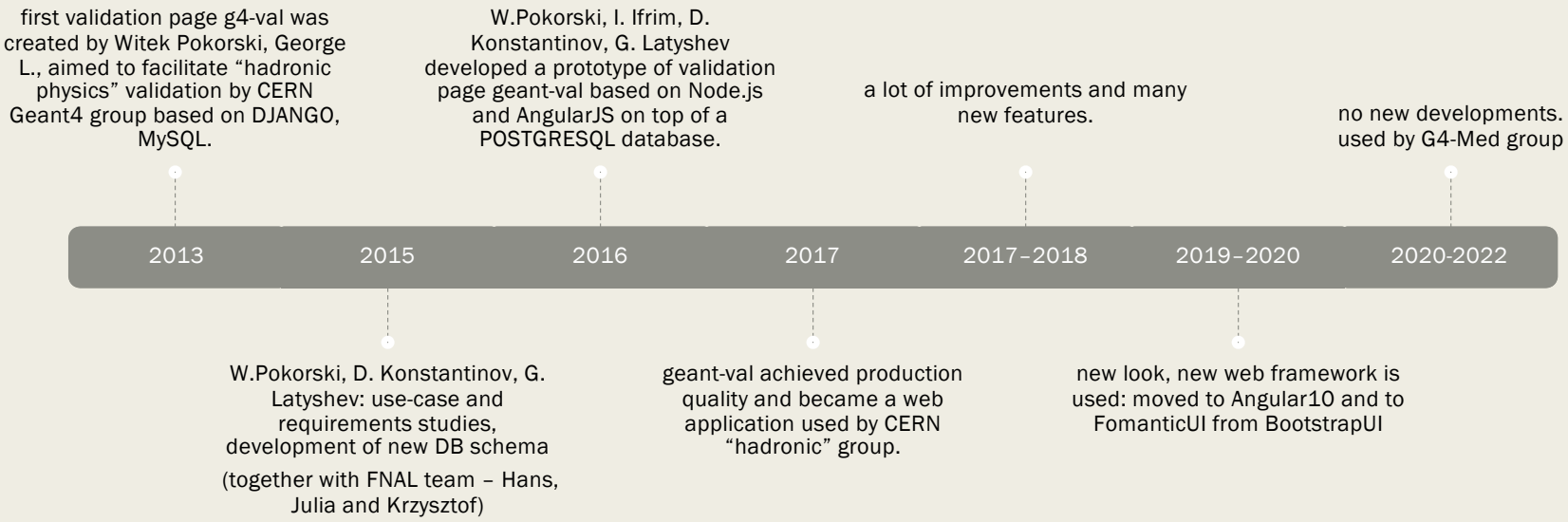
GEANT-VAL - WEB APPLICATION FOR GEANT4 VALIDATION UPDATE.

Dmitri Konstantinov

Brief history of Geant4 validation tools



Capabilities of *geant-val*

The *geant-val* is facilitating validation of Geant4 and has intuitive user interface, nice graphics

The app provides:

- consistent storage of test results
- overlaying plots and ratio plots for regression testing
- possibility for comparison with experimental data
- simple statistical evaluation for regression testing
- used by several Geant4 teams – Hadronic group and G4Med.
- geant-val IS NOT data base of experimental data:
  - *it contains such data for reference plots only.*

geant-val users

- Alberto for **“hadronic” validation** of monthly reference releases.
- Lorenzo Pezzotti integrated several physics validation simulations based on test beam simulations of different experiments (see his talk on this topic).
- **G4Med group** (Susanna), run for stable releases and for betas.
- Many EM tests (Vladimir I) of electromagnetic validation are integrated but not used, run occasionally following Mihaly’s requests.

Ideal validation workflow....

But in reality .... G4 apps are different: different ways to configure and run, different output (ascii files, ROOT files), no axis names, strange histogram names, no titles, no post-process scripts...
And instead of modifying/updating G4 applications, we wrote *geant-config-generator* *mitigating* many internal problems of G4 tests.

Geant-config-generator is Python-based utility managing user's physics tests:

- facilitates and makes more transparent the creation of test configurations from test configuration template and steering file.
- submits jobs to a batch system(HTCondor).
- parses and combines produced results.
- add missing meta information
- pass info about parameters used by G4 application to geant-val
- creates input JSON files for geant-val.

Question: was it right move to create geant-config-generator?

- yes….
  - *Quick integration of G4 apps gave a better understanding of the zoo of tests and first ideas about what we need for geant-val.*
- but now… I think it is time to slow down, sit and think of how to make our applications used for validation more uniform…

Validation Applications should:

- Be well documented and revised and kept in official (or semi-official) neat and well supported Geant4 repo.
  - *https://gitlab.cern.ch/geant4/g4tests-verification is not a good example.*
- have a script performing merging of results from multiple runs produced with different random seeds:
  - *./merge.py file1.root file2.root file3.root*
- have a script performing post-processing (analysis) run with simple syntax (to be run later inside geant-val)
- have CMakeLists.txt which installs the test executable into test directory altogether with all necessary scripts and data.
- be part of Geant4 CI – will help to keep apps up to date.

Yes, it could be boring and time-consuming….But it will improve quality and self-consistency of our application…

List of possible improvements for geant-val.

- migration from Angular 10 to Angular 14
- regular automated checks for critical vulnerabilities (for node modules and js libraries).
- more sophisticated and well thought statistical comparison of histograms.
- revision of DB schema and investigation of prospects to use NoSQL DB.
- “Admin” Web Interface for histogram deletion.
- documentation of geant-val
- new “thin target” test aggregating many tests written V. Uzhinsky were ready in February this year, but it is already not up to date with the last releases.

Summary

- Geant-val is the only web application used by Geant4 collaboration for Geant4 validation.
- There are many places for improvement in geant-val
- Also there are many places for improvements in G4 applications used for validation.

To move forward we would need to have a centrally managed cross working effort…