

# Utilitaire de traitement de photo astronomiques

---

## Cahier des clauses techniques particulières (CCTP)

---

### Généralités

---

L'astrophotographie est une discipline spécifique de la photographie qui requiert une certaine technicité. Notamment, la photographie des objets du ciel profond nécessite des techniques particulières du fait que ces objets - nébuleuses, amas, galaxies, etc. - sont la plupart du temps invisibles à l'oeil nu voire difficilement observable à l'aide d'un télescope.

Les techniques d'astrophotographie consistent de manière générale à prendre un grand nombre de clichés avec de longs temps de pause puis à les "empiler" intelligemment pour révéler l'objet visé. Une présentation succincte du procédé est donnée en annexe2 de ce document.

Le format de fichier images le plus utilisé est *FITS (Flexible Image Transport System)*, spécialement conçu pour ce type d'applications. Une description est fournie en annexe 1 et la documentation de référence est fournie par ailleurs.

Le présent document décrit les fonctionnalités et contraintes pour le développement d'un utilitaire permettant la lecture et le traitement simple des fichiers images bruts.

### Évaluation

Vous serez évalués par groupe et individuellement.

**Par groupe** sur le respect des exigences :

- les exigences primordiales sont notées sur 16 points au total. Cela revient à attribuer 2 points par exigence hormis les deux premières qui sont globales à tout le projet (*PRIM\_00* et *PRIM\_05*) ;
- les exigences souhaitables (*OPT\_xx*) sont notées sur 4 points au total, soit 1 point par exigence.

**Attention** les exigences souhaitables ne seront prises en considération que si les primordiales sont respectées. En d'autres termes, faire une super IHM ne compte que si toutes les exigences *PRIM\_xx* sont respectées.

**Par groupe** sur la qualité des livrables et la présentation.

**Individuellement** : à l'issue de la présentation de votre groupe, il vous sera demandé de rédiger en 15 minutes et sur papier un code en relation avec les développements dont vous avez été responsable dans le groupe et de l'expliquer.

## terminologie

Les termes suivants seront employés dans la suite du document :

- *commanditaire* : désigne l'établissement émettant la présente spécification (en la circonstance : L'ISEN). C'est en particulier le commanditaire qui réceptionnera le produit et évaluera la prestation.
- *dark* : fichier image de calibration du bruit lumineux.
- *empilement* : opération au choix de sommer ou moyenner un ensemble de fichiers, pixel par pixel. On obtient un fichier *master*.
- *flat* : fichier image de calibration du maximum de lumière.
- *fournitures* : éléments fournis par le commanditaire.
- *light* : fichier de prise de vue de l'objet à imager.
- *livrable* : élément devant être fourni par le titulaire à l'exécution du contrat (*i.e. le rendu de votre projet*) : documents, présentation, essais, executables, etc.
- *l'outil* : désigne l'ensemble des executables nécessaires répondant au présentes spécifications.
- *master* : fichier image résultant de l'empilement de fichiers sources. l'image master a les mêmes dimensions que ses sources et chacun de ses pixels est l'empilement des pixels correspondant dans les fichiers sources.
- *offset* : fichier image de calibration du bruit électronique.
- *titulaire* : groupe constitué d'étudiants chargé de développer l'outil.

## Spécification technique de besoin (STB)

---

### Exigences Primordiales

Les exigences de ce paragraphe sont obligatoires.

- **PRIM\_00** : tous les développements se feront en langage C.
- **PRIM\_05** : l'outil doit permettre d'effectuer les opérations *PRIM\_30* à *PRIM\_80* de manière séparée. ex: réaliser la moyenne de tous les fichiers du dossier *Dark*
- **PRIM\_10** : l'outil doit travailler avec des fichiers *FITS* provenant de différentes sources, avec des formats de capteurs différents. Notamment, les champs *BITPIX*, *NAXIS1*,

*NAXIS2*, *BZERO* et *BSCALE* sont susceptible de varier.

- **PRIM\_20** : l'outil doit générer ses résultats au format *FITS*
- **PRIM\_30** : l'outil doit lire et afficher les entêtes des fichiers *FITS*
- **PRIM\_40** : l'outil doit convertir les données (hors entête) du format *FITS* au format *csv* avec séparateur ';' (point-virgule)
- **PRIM\_50** : Sur la base d'un ensemble de fichiers sources *FITS*, l'outil doit générer un fichier cible *FITS* dont les données sont la **somme**, pixel par pixel, des fichiers sources. Pour chaque couple (x,y), le pixel (x,y) du fichier cible doit correspondre à la somme des pixels (x,y) de tous les fichiers sources.
- **PRIM\_60** : Sur la base d'un ensemble de fichiers sources *FITS*, l'outil doit générer un fichier cible *FITS* dont les données sont la **moyenne**, pixel par pixel, des fichiers sources. Pour chaque couple (x,y), le pixel (x,y) du fichier cible doit correspondre à la moyenne des pixels (x,y) de tous les fichiers sources.
- **PRIM\_70** : l'outil doit générer un fichier cible *FITS* dont les données sont la **différence**, pixel par pixel, entre deux fichiers sources. Pour chaque couple (x,y) le pixel (x,y) du fichier cible doit correspondre à la différence entre le pixel (x,y) du premier fichier avec le pixel (x, y) du second fichier
- **PRIM\_80** : l'outil doit générer un fichier cible *FITS* dont les données sont la **division**, pixel par pixel, entre deux fichiers sources. Pour chaque couple (x,y) le pixel (x,y) du fichier cible doit correspondre à la division entre le pixel (x,y) du premier fichier avec le pixel (x, y) du second fichier

## Exigences souhaitables (non obligatoires)

Les exigences qui suivent sont complémentaires aux précédentes. Elles ne sauraient les remplacer.

- **OPT\_10** : l'outil doit permettre d'effectuer en une seule commande l'ensemble du processus de génération de l'image finale dans cet ordre (cf. précisions sur le traitement en annexe 2):
  - i. génération du *master\_dark* correspondant à l'opération d'empilement sur tous les fichiers dans le dossier *Dark*
  - ii. génération du *master\_flat* correspondant à l'opération d'empilement sur tous les fichiers dans le dossier *Flat*
  - iii. génération du *master\_offset* correspondant à l'opération d'empilement sur tous les fichiers dans le dossier *Offset*
  - iv. prétraitement de chaque fichier *light* (cf. annexe2)
  - v. empilement des fichiers prétraités à l'étape précédente.
- **OPT\_20** : les opérations d'empilement doivent pouvoir être choisies par l'utilisateur indépendamment pour chaque étape. ex. le *master\_dark* et l'empilement final par

sommation et les autres masters par moyennage.

- **OPT\_30** : l'outil doit être pilotable via une interface interactive
- **OPT\_40** : l'outil doit convertir les fichiers *FITS* au format *png*.

## Composition détaillée des prestations et fournitures (CDPF)

---

### Livrables

Le titulaire fournira à la fin du projet les livrables suivants :

- **L1** : dossier de conception présentant les choix d'architecture logicielle, les différents modules et leurs dépendances. Ce document fera office de mode d'emploi. Le document aura une taille maximale de 5 pages.
- **L2** : matrice de conformité aux exigences présentant sur chaque ligne les exigences et sur les colonnes les modules. la matrice décrira dans ses cases par quel module et comment l'exigence est remplie (ex. nom d'une fonction ou d'une commande).
- **L3** : l'ensemble du code et le makefile associé permettant sa compilation. Le commanditaire se réserve le droit de faire un test de compilation en séance.
- **L4** : une présentation des livrables *L1* et *L2* incluant une démonstration du fonctionnement de l'applicatif. L'ensemble (présentation et démonstration) durera 15 à 20 minutes.

### Fournitures

Au lancement du projet, le commanditaire fournit :

- **FC1** : le présent CCTP.
- **FC2** : le document de définition du format FITS (FITS Standard - V3.0 - 10 juillet 2008).
- **FC3** : une archive contenant un ensemble de prises de vue au format FITS et suffisant pour la production d'une image "astro" du grand amas d'Hercule (Messier 13 ou NGC6205). (*Crédit Photo : Pierre SAGNIÉRES - club astronomique Véga - Ollioules*).
- **FC3** : une description synthétique du format FITS et réduite au cadre du présent utilitaire.(cf. Annexe 1).
- **FC4** : un utilitaire d'affichage du fichier CSV brut. (language python). exemple d'utilisation : `./AfficheCSV.py M13_001.csv` .

## Annexes

---

## Annexe 1 - Description brève du format *FITS*

Les paragraphes qui suivent présentent succinctement le format FITS. Pour plus de détails, se référer à la documentation officielle fournie (*FC2*).

### Outil

Pour vous aider dans vos développements, **les fichiers peuvent être inspectés avec l'outil `od`**. par exemple, `od --endian=big -w20 -N 2880 -A d -t x2z -v r_lights_00015.fit` affiche les 2880 premiers octets du fichier `r_lights_00015.fit` avec 20 octets par ligne. Les lignes ressemblent à cela (ex sur deux lignes) :

```
0000080 4249 5450 4958 2020 3d20 2020 2020 2020 2020 2020  >BITPIX  =
0000100 2020 2020 2020 2020 3136 202f 206e 756d 6265 7220  >          16 / number
```

- première colonne : l'adresse du premier octet en décimales (ici 80 et 100).
- colonnes 2 à 11 : les valeurs des 20 octets à partir de l'adresse indiquée. Affichage en hexadécimal et regroupé par couple d'octets. ex, sur la ligne 1, `5450` correspond à la valeur hexa des octets 82 et 83.
- à droite `od` écrit le caractère ASCII de l'octet correspondant (s'il est imprimable). Dans l'exemple, `4249 5450 4958` donne le mot `BITPIX` pour les caractères 42 (B), 49 (I), 54 (T), etc.

Pour plus d'informations sur la commande `od` vous vous réfèrerez au manuel : `man od`.

### *FITS structures et FITS blocks*

- Une **structure** est composée d'un nombre entier de *blocks*. Il y a trois types de structures possibles mais les fichiers que nous manipulons n'en ont qu'une : **Primary Header and Data Unit (HDU)**.
- la longueur des *structures* (i.e. le nombre de *blocks*) n'est pas limité.
- le **block** est l'unité de base. Sa taille est fixe : 2880 octets. Il y a deux types de blocks : Header et Data, détaillés ci-après.
- Il n'y a **que des blocks entiers**. Au besoin, les blocks sont complétés par des zéros. Par exemple, pour écrire un block Header qui ne contient qu'un seul enregistrement de 80 caractères, on écrira tout un bloc de 2880 octets dont les 2800 derniers caractères sont à 0. De la même manière, il faut 2 blocks (5760 octets) pour écrire 2881 octets de données : un bloc complet et un bloc complété de 2799 octets à 0.

### Header du fichier FITS (*primary header*)

## Description du Header

Cet entête est composé d'un bloc (2880 octets). Son contenu est en ASCII restreint aux caractères imprimables. Il est constitué de 36 enregistrements de 80 caractères. Chaque enregistrement commence par un mot clef sur 8 octets, suivi de "=" (sur les octets 9 et 10), suivi de la valeur du paramètre sur les octets 11 à 80. Cette valeur peut être suivie d'un commentaire introduit par un '/'. Le commentaire est inclus dans les octets 11 à 80, après la valeur et n'est pas pris en compte dans la valeur.

Ex : "NAXIS = 3 / Dimensionality" signifie que la variable NAXIS vaut 3. Le commentaire rappelle que NAXIS est le nombre de dimensions (c'est à dire d'axes) dans les données.

## NOTES :

- les doubles côtes ont été rajoutées ici pour délimiter l'extention de l'enregistrement (80 caractères). elles ne sont pas incluses dans le fichier FITS.
- les valeurs sont aussi en ASCII. Ainsi la valeur 32 est codée sur deux octets par les caractères '3' et '2' et non un octet par la valeur binaire correspondante (0x20).

## Exemple commenté

- "SIMPLE = T / C# FITS: 03/21/2021 00:23:54" : premier enregistrement du header. Indique que le fichier se conforme à la norme FITS
- "BITPIX = 16" : longueur en bits de la donnée. ici 16 bits, c'est à dire 2 octets.
- "NAXIS = 3 / Dimensionality" : nombre d'axes dans les données. Ici 3, c'est à dire horizontal et vertical, et couche RGB. Pour information il peut y avoir jusqu'à 999 dimensions.
- "NAXIS1 = 4144" : nombre de données sur le premier axe : 4144 colonnes
- "NAXIS2 = 2822" : nombre de données sur le second axe : 2822 lignes
- "NAXIS3 = 3" : nombre de données sur le troisième axe : 3 couches, une par couleur.
- "GAIN = 260 / " : le gain d'emplification, appliqué par le système. Inutile pour la présente application.
- "DATE-OBS= '2021-03-20T23:23:37.3047052' / System Clock:Est. Frame Start" : la date à laquelle le fichier a été créé. Inutile pour la présente application.

( champs inutiles pour la présente application ... )

Bayer. Nécessaire pour \*SOUH\_10\*.

- "BAYERPAT= 'GBRG ' / NOTE: Use RGGG on some software (eg PixInsight)" : motif de la matrice de Bayer. Inutile pour la présente application

- “ COLORTYP= 'GBRG ' / NOTE: Use RGGG on some software (eg PixInsight) ” : Inutile pour la présente application
- “ CCD-TEMP= -23.2 / ” : température du capteur. En astronomie, on utilise souvent des capteurs refroidis pour éviter le bruit thermique. Inutile pour la présente application

( champs inutiles pour la présente application ... )

Les deux enregistrements suivants sont utilisés conjointement pour calculer la valeur physique réelle à partir de la valeur stockée dans les data :  $\text{valeur\_réelle} = \text{BZERO} + \text{BSCALE} \times \text{valeur\_enregistrée}$ , avec :

- *valeur\_réelle* : la valeur physique réellement mesurée (ici une charge électrique correspondant à l'illumination). C'est une valeur positive.
- *valeur\_enregistrée* : la grandeur stockée dans les data.

Dans notre cas, les valeurs sont stockées en entiers signés sur 2 octets ( $\text{BITPIX} = 16$ ) or la valeur physique est une valeur de luminance positive. Il faut donc rajouter 32768 pour obtenir la valeur physique réelle.

- “ BSCALE = 1 / ” : le facteur d'échelle pour calculer la valeur physique réelle à partir de la valeur stockée.
- “ BZERO = 32768 / ” : les facteur de décallage
- “ EXTEND = T / Extensions are permitted ” : Inutile pour la présente application
- “ INSTRUME= 'ZWO ASI294MC Pro' / ”
- “ END ” : ce mot clef est en fin de primary header. la suite du block n'est pas prise en compte.

## Primary Data Unit

Cette section contient les données proprement dites. Elle suit le header qui lui correspond. Les données y sont stockées à la suite les unes des autres, en binaire, suivant les indications du header :

- par mots de 16 bits ou 2 octets ( $\text{BITPIX} = 16$ ). La convention est **Big-endian**.
- l'index de l'axe 1 varie avant celui de l'axe 2 etc. ex :
  - les 4144 ( $\text{NAXIS1} = 4144$ ) premiers mots de 2 octets correspondent à la première ligne de la première couche,
  - puis les 4144 suivants à la deuxième ligne de la première couche, etc.
  - le 1er octet de la deuxième couche est à la 11694369ème position ( $4144 \times 2822 + 1$ )
- Il y a toujours un nombre entier de blocks. Au besoin on complète par des 0.
- la section données comporte donc 70168320 octets :



- $4144 * 2822 \text{ pixels} * 3 \text{ couleurs} * 2 \text{ octets par valeurs}$ . Cela ne donne pas un nombre entier de blocs. Le dernier est partiellement rempli et complété de 768 octets à 0.
- cela donne 24364 blocs.

## Annexe 2 - Démarche générale d'astrophotographie

---

Les grandes lignes de l'astrophotographie sont présentées ici pour aider à la compréhension du besoin.

Le principal problème des objets du ciel profond est leur très faible luminosité. Une simple amplification ne suffit pas car elle amplifie aussi les autres sources de lumière, parasites, que l'on qualifie de "bruit" (pollution lumineuse par l'éclairage public, passage de satellites et avions, luminosité de la lune ...).

Le principe est de faire des poses très longues (de plusieurs secondes) et d'accumuler le plus de lumière possible. Au-delà des problématiques de bougé, il faudrait des poses de plusieurs heures. On "contourne" le problème en accumulant plusieurs poses successives sur le même sujet. Il reste alors à éliminer le bruit lumineux et les défauts du capteurs. On réalise alors 4 séries de photos :

- les *light* : clichés de l'objet à photographier. Il y en a plusieurs dizaines. Dans l'exemple 50 clichés de 16 secondes (soit l'équivalent de 13 minutes de pose).
- les clichés de calibration : *darks*, *flats* et *offsets* : série de clichés complémentaires pour éliminer les défauts. Il y a aussi plusieurs prises du même cliché.

Le traitement consiste à "empiler" les clichés de la façon suivante :

1. empiler les fichiers de calibration, chacun dans sa catégorie, pour obtenir une image *master* (ex. *master dark*).
2. prétraiter chaque fichier *light* :  $\text{ImageBrutPretraitee} = (\text{imageBrut} - \text{masterDark}) / (\text{masterFlat} - \text{masterOffset})$
3. empiler les ImagesBrutPretaitées pour obtenir le résultat final.

L'empilement peut se faire de plusieurs façons au choix du photographe : moyenne, moyenne pondérée, addition ... .