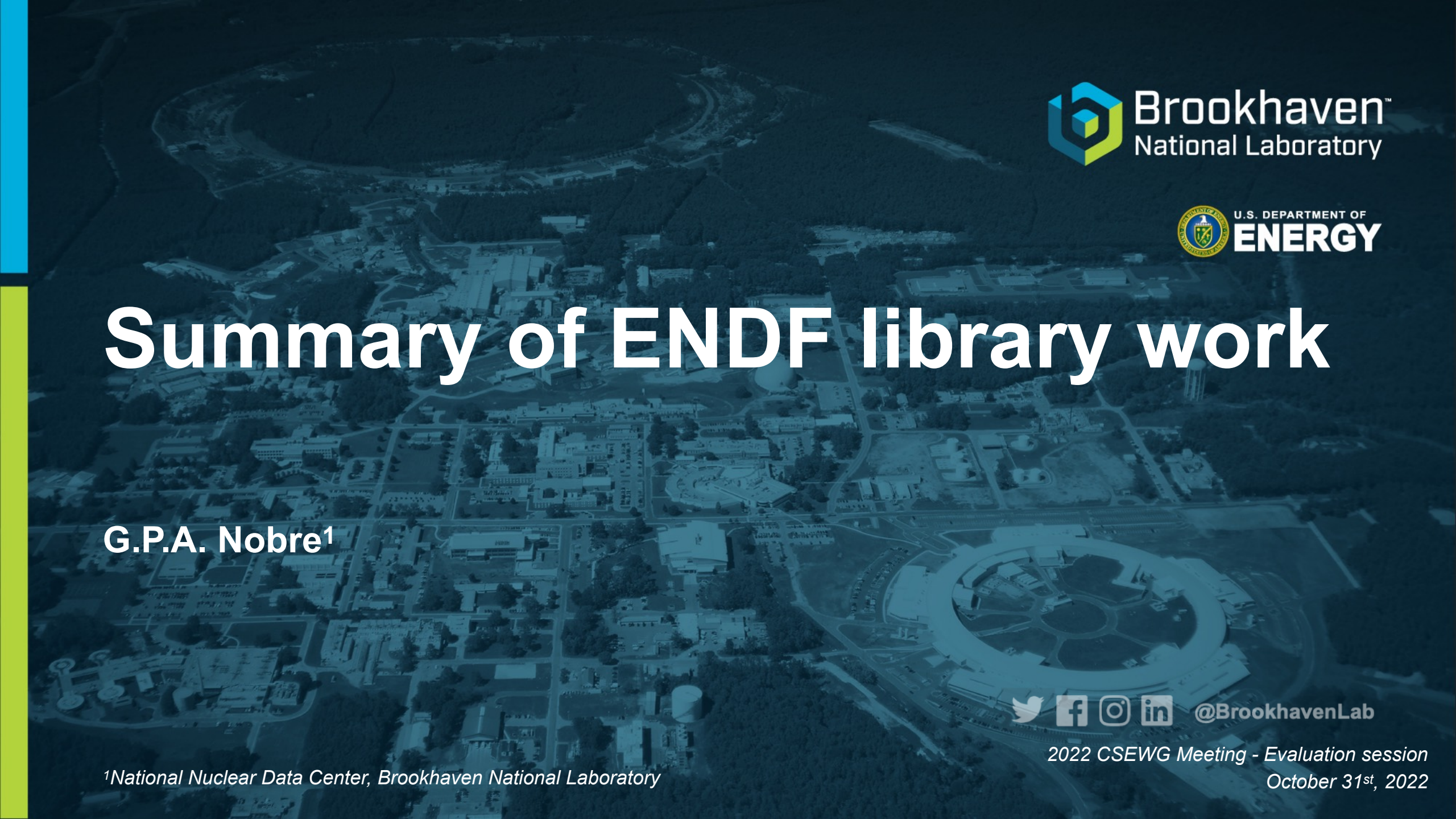


Summary of ENDF library work

G.P.A. Nobre[1]

[1]National Nuclear Data Center, Brookhaven National Laboratory

2022 CSEWG Meeting - Evaluation session
October 31st, 2022

Topics

- Release Timeline
- 2022 Hackathon
- ENDF/B-VIII.1-Beta0
- Review status for neutron sub library
- A few words on reproducibility
- A couple of things to keep in mind
- Reviewers wanted

Release timeline

ENDF/B VIII.1

Jan - Mar 2023
- Collect feedback from the community on the beta version
- Wrap up "lesser impactful" files

Oct 31 - Nov 4 2022
CSEWG
- Validation results of ^{239}Pu and INDEN
- Status of reviews (what's lagging?)

Sep 2022
- Distribute "β_0" (VIII.0+INDEN) for testing

June 2023
- Release second beta
- First complete draft of Big Paper

Sep 2023
- FPY files w/ covariances tested

2022 2023 2024

Aug-Oct 2022
- Tests of ^{239}Pu and INDEN
- Review process of TSL

Sep. 19-23 2022
- Hackathon

Nov-Dec 2022
- Address eventual issues in 239Pu, INDEN, TSL found during CSEWG

Apr - May 2023
- Address issues found

Jan 2023
- Covariances ready for validation

Dec 2022
- Release first beta version for general testing, containing AT LEAST 239Pu + INDEN + TSL

February 2024
- Release VIII.1
- Submission of Big Paper to NDS

Release timeline

ENDF/B VIII.1

Jan - Mar 2023
- Collect feedback from the community on the beta version
- Wrap up "lesser impactful" files

Oct 31 - Nov 4 2022
CSEWG
- Validation results of ^{239}Pu and INDEN
- Status of reviews (what's lagging?)

Sep 2022
- Distribute "β_0" (VIII.0+INDEN) for testing

YOU ARE HERE

June 2023
- Release second beta
- First complete draft of Big Paper

Sep 2023
- FPY files w/ covariances tested

2022 2023 2024

Aug-Oct 2022
- Tests of ^{239}Pu and INDEN
- Review process of TSL

Sep. 19-23 2022
- Hackathon

Nov-Dec 2022
- Address eventual issues in 239Pu, INDEN, TSL found during CSEWG

Apr - May 2023
- Address issues found

Jan 2023
- Covariances ready for validation

Dec 2022
- Release first beta version for general testing, containing AT LEAST 239Pu + INDEN + TSL

February 2024
- Release VIII.1
- Submission of Big Paper to NDS

Release timeline

ENDF/B VIII.1

Jan - Mar 2023
- Collect feedback from the community on the beta version
- Wrap up "lesser impactful" files

Oct 31 - Nov 4 2022
CSEWG
- Validation results of ^{239}Pu and INDEN
- Status of reviews (what's lagging?)

June 2023
- Release second beta
- First complete draft of Big Paper

Sep 2023
- FPY files w/ covariances tested

Sep 2022
- Distribute "β_0" (VIII.0+INDEN) for testing

YOU ARE HERE

2022 2023 2024

Aug-Oct 2022
- Tests of ^{239}Pu and INDEN
- Review process of TSL

Apr - May 2023
- Address issues found

February 2024
- Release VIII.1
- Submission of Big Paper to NDS

Sep. 19-23 2022
- Hackathon

Jan 2023
- Covariances ready for validation

Nov-Dec 2022
- Address eventual issues in 239Pu, INDEN, TSL found during CSEWG

Dec 2022
- Release first beta version for general testing, containing AT LEAST 239Pu + INDEN + TSL

Release timeline

ENDF/B VIII.1

Jan - Mar 2023
- Collect feedback from the community on the beta version
- Wrap up "lesser impactful" files

Oct 31 - Nov 4 2022
CSEWG
- Validation results of ^{239}Pu and INDEN
- Status of reviews (what's lagging?)

Sep 2022
- Distribute "β_0" (VIII.0+INDEN) for testing

YOU ARE HERE

June 2023
- Release second beta
- First complete draft of Big Paper

Sep 2023
- FPY files w/ covariances tested

2022 2023 2024

Aug-Oct 2022
- Tests of ^{239}Pu and INDEN
- Review process of TSL

Sep. 19-23 2022
- Hackathon

Nov-Dec 2022
- Address eventual issues in 239Pu, INDEN, TSL found during CSEWG

Apr - May 2023
- Address issues found

Jan 2023
- Covariances ready for validation

Dec 2022
- Release first beta version for general testing, containing AT LEAST 239Pu + INDEN + TSL

February 2024
- Release VIII.1
- Submission of Big Paper to NDS

Still need to squeeze in a mini-CSEWG and another Hackathon…

Release timeline

ENDF/B VIII.1

Jan - Mar 2023
- Collect feedback from the community on the beta version
- Wrap up "lesser impactful" files

Oct 31 - Nov 4 2022
CSEWG
- Validation results of ^{239}Pu and INDEN
- Status of reviews (what's lagging?)

Sep 2022
- Distribute "β_0" (VIII.0+INDEN) for testing

YOU ARE HERE

June 2023
- Release second beta
- First complete draft of Big Paper

Sep 2023
- FPY files w/ covariances tested

2022 2023 2024

Aug-Oct 2022
- Tests of ^{239}Pu and INDEN
- Review process of TSL

Sep. 19-23 2022
- Hackathon

Nov-Dec 2022
- Address eventual issues in 239Pu, INDEN, TSL found during CSEWG

Apr - May 2023
- Address issues found

Jan 2023
- Covariances ready for validation

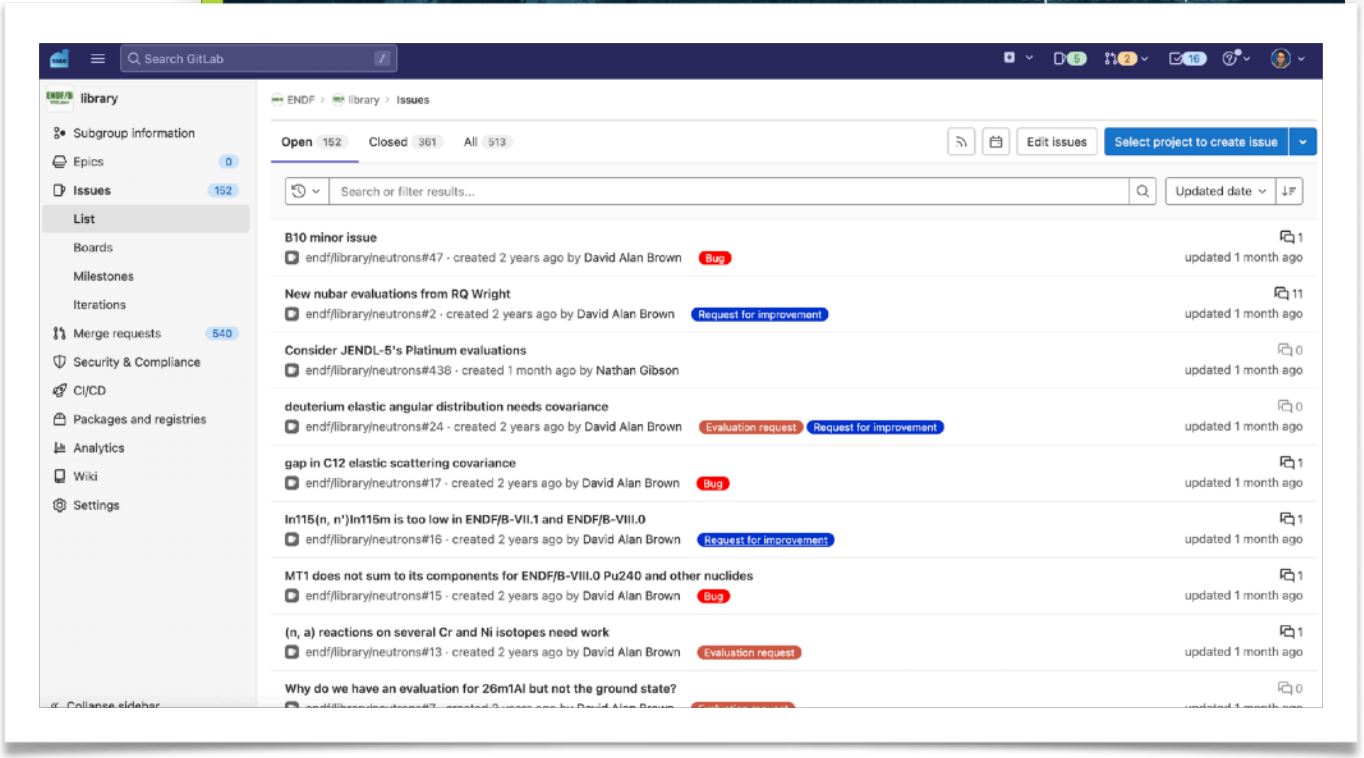
Dec 2022
- Release first beta version for general testing, containing AT LEAST 239Pu + INDEN + TSL

- Relea
- Subm

Still need to squeeze in a mini-CSEWG and another Hackathon…

2022 Hackathon

- Held on September 19-23, 2022 at BNL
- About 10 participants in person and ~5 remote
- Tracked issues resolved:
  - 22 issues for neutrons
  - 7 in decay
- More issues were found and logged
  - ~55 issues updated/opened
- Exit distributions: see Ian/Hye Young's talk
- We will plan one for 2023
  - After Beta1, before Beta2
  - Before mini-CSEWG?



Brookhaven National Laboratory

ENDF/B-VIII.1 Beta0 released (Plural!)

ENDF/B VIII.1

- Needed a Beta version for preliminary validation ahead of 2022 CSEWG
- No time yet for full review: Focused on neutron sub library and materials from INDEN collaboration (some degree of internal review)
- Indicate what should be the general trend of the VIII.1 release
- 3 “sub-releases”: Different ^{239}Pu candidates
- Planned Beta1: December 2022
  - Single Pu file
  - Fully reviewed files
  - All sub libraries
  - Substantial testing
    - CSEWG
    - Internal validation
- Beta2: tested against crits, LPS, by LANL and LLNL users, to ensure performance is as expected.

Changes* in VIII.1 Beta0 from VIII.0:

- ^{28,29,30}Si
- ^{50,52,53,54}Cr
- ^{54,56,57}Fe
- ^{63,65}Cu
- ^{233,235,238}U
- ^{239}Pu
  - Beta0a: INDEN
  - Beta0b: LANL
  - Beta0c: LLNL

Small fixes in Beta0.1:

- Release readme and change log
- ^{239}Pu MF1/MT458
  - Beta0a.1: INDEN
  - Beta0b.1: LANL

Changes* in VIII.1 Beta0.2 from VIII.0:

= Changed from Beta0.1

- ^{19}F
- ^{28,29,30}Si
- ^{50,52,53,54}Cr
- ^{54,56,57}Fe
- ^{63,65}Cu
- ^{233,235,238}U
- ^{239}Pu
  - Beta0a.2: INDEN
  - Beta0b.2: LANL (10/17)
  - Beta0c.2: LLNL

**There were additional changes done shortly after the VIII.0 release, but before the whole evaluated file repository was migrated to GitLab, which are now part of ENDF/B-VIII.1Beta0:*

- ^{10}B: ENDF/B-VIII.0 errata
- ^{156,158,160,161,162,163,164}Dy: Set of ORNL evaluations
- ^{192}Pt: Tweaked energy of first resonance
- ^{240}Pu: Fix of unitarity issue by LANL

Brookhaven National Laboratory

Review status - neutron sub library

Actinides

- ^{239}Pu
  - On-going collaborative process
  - Multiple evaluations are converging
  - CSEWG Executive Committee should decide the best course of action towards Beta1
  - Recommended evaluation in Beta1 should be better than when process started
- ^{235,238}U: waiting for reviewer assignment
- ^{233}U: new INDEN file is expected soon

Main updates for VIII.1 - neutron sub library

INDEN

- ^{235}U
- ^{239}Pu
- ^{238}U
- ^{233}U
- ^{54,56,57}Fe
- ^{28,29,30}Si
- ^{55}Mn
- ^{50,52,53,54}Cr
- ^{63,65}Cu
- ^{18}O - ^{16}O not ready for submission
- ^{10,11}B - not ready for submission
- ^{139}La

= Submitted
= Under review
= Not submitted yet
= Approved

Non-INDEN

- ^{140,142}Ce (ORNL)
- ^{103}Rh (RPI/IRSN)
- ^{86}Kr (BNL)
- ^{6}Li (LANL)
- ^{106,108,110,111,112,114,116}Cd (LANL)
- ^{6}Li, ^{234}U (LANL)
- ^{181}Ta (RPI/ORNL/ LANL)
- Fission products (RQW+BNL): ^{78}Se, ^{84}Kr, ^{85}Rb, ^{97}Mo, ^{99}Tc, ^{102}Pd, ^{109}Ag, ^{113,115}In, ^{115,119}Sn, ^{127}I, ^{122,124}Te, ^{133,134}Cs, ^{130,134,137}Ba, ^{138}La, ^{143}Pr, ^{147}Pm, ^{148,150}Nd, ^{153}Sm, ^{155}Eu, ^{160}Gd, ^{159}Tb, ^{166,168,170}Er, ^{175,176}Lu, ^{168,176}Yb, ^{174,176,177,178,179,180}Hf
- Other small fixes

Brookhaven National Laboratory

A wish on evaluation reproducibility…

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."
- "Here's a detailed, quantified, well-documented description of what was done."

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."
- "Here's a detailed, quantified, well-documented description of what was done."
- "Here's all that **AND** a portable script that does the file assembly."

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."
- "Here's a detailed, quantified, well-documented description of what was done."
- "Here's all that **AND** a portable script that does the file assembly."
- "Here's a container/virtual environment with all the evaluation and assembly codes, with corresponding inputs, which assembles the final evaluation with a single command."

A wish on evaluation reproducibility…

- Let's say I bring in a delicious cake that I baked…
- You love it, but think it would be better with more whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

Any step in this direction is a welcome one!

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."
- "Here's a detailed, quantified, well-documented description of what was done."
- "Here's all that **AND** a portable script that does the file assembly."
- "Here's a container/virtual environment with all the evaluation and assembly codes, with corresponding inputs, which assembles the final evaluation with a single command."

A wish on evaluation reproducibility…

- Let's say I bring in a delicious c
  baked…
- You love it, but think it would be
  whipped cream on top and chocolate chips in it
- I say, "No problem, here's the cake! Put it in yourself!"
- I say, "The ingredients are eggs, flour, etc…"
- "Ok, here is the recipe!"

So we don't have to reverse engineer the baking of a cake…

Any step in this direction is a welcome one!

Similarly, for reproducibility's sake and streamlining improvements, a recipe should be given on how the file was assembled.

- "Here's the file! Put your improvement in!"
- "Here's the qualitative description of what was done."
- "Here's a detailed, quantified, well-documented description of what was done."
- "Here's all that **AND** a portable script that does the file assembly."
- "Here's a container/virtual environment with all the evaluation and assembly codes, with corresponding inputs, which assembles the final evaluation with a single command."

Two general observations

- We've been seeing new evaluations
  - overwriting or ignoring existing gamma-spectra information
  - not formatting gamma spectra
- Many applications are particularly sensitive to that
- Ideally, gamma spectra should be improved consistently with the rest of the evaluation, or at least have the previous one preserved.

Two general observations

- We've been seeing new evaluations
  - overwriting or ignoring existing gamma-spectra information
  - not formatting gamma spectra
- Many applications are particularly sensitive to that
- Ideally, gamma spectra should be improved consistently with the rest of the evaluation, or at least have the previous one preserved.

- Many evaluators, for good reasons, will keep on working on their files before the next release
- Beware, though, that there may have been small fixes or corrections to your evaluation after it has been sent.
- It's worth checking the **history** of your file in phase1/phase2 branches so you don't accidentally overwrite an eventual correction or format fix, forcing it to be done again.

Pardon my recruiting…

- We need volunteer reviewers!
- Specially for the decay, fission, and **neutron**
- The number of reviewers has been the main bottleneck
- To have a major portion of submitted evaluations included in Beta1, we have to move fast

… to be a volunteer reviewer!