BLonD meeting 2017-05-05

Guidelines for code clean-up

- Group parameters in CLASSES (not in functions)
    - See "Particle" in Beams
    - Or new "Preprocess" class
- Create separate beam objects for different "Particle" types
    - Remove secondary particle
- Declaration of parameter type:
    - For new parameter, define int/float ect explicitly
    - For derived parameter, inherit simply
- RuntimeError format
    - ERROR in ClassName: Error message
    - Break lines inside the error message with "+"
- PEP8 convention for naming
    - Classes CamelTypeName
    - Functions and parameters: lowercase, separated by underscore
        - E.g. omega_rf
    - E.g. Qs -> tune_s
    - Script to verify
    - 4 spaces instead of tabs
- If conditions with string
    - Compare with "==" rather than "is"
- Header message
    - Coding: UTF8
    - Copyright 2014-2017
- RFSectionParameters
    - phi_offset renamed to phi_rf_design; will have to be propagated
    - Removed dphi_rf_steering
    - Separate phase noise
    - Revise BeamBasedFeedback
    - Will need the Assembler to have correct tracking...
    - Alex will clean up the input check
- Documentation: now with "napoleon" extension instead of "numpydoc"
    - Follow example in GeneralParameters

TO DO:
Helga

Will push GeneralParameters and finish RFSectionParameters
Propagate naming changes to tracker
Long-term: Attack BeamBasedFeedback and Noise

Joël

Finish and push Beams
Unittests for RFSectionParameters

Alex

Input check in RFSectionParameters
Potential well calculation

Markus

Unittests for GeneralParameters

All

Assembler…

Tour de Table

Alex

"Satellite packages" around BLonD: impedance, toolbox and MD scripts…
Super-object for beams….
"Translator" classes between BLonD and packages
BLonD packages under one namespace when forks are merged

Next meeting: 2nd June in the JBA room