Parallel session IV: Hadronic Code Improvements

- Hadronic working group members undertake homework and report at the session their observations and proposals for improvements
- G.Folger: currently inside hadronics Z and A are double that is not natural and it was agreed to make a migration, which will allow to simplify and probably speedup hadronic code:
  - Start in January
  - Strategy "from top classes to bottom"
  - Add new interfaces for this migration, remove old interfaces for the major release
- T.Koi: currently base G4HadronicProcess class performs random rotation of all secondary particle directions around projectile direction;
  - hadronic models are not aware about spin of projectile and target; a secondary particle has no polarisation
  - In the case of a serious requirement a design iteration should be done in hadronic framework
  - No actions for the time being

- D.Wright: in some part of hadronic code "hand-made" non-standard methods are used, in particular:
  - Sampling of random function
  - Computation of Lorentz boost
  - Usage of units
  - Using internal particle types for a model
  - Unnecessary new/delete of objects
- These problems make some part of code hardly readable and not potentially effective
- In 2010 developers will be asked to perform code review and replace problematic peace of code by standard implementations
  - Important requirement to have corresponding tests to reduce risk of extra bugs

- V.Ivanchenko: performed review of PostStepDoIt method of the base class *G4HadronicProcess*
- There are 3 approaches for handing exceptions (18 exceptions inside this class) but cout not complete and in practice not easy to understand what happens in reality; G4HadronicWtiteBoard at each step received several strings that may be an extra overhead
  - It agreed improve situation before 9.3:
    - To use G4HadronicException
    - Do not use G4HadReenteringException
    - To examine further G4HadronicWriteBoard and be sure if it is really needed
    - Improve cout in the case of exception
- Move check on environment variable from run time methods to initialisation
  - There are 2 places in PostStepDoIt, which should be fixed for 9.3
  - Strong request to do similar cleanup for all developers
- In 2010 extra cleanup may be introduced:
  - Initialise() method can be included into G4HadronicInteraction and derived model classes to make code more transparent
  - Isotope production mechanism can be moved to LHEP models from the base class
  - Existing cross section biasing mechanism can be reviewed