

# Code d'analyse AnaPAON4

R. Ansari

---

*6 Avril 2018*

# AnaPAON4 (1)

[GIT@GITLAB.IN2P3.FR:BAORADIO/ANAPAON4.GIT](https://gitlab.in2p3.fr/baoradio/anapaon4.git)

- \* Code de lecture et traitement des fichiers de visibilité
- \* Répertoire **Satellites** : Calcul de position / vitesses des satellites (C. Magneville)
- \* Répertoire **AnaFringes** : Ajustement (avec Minuit) des franges (J.E.Campagne)
  - \* Calibration en phase et géométrie des antennes  $\neg$
- \* Répertoire **Script**
- \* Classes utilitaires dans p4utils.h .cc :
  - \* P4FreqBand , P4VisiNumEncoder , P4AnaParams , P4Coords , P4RAMapUtil
- \* Classes de lecture des visibilité (visip4reader.h .cc) :
  - \* VisiP4ReaderBase , VisiP4ReaderNoDB , VisiP4ReaderDB
- \* Classes de correction de gain  $g(\nu)$  ( p4gnugain.h .cc ) et  $G(t)$  ( p4gvcor.h .cc )
  - \* P4gnuGain , P4gvCor

# AnaPAON4 (2)

[GIT@GITLAB.IN2P3.FR:BAORADIO/ANAPAON4.GIT](https://gitlab.in2p3.fr/baoradio/anapaon4.git)

- \* Classes de lecture des visibilitées (`visip4reader.h .cc`):
  - \* `VisiP4ReaderBase` , `VisiP4ReaderNoDB` , `VisiP4ReaderDB`
- \* Classes de correction de phase (nu) ( `p4phcor.h .cc` )
  - \* `P4PhaseCor`
- \* Classe de lecture de plus haut niveau, avec fenêtre en temps, avec la possibilité de prise en charge des classes de correction de gain et de phase ( `visp4winreader.h .cc` )
  - \* `VisiP4WindowReader`
- \* Quelques programmes de lectures permettant de faire des cartes temps-fréquence (ou ra-fréquence)
  - \* `visiavg.cc` `visi2ra.cc` `visi2dtacx.cc` `visi2nt.cc`

# JSkyMap

[GIT@GITLAB.IN2P3.FR:SCOSMOTOOLS/JSKYMAP.GIT](https://gitlab.in2p3.fr:SCosmoTools/JSkyMap.git)

- \* Code de simulation de visibilité à partir de cartes et reconstruction des cartes
- \* Travail de thèse de J. Zhang
- \* Codes [vis2map.cc](#) , [p4src2vis.cc](#) , [map2vis.cc](#)
- \* Fonction de reconstruction de cartes en ra par combinaison linéaire (phase-array) des visibilités ds [p4src2vis.cc](#)

```
TArray<double> make_map_from_visibilities(vector< TArray<  
complex<double> > > & vec_visarr, bool useauto=false);
```

- \* Calcul de visibilités pour PAON4 à partir d'une carte sphérique ou une liste de sources

# JSkyMap : Map making software (II)

(J. Zhang PhD)

- Do not yet handle polarisation, but extension is rather easy
- Except for the computation of polarised beam responses...
- The code is rather simple, built around few classes, but relies on the SOPHYA class library (<http://www.sophya.org>)
- Main classes used in JSkyMap :
  - BeamTP , BeamLM and BeamVis
  - SphCoordTrans , PseudoInverse<T>
  - JSphSkyMap
  - JSkyMap and BeamUV for planar geometry
  - Some utility functions

GIT repo: <https://gitlab.in2p3.fr/SCosmoTools/JSkyMap>