

L'extension `fits-astro`

v1.0.1

Valentin Dao*

9 juillet 2026

L'extension `fits-astro` offre une interface simple permettant de lire des fichiers *Flexible Image Transport System* (FITS) et d'accéder à leur en-tête afin d'y récupérer des informations. Elle dépend d'un script Lua personnalisé et requiert donc le moteur Lua¹LaTeX. Contrairement à des bibliothèques plus complexes comme `astropy.io.fits` en Python, cette extension est uniquement conçue pour récupérer directement les paires clé-valeur des en-têtes. Elle n'est *pas* destinée à les modifier ou à corriger les erreurs de standardisation, et encore moins à lire la partie des données. Veuillez également noter qu'à l'heure actuelle, le paquet ne *prend pas* en charge les fichiers comportant plusieurs *Header Data Units* (HDUs), et seul l'en-tête principal sera lu; c.-à-d. que les extensions seront ignorées.

1 Importer le paquet

Le paquet peut être importé de l'une des manières suivantes :

- `\usepackage{fits-astro}` si compilation avec Lua²LaTeX.
- `\input{fits-astro.tex}` si compilation avec LuaTeX.

2 Macros du paquet

`\loadheader` `\loadheader{<identification FITS>}{<chemin du fichier FITS>}`

Cette macro lit le fichier FITS et alloue l'en-tête à l'identification FITS spécifiée, qui peut prendre la forme d'un *string* ou d'une séquence de contrôle. Dans ce dernier cas, `\csstring` est utilisée pour retirer le caractère d'échappement pour le stockage interne. De ce fait, si une séquence de contrôle est utilisée, elle ne contient en réalité pas l'en-tête. Ainsi, les deux formats d'argument sont interchangeables. La macro utilise également la bibliothèque `Kpathsea`, et il n'est donc pas nécessaire que les fichiers se trouvent dans le répertoire courant. Cette documentation utilisera une **image publique** pour démontrer les macros du paquet (l'image utilisée est la première dans le tableau).

```
\loadheader\HST{WFPC2u5780205r_c0fx.fits}
\loadheader{HST}{WFPC2u5780205r_c0fx.fits} % Façon alternative de charger le
fichier
```

*E-mail: vdao.texdev@gmail.com

`\showheader` `\showheader {<identification FITS>}`

Cette macro affiche le contenu de l'en-tête de l'identification FITS dans le fichier log.

```
\showheader\HST
\showheader{HST} % Produirait le même résultat
```

`\headerkey` `\headerkey {<identification FITS>} {<clé>}`

Cette macro récupère la valeur associée à la clé désignée dans l'en-tête de l'identification FITS donnée.

```
\headerkey\HST{FITSDATE}\par      2004-01-09
\headerkey\HST{TELESCOP}\par      HST
```

`\setheaderkey` `\setheaderkey {<FITS identification>} {<key>} {<control sequence>}`

`\gsetheaderkey`

Cette macro récupère la valeur associée à la clé désignée dans l'en-tête de l'identification FITS donnée et l'assigne à la séquence de contrôle spécifiée. L'assignation est soit locale au groupe courant, soit globale.

```
\ttfamily
\begingroup
\setheaderkey\HST{RA_TARG}\ra      macro:->182.6355
\setheaderkey\HST{DEC_TARG}\dec    macro:->39.40576666667
\gsetheaderkey\HST{EXPTIME}\exptime macro:->300.0
\meaning\ra\par                   macro:->300.0
\meaning\dec\par                   undefined
\meaning\exptime\par               undefined
\endgroup
\meaning\exptime\par
\meaning\ra\par
\meaning\dec
```

Historique des modifications

1.0.1 (09-07-2026)

Ajouté

- Documentation française.

Corrigé

- Amélioration du script Lua et de la documentation.

1.0.0 (06-04-2026) – Version initiale.

3 Implémentation

3.1 Package T_EX

```
1 <*package-tex>
2 \ifdefined\loadheader\expandafter\endinput\fi
3
4 \def\FITSerr#1{\errmessage{Package fits Error : #1}}
5 \def\FITSwar#1{\message{Package fits Warning : #1}}
6
7 \unless\ifdefined\directlua\FITSerr{LuaTeX required}\fi
8
9 \directlua{dofile('fits-astro.lua')}
10
11 \def\loadheader#1#2{
12   \directlua{
13     local file = '\csstring#1'
14     if FITS_data[file] then
15       tex.print('\luaescapestring{\FITSerr{FITS file '#2' already loaded}}')
16     else
17       FITS_data[file] = load_FITS_header('#2')
18     end
19   }
20 }
21
22 \def\showheader#1{
23   \directlua{
24     show_fits(FITS_data['\csstring#1'])
25   }
26 }
27
28 \def\headerkey#1#2{
29   \directlua{
30     if FITS_data['\csstring#1']['#2'] then
31       get_header_key(FITS_data['\csstring#1'], '#2')
32     else
33       tex.print('\luaescapestring{\FITSwar{Header key '#2' not found}}')
34     end
35   }
36 }
37
38 \def\setheaderkey#1#2#3{
39   \directlua{
40     if FITS_data['\csstring#1']['#2'] then
41       token.set_macro('\csstring#3', FITS_data['\csstring#1']['#2'])
42     end
43   }
44 }
45
46 \def\gsetheaderkey#1#2#3{
47   \directlua{
48     if FITS_data['\csstring#1']['#2'] then
49       token.set_macro('\csstring#3', FITS_data['\csstring#1']['#2'], 'global')
50     end
51   }
52 }
53
```

```
54 </package-tex>
```

3.2 Package \LaTeX

```
1 <*package-latex>
2 \NeedsTeXFormat{LaTeX2e}
3
4 \def\fitsastro@module{fits-astro}
5 \def\fitsastro@version{v1.0.1}
6 \def\fitsastro@date{2026-07-09}
7 \def\fitsastro@description{Retrieving FITS file header information in LuaTeX}
8
9 \ProvidesPackage{fits-astro}[\fitsastro@date\space\fitsastro@version\space\fitsastro@des
10
11 \input{fits-astro.tex}
12 </package-latex>
```

3.3 Script Lua

```
1 <*package-lua>
2 tex = tex or {}
3 texio = texio or {}
4 FITS_data = {}
5
6 function load_FITS_header(file)
7   local file_path = kpse.find_file(file)
8   if not file_path then
9     tex.error("FITS file not found : " .. file)
10  else
11    local FITS_file = io.open(file_path, "rb")
12    local header = {}
13    repeat
14      local header_line = FITS_file :read(80)
15      if not header_line or #header_line < 80 then break end
16
17      -- string
18      local key, value = header_line :match("([%u%-%d_]+)%s*=%s*'([^\']*)*'")
19      if not key then
20        -- number or boolean
21        key, value = header_line :match("([%u%-%d_]+)%s*=%s*([~/]+)")
22      end
23
24      if key and value then
25        value = sanitize_FITS_key(value)
26        if value :match("^%s*$") then
27          header[key] = nil
28        else
29          header[key] = parse_FITS_key(key, value)
30        end
31      end
32    until header_line :sub(1, 3) == "END"
33    FITS_file :close()
34    return header
35  end
36 end
37
38 function parse_FITS_key(key, value)
39   -- converts FITS standard status to a boolean
40   if key == "SIMPLE" then
```

```

41     if value == "T" then
42         value = true
43     else
44         value = false
45     end
46     return value
47 end
48 -- converts numbers into proper integers
49 local num = tonumber(value)
50 if num then value = num end
51 return value
52 end
53
54 function sanitize_FITS_key(value)
55     value = value :match("^%s*(.)%s*$") or value
56     return value
57 end
58
59 function get_header_key(header, key)
60     local value = header[key]
61     if value then
62         tex.print(value)
63     end
64 end
65
66 function show_fits(header)
67     texio.write_nl("FITS header information :")
68     for key, value in pairs(header) do
69         texio.write_nl(string.format("%s = %s", key, tostring(value)))
70     end
71 end
72 </package-lua>

```

Index

The *italic* numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

C		L	
<code>\csstring</code> . . .	13, 24, 30, 31, 40, 41, 48, 49	<code>\loadheader</code>	2, 11
		<code>\luaescapestring</code>	15, 33
D		M	
<code>\def</code>	4, 4, 5, 5, 6, 7, 11, 22, 28, 38, 46	<code>\message</code>	5
<code>\directlua</code>	7, 9, 12, 23, 29, 39, 47		
E		N	
<code>\endinput</code>	2	<code>\NeedsTeXFormat</code>	2
<code>\errmessage</code>	4		
<code>\expandafter</code>	2	P	
		<code>\ProvidesPackage</code>	9
F		S	
<code>\fi</code>	2, 7	<code>\setheaderkey</code>	38
<code>\FITSErr</code>	4, 7, 15	<code>\showheader</code>	22
<code>\FITSwar</code>	5, 33	<code>\space</code>	9
G		T	
<code>\gsetheaderkey</code>	46	TeX and L ^A T _E X 2 _ε commands:	
H		<code>\fitsastro@date</code>	6, 9
<code>\headerkey</code>	28	<code>\fitsastro@description</code>	7, 9
		<code>\fitsastro@module</code>	4
I		<code>\fitsastro@version</code>	5, 9
<code>\ifdefined</code>	2, 7		
<code>\input</code>	11	U	
		<code>\unless</code>	7