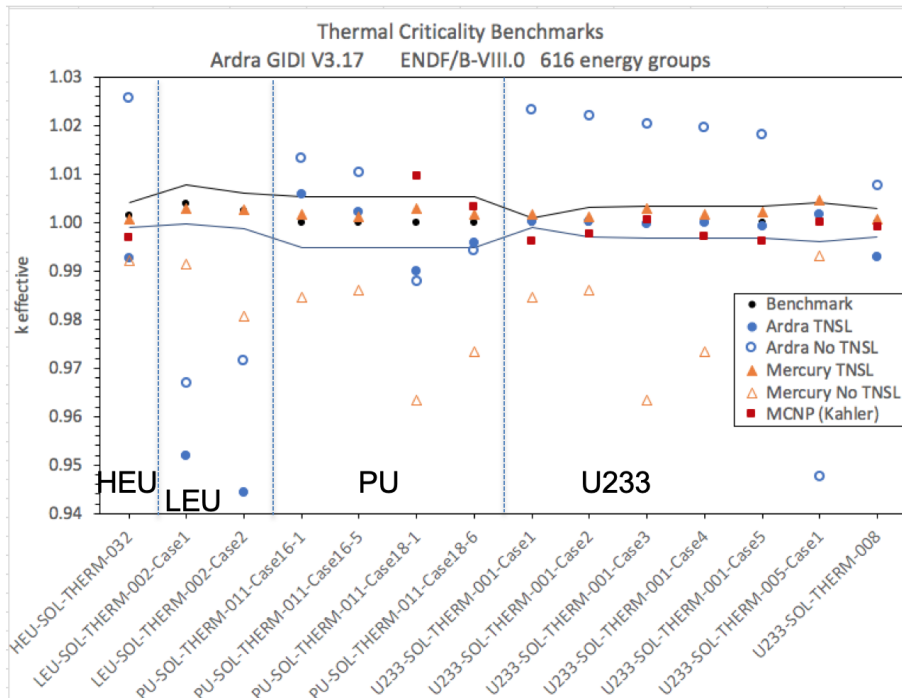
Checking

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

- Continue implementations as needed
- Extract actual APIs from working implementations
    - Post-it in NEA-Gitlab
    - Compare them for consistencies/inconsistencies
- Start draft report
- Situation assessment before end 2019

Commissariat à l'énergie atomique et aux énergies alternatives - www.cea.fr