Protokoll vom 29. Mai 2013

**Flow Chart der Module und des projektes von (cb) liegt im rsrc-Ordner**

**(ur): symmetryOrder soll nur noch im Covering sein; zZt. Auch in PgParamGeom, PmSparseFieldGenerator,…**

**OK (kp, 2013-06-01) Allerdings in PmSparseFieldGenerator wg. Initialisierung noch belassen.**

**(pb): Angleichen der Namen in PwParameterizerBase, Modulname, im Menü, …**

**(kp): Methoden set/getStatusString aufräumen**

**OK (kp, 2013-06-01)**

**(cb): externen FieldExtender-Test-Workshop bauen nach dem Vorbild von qcTest**

**FieldExtender: splitten? Nachrangig!**

**Start() soll eigentlich nur konfigurieren, ist aber in der QC-Logik die ausführende Methode -> lassen**

**Bessere Strategie (kp, 2013-06-01):**

* **setGeometry() konfiguriert und checked schon mal notwendig instanzen variablen**
* **start(): ggfs. Weitere checks, ansonsten geht hier das Berechnen los**
* **habe diesbzgl. schon leicht überarbeitet und z.B. Checks nach Oberklasse PnModule**

**hochgezogen.**

**Alle Module sollen durchgegangen werden (Aufteilung siehe unten). Dabei sind folgende Punkte zu beachten:**

* Doc vor der Klasse und vor jeder Methode
* Elementare Formatierungen:
  + Initialisierung mit Werten nur im Constructor
  + Auskommentieren immer direkt vor dem Code (nicht am Zeilenanfang) und dafür „//“ oder „/\*…\*/“ verwenden; Kommentare begründen („funktioniert nicht“, „war ein test“, …)
  + Bei geschachtelten Schleifen und Blöcken immer Klammern verwenden!
  + Tabbing immer mit Weite 3 statt 4
  + Bei Deklarationen: nach „protected“ ein Tab, nach „private“ und „public“ zwei Tabs
  + Deklarationen mittels Tabbing bündig anordnen, dabei sollen die Gleichheitszeichen untereinander stehen!
  + Speicherallokation wenn möglich frühzeitig und zusammengefasst und nicht innerhalb einer Schleife nach und nach
  + Kommentare über oder unter einer Zeile und bündig links anstatt hinter der Zeile
  + Unnötiges Erschaffen von neuen Variablen vermeiden (z.B. anstatt subNew in einer Schleife, eine neue Variable davor einfügen und in der Schleife sub verwenden)
  + Methoden getX und setX,… immer direkt hintereinander schreiben für bessere Übersicht
  + Neuer tag „@MOVE ?“, um zu kennzeichnen, dass eine methode woanders evtl. besser aufgehoben wäre
* Zu Beginn jeden Moduls testen, ob benötigte Daten vorhanden sind
* Am Ende jeden moduls testen, ob Daten, die erstellt werden sollten erstellt worden sind
* Funktionsweise des Moduls verstehen und dokumentieren
* Einen geeigneten Namen finden und (pb) mitteilen
* Flow-Chart überprüfen und evtl. (cb) Änderungen mitteilen

**Aufteilung der Module:**

* **(cb) fertig**
  + PmFrameAngleSmoother (X)
  + PmSparseFieldEditor (X)
  + PmSparseFieldGenerator (X)
  + PmVectorAngleSmoother (X)
  + PmFieldExtender (X)
  + PwExtenderAngle (X)
  + PwExtenderParallel (X)
* **(fr)fertig**
  + PmAlignmentOptimizer(x)
  + PmCoveringGenerator (x)
  + PmConstraintsGenerator (x)
  + PwCoveringEditor (x)
  + PmCoveringEditor (x)
  + PmMeshGenerator (x)
* **(kp) [kp erledigt, 2013-06-02]**
  + PmGapEstimator
  + PmGapRounder
  + PmGraphBuilder
  + PmPositiveOrienter
* **(pb)fertig**
  + PmAnisoFrameFieldScaler
  + PmAnisoScalerByCurvature
  + PmFieldScaler
  + PmFrameFieldScaler
  + PmContinuousmaker
  + PmDegenerationDetector
* **(ur)**
  + PmPrerelaxer
  + PmRelaxAlongGradient
  + PmRelaxEnergyConformal
  + PmRelaxEnergyCurl
  + PmRelaxEnergyIf
  + PmRelaxEnergyModuleCaller
  + PmRelaxEnergySmooth