

# Acquisition de données pour CTAO

À l'aide de librairies standard et d'implementations maison

Etienne Lyard, Université de Genève - avec beaucoup d'aide de l'organisation centrale de CTAO et de son consortium  
[Etienne.Lyard@unige.ch](mailto:Etienne.Lyard@unige.ch)

- 
- 1 Introduction - CTAO
  - 2 ACADA-ADH
  - 3 Bibliothèques standard
  - 4 Implementations maison
  - 5 Conclusion

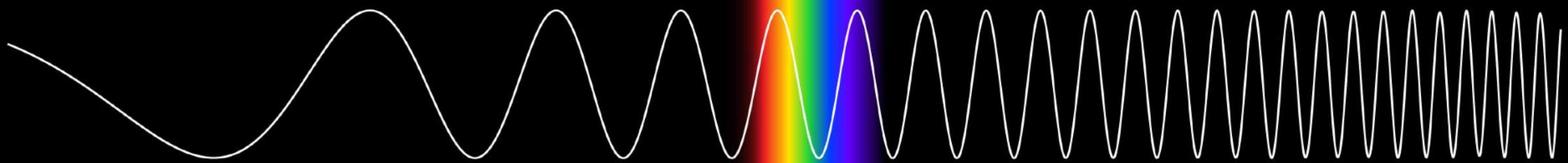
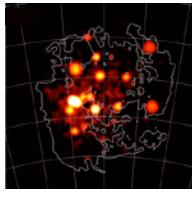
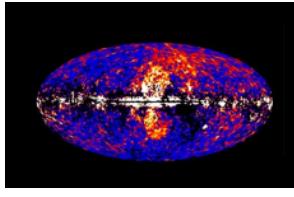
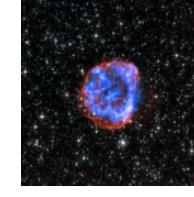
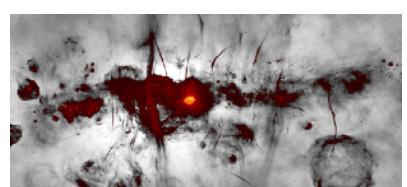
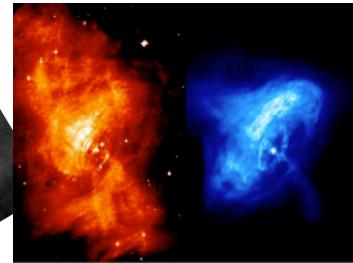
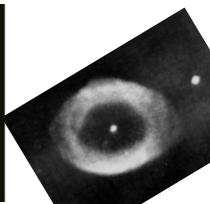
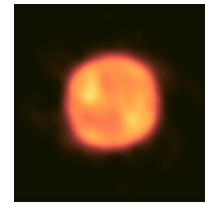
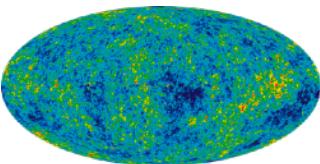
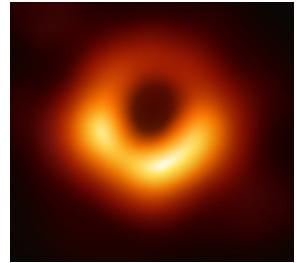
# CTAO c'est...

- L'Observatoire de réseau de Télescopes Tcherenkov
- L'observatoire **le plus grand et le plus sensible** aux rayons gammas
- La somme de l'organisation centrale CTAO ERIC et de **tous ses partenaires** – contributeurs en nature qui développent les technologies nécessaires ou groupes de scientifiques du consortium CTAO.
- la **prochaine évolution** dans l'exploration des événements les plus cataclysmiques de l'univers, permettant d'étudier les rayons cosmiques, trous noirs super-massifs et la matière noire.
- le premier observatoire de ce type qui permettra un accès ouvert à ses données et produits de haut niveau.
- un **projet "landmark"** du Forum stratégique Européen sur les infrastructures de recherche (ESFRI), classé à **la plus haute priorité** parmi les nouvelles infrastructures terrestres de la feuille de route ASTRONET 2022-2035.

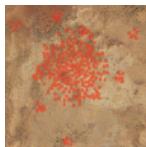
CTAO Director General, Stuart  
McMuldroch, with Italian  
Minister of University and  
Research, Anna Maria Bernini,  
durant l'inauguration du ERIC  
de CTAO. Credit: INAF



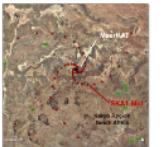
# La lumière que l'on voit (ou pas)



RADIO



SKA Low



MeerKAT / SKA Mid

MICROWAVE

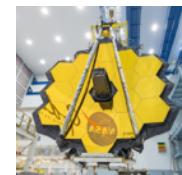


ALMA



Herschel

INFRARED



WST

VISIBLE



HST

ULTRAVIOLET



GALEX

X-RAY



Chandra



Fermi

GAMMA RAY



CTAO



EHT



Planck

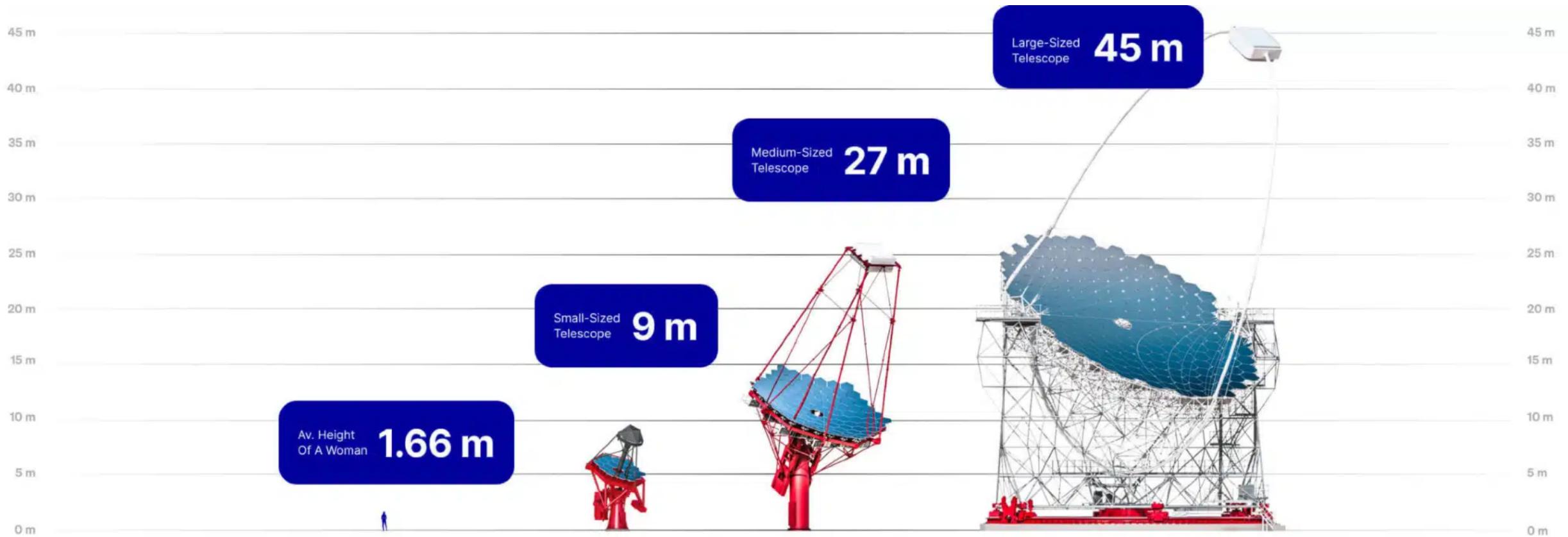


VLT



Yerkes

# Les télescopes





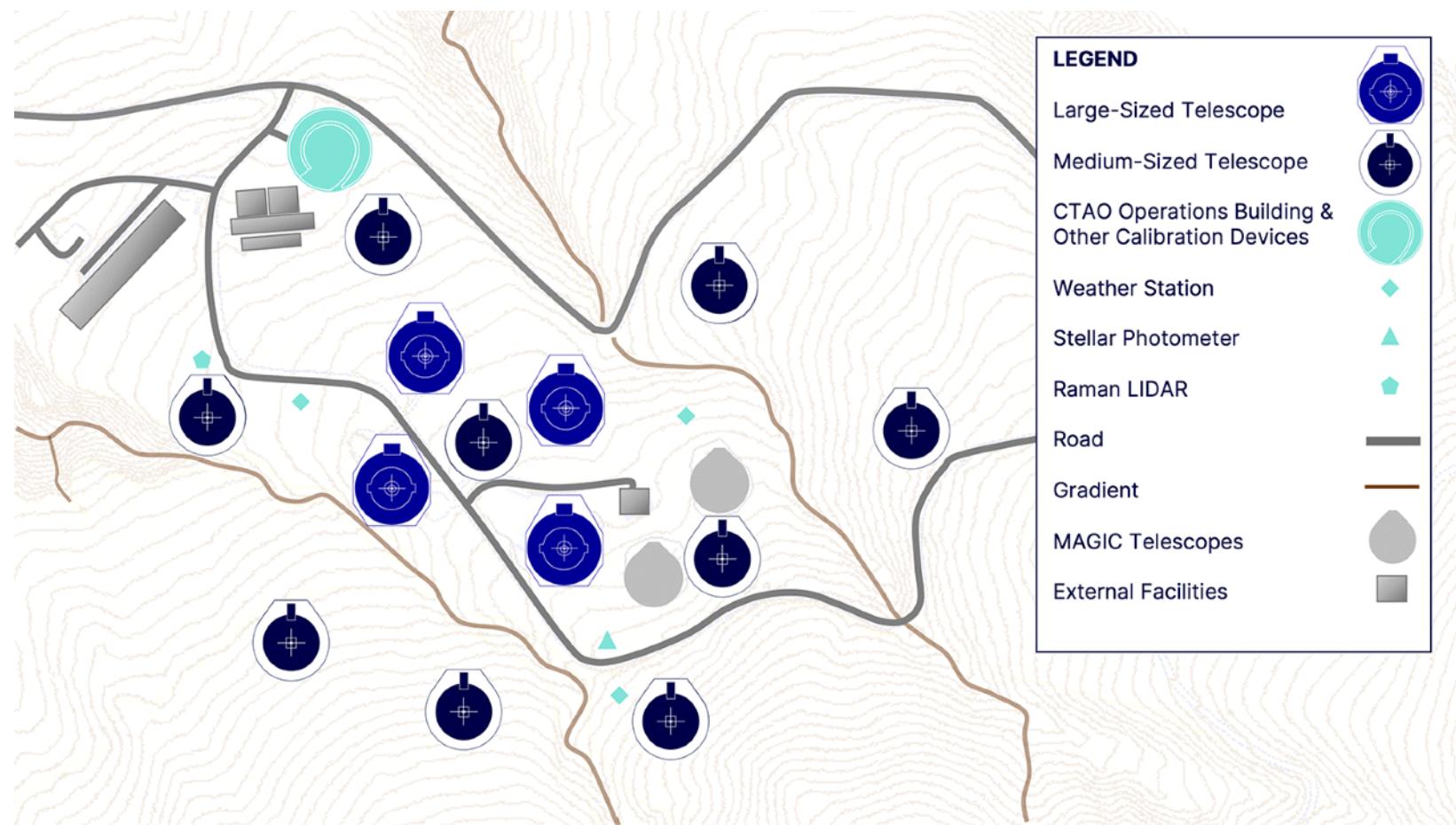
# CTAO-Nord

- Situé sur l'île de **La Palma** en Espagne, à 2200 mètres d'altitude
- Hébergé par l'institut d'astrophysique des Canaries (IAC) à l'**observatoire du Roque de los Muchachos**
- LST-1 est en phase de mise en service par le consortium L S T . 3 de plus en construction, mise en service prévue en 2026



Credit: J. Cortina - LST Collaboration

# Configuration du réseau



- Optimisé pour la sensibilité à basses et moyennes énergies, entre 20 GeV et 5 TeV pour la physique extra-galactique

# CTAO-Sud

- Situé dans le **désert de l'Atacama** au nord du Chili
- Hébergé par **l'ESO** entre l'observatoire de Paranal et l'EELT
- Pour l'instant seule l'infrastructure de base est en construction: fondations des télescopes, routes, lignes à haute-tension...
- Les premiers télescopes arriveront en 2026



## Announcement

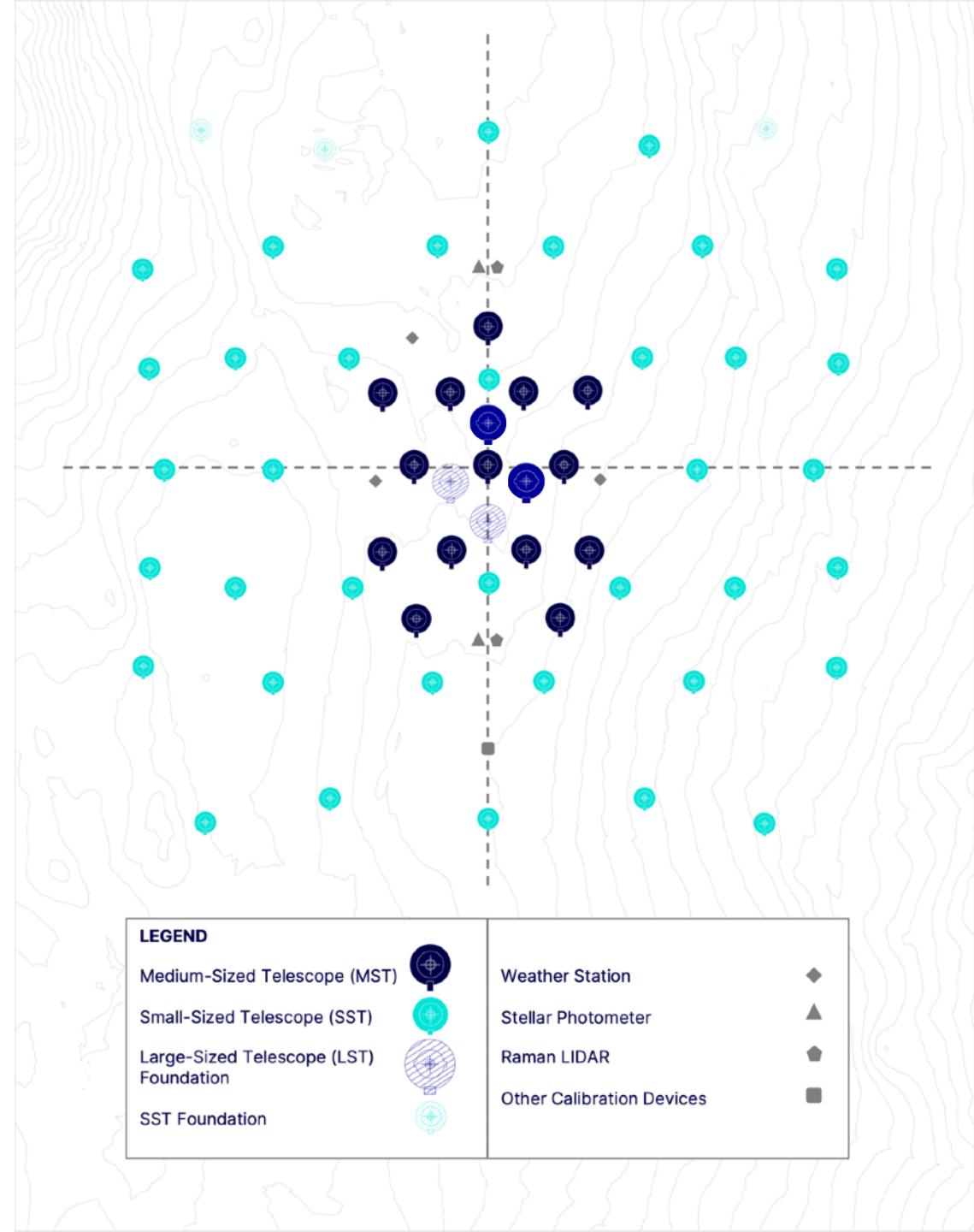
Contract signed for construction of CTAO-South roads and telescope foundations

3 July 2025



# Configuration du réseau

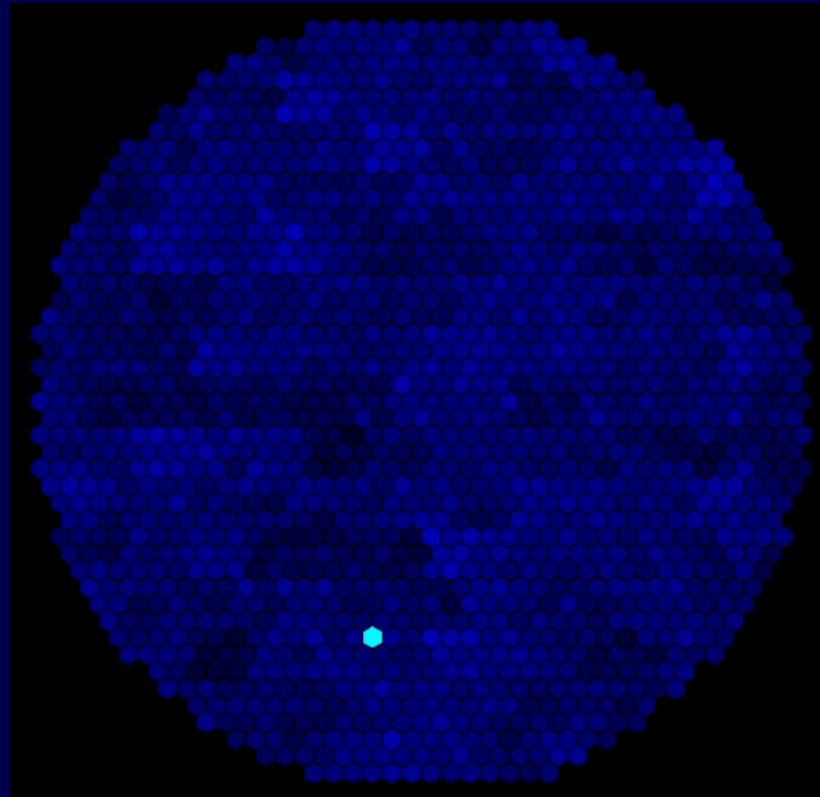
- Inclue 14 MSTs et 37 SSTs
- Les fondations pour les futurs télescopes seront construites.
- Optimisé pour la sensibilité à moyenne et haute énergie, de 150 GeV à 300 TeV pour les sources galactiques.



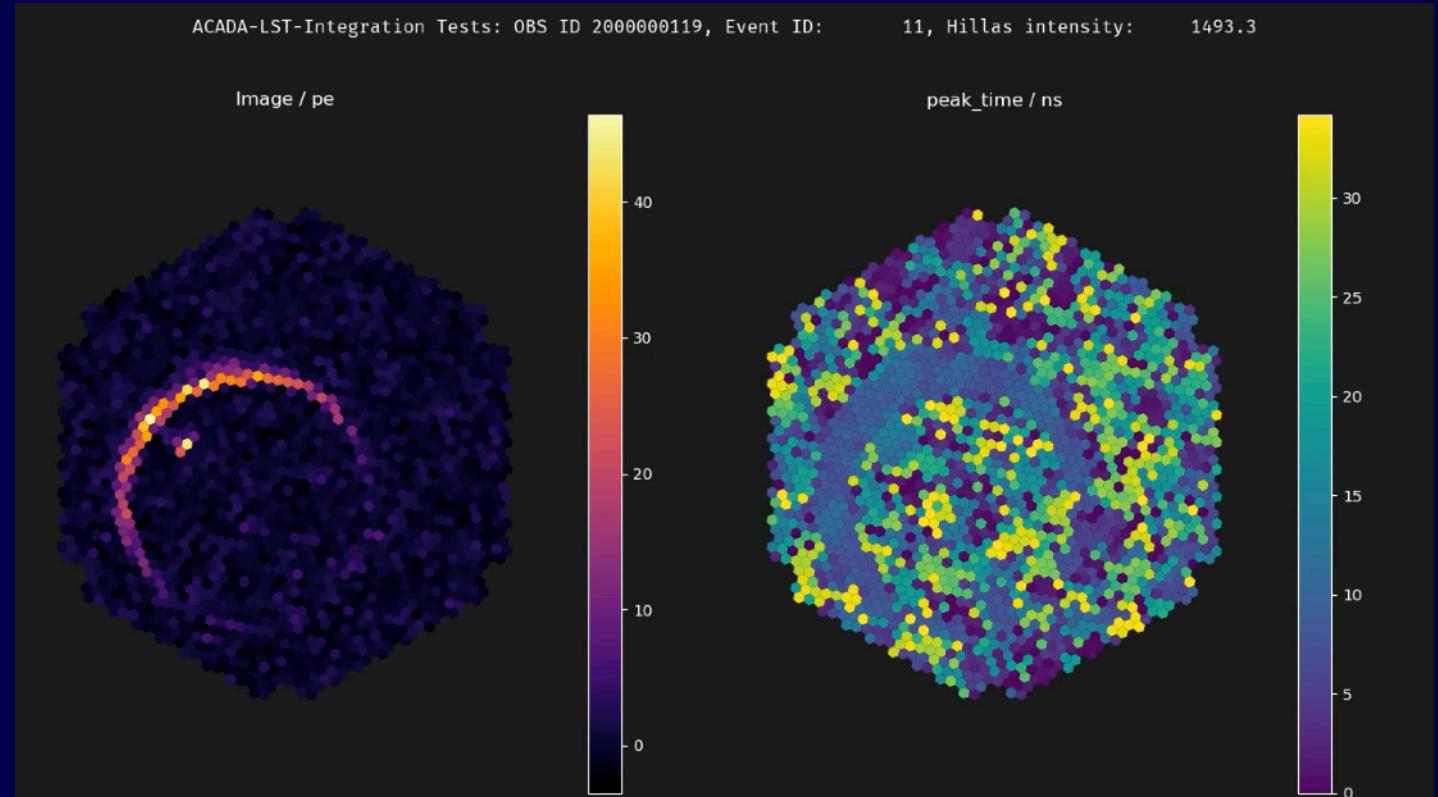
# Comment est-ce que CTAO fonctionne ?



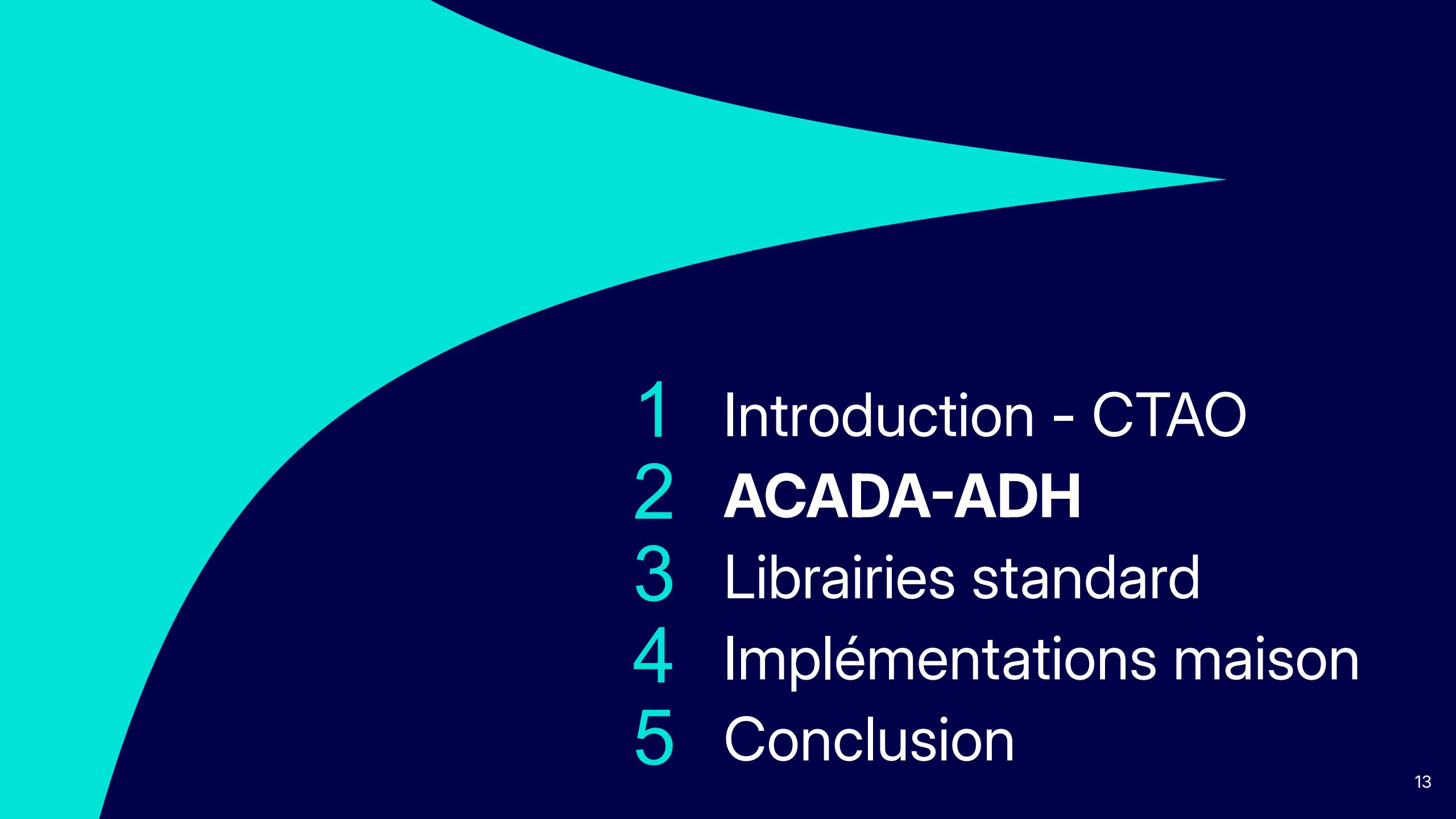
# Qu'est-ce que voit CTAO ?



Télescope FACT - 2 GSPS



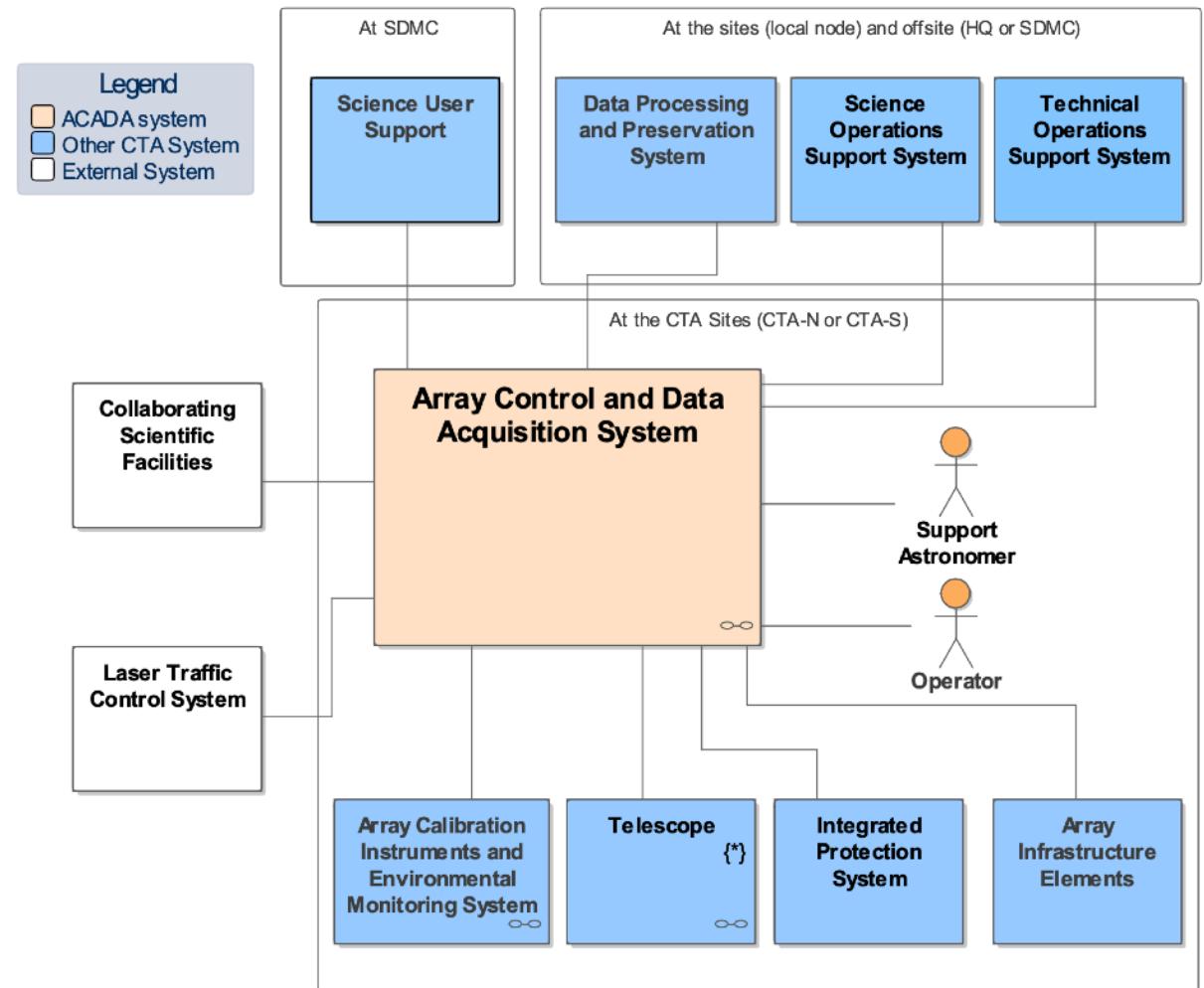
Télescope LST1 - 1 GSPS

- 
- 1 Introduction - CTAO
  - 2 **ACADA-ADH**
  - 3 Librairies standard
  - 4 Implémentations maison
  - 5 Conclusion

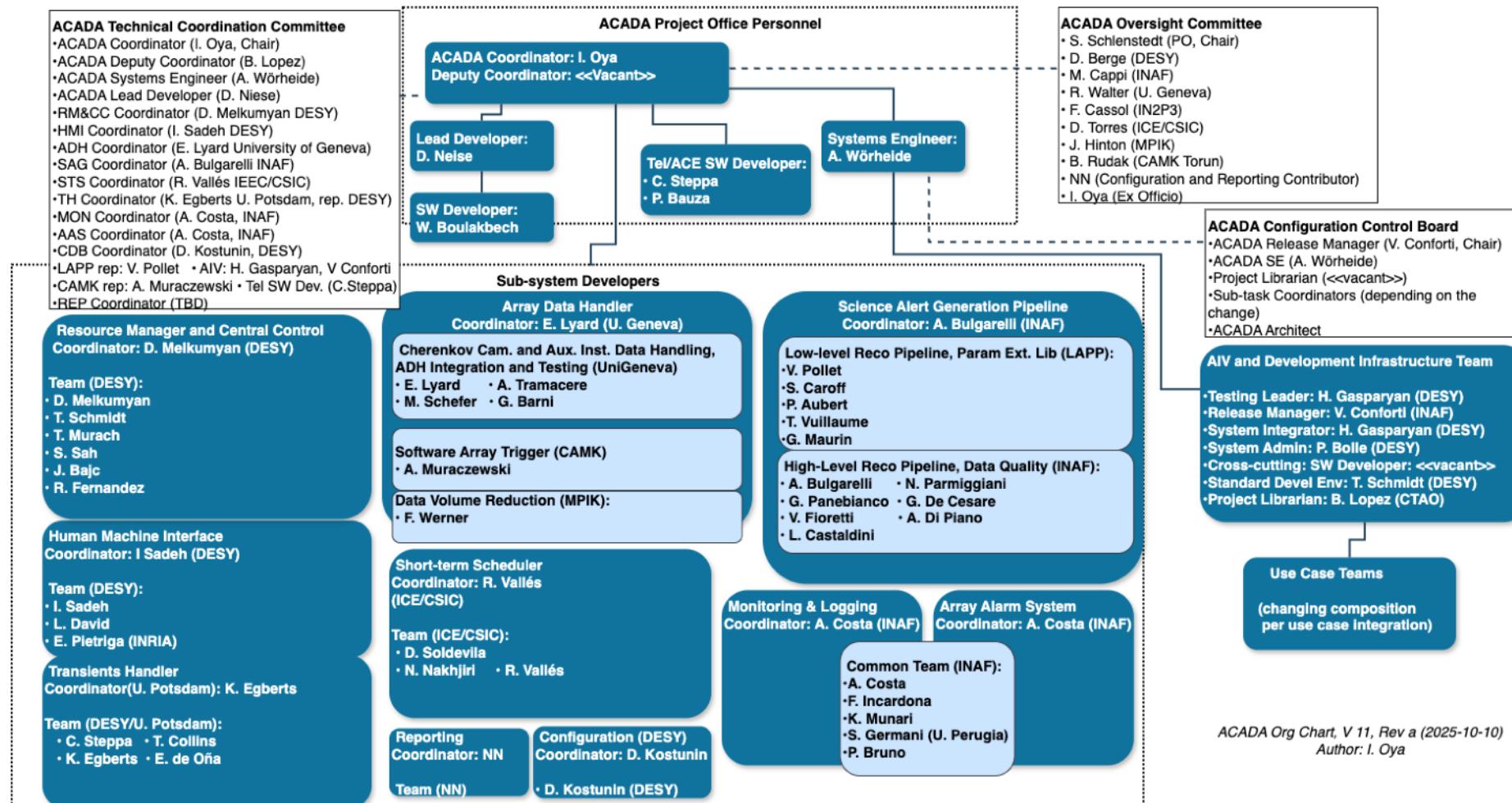
# Array Control and Data Acquisition

## Vue d'ensemble

- **Contrôle central et acquisition de données** pour l'ensemble des télescopes et les instruments auxiliaires sur les deux sites
- Utilise le ALMA Common Software (ACS)
  - Permet une architecture distribuée
  - Offre les fonctionnalités de surveillance des composants
  - Maintenu par ALMA



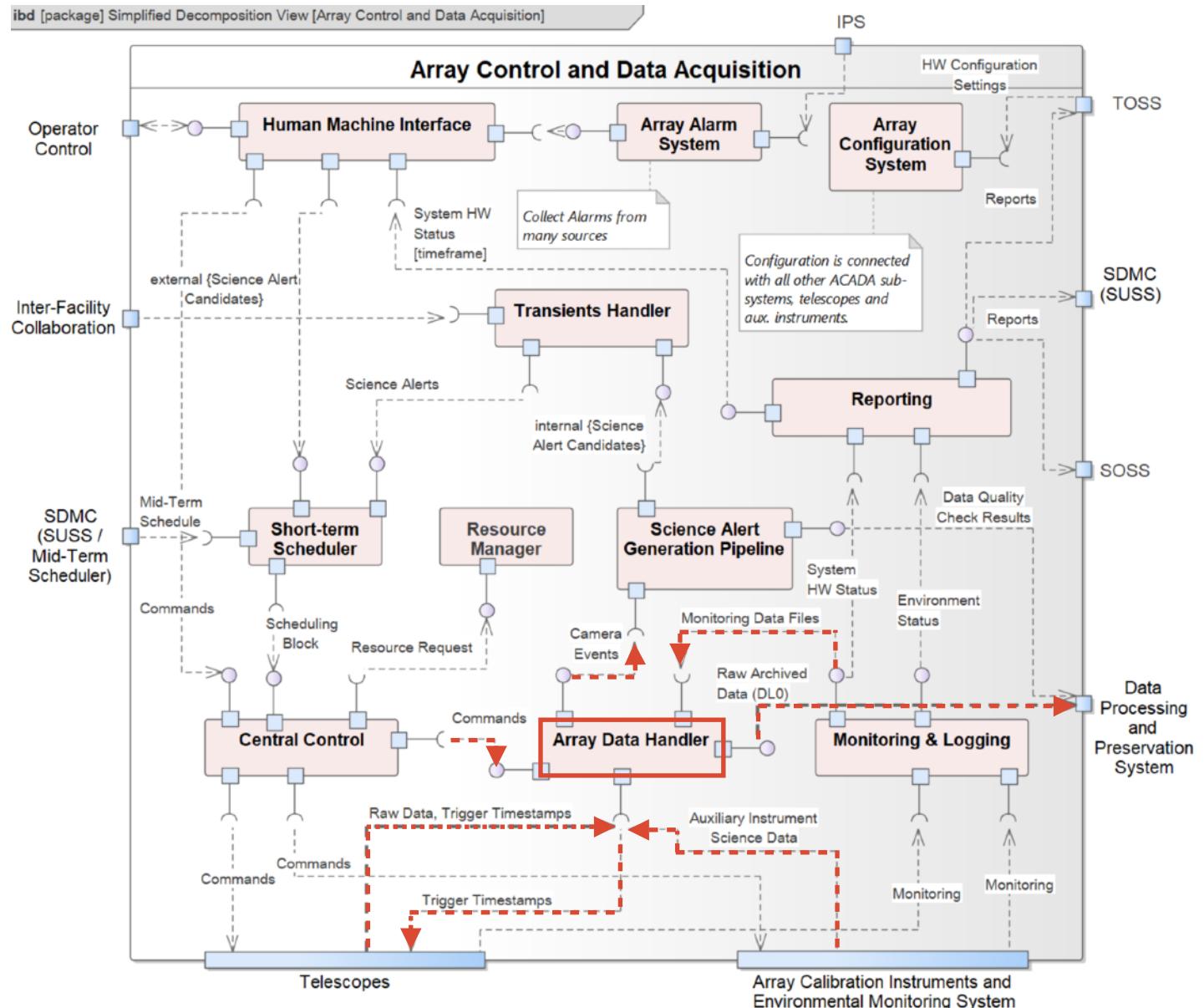
# Array Control and Data Acquisition Organigramme



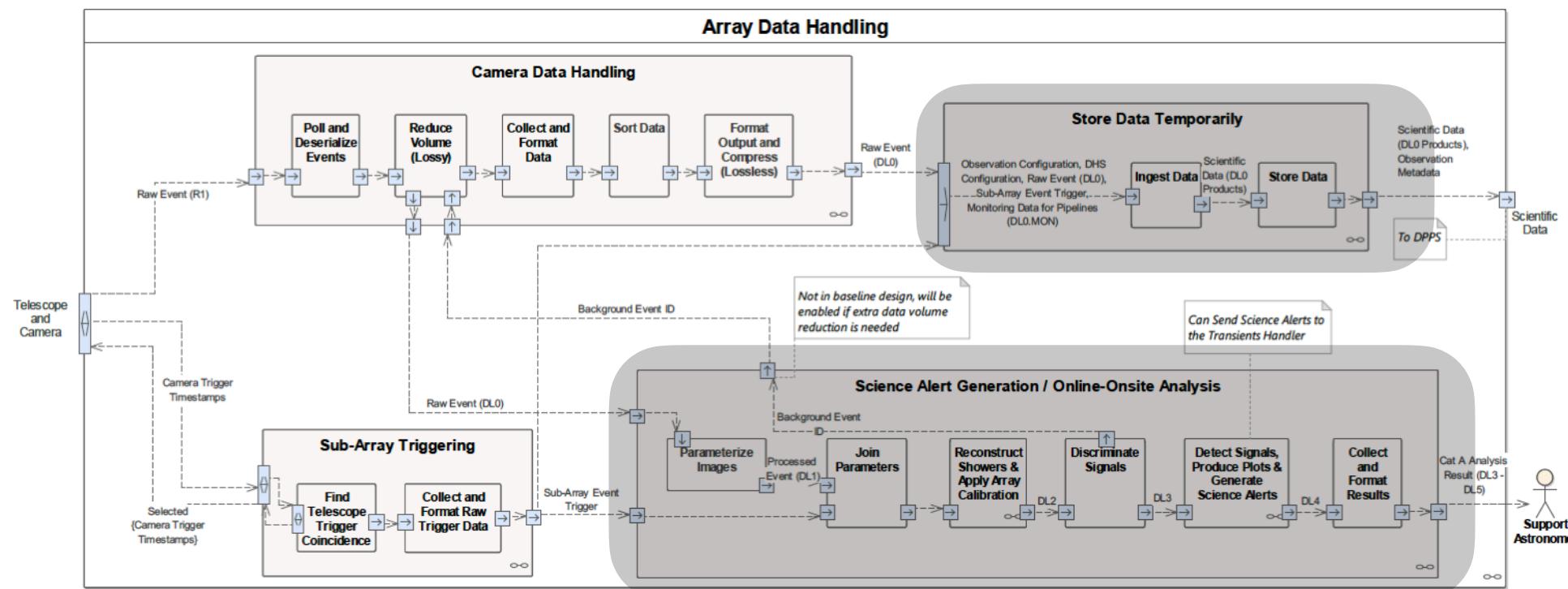
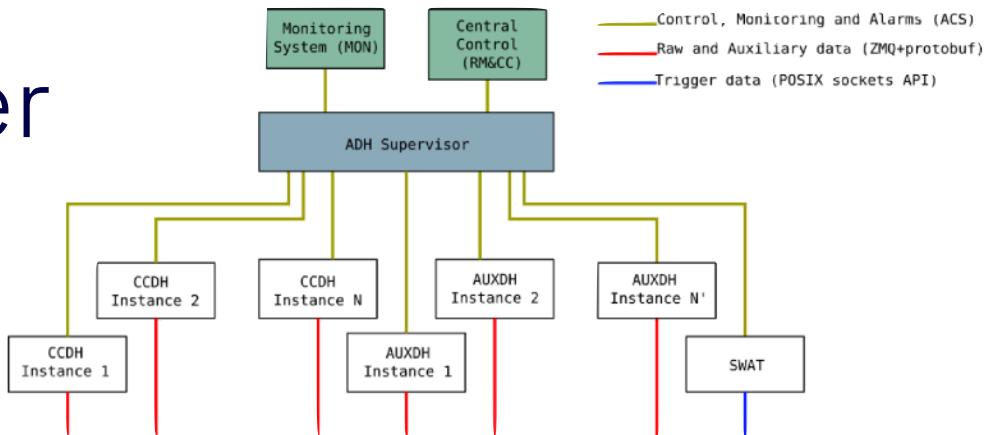
# Array Data Handler

## Fonctionnalités

- Déclenche et enregistre les données brutes venant des caméras de Tcherenkov
- Réduit le volume de données
- Comprime, formate et écrit les événements sur le stockage sur site.
- Fait suivre les données vers l'analyse en temps réel
- Écrit les données auxiliaires, e.g. FRAM
- Écrit chaque jour les données de supervision vers le format d'archivage

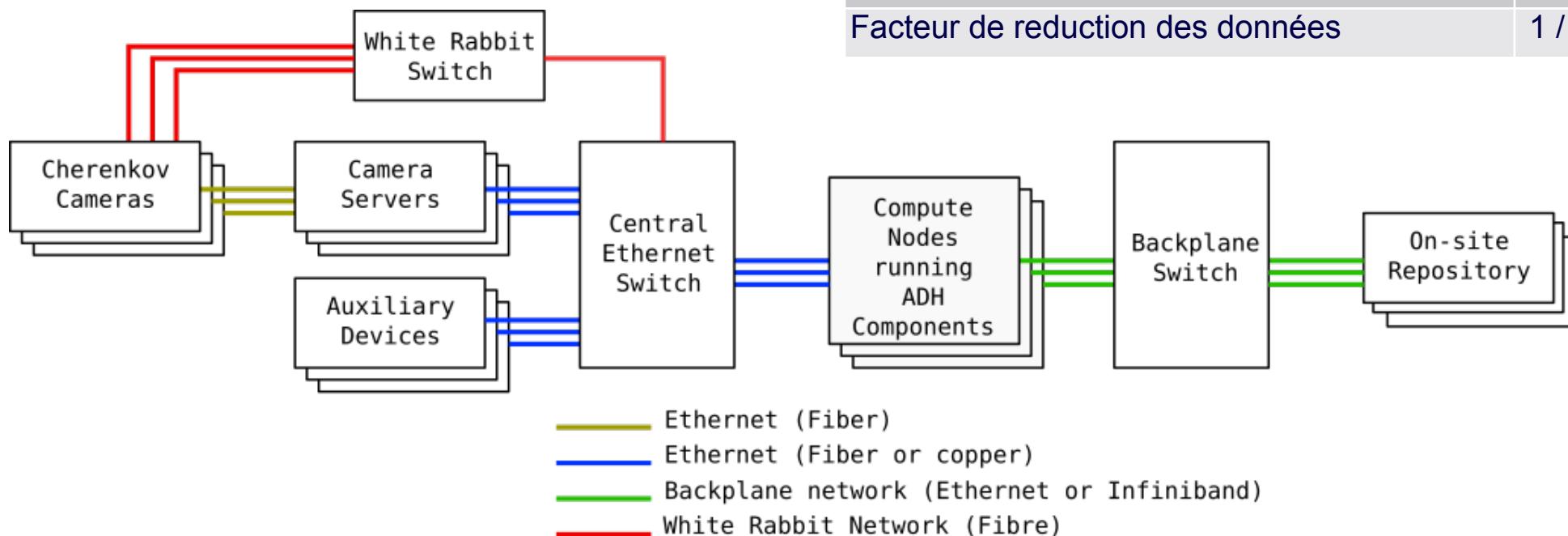


# Array Data Handler Architecture



# Array Data Handler

## Déploiement physique



Volume de données	24 Gb/s / LST (x4)
	12 Gb/s / MST (x25)
	2 Gb/s / SST (x70)
Frequence des evenements monoscopiques	15 kHz / LST (x4)
	14 kHz / MST (x25)
	1.2 kHz / SST (x70)
Frequence des declenchements du reseau	< 40 kHz
Facteur de reduction des données	1 / 50

- 1 Introduction - CTAO
- 2 ACADA-ADH
- 3 **Librairies standard**
- 4 Implémentations maison
- 5 Conclusion

# Utilisation dans ADH

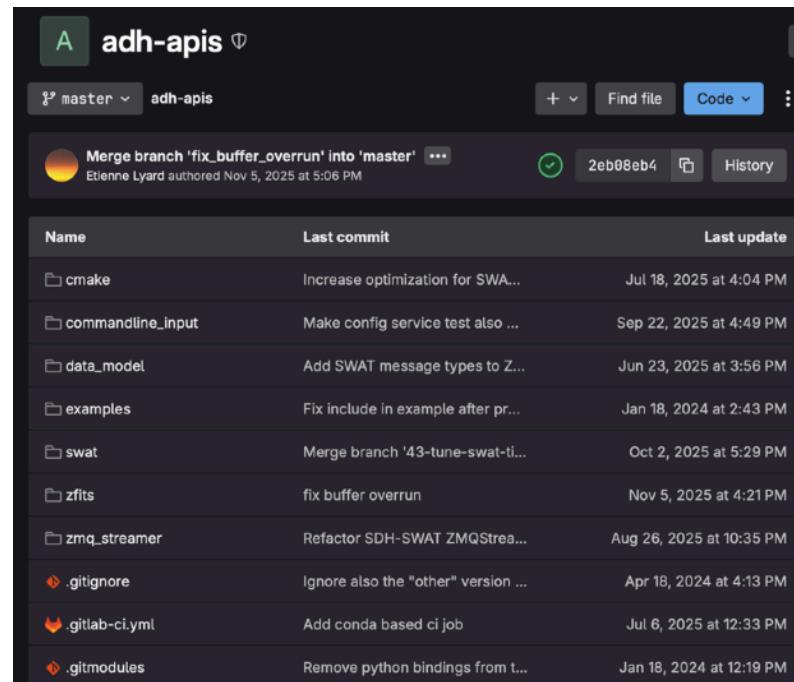
- Utiliser les librairies standard si...
  - Elles existent
  - Sont maintenues
  - Si leur code source est ouvert
  - Sont gratuites
  - Offrent des performances suffisantes
  - ADH utilise STL, **ZEROMQ**, **Protocol Buffers**, ZStandard, astropy...
- Utiliser des produits commerciaux si...
  - Aucun dans ADH pour l'instant (à part les protobuf)

# Utilisation dans ADH

- Pour:
  - Le travail est déjà fait
  - Maintenu par la communauté
- Contre:
  - On ne peut pas facilement ajouter de nouvelles fonctionnalité
    - Cependant on peut contribuer à certaines librairies
- Leur utilisation est toujours abstraite
  - Permet de survivre à un passage vers une autre librairie
  - Pour autant que l'API soit suffisamment similaire...
  - L'utilisateur peut choisir entre l'abstraction, librairie elle-même et l'encodage binaire

# Utilisation dans ADH

- Tout est empaqueté dans un projet interface: adh-apis
  - L'utilisateur peut soit utiliser ce projet directement comme ctao-dpps
  - Soit créer sa propre librairie a partir du projet comme lst1-evb

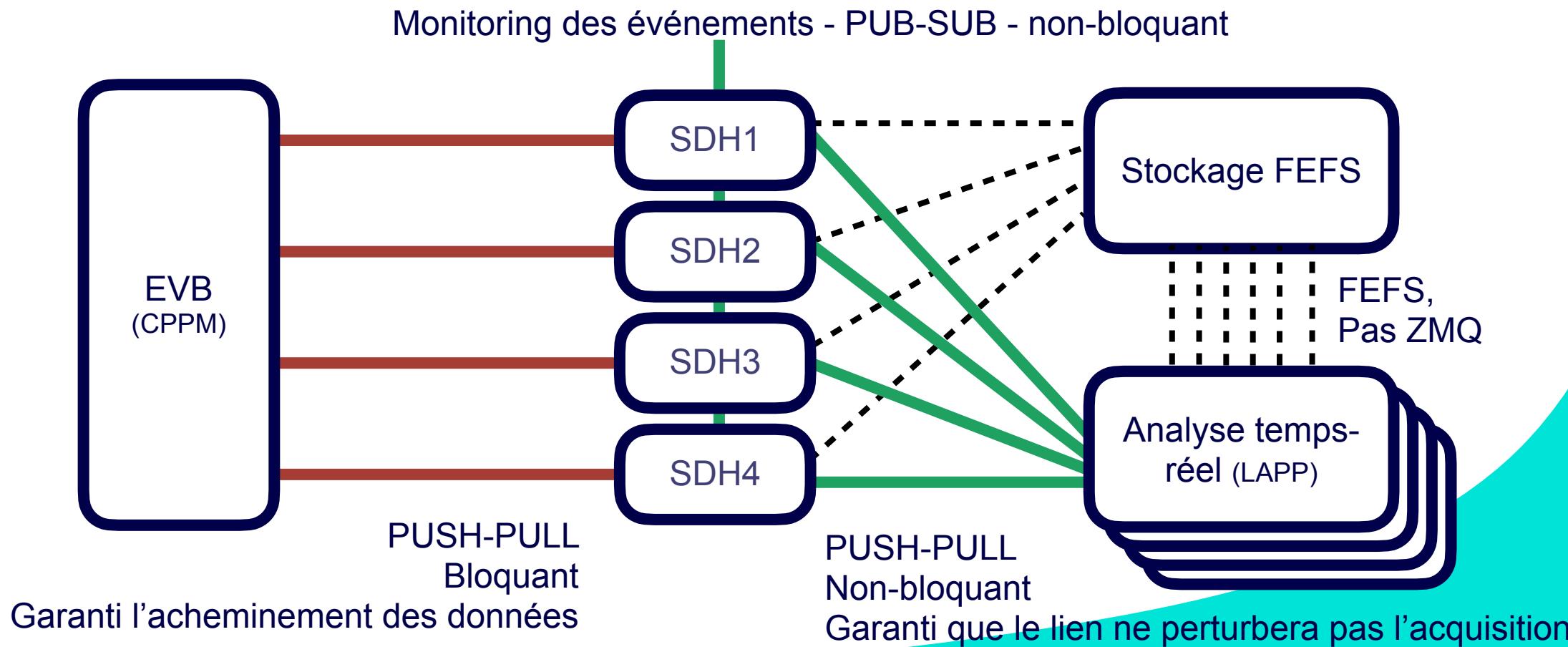


# ZeroMQ

- Zero Message Queue
  - Sur-couche des sockets UNIX
  - Simplifie grandement leur utilisation
  - Implémenté dans different languages
  - Largement utilisé: CERN, Thales, Lockheed...
- Utilisé uniquement pour le transfert des données brutes O(10Gbps)
  - Les échanges de control ou de monitoring utilisent ACS (Alma Common Software)
- De nombreux clones plus ou moins similaires existent

# ZeroMQ

## Utilisation avec l'assembleur d'évènements de LST1

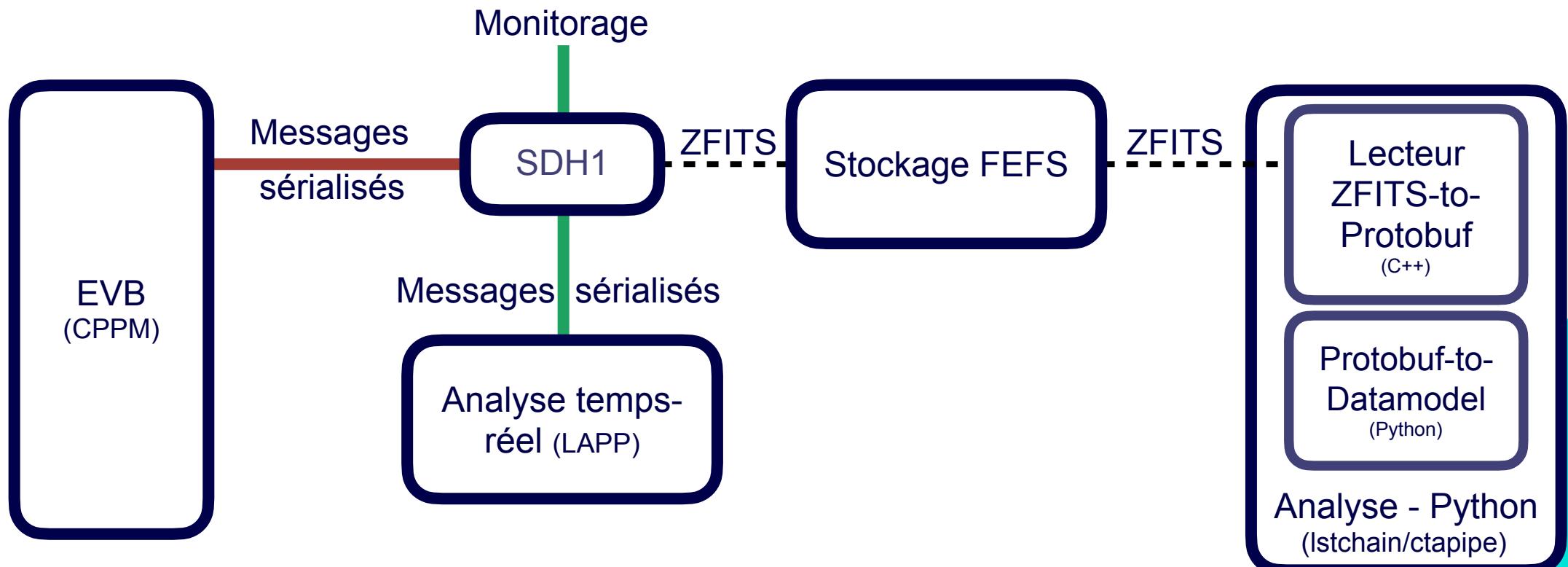


# Protocol Buffers

- De Google
  - Serialisation des données à partir d'un modèle
  - Permet d'avoir des champs optionnels (v2) ou avec valeur par défaut (v3)
  - Permet d'avoir plusieurs versions du modèle qui cohabitent
  - Implémenté dans différents langages
  - Largement utilisé: Google
- Utilisé uniquement pour le transfert des données brutes O(10Gbps)
  - Les échanges de contrôle ou de monitoring utilisent ACS (Alma Common Software)
  - La gestion des tableaux a du être contournée car pas assez performante
- De nombreux clones plus ou moins similaires existent

# Protocol Buffers

## Utilisation dans la chaîne d'acquisition de LST1



- 
- 1 Introduction - CTAO
  - 2 ACADA-ADH
  - 3 Librairies standard
  - 4 **Implémentations maison**
  - 5 Conclusion

# Utilisation dans ADH

- Utiliser des implémentations maison si...
  - Elles répondent à des besoins spécifiques
  - Elles améliorent les performances de manière déterminante
  - ADH utilise un **module d'écriture de ZFITS** fait maison pour économiser les ressources informatiques
    - Les données sont compressées à la volée
    - Économise une passe de lecteur/écriture de/vers le stockage
    - Utilise des algorithmes plus adaptés à nos données que gzip
      - Pré-traitement + Huffman 16-bits
    - Gère les tailles variables de tableaux
      - Gain d'espace estimé à environ 20%
    - Gain de performance estimé à environ un facteur 10

# ZFITS

- Standard pour compresser des données de manière native dans le format FITS
  - Par opposition à la coutume de gzipper les fichiers FITS
- Apporte un gain de performance car les données sont compressées par colonne
- Support limité dans la librairie CFITSIO
  - nécessité d'utiliser l'outil fpack pour compresser ou décompresser des fichiers



## CFITSIO and Data Compression

CFITSIO supports data compression on 2 different levels:

- External file compression
- Internal tiled-image compression

The [imcopy](#) example program that is included in the CFITSIO distribution and is also available in the [FITS tools](#) suite of utility programs can be used to compress any existing FITS image into the tile-compressed format, or to uncompress a tile-compressed image into a normal FITS image. This program can be used to experiment with the various compression options on existing images.

### External File Compression

In the first type of data compression, the entire FITS file may be externally compressed with the gzip or Unix compress algorithm, producing a \*.gz or \*.Z file, respectively. When reading compressed files of this type, CFITSIO first uncompresses the entire file into memory before performing the requested read operations. Output files can be directly written in the gzip compressed format if the user-specified filename ends with '.gz'. In this case, CFITSIO initially writes the uncompressed file in memory and then compresses it and writes it to disk when the FITS file is closed, thus saving user disk space. Read and write access to these compressed FITS files is generally quite fast; the main limitation is that there must be enough available memory (or swap space) to hold the entire uncompressed FITS file.

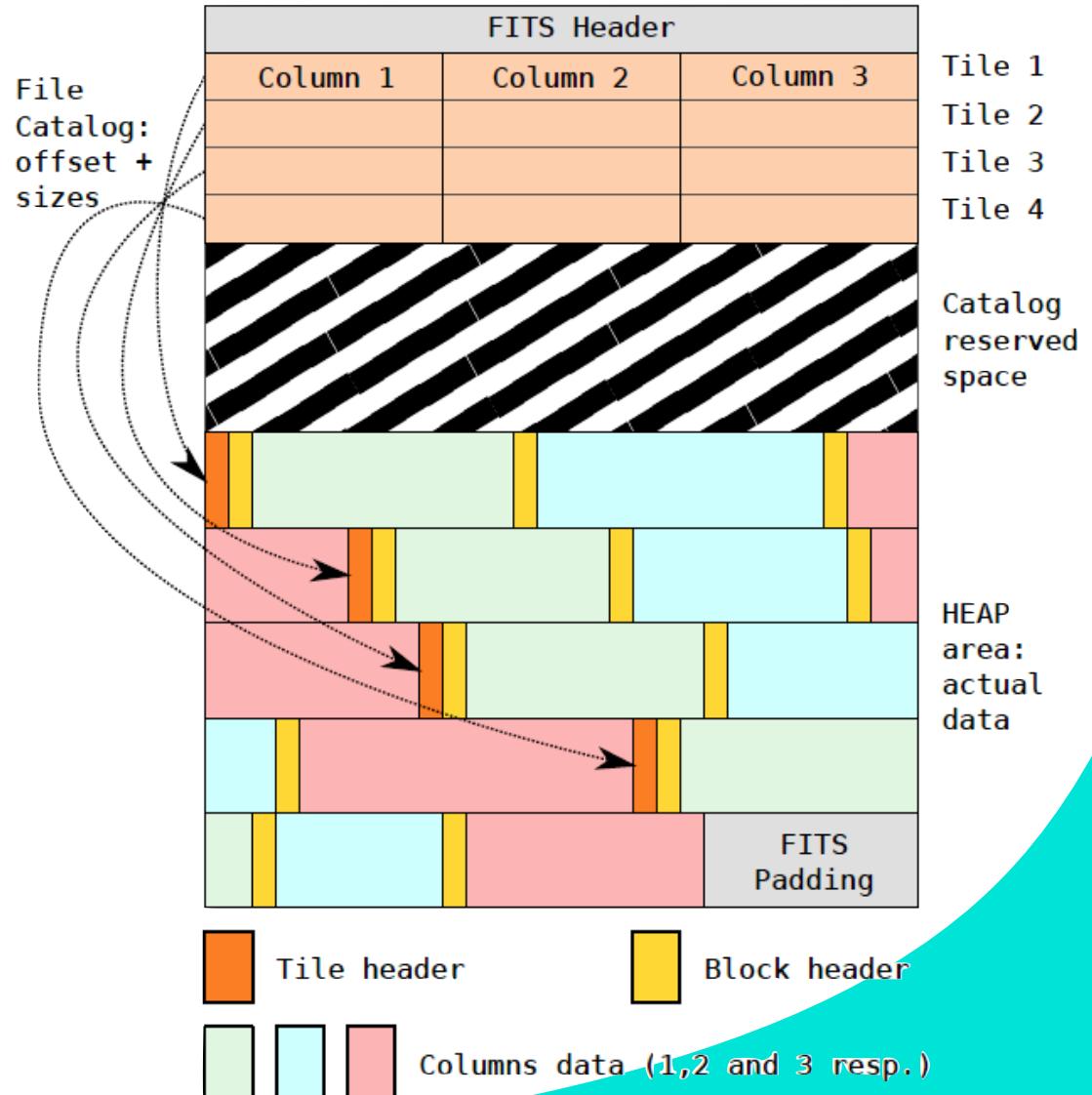
### Internal Tiled-Image Compression

CFITSIO also supports the tiled image compression format in which the image is divided into a grid of rectangular tiles, and each tile of pixels is individually compressed. The compressed tiles are stored in rows of a variable length array column in a FITS binary table, but CFITSIO recognizes that the binary table extension contains an image and treats it as if it were an IMAGE extension. This tile-compressed format is especially well suited for compressing very large images because a) the FITS header keywords remain uncompressed for rapid read access, and because b) it is possible to extract and uncompress sections of the image without having to uncompress the entire image. This format is also much more effective in compressing floating point images (using a lossy compression algorithm) than simply compressing the image using gzip or compress.

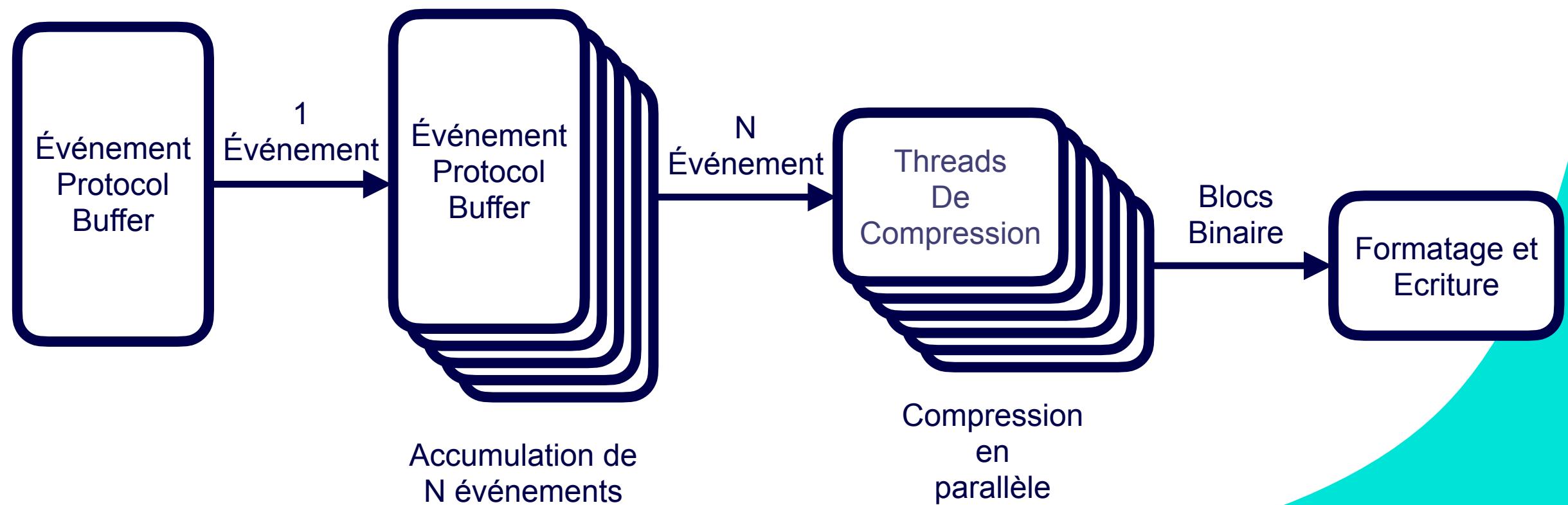
A [detailed description](#) of this compressed image format can be found in the [Registry of FITS Conventions](#).

# ZFITS

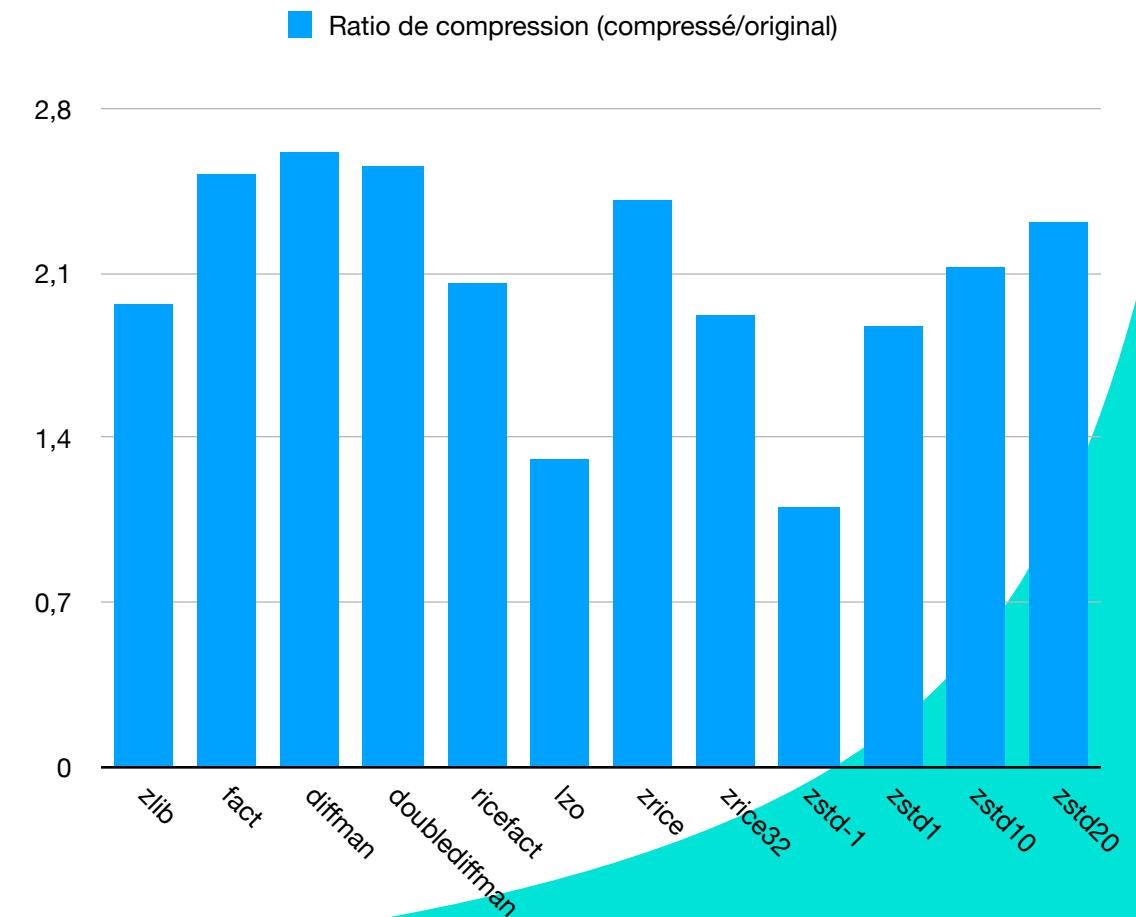
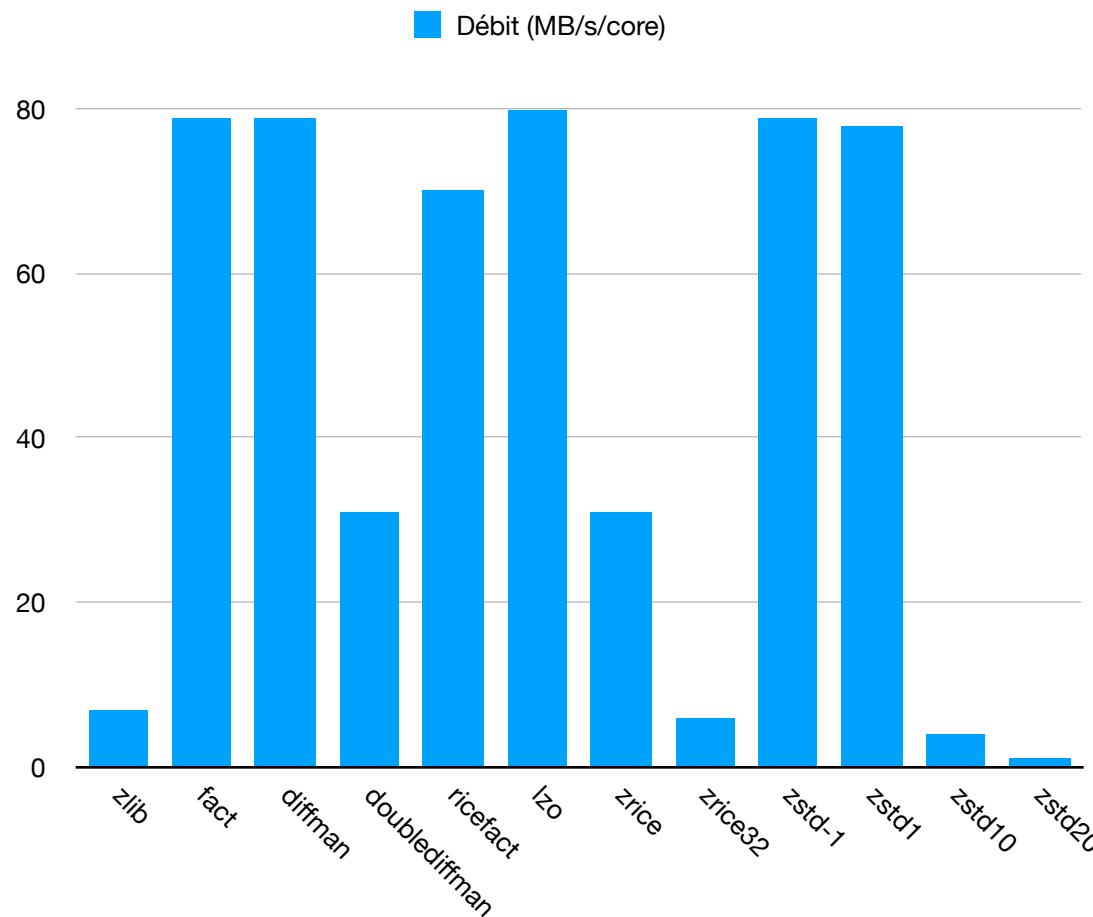
- Les nombre total de lignes doit être connu d'avance
  - Avec un mécanisme de secours qui permet de dépasser cette limite
- Les données des colonnes sont déplacées vers la zone HEAP, après le catalogue
- On insère des marqueurs en plus
  - Tile/Block header pour pouvoir reconstruire les données sans catalogue
  - Des métadonnées pour reconstruire les objets protocol buffer



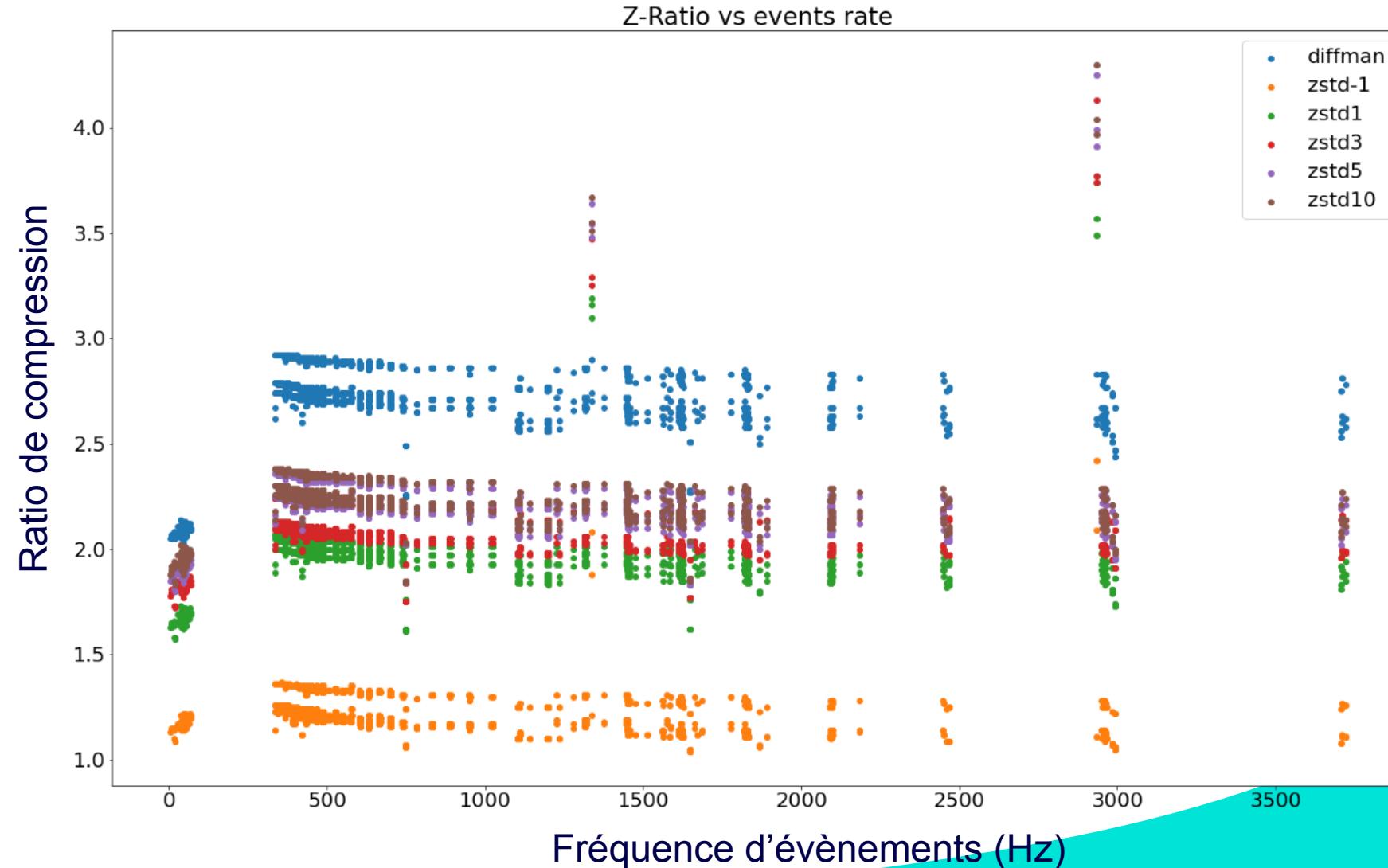
# Le ZFITSWriter



# Tests de performance

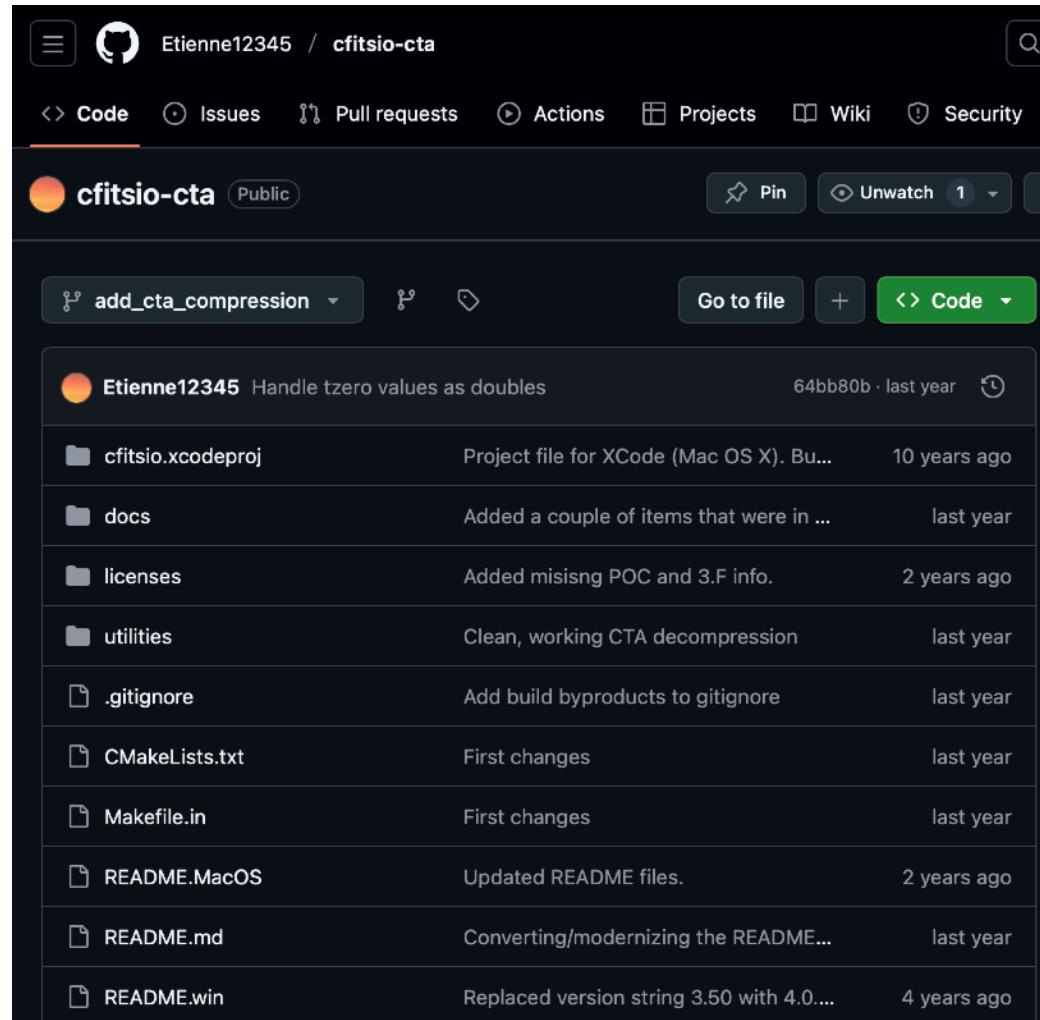


# Performance vs fréquence d'événements



# Intégration dans CFITSIO

- Fork de CFITSIO dans GitHub
- Implémentation de la décompression uniquement
- Discussions avec le comité FITS en attente



- 
- 1 Introduction - CTAO
  - 2 ACADA-ADH
  - 3 Librairies standard
  - 4 Implémentations maison
  - 5 Conclusion

# Conclusion

- CTAO est en phase de construction
- Le développement logiciel est en cours
  - En collaboration étroite avec les acteurs du CTAO consortium
- Les choix fait jusqu'à présent sont adaptés à nos usages
  - Utilisation de librairies standard si possible
  - Utilisation de librairies maison si nécessaire
- Merci pour votre attention !
  - Questions ?

