

Les HiPS 3D dans Aladin Lite

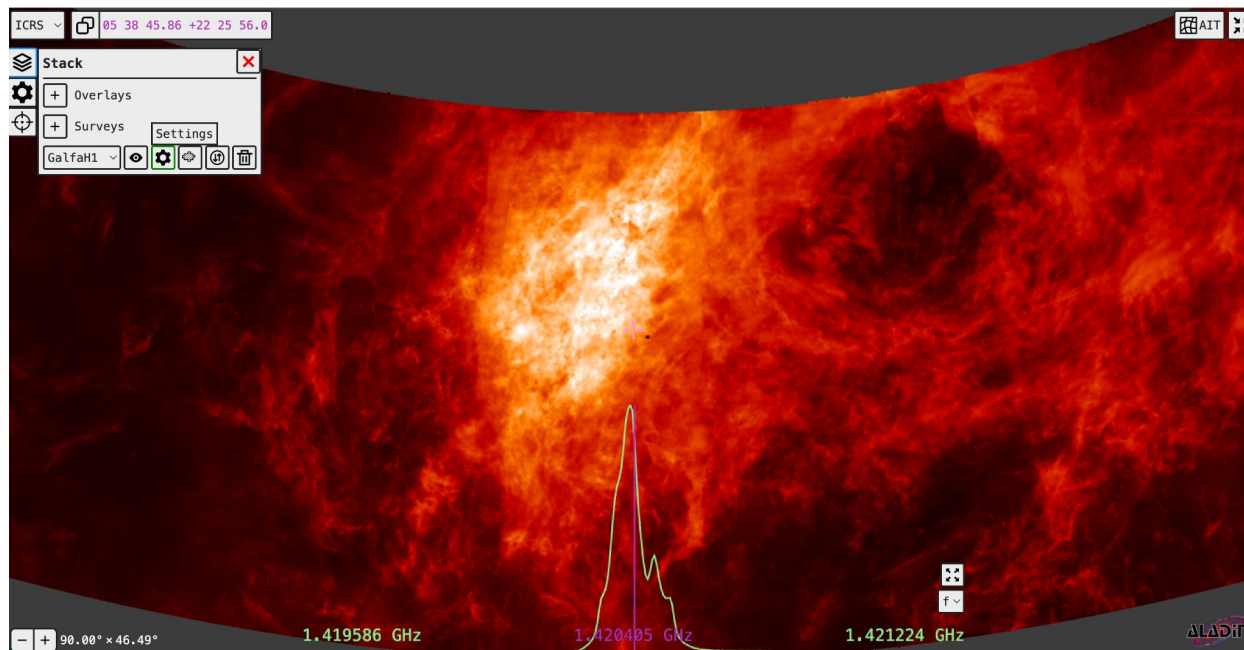
Matthieu Baumann

Ingénieur de Recherche CNRS
Centre de Données astronomique de Strasbourg



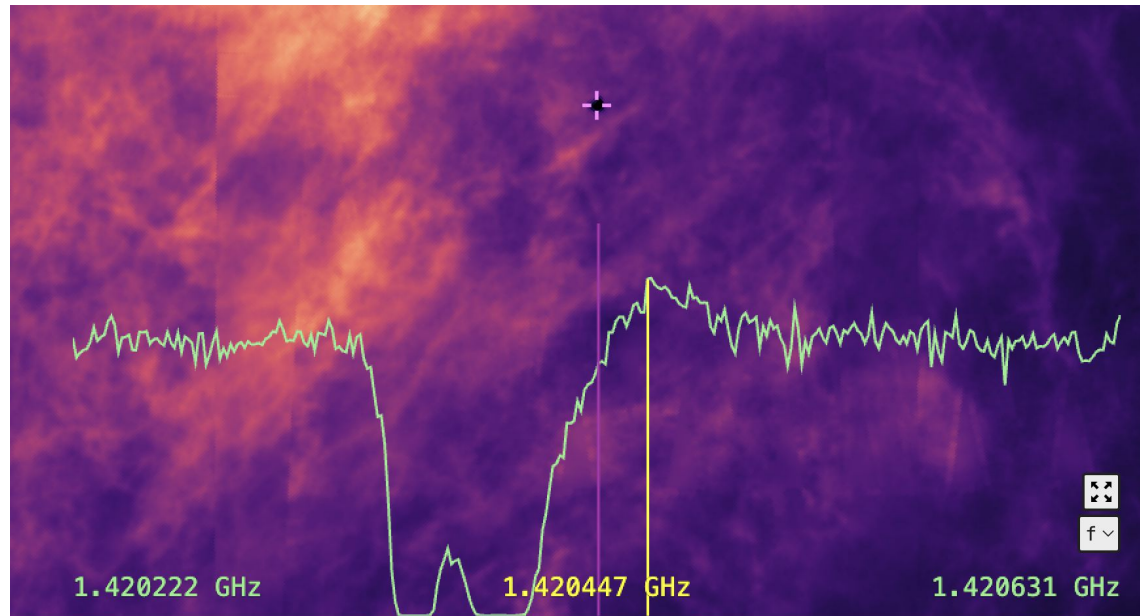
Les fonctionnalités actuelles de l'implémentation des HiPS3D dans Aladin Lite

- Implémentation vers un support HiPS 2.0 dans Aladin Lite
 - Disponible depuis la version ≥ 3.7 (sortie le 12 septembre 2025)
 - Le support des HiPS3D
 - i. Tuiles (FITS, WEBP, PNG, JPG/JPEG), FITS RICE en cours
 - ii. Un widget javascript d'affichage d'une fenêtre spectrale autour du curseur
 - iii. Une API JS pour manipuler des HiPS3D (objet HiPS3D)



Un nouvel outil d'affichage de spectre

- Permet l'affichage d'une fenêtre spectrale sous la position du curseur central
- Plusieurs unités disponibles:
 - Fréquence (Hz)
 - Longueur d'onde (m)
 - Vitesse (km/s) si **obs_rest_freq** se trouve dans le /properties du HiPS3D
- Bouton pour rescale le spectre
- Navigation dans la dimension spectrale par:
 - glisser (drag) de la ligne centrale (violet)
 - click sous le spectre



Présentation de l'API JS autour des HiPS3D

- La création d'un HiPS3D (similaire aux HiPS image traditionnels)

```
let hips3d = aladin.setBaseImageLayer("https://alasky.cds.unistra.fr/HIPS3D/GalfaHI")
```

- Les méthodes disponibles sur l'objet HiPS3D
 - GET la fréquence observée: **getFrequency**
 - GET la fenêtre de fréquence observée: **getFrequencyWindow**
 - SET de la fréquence à observer

```
hips3d.setFrequency({  
  value: 1420302592,  
  unit: 'Hz'  
})
```

Name	Type	Attributes	Default	Description
value	number	<optional>		= The frequency value expressed in `options.unit`
unit	"Hz" "m" "m/s"	<optional>	"Hz"	The unit of the frequency passed
restFreq	number	<optional>		"The rest frequency (in Hz) to use for computing the velocity in m.s-1"

Démonstration

Exploration du relevé cubique GALFA H1

- Accès à l'exemple

<https://aladin.cds.unistra.fr/AladinLite/HiPS3D-testing/al-hips3d-galfah1.html>

- Affichage du spectre dynamique sous le curseur central
- Observation à différentes fréquences
- Raffinement du spectre en zoomant spatialement

Démonstration

Exploration de M31 dans le survey radio LGLBSHI: The Local Group L-Band Survey

- Accès à l'exemple:
<https://aladin.cds.unistra.fr/AladinLite/HiPS3D-testing/al-hips3d-m31.html>
- Détermination de la vitesse radiale moyenne de M31

Démonstration

Identification de raies spectrales pour un cube MUSE autour de NGC 5806:

- Url de test

<https://aladin.cds.unistra.fr/AladinLite/HiPS3D-testing/al-hips3d-muse.html>

- Détermination de raies d'observation (**λ_{obs}**)
- Détermination de **z** de NGC 5806
- Quels sont les raies d'émission ?

Formule:

$$\lambda_{\text{emit}} = \lambda_{\text{obs}} / (1+z)$$

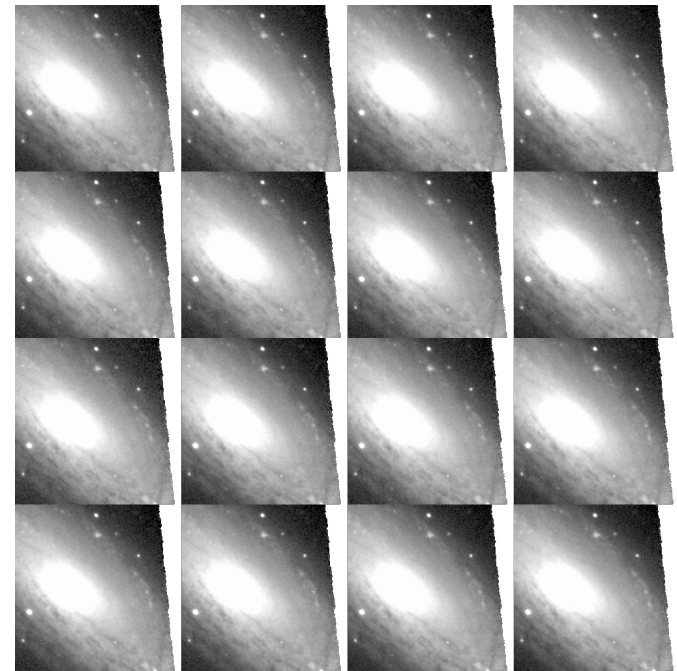
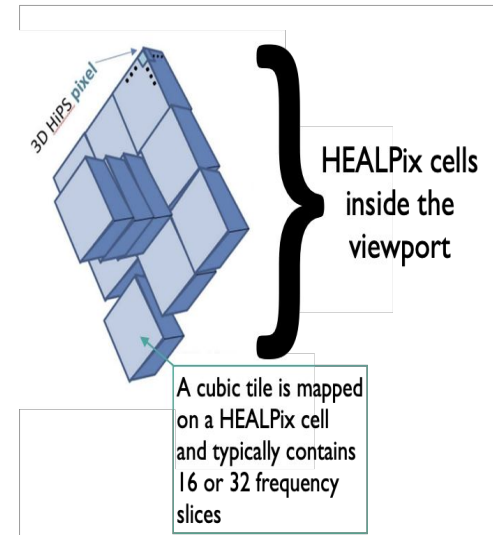
Avec:

- λ_{emit} = longueur d'onde émise (au repos)
- λ_{obs} = longueur d'onde observée
- z = redshift

4504.450	0.0	0.0	O III
4862.68	2.0	4.0	H β
4932.603	0.0	0.0	O III
4960.295	2.0	2.0	O III
5008.240	3.0	2.0	O III
6302.046	0.0	0.0	O I
6365.536	0.0	0.0	O I
6529.03	0.0	0.0	N I
6549.86	3.0	0.0	N II
6564.61	8.0	8.0	H α
6585.27	3.0	0.0	N II
6718.29	3.0	0.0	S II
6732.67	3.0	0.0	S II

Détails de mise en oeuvre

- Lecture du SFMOC du HiPS3D (crate [moc](#) >= 0.18.0)
- ~10 tuiles téléchargées (dans la vue spatiale et autour d'une fréquence)
- Une tuile HiPS3D est stockée dans un objet WebGL2: Texture3D
 - FITS: le cube est directement obtenu des data
 - PNG/WebP: le cube est reconstruit à partir d'une image 2D
 1. reconstruction dans un worker (sinon blocage de ~100ms/cube sur les basses configs)
 2. transfert du cube à la mémoire WASM (cf. crate [wasm_bindgen_futures](#))
 3. création de la texture 3D
 4. envoi au GPU
- Affichage tuile par tuile dans la vue (shader exécuté 10x pour les ~10 cellules dans la vue)



Tuile HiPS3D (16 slices) au format PNG/WebP

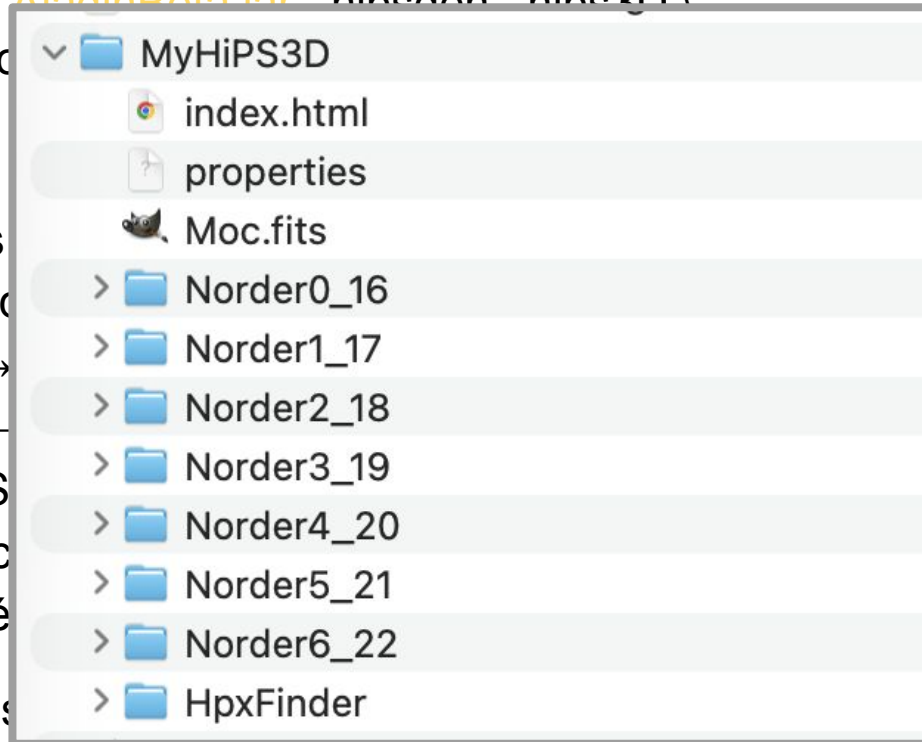
Créer un HiPS3D avec HiPSGen et visualiser dans Aladin Lite

Exemple de commande:

```
java -Xmx2G -jar AladinBeta.jar hpsgen -hips3D \
in=YourCube.fits
```

Les options:

- YourCube.fits : nom du cube de données spatiale & fréquentielle
- MyHiPS3D → nom du HiPS3D
- AUTH/C/xxx → nom de l'auteur
- INDEX TILES → nom des fichiers index
- -hips3D → Activation du HiPS3D
- -Xmx2G → mémoire maximale



PNG

s) avec une calibration

duit

es FITS et PNG

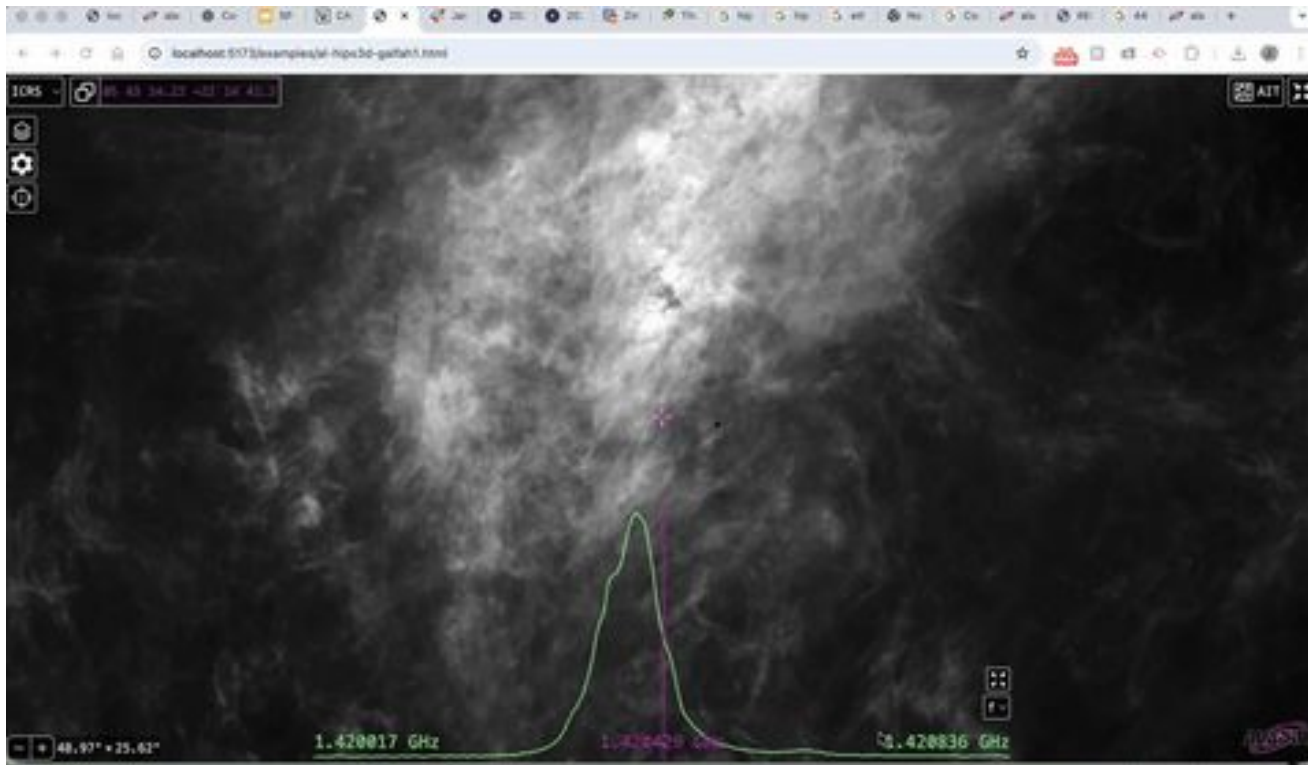
rtant!)

Pour plus d'options

Un index.html est généré à la racine du HiPS permettant de visualiser le HiPS dans Aladin Lite

Les pistes de développement futurs:

- (Dé)zoomer dans le spectre quelque que soit le niveau de zoom spatial.
- Accès au service d'extraction de l'entièreté du spectre pour une position (cf. présentation R&D Thomas)
- Support des tuiles FITS compressées RICE (worker ?)
- Exploration de rendu 3D (cf. fits3) d'un HiPS3D
- Affichage du spectre moyenné dans une région autour du curseur.
- Rendu colorimétrique par combinaison de plusieurs slices



Questions ?

Contacts:

- GitHub: <https://github.com/cds-astro/aladin-lite>
- email: cds-question@astro.unistra.fr

