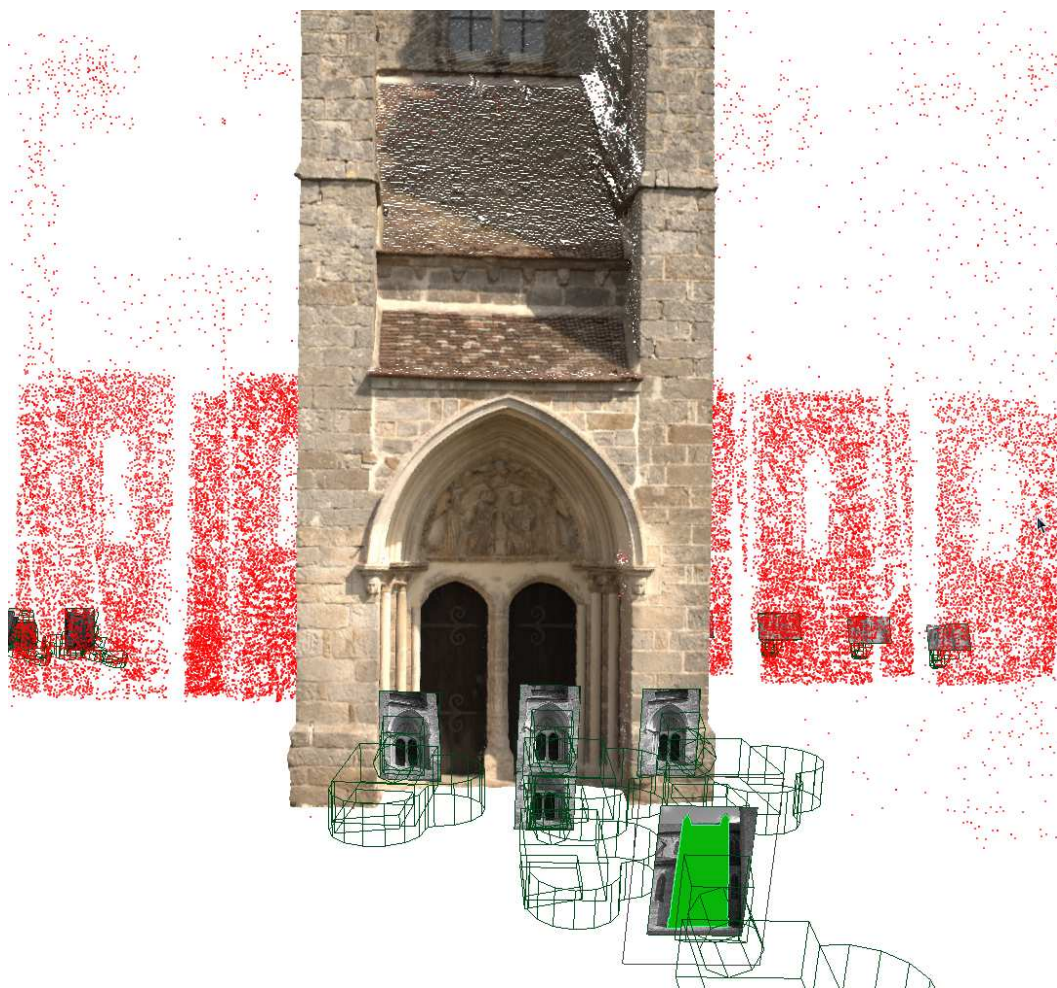


## GUIDE PROGRAMMEUR DE PHOTOCLOUD



Aymeric GODET

# SOMMAIRE

---

Ce document a pour but de présenter au futur programmeur de PhotoCloud l'architecture générale du logiciel.

|     |  |    |
|-----|--|----|
| I.  | Installation .....                               | 3  |
| 1.  | Installation de la chaine de traitement PAM..... | 3  |
| a)  | Les sources : .....                              | 3  |
| b)  | Les compilations.....                            | 3  |
| 2.  | Installation de PhotoCloud .....                 | 4  |
| II. | PhotoCloud.....                                  | 5  |
| 1.  | Structure Générale.....                          | 5  |
| 2.  | PhotoCloud.mel .....                             | 5  |
| 3.  | PC_Aff .....                                     | 6  |
| 4.  | PC_ErrSys .....                                  | 6  |
| 5.  | PC_Lib .....                                     | 6  |
| 6.  | PC_Sav.....                                      | 6  |
| 7.  | PC_Trait.....                                    | 6  |
| a)  | TraitImg.mel : .....                             | 7  |
| b)  | TraitPastis.mel : .....                          | 7  |
| c)  | TraitApero.mel : .....                           | 7  |
| d)  | PbApero.mel .....                                | 8  |
| e)  | TreeCor.mel .....                                | 9  |
| f)  | Masq.mel.....                                    | 9  |
| g)  | TraitMICMAC.mel.....                             | 10 |
| 8.  | Commande système utile: .....                    | 10 |

# I. INSTALLATION

---

Le guide d'installation présenté ici est pour une machine ayant un système d'exploitation de la distribution FEDORA.

## 1. Installation de la chaine de traitement PAM

### a) Les sources :

Afin d'assurer l'utilisation des fichiers binaires adéquats correspondant à la machine de travail (32 ou 64 bits) il est conseillé de recompiler à partir des derniers fichiers sources disponible sur le site :

<http://www.micmac.ign.fr/svn>

La commande `svn checkout` permet de tous télécharger en une fois.

Il faut également télécharger les fichiers binaires auxiliaires sur le lien

[http://www.micmac.ign.fr/svn/micmac\\_data/trunk/aux\\_bin/](http://www.micmac.ign.fr/svn/micmac_data/trunk/aux_bin/)

### b) Les compilations

La première étape consiste à compiler la librairie élise contenue dans le fichier Makefile. La commande à réaliser à la console est donc :

```
make bin/elise.a -f Makefile
```

Ensuite il faut compiler Myrename :

```
make bin/MyRename -f Makefile
```

MpDcraw :

```
make all -f MakeMpDcraw
```

MapCmd :

```
make bin/MapCmd -f Makefile
```

PASTIS :

```
make bin/Pastis -f Makefile
```

APERO :

```
make all -f AperiMake
```

MICMAC :

```
make all -f MakeMICMAC
```

Il faut également compiler dcraw.

```
gcc -o dcraw -O4 dcraw.c -lm -DNO_JPEG -DNO_LCMS
```

Il faut enfin compiler le projet Qt SIFT3D

Nous avons à présent tous les logiciels pour réaliser les traitements. Les binaires doivent être placés dans le répertoire /usr/bin afin que les commandes puissent être reconnues n'importe où par la console. Il reste cependant des dépendances « locales » qui oblige à placer dans le répertoire /usr/Pastis les fichiers auxiliaires \*.LINUX nécessaires à PASTIS (ils seront automatiquement recopier dans les repertoire de projet NUBES par PhotoCloud).

## **2. Installation de PhotoCloud**

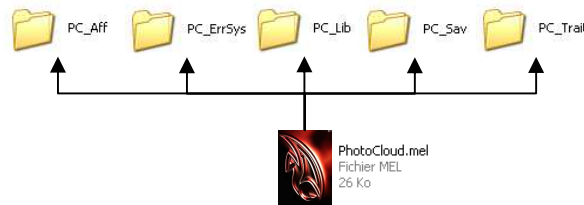
Le dossier d'installation de PhotoCloud doit être placé dans le dossier racine de NUBES. Il faut également remplacer le fichier « interface.mel » afin d'avoir l'intégration de PhotoCloud dans NUBES.

## II. PHOTOCLOUD

---

### 1. Structure Générale

Dans le dossier d'installation de PhotoCloud sont placés différents dossiers qui regroupent des scripts MEL classés par thématiques :



PC\_Aff : Regroupe les fonctions d'affichage de l'interface.

PC\_ErrSys : Fonctions de contrôle.

PC\_Lib : Les bibliothèques Mel extérieures utilisées.

PC\_Sav : Les fonctions de sauvegarde d'étape de PhotoCloud.

PC\_Trait : Regroupe les différentes étapes de Traitement de la chaîne PAM.

### 2. PhotoCloud.mel

Ce fichier contient les éléments de description de l'interface graphique ainsi que les variables globales utilisées.

global proc ChantierPath() :

Fonction pour afficher le chemin source des images, l'extension du format RAW et charge la liste des images dans le bloc image brute de l'interface.

global proc string pathpart( string \$path ) :

Fonction récupérant seulement le chemin d'une adresse

global proc string filepart( string \$path ) :

Fonction récupérant seulement le nom de fichier d'une adresse

global proc PhotoCloud(string \$path, string \$NUBEScloud,string \$NUBESimg) :

Procédure regroupant tous les éléments de l'interface de PhotoCloud. Chaque étape est identifiée par un préfixe NC(i) comme **N**ouveau **C**hantier **i**ème étapes.

\$path : Chemin du chantier NUBES

\$NUBEScloud : Identifiant du widget de la liste des Nuages de points dans NUBES

\$NUBESimg : Identifiant du widget de la liste des Images dans NUBES

### **3. PC Aff**

#### **Affstep.mel :**

Ce script contient les fonctions qui gèrent la visibilité et l'accessibilité aux différents éléments de l'interface en fonction de l'étape du traitement. C'est également ici que sont lancées les fonctions de traitements.

#### **anc2.mel et anc6.mel :**

Scripts pour ordonner dynamiquement les listes des images des étapes 2 (sélection des images brutes) et 6 (Tri dans les trois blocs Calibration, Globale, Locale).

#### **Camera.mel :**

Script qui sauvegarde les caméras du chantier et qui retransmet les positions et orientations dans le système de Maya. Elle permet de réaliser l'affichage du plan de situation dans le viewer de Maya après calcul.

### **4. PC ErrSys**

Ce dossier a été appelé erreur système car lors de la programmation de PhotoCloud les différentes stratégies pour définir les conditions de fin dans les boucles de contrôle d'avancement généraient des erreurs. Il ne reste à présent que le critère d'activité du processus dans le script Zombies.mel. La fonction prend en entrée un PID et retourne en sortie un binaire en fonction de l'état zombie ou non du processus.

### **5. PC Lib**

Ce dossier contient la librairie MEL de gestion des fichiers XML pour atteindre facilement les nœuds d'un fichier.

### **6. PC Sav**

Ce dossier contiendra les scripts de sauvegarde des différents éléments d'un traitement par PhotoCloud : l'état d'avancement des traitements des sites de corrélation (Masque et plan d'accroche) ainsi qu'un point de sauvegarde après le traitement réalisé par Pastis.

### **7. PC Trait**

Dans ce dossier central sont regroupés tous les scripts de gestion des étapes de la chaîne de traitement.

### **a) TraitImg.mel :**

proc int convertTifFolder(string \$path,int \$nbtot) :

Cette procédure convertit automatiquement toutes les images RAW du dossier \$path en TIF. \$nbtot est le nombre d'image total à traiter.

La procédure renvoi 1 si l'ensemble de la conversion c'est bien passé. 0 sinon. Le retour d'information est important car il arrive que les conversions soient bloquées par le système. Du coup la procédure est parfois réitérée.

global proc IniImg() :

Nous commandons ici les différents prétraitements sur les images :

- Copie des fichiers RAW dans le fichier du projet NUBES.
- Détection des focales en lisant les EXIF
- Génération des tifs
- Générations des jpg à moyenne et faible résolution.

### **b) TraitPastis.mel :**

Dans ce fichier est contenue la procédure globale, TraitPastis(), qui pilote PASTIS. Le traitement étant linéaire les deux résolutions sont traitées à la suite. Les différentes étapes effectuées sont :

- Génération des calibrations initiales des caméras Cal-FXX-Init.xml
- Génération du fichier descripteur du chantier MicMac-LocalChantierDescripteur.xml. Ce fichier définit les clés qui sont utilisées dans le traitement de la chaine PAM (cf Paramétrage de la chaine PAM).
- Lancement du calcul de Pastis à faible résolution.
- Lancement du calcul de Pastis à moyenne résolution.

### **c) TraitApero.mel :**

Ce script permet de commander Apero avec différentes procédures :

global proc TraitApero()

Nous générons ici le fichier de paramétrage d'Apero pour chaque calibration des caméras et nous lançons le calcul. Dans le cas des téléobjectifs la fonction est redirigée vers la procédure CalTele() du fichier PbApero.mel. Le cas de calibration des fisheyes est intégré dans ce processus car les modifications du fichier de paramètre de calibration n'est pas aussi lourde que dans le cas des téléobjectifs (avec deux focales pour avoir une stabilité dans le calcul).

#### global proc Aperochant()

Nous générons dans un premier temps le fichier de paramètres XML. Dans ce fichier les objectifs « moyen » et les téléobjectifs ont le même rôle d'un point de vue des paramètres. En revanche s'il y a un fisheye sur le chantier, la balise de libération des contraintes doit contenir en plus une clé pour savoir quelle caméra est libérée. Il s'agit de :

<PatternSel> </PatternSel>

Dans un second temps le calcul est lancé.

#### global proc siftCloud(string \$nImp,string \$pC)

Cette procédure est la dernière du traitement avant la corrélation. On génère les systèmes de particule SIFT dans Maya. Le principe est de faire un nuage de point par paire de photo classée dans le bloc images globales. Le calcul est réalisé en utilisant le logiciel SIFT3D par une commande system().

Nous rédigeons et exportons ensuite dans les dossiers images et nuages du projet NUBES les caméras du chantier au format Maya ascii.

Enfin on sauvegarde l'arbre de corrélation dans le fichier listCorel.txt dans le dossier PhotoCloud. Le formalisme utilisé dans le fichier texte est le suivant :

```
+ image maitresse
Image fille
Image fille ...
Image fille
+ Nouvelle image maitresse
Image fille ...
```

#### **d) PbApero.mel**

Les fonctions dans ce fichier servent à gérer les cas où les résidus après Aperochant sont forts :

#### global proc affResSel()

Affiche dans l'interface le résidu de la calibration sélectionnée

#### global proc Journal(int \$cmd)

Ouvrir le journal des calculs d'APEROC.

#### global proc ReCalib()

Relancer un calcul de calibration. Cas où l'opérateur a modifié le tri des images du bloc de calibration.

#### global proc ReChant()

Relancer le calcul d'aérottriangulation global du chantier. Cas où l'opérateur a modifié le tri des images du bloc globale.



global proc ReitCal()

Réitération du calcul de calibration. Cas où la calibration initiale utilisée est le résultat d'une première passe de calibration avec le même jeu de donnée.

global proc ReitChant()

Réitération du calcul d'aérottriangulation global du chantier. Cas où la calibration initiale utilisée est le résultat d'une première passe d'aérottriangulation global avec le même jeu de donnée.

global proc CalTele(int \$ifocTele) :

Nous rédigeons ici le fichier de paramètre de calibration du téléobjectif en utilisant également un site de calibration d'un objectif « moyen » du chantier.

### **e) TreeCor.mel**

Ce script est relatif à la gestion de l'arbre de corrélation dans NUBES. Les fonctions sont :

global proc SaveTree(string \$projet)

Nous mettons ici à jour le fichier listCorel.txt en fonction des modifications faites par l'opérateur.

global proc TreeButton (string \$s, string \$proj)

En fonction de l'image sélectionnée dans l'arbre de corrélation nous rendons accessible ou non les outils en fonction de son état d'avancement.

global proc int CorVue(string \$proj,string \$toolbar,string \$s,int \$i)

En fonction de l'image sélectionnée et de la structure de l'arbre nous gérons ici l'affichage des images (vue principale et bandeau supérieur des images filles).

global proc RefreshTree(string \$proj)

On charge l'arbre de corrélation à partir du fichier listCorel.txt du chantier NUBES courant.

### **f)Masq.mel**

Ce script est relatif aux différentes étapes de la saisie d'un masque

global proc ValidMasq(int \$new)

Correspond à l'étape de validation ou suppression du masque.

global proc SaisiMasq()

Correspond à l'étape de saisi du masque avec la création du plan aimanté (*makelive*) et du dessin du masque (*CreatePolygonTool*)

global proc MasqGen(string \$proj)

Nous lançons ici le calcul de rendu de Maya en filtrant les objets avec la couche de rendu associée à l'image maitresse. L'image est ensuite convertie pour être compatible avec MicMac.

## **g) TraitMICMAC.mel**

Ce script contient la procédure globale MicmacLauncher(). On rédige dans un premier temps le fichier de paramètre MicMac en questionnant l'arbre de corrélation pour savoir la structure du site.

## **8. Commandes systèmes utiles :**

Il y a dans PhotoCloud un certain nombre de commandes systèmes qui sont utilisées pour déplacer, créer ou supprimer des fichiers et dossiers. Il y en a d'autres un peu plus spécifiques qui sont par exemple contenues dans les boucles de contrôles d'avancement. Je les présente ici pour en expliquer le fonctionnement et aider le futur programmeur qui sera peut être amené à rendre ce logiciel compatible sur Mac ou Windows.

### Structuration des fichiers et dossier

|         |                                     |
|---------|-------------------------------------|
| cp :    | copier un fichier                   |
| mkdir : | créer un répertoire                 |
| mv :    | déplacer un fichier                 |
| rm :    | supprimer un fichier                |
| rm -r : | supprimer un dossier et son contenu |

### Commandes spécifiques :

|                          |  |
|--------------------------|--|
| Commande > journal.txt : | Renvoyer le flux de sortie d'une commande dans un fichier texte      |
| chmod 777 p1 :           | rendre le fichier p1 accessible en lecture écriture                  |
| convert :                | Change le format des images avec ImageMagick                         |
| gnome-open :             | Commande pour ouvrir un document avec le logiciel adéquat par défaut |
| grep p1 p2 :             | Rechercher le texte p1 dans l'espace p2                              |
| ls -l p1 :               | liste les éléments en filtrant avec p1                               |

ps PID :

Affiche les informations concernant le statut du processus PID