

Extensions FITS et règles de gestion des acquisitions et traitements des séquences d'images en astrophotographie

<https://github.com/euldulle/seqfits>
fm@dulle.fr

3 novembre 2018

Résumé et motivations

(Ce document est principalement un document de travail orienté développement ; pour l'observateur, retenir que le projet vise à rendre possible le pilotage des acquisitions et l'automatisation des traitements à partir d'un minimum d'informations -voir les 2 lignes "observateur" du Makefile).

Pour un certain nombre de raisons, l'objet de base manipulé en astrophotographie amateur n'est plus une image mais une séquence d'images. C'est valable autant pour les images de calibration que pour les images "signal" et cela impacte les processus d'acquisition comme ceux de traitement.

L'approche naturelle qui prévaut est celle consistant à coder dans le nom des fichiers manipulés les informations nécessaires au traitement (type d'image, numéro dans la séquence...) Cette approche souffre d'une part d'un manque de cadre systématique (2 logiciels d'acquisition différents, ou 2 observateurs différents vont créer des fichiers aux noms différents pour le même objet), et de la fragilité intrinsèque d'une telle approche (quand le simple renommage d'un fichier suffit à le rendre inexploitable, il est légitime de penser qu'il y a un problème conceptuel).

Dans cette approche la construction d'une image avec 4 ou 5 bandes de fréquence, et potentiellement 1, 2 ou 3 temps de pose différents est un casse-tête organisationnel, si ce n'est au moment de l'acquisition, en tous cas au moment du traitement : l'observateur se retrouve à devoir faire à la main des opérations (renommage, déplacement de fichiers, création de répertoire) fastidieuses et sans aucun intérêt. C'est d'autant plus paradoxal qu'on a sous la main des puissances de calcul... disons astronomiques.

Ce document propose une approche systématique de la construction d'une image finale (le projet) où toutes les informations pertinentes à la réalisation du projet sont encodées dans les fichiers FITS au moment de l'acquisition (ou de la conversion en fits pour les acquisitions APN).

L'objectif est de déterminer le plus petit ensemble d'informations que doit fournir l'observateur pour que, moyennant quelques règles communes, logiciels d'acquisition et de traitement puissent faire leur travail sans que l'observateur ait à intervenir sur les détails avant la production des images finales.

Le premier élément de la proposition est un Makefile dont l'objectif est la construction d'une liste de tâches (de 2 types, acquisition ou traitement) à accomplir, et d'accompagner chacune de ces tâches des paramètres utiles à la traçabilité des images produites à chaque étape. Le Makefile proposé est générique, l'input observateur se limite au nom du projet et aux composantes de l'image.

La traçabilité des images s'appuie sur des mots-clés (KEYWORDS) du standard FITS ; 3 sont déjà largement utilisés (IMAGETYP, FREQ, EXPTIME), et le quatrième, SEQNUM, doit permettre de gérer l'aspect séquence. On peut aussi introduire un mot-clé PROJECT, mais il n'est pas strictement nécessaire.

Le deuxième élément c'est la liste des tâches (ACQ et PROC) à effectuer pour mener le projet à bien. Dans ce document les tâches sont décrites de manière symbolique, l'essentiel étant la présence des paramètres permettant d'assurer la traçabilité.

Le troisième élément c'est les règles à suivre par l'acquisition et le traitement pour que le premier fasse ce qu'il faut pour

que le deuxième puisse retrouver ses petits sans rien demander à personne.

1 Exemple symbolique, Makefile, génération des tâches

Donc voilà le Makefile, suivi de la liste des tâches générées quand on mouline ce Makefile par un simple make.

```
#####
#
# Section définie par l'utilisateur : nom du projet et composantes
#
# contraintes (amendables) d'implémentation choisies pour ce premier jet:
#
#     chaque composante est décrite par un mot composé d'une lettre
#     définissant la fréquence, et d'un nombre définissant le
#     temps d'exposition en secondes. Les règles du Makefile utilise
#     ces contraintes donc si on les modifie (si on veut plus de caracteres
#     pour FREQ par exemple), il faut modifier les recettes du Makefile
#     en conséquence. L = NOFILTER
#
#                                     R = RED
#                                     G = GREEN
#                                     B = BLUE
#                                     O = OIII
#                                     H = Halpha
#                                     A = Aucune idée
#                                     ...
#
#
project=projectname
components=L300 L120 L30 R60 G60 B60 A60 H120 K60
#
# Fin de la section observateur
#####
# Section générique : elle traduit une
# stratégie d'acquisition/traitement classique en tâches ACQ et PROCESS.
#
# On crée 3 fichiers séparés, pour la facilité de lecture :
# le fichier .tasks contient toutes les tâches ACQ et PROCESS, dans l'ordre
# où elles sont générées par le Makefile, donc c'est un ordre consistant :
# les tâches PROCESS apparaissent apres que les acquisitions nécessaires
# aient été programmées.
#
# le fichier projectname.acq ne contient que les tâches d'acquisition
#
# le fichier projectname.process ne contient que le tâches de traitement,
# c'est un squelette symbolique du script final de traitement.
#
.SILENT:project
SHELL := /bin/bash
explist := $(shell for i in $(components); do echo $$i:1; done | sort | uniq)
freqlist := $(shell for i in $(components); do echo $$i:0:1; done | sort | uniq)
darklist := $(shell for i in $(explist); do echo Master_Dark_$$i; done)
lightlist := $(shell for i in $(components); do echo Master_Light_$$i; done)
flatlist := $(shell for i in $(freqlist); do echo Master_Flat_$$i; done)
```

```

biaslist := Master_Bias
acq := $(project).acq
process := $(project).process
tasks := $(project).tasks
rm := $(shell rm $(acq) $(process) $(tasks) 2>/dev/null)

project: $(biaslist) $(darklist) $(flatlist) $(lightlist)
    @echo NEWPROCESS composite ${lightlist} \
        |tee -a $(process)|tee -a $(tasks)

Master_Bias: Seq_Bias
    @echo NEWPROCESS makemaster IMAGETYP="Bias" \
        |tee -a $(process)|tee -a $(tasks)

Seq_Bias:
    @echo NEWACQ Seq IMAGETYP='Bias' \
        |tee -a $(acq)|tee -a $(tasks)

Master_Dark_%: Seq_Dark_%
    @echo NEWPROCESS makemaster IMAGETYP="Dark" EXPTIME=$* \
        |tee -a $(process)|tee -a $(tasks)

Seq_Dark_%:
    @echo NEWACQ Seq IMAGETYP='Dark' EXPTIME=$* \
        |tee -a $(acq)|tee -a $(tasks)

Master_Flat_%: Seq_Flat_%
    @echo NEWPROCESS makemaster IMAGETYP="Flat" FREQ=$* \
        |tee -a $(process)|tee -a $(tasks)

Seq_Flat_%:
    @echo NEWACQ Seq IMAGETYP='Flat' FREQ=$* \
        |tee -a $(acq)|tee -a $(tasks)

Master_Light_%: freq = $(shell i=$(*) ; echo ${i:0:1})
Master_Light_%: exp = $(shell i=$(*) ; echo ${i:1})
Master_Light_%: Seq_Light_%
    @echo NEWPROCESS makemaster IMAGETYP="Light" FREQ=$(freq) EXPTIME=$(exp) \
        |tee -a $(process)|tee -a $(tasks)

Seq_Light_%: freq = $(shell i=$(*) ; echo ${i:0:1})
Seq_Light_%: exp = $(shell i=$(*) ; echo ${i:1})
Seq_Light_%:
    @echo NEWACQ Seq IMAGETYP='Light' EXPTIME=$(exp) FREQ=$(freq) \
        |tee -a $(acq) |tee -a $(tasks)

```

— Le fichier `projectname.tasks` (toutes les tâches dans l'ordre de création par le Makefile, donc dans l'ordre où il rencontre les contraintes de construction) :

```

NEWACQ Seq IMAGETYP=Bias
NEWPROCESS makemaster IMAGETYP=Bias
NEWACQ Seq IMAGETYP=Dark EXPTIME=120
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=120
NEWACQ Seq IMAGETYP=Dark EXPTIME=30
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=30
NEWACQ Seq IMAGETYP=Dark EXPTIME=300

```

```

NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=300
NEWACQ Seq IMAGETYP=Dark EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=60
NEWACQ Seq IMAGETYP=Flat FREQ=A
NEWPROCESS makemaster IMAGETYP=Flat FREQ=A
NEWACQ Seq IMAGETYP=Flat FREQ=B
NEWPROCESS makemaster IMAGETYP=Flat FREQ=B
NEWACQ Seq IMAGETYP=Flat FREQ=G
NEWPROCESS makemaster IMAGETYP=Flat FREQ=G
NEWACQ Seq IMAGETYP=Flat FREQ=H
NEWPROCESS makemaster IMAGETYP=Flat FREQ=H
NEWACQ Seq IMAGETYP=Flat FREQ=K
NEWPROCESS makemaster IMAGETYP=Flat FREQ=K
NEWACQ Seq IMAGETYP=Flat FREQ=L
NEWPROCESS makemaster IMAGETYP=Flat FREQ=L
NEWACQ Seq IMAGETYP=Flat FREQ=R
NEWPROCESS makemaster IMAGETYP=Flat FREQ=R
NEWACQ Seq IMAGETYP=Light EXPTIME=300 FREQ=L
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=300
NEWACQ Seq IMAGETYP=Light EXPTIME=120 FREQ=L
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=120
NEWACQ Seq IMAGETYP=Light EXPTIME=30 FREQ=L
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=30
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=R
NEWPROCESS makemaster IMAGETYP=Light FREQ=R EXPTIME=60
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=G
NEWPROCESS makemaster IMAGETYP=Light FREQ=G EXPTIME=60
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=B
NEWPROCESS makemaster IMAGETYP=Light FREQ=B EXPTIME=60
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=A
NEWPROCESS makemaster IMAGETYP=Light FREQ=A EXPTIME=60
NEWACQ Seq IMAGETYP=Light EXPTIME=120 FREQ=H
NEWPROCESS makemaster IMAGETYP=Light FREQ=H EXPTIME=120
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=K
NEWPROCESS makemaster IMAGETYP=Light FREQ=K EXPTIME=60
NEWPROCESS composite Master_Light_L300 Master_Light_L120 Master_Light_L30 \
    Master_Light_R60 Master_Light_G60 Master_Light_B60 \
    Master_Light_A60 Master_Light_H120 Master_Light_K60

```

— Le fichier projectname.acq (uniquement les tâches acquisition) :

```

NEWACQ Seq IMAGETYP=Bias
NEWACQ Seq IMAGETYP=Dark EXPTIME=120
NEWACQ Seq IMAGETYP=Dark EXPTIME=30
NEWACQ Seq IMAGETYP=Dark EXPTIME=300
NEWACQ Seq IMAGETYP=Dark EXPTIME=60
NEWACQ Seq IMAGETYP=Flat FREQ=A
NEWACQ Seq IMAGETYP=Flat FREQ=B
NEWACQ Seq IMAGETYP=Flat FREQ=G
NEWACQ Seq IMAGETYP=Flat FREQ=H
NEWACQ Seq IMAGETYP=Flat FREQ=K
NEWACQ Seq IMAGETYP=Flat FREQ=L
NEWACQ Seq IMAGETYP=Flat FREQ=R
NEWACQ Seq IMAGETYP=Light EXPTIME=300 FREQ=L
NEWACQ Seq IMAGETYP=Light EXPTIME=120 FREQ=L

```

```

NEWACQ Seq IMAGETYP=Light EXPTIME=30 FREQ=L
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=R
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=G
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=B
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=A
NEWACQ Seq IMAGETYP=Light EXPTIME=120 FREQ=H
NEWACQ Seq IMAGETYP=Light EXPTIME=60 FREQ=K

```

— Le fichier `projectname.process` (uniquement les tâches traitement) :

```

NEWPROCESS makemaster IMAGETYP=Bias
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=120
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=30
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=300
NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Flat FREQ=A
NEWPROCESS makemaster IMAGETYP=Flat FREQ=B
NEWPROCESS makemaster IMAGETYP=Flat FREQ=G
NEWPROCESS makemaster IMAGETYP=Flat FREQ=H
NEWPROCESS makemaster IMAGETYP=Flat FREQ=K
NEWPROCESS makemaster IMAGETYP=Flat FREQ=L
NEWPROCESS makemaster IMAGETYP=Flat FREQ=R
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=300
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=120
NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=30
NEWPROCESS makemaster IMAGETYP=Light FREQ=R EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Light FREQ=G EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Light FREQ=B EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Light FREQ=A EXPTIME=60
NEWPROCESS makemaster IMAGETYP=Light FREQ=H EXPTIME=120
NEWPROCESS makemaster IMAGETYP=Light FREQ=K EXPTIME=60
NEWPROCESS composite Master_Light_L300 Master_Light_L120 Master_Light_L30 \
    Master_Light_R60 Master_Light_G60 Master_Light_B60 \
    Master_Light_A60 Master_Light_H120 Master_Light_K60

```

2 Traduction des contraintes pour les logiciels acquisition et traitement

On prend la liste des tâches et on traduit ce que doit faire chaque logiciel.

2.1 Acquisition

On prend les tâches une à une et on voit ce qu'on doit en faire :

— NEWACQ Seq IMAGETYP=Bias

Le logiciel d'acquisition propose cette tâche à l'observateur, l'observateur règle le temps d'exposition et le nombre d'images ; à l'exécution, chaque image est enregistrée avec les mots-clés FITS :

IMAGETYP='Bias' SEQNUM = numéro de l'image dans la séquence

peut-être, si c'est utile (ça peut l'être pour des raisons d'efficacité) quelque chose comme :

FREQ = ALL

et optionnellement :

PROJECT = 'projectname'

Dans la suite on ne rementionnera pas la possibilité d'inclure ce champ, mais si on choisit a priori de l'implémenter,

il faut le mettre dans toutes les images.

— NEWACQ Seq IMAGETYP=Dark EXPTIME=120

Champs à inclure :

1. IMAGETYPE='Dark'
2. EXPTIME=120
3. FREQ='ALL'
4. SEQNUM=numéro d'image

Pour tous les darks on fait la même chose, avec un champ EXPTIME adapté.

On passe aux flats, le champ FREQ devient pertinent :

— NEWACQ Seq IMAGETYP=Flat FREQ=A

Champs à inclure :

1. IMAGETYPE='Flat'
2. FREQ='A'
3. SEQNUM=numéro d'image

on fait la même chose pour tous les flats, en adaptant le champ FREQ.

On passe aux lights :

— NEWACQ Seq IMAGETYP=Light EXPTIME=300 FREQ=L

Champs à inclure en sortie :

1. IMAGETYPE='Light'
2. EXPTIME=300
3. FREQ='L'
4. SEQNUM=numéro d'image

on fait la même chose pour tous les lights, en adaptant les champs FREQ et EXPTIME.

2.2 Traitement

On n'impose rien sur le nom du fichier, c'est au logiciel de traitement de choisir ce qui lui paraît pertinent comme nom. Tout ce qu'on impose c'est des mots-clés FITS embarqués dans le fichier.

— NEWPROCESS makemaster IMAGETYP=Bias

La stratégie de construction de la séquence par le logiciel de traitement consiste à lister les images du répertoire de travail dont le champ IMAGETYP vaut Bias et le champ SEQNUM est non nul. Une fois que c'est fait, c'est terminé, le traitement est le même qu'avant.

makemaster est une instruction symbolique disant qu'il faut stacker une séquence pour en faire un master. Dans le cas du Bias, les mots-clés à inclure dans l'image master de sortie :

1. IMAGETYP='MasterBias'
2. FREQ='ALL'

3. SEQNUM = 0 les champs FREQ et SEQNUM ns sont pas nécessaires ; il peut être utile de les ajouter soit pour des raisons de cohérence soit pour des raisons d'efficacité.

— NEWPROCESS makemaster IMAGETYP=Dark EXPTIME=120

Dans le cas du Dark, les mots-clés à inclure dans l'image master de sortie :

1. IMAGETYP='MasterDark'
2. EXPTIME=120
3. FREQ='ALL'
4. SEQNUM = 0

Tous les darks sont traités de la même manière, avec un champ EXPTIME ajusté.

— NEWPROCESS makemaster IMAGETYP=Flat FREQ=L En ce qui concerne les flats, le MasterBias est la seule image avec IMAGETYP=MasterBias ; on a toutes les informations pour faire le MasterFlat L. Idem pour toutes les fréquences.

En sortie :

1. IMAGETYP='MasterFlat'
2. FREQ='L'
3. SEQNUM = 0

— NEWPROCESS makemaster IMAGETYP=Light FREQ=L EXPTIME=300

En ce qui concerne les Lights, le MasterBias est la seule image avec IMAGETYP=MasterBias ; il y a une seule image MasterDark/ExpTime=300, il y a une seule image MasterFlat/FREQ=L, donc on a toutes les informations pour contruire le MasterLight/L/300. Idem pour toutes les fréquences et tous les temps de pose.

En sortie :

1. IMAGETYP='MasterLight'
2. FREQ='A'
3. EXPTIME='300'
4. SEQNUM = 0

on fait la même chose pour tous les "Lights".

Enfin pour la dernière étape,

— NEWPROCESS composite Master_Light_L300 Master_Light_L120 Master_Light_L30 ...

on peut simplement rendre la main à l'observateur pour la partie la plus intéressante du traitement.

3 Contraintes/suggestions d'implémentation

3.1 Spécifications

Le format des mots-clés FITS utilisés, notamment les valeurs légitimes pour chacun d'eux doivent être standardisés pour autoriser l'interopérabilité des logiciels souhaitant implémenter la proposition.

3.2 Acquisition

La liste des tâches d'acquisition peut être implémentée comme liste déroulante à proposer à l'observateur. Des informations diverses sur l'état de chaque tâche peuvent accompagner cette liste, comme le nombre d'images déjà existantes dans telle ou telle séquence ; la sélection d'une tâche concernant une séquence déjà existante peut donner le choix entre ajouter des images à la séquence (SEQNUM adapté) ou écraser l'existant (ça nécessite de faire le ménage pour éviter des ambiguïtés au traitement).

Une tâche peut être marquée comme terminée par l'observateur, ça peut être une facilité pour le logiciel de traitement.

...

3.3 Traitement

La construction des séquences et de manière plus générale l'identification des images déjà acquises dans le cadre du projet est la principale nouveauté ; ça peut paraître lourd à implémenter parce que ça implique de passer en revue les headers de tous les fichiers fits du répertoire de travail, mais d'une part les bénéfices du point de vue utilisateur sont inappréciables et d'autre part, bien organisé c'est sans doute complètement indolore en terme de temps de calcul par rapport au traitement lui-même.

Sur le traitement lui-même, chaque tâche du fichier .tasks doit donner lieu à une partie de script ; ce n'est pas forcément trivial à faire, mais toutes les informations pour produire le script final sont disponibles dans les images et les tâches de traitement.