天文科学数据基础

何勃亮

国家天文台 国家天文科学数据中心

2020-12-8

National Astronomical Data Center
国家天文科学数据中心

提 纲

- 天文数据源流
- 普适图像传输系统标准（FITS）
- 虚拟天文台表格 （VOTable）
- 文本型数据（TXT、CSV等）

天文数据源流

史記天官書　原卷二十七

〖索隱〗案：天文有五官。官者，星官也。星座有尊卑，若人之官曹列位，故曰天官。〖正義〗張衡云：「文曜麗乎天，其動者有七，日月五星是也。日者，陽精之宗；月者，陰精之宗；五星，五行之精。衆星列布，體生於地，精成於天，列居錯峙，各有所屬，在野象物，在朝象官，在人象事。其以神著有五列焉，是有三十五名：一居中央，謂之北斗；四布於方各七，爲二十八舍；日月運行，曆示吉凶也。」

中宮㈠天極星，㈡其一明者，太一常居也；㈢旁三星三公，㈣或曰子屬。後句四星，㈤末大星正妃，㈥餘三星後宮之屬也。環之匡衛十二星，藩臣。皆曰紫宮。㈦

㈠〖索隱〗姚氏案：春秋元命包云「官之爲言宣也，宣氣立精爲神垣」。又文耀鉤曰「中宮大帝，其精北極星。含元出氣，流精生一也」。

㈡〖索隱〗案：爾雅「北極謂之北辰」。又春秋合誠圖云「北辰，其星五，在紫微中」。楊泉物理論云「北極，天之中，陽氣之北極也。極南爲太陽，極北爲太陰。日、月、五星行太陰則無光，行太陽則能照，故爲昏明寒暑之限極也」。

㈢〖索隱〗案：春秋合誠圖云「紫微，大帝室，太一之精也」。〖正義〗泰一，天帝之別名也。劉伯莊云「泰一，天神之最尊貴者也。」

唐開元占經卷六十五

石氏中官占上

唐 瞿曇悉達 撰

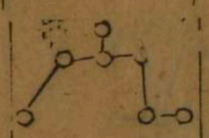
攝提占一

石氏曰攝提六星夾大角入角八度少去北極五十九度半在黃道內三十二度太

一名環樞一名天樞一名闕丘一名致法一名三老一名天鉞一名天獄一名天楯一名天武一名天兵星東

星傾天子不安失位也訣曰火入紫微宮中天下大亂帝王失位

北斗



北斗星謂之七政天之諸侯亦爲帝車魁四星爲璇璣杓三星爲玉衡齊七政斗爲人君號令之主出號施令布政天中臨制四方第一名天樞爲土星主陽德亦曰政星也是太子像星暗若經七日則大災第二名旋主金刑獄女主之位主月及法若星暗經六

欽定四庫全書

靈臺秘苑卷十三　北周　庾季才　原撰

宋　王安禮等　重修

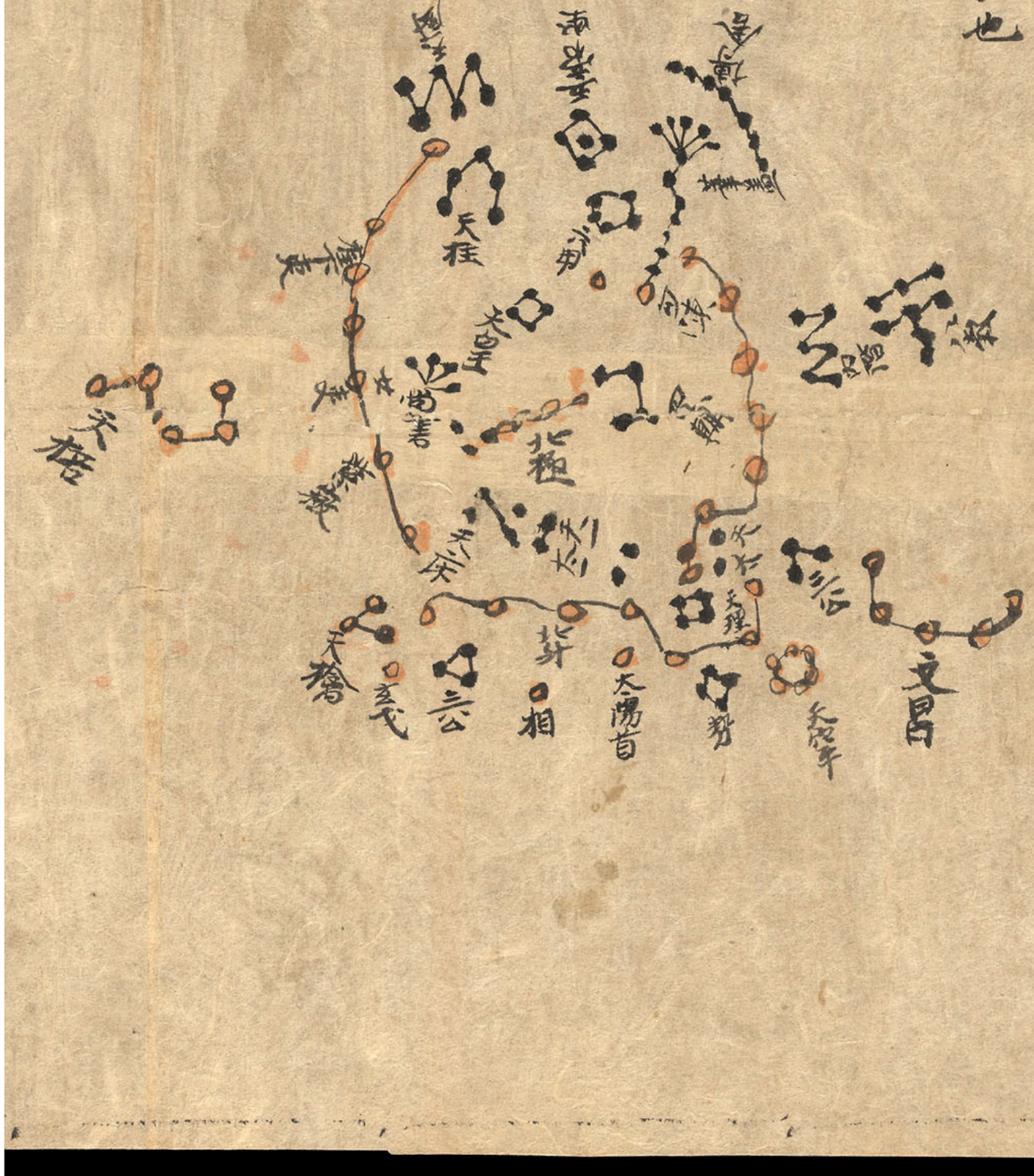
西方七宿 中外官附

奎宿

奎宿距西南大星去極七十二度半　星欲明則吉　若德政虧則星有角　動為兵　其中星明則大水月犯之其分亂邊兵水灾貴人憂民不安

自牛二度析木尽七度於辰在丑為星之紀記者言統已万物之終故曰星紀矣

越之分也



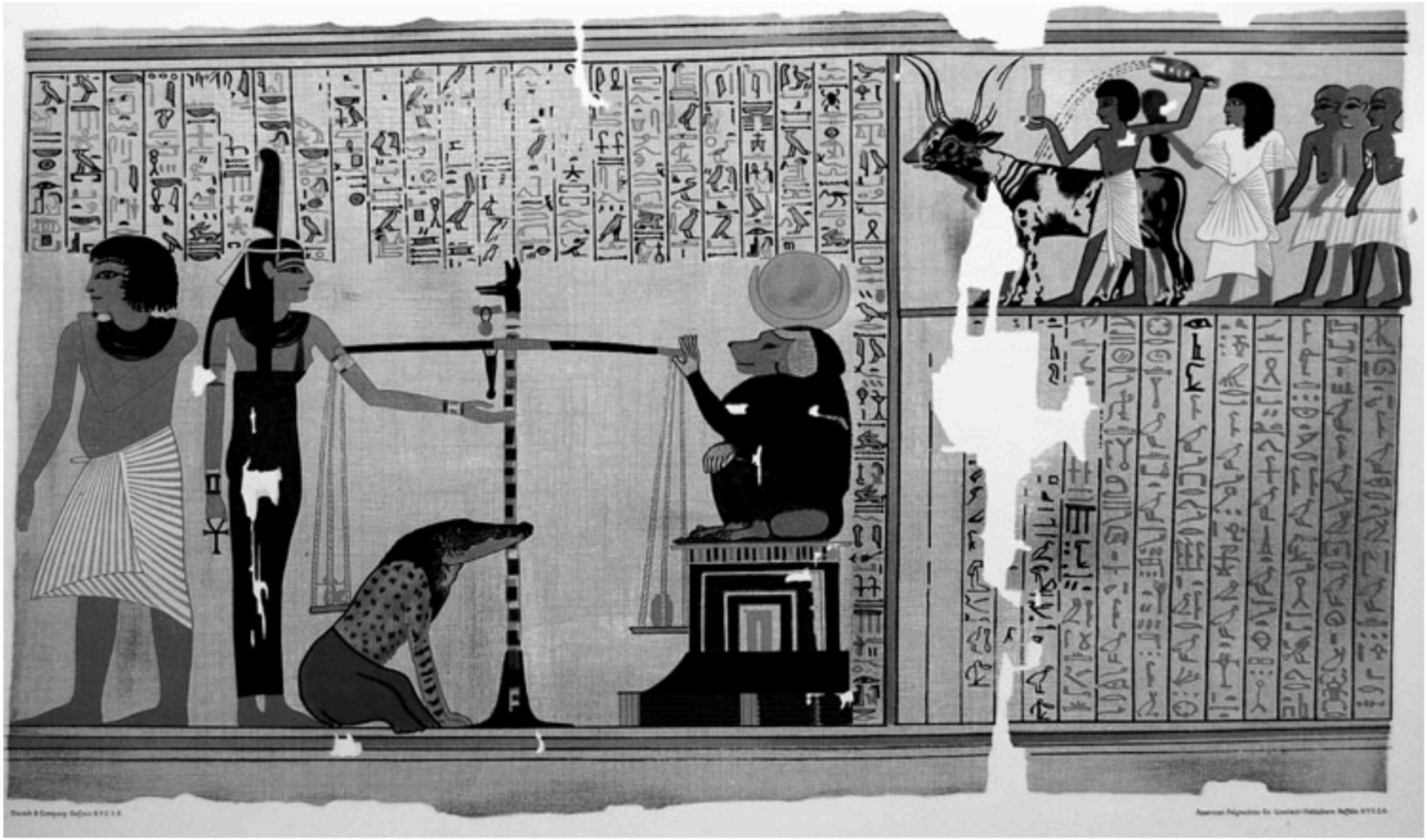
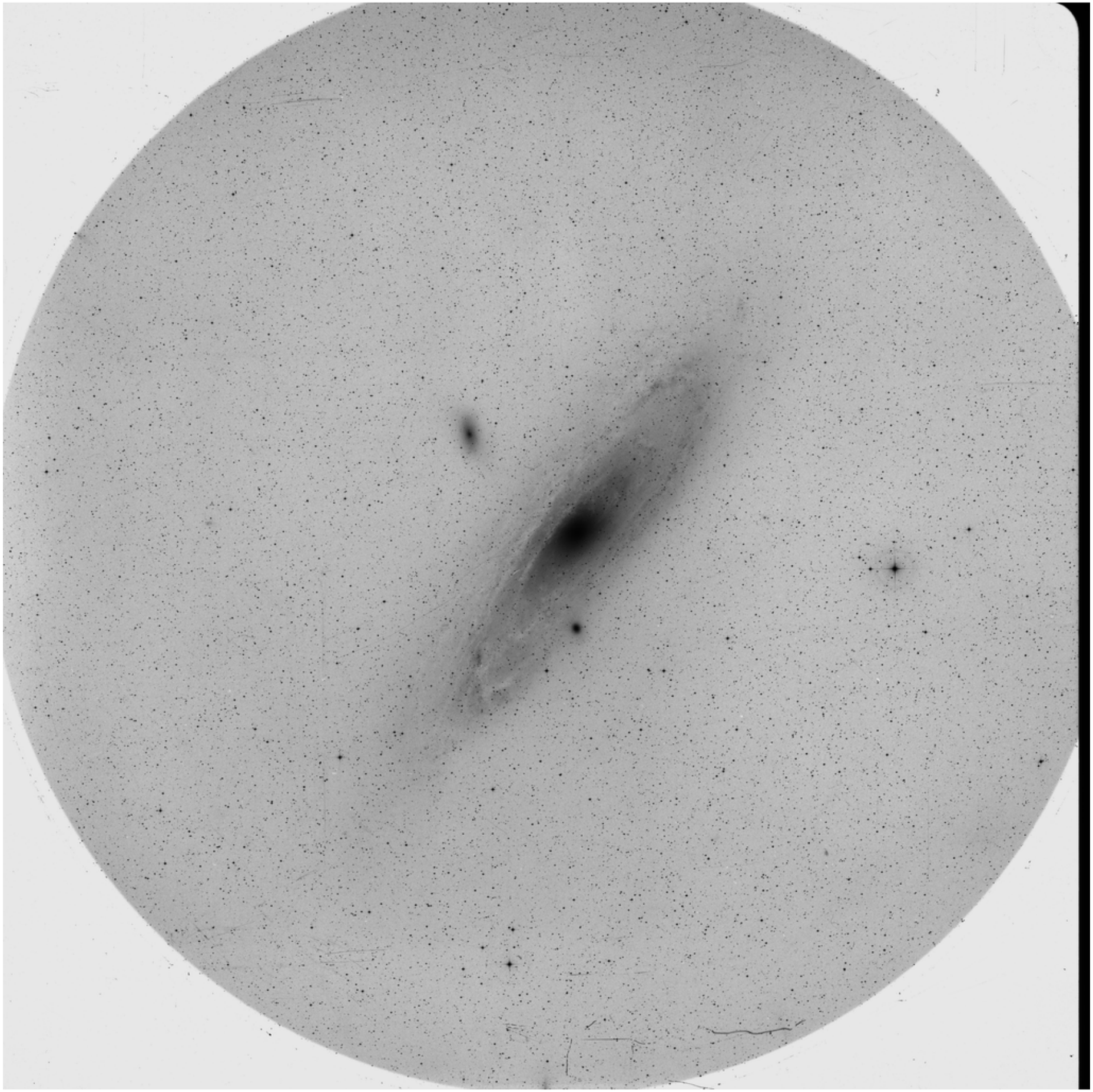


Figure 2.5. Chromolithograph of an Egyptian papyrus "Judgment of the Dead", from Binion's (1887) *Ancient Egypt or Mizraim*. 21 × 43.6 cm. Note the goddess of truth, Maat, presiding over the weighing of the heart of the deceased man on the left. Thoth, the god of wisdom and writing, is ready to record the outcome, and the "devourer" is waiting to destroy the heart if there is an unfavorable outcome. Being a lunar deity, Thoth has a Moon over his head. *See also* Color Section 1.



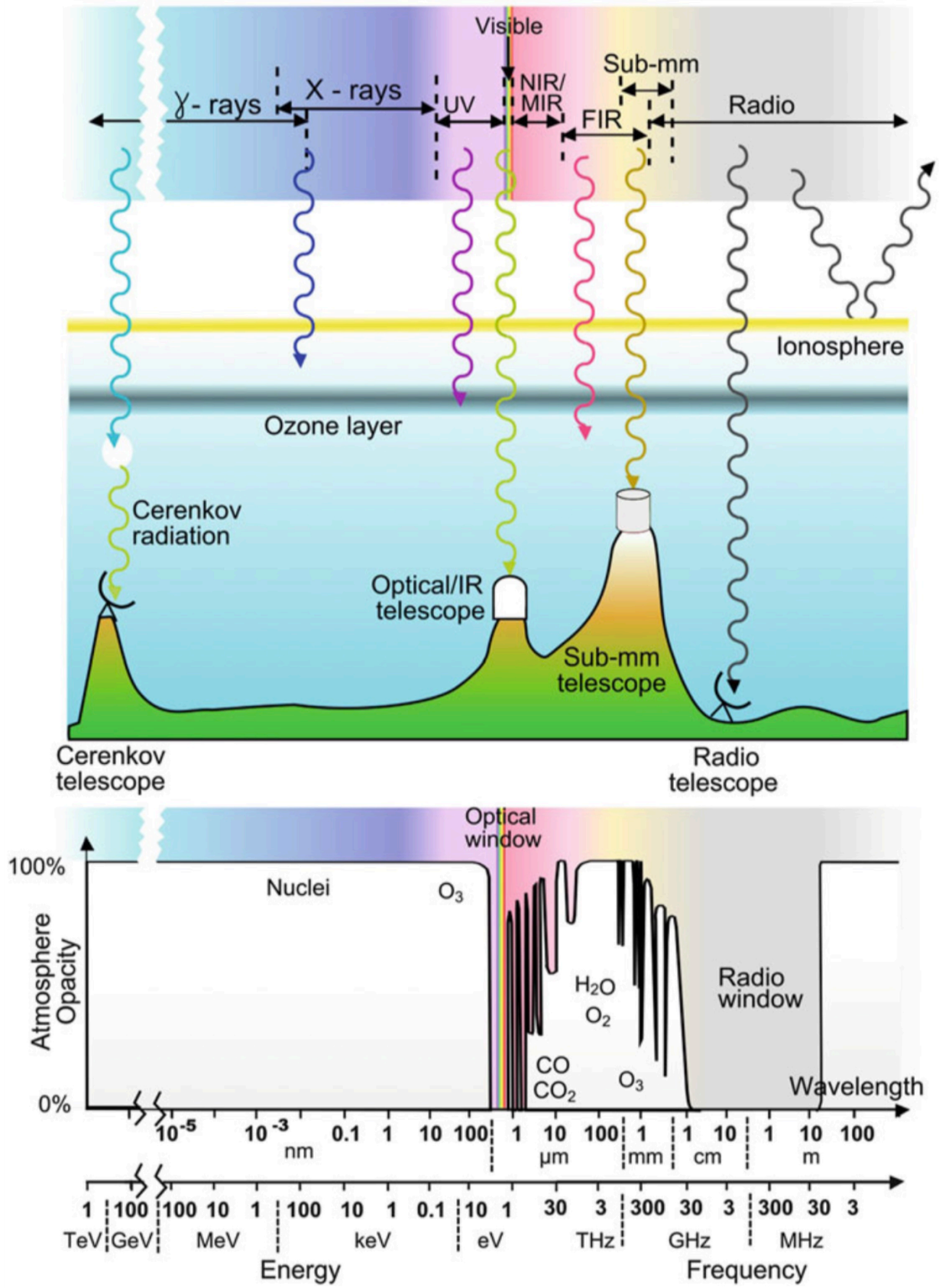
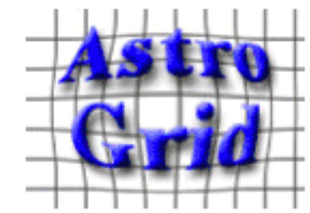


Fig. 2.1 Cartoon indicating absorption by the atmosphere versus wavelength, and the height needed to carry out astronomy at different wavelengths (Figure kindly provided by Dr Adam Woodcraft)


Astro
Grid


ArVO
Armenian Virtual Observatory


Aus-VO


BRAVO


China-VO

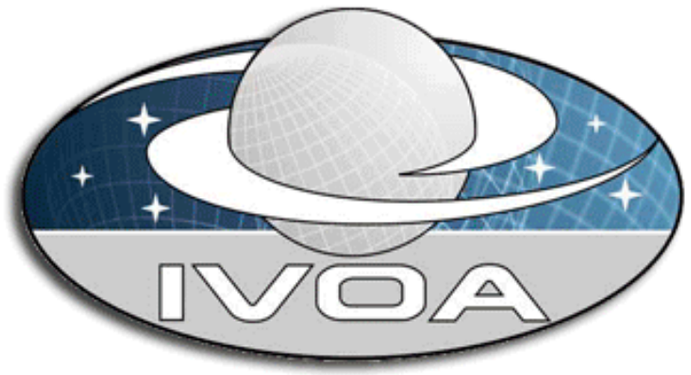

ChiVO
CHILEAN
VIRTUAL
OBSERVATORY


SVO
Spanish Virtual Observatory


VObs.it


esa


CVO


IVOA


USVOA


France


vo-i


EURO VO


UKR
VO


HVO
Hungarian Virtual Observatory


GERMAN ASTROPHYSICAL
GAVO
VIRTUAL OBSERVATORY


JVO
NOVA


RVO


NRF
National Research
Foundation
SA3
South African
Astroinformatics Alliance

Chinese Astronomical Data Center

input coordinate or target name | 30 arcsec | Find Dataset

Sample: 15.51967109 -0.28502335, m31

Domestic Dataset

- AST3 Image
- AST3 Light Curve
- AST3 Survey
- BASS DR1 gradd
- BASS DR1 Images
- BASS DR2 gradd
- BASS DR2 Images
- BASS DR2 Stack
- BASS EDR
- CSTAR
- GMG2.4m
- MMData SLFITS
- SCUSS Catalogue
- SCUSS Image
- SCUSS Proper Motion
- SHAO156 2013
- SHAO156 2014
- SHAO 65M

Solar Dataset

- Huairou SOS
- NVST
- SBRS

Other

- TwoMass
- UCAC
- WISE
- Dragon on Silkroad
- GAIA DR1
- GAIA DR2
- Casjobs
- SDSS DR12
- Vizier
- Historical Plates

Links

IVOA China-VO NAOC CDS NASA/IPAC NED NASA/IPAC IRSA MAST Portal NASA/ADS arXiv.org (Astrophysics)

Certified **Member of**

Portal Simbad VizieR Aladin X-Match Other Help

VizieR

VizieR provides access to the most complete library of published astronomical catalogues and data tables available on line organized in a self-documented database. Query tools allow the user to select relevant data tables and to extract and format records matching given criteria. Currently, 17489 catalogues are available. more info

VO compatibility

Free text search catalogue name, aut

Position position or object na

Go to the classic form Advanced search

VizieR

- How to publish my catalog
- Help and tutorials
- View large catalogs
- Rules of usage
- Mirrors

Oth

News

SDSS

Data Surveys Instruments Collaboration Results Education The Future Contact

This is Data Release 15.

Search www.sdss.org Search

The Sloan Digital Sky Survey: Mapping the Universe

The Sloan Digital Sky Survey has created the most detailed three-dimensional maps of the Universe ever made, with deep multi-color images of one third of the sky, and spectra for more than three million astronomical objects. Learn and explore all phases and surveys—past, present, and future—of the SDSS.

EXPLORE OUR DATA

Go to Data Access

News

SDSS Press Releases

The Sloan Digital Sky Survey names its

普适图像传输系统（FITS）

Flexible Image Transport System

表 1: *FITS* 发展过程中的标志性里程碑

日期	里程碑
1979	Initial *FITS* Agreement and first interchange of files
1981	Published original (single HDU) definition (Wells et al. 1981)
1981	Published random-groups definition (Greisen & Harten 1981)
1982	Formally endorsed by the IAU (IAU 1983)
1988	Defined rules for multiple extensions (Grosbøl et al. 1988)
1988	IAU *FITS* Working Group (IAUFWG) established
1988	Extended to include ASCII-table extensions (Harten et al. 1988)
1988	Formal IAU approval of ASCII tables (IAU 1988)
1990	Extended to include IEEE floating-point data (Wells & Grosbøl 1990)
1994	Extended to multiple `IMAGE`-array extensions (Ponz et al. 1994)
1995	Extended to binary-table extensions (Cotton et al. 1995)
1997	Adopted four-digit-year date format (Bunclark & Rots 1997)
2002	Adopted proposals for world-coordinate systems (Greisen & Calabretta 2002)
2002	Adopted proposals for celestial coordinates (Calabretta & Greisen 2002)
2004	Adopted MIME types for *FITS* data files (Allen & Wells 2005)
2005	Extended to support variable-length arrays in binary tables
2005	Adopted proposals for spectral-coordinate systems (Greisen et al. 2006)
2005	Extended to include 64-bit integer data type
2006	Adopted WCS HEALPix projection (Calabretta & Roukema 2007)
2006	Established *FITS* convention registry
2014	Adopted proposals for time coordinates (Rots et al. 2015)
2016	Adopted proposals for compressed data
2016	Adopted various registered conventions
2018	General language editing

版本历史

表 2: 标准的版本历史

版本	日期	状态
NOST 100-0.1	1990-12	第一版草案标准
NOST 100-0.2	1991-06	第二版修订草案标准
NOST 100-0.3	1991-12	第三版修订草案标准
NOST 100-1.0	1993-06	NOST 标准
NOST 100-1.1	1995-09	NOST 标准
NOST 100-2.0	1999-03	NOST 标准
IAUFWG 2.1	2005-04	IAUFWG 标准
IAUFWG 2.1b	2005-12	IAUFWG 标准
IAUFWG 3.0	2008-07	IAUFWG 标准
IAUFWG 4.0	2016-07	IAUFWG 标准（批准）
IAUFWG 4.0	2018-08	IAUFWG 标准（语言编辑版）

FITS文件结构

- Primary header and data unit (HDU).
- Conforming Extensions (*optional*).
- Other special records (*optional*, *restricted*).

FITS 类型

- `SIF` (Single Image *FITS*)

- `MEF` (Multi-Extension *FITS*)

FITS BLOCKS

- 每个FITS文件都是由若干BLOCKS组成的，而每个块的固定大小是 2880 字节（23040位）

- 2880 = 80 x 36

头信息（Header）

- ASCII字符（0x20-0x7E）

```
SIMPLE  =                    T /FITS STANDARD
```

- 2880 = 80 x 36
- 结构：
  - 1-8 字节：关键字
  - 9-10 字节：=
  - 11-80 字节：值和注释，注释以 / 开头
- END结尾头信息，之后如不足2880，以空格填充。

- CARD
- RECORD
- KEY
- VALUE
- COMMENT

```
EXPTIME = 1200. / [s] exposure time in seconds
```

头信息—值（字符型）

- 单记录字符型

```
KEYWORD1= ''                    / null string keyword
KEYWORD2= '    '                / empty string keyword
KEYWORD3=                       / undefined keyword
```

- 长字符型

头信息—值（逻辑型）

- `T` (True)
- `F` (False)

头信息—值（整数型）

- 11-30字节：右对齐
- 正数的 '+' 可忽略

头信息—值（浮点数型）

- 11-30字节：右对齐
- 指数’E’ ’D’需要大写

头信息—值（复数型）

- 整形复数　(12, 34)

- 浮点型复数　(12.234, 34.5678)

头信息—值（日期型）

- ISO-8601格式　(YYYY-MM-DDThh:mm:ss[.sss])
  - DATE 关键字

- JD、MJD等格式　(2345675.22222, 23456.234)

单位

表 3: IAU-recommended basic units.

Quantity	Unit	Meaning	Notes
SI base & supplementary units			
length	`m`	meter	
mass	`kg`	kilogram	`g` gram allowed
time	`s`	second	
plane angle	`rad`	radian	
solid angle	`sr`	steradian	
temperature	`K`	kelvin	
electric current	`A`	ampere	
amount of substance	`mol`	mole	
luminous intensity	`cd`	candela	
IAU-recognized derived units			
frequency	`Hz`	hertz	s^{-1}
energy	`J`	joule	N m
power	`W`	watt	$J\ s^{-1}$
electric potential	`V`	volt	$J\ C^{-1}$
force	`N`	newton	$kg\ m\ s^{-2}$
pressure, stress	`Pa`	pascal	$N\ m^{-2}$
electric charge	`C`	coulomb	A s
electric resistance	`Ohm`	ohm	$V\ A^{-1}$
electric conductance	`S`	siemens	$A\ V^{-1}$
electric capacitance	`F`	farad	$C\ V^{-1}$
magnetic flux	`Wb`	weber	V s
magnetic flux density	`T`	tesla	$Wb\ m^{-2}$
inductance	`H`	henry	$Wb\ A^{-1}$
luminous flux	`lm`	lumen	cd sr
illuminance	`lx`	lux	$lm\ m^{-2}$

表 4: Additional allowed units.

Quantity		Unit	Meaning	Notes
plane angle		deg	degree of arc	$\pi/180$ rad
		arcmin	minute of arc	$1/60$ deg
		arcsec	second of arc	$1/3600$ deg
		mas	milli-second of arc	$1/3\,600\,000$ deg
time		min	minute	60 s
		h	hour	60 min = 3600 s
		d	day	86 400 s
	†	a	year (Julian)	31 557 600 s (365.25 d), peta a(Pa) forbidden
	†	yr	year (Julian)	a is IAU-style
energy*	†	eV	electron volt	$1.6021765 \times 10^{-19}$ J
	‡	erg	erg	10^{-7} J
		Ry	rydberg	$\frac{1}{2}\left(\frac{2\pi e^2}{hc}\right)^2 m_e c^2 = 13.605692$ eV
mass*		solMass	solar mass	1.9891×10^{30} kg
		u	unified atomic mass unit	$1.6605387 \times 10^{-27}$ kg
luminosity		solLum	Solar luminosity	3.8268×10^{26} W
length	‡	Angstrom	angstrom	10^{-10} m
		solRad	Solar radius	6.9599×10^{8} m
		AU	astronomical unit	1.49598×10^{11} m
		lyr	light year	9.460730×10^{15} m
	†	pc	parsec	3.0857×10^{16} m
events		count	count	
		ct	count	
		photon	photon	
		ph	photon	
flux density	†	Jy	jansky	10^{-26} W m^{-2} Hz^{-1}
	†	mag	(stellar) magnitude	
	†	R	rayleigh	$10^{10}/(4\pi)$ photons m^{-2} s^{-1} sr^{-1}
magnetic field	†‡	G	gauss	10^{-4} T
area		pixel	(image/detector) pixel	
		pix	(image/detector) pixel	
	†‡	barn	barn	10^{-28} m^2
Miscellaneous units				
		D	debye	$\frac{1}{3} \times 10^{-29}$ C.m
		Sun	relative to Sun	e.g., abundances
		chan	(detector) channel	
		bin	numerous applications	(including the one-dimensional analog of pixel)
		voxel	three-dimensional analog of pixel	
	†	bit	binary information unit	
	†	byte	(computer) byte	eight bits
		adu	Analog-to-digital converter	
		beam	beam area of observation	as in Jy/beam

Notes. (†)Addition of prefixes for decimal multiples and submultiples are allowed. (‡)Deprecated in IAU Style Manual (McNally 1988) but still in use. (*)Conversion factors from CODATA Internationally recommended values of the fundamental physical constants 2002 (http://physics.nist.gov/cuu/Constants/).

单位

表 5: Prefixes for multiples and submultiples.

Submult	Prefix	Char	Mult	Prefix	Char
10^{-1}	deci	d	10	deca	da
10^{-2}	centi	c	10^{2}	hecto	h
10^{-3}	milli	m	10^{3}	kilo	k
10^{-6}	micro	u	10^{6}	mega	M
10^{-9}	nano	n	10^{9}	giga	G
10^{-12}	pico	p	10^{12}	tera	T
10^{-15}	femto	f	10^{15}	peta	P
10^{-18}	atto	a	10^{18}	exa	E
10^{-21}	zepto	z	10^{21}	zetta	Z
10^{-24}	yocto	y	10^{24}	yotta	Y

```
EXPTIME = 1200. / [s] exposure time in seconds
```

头结构

表 7: Mandatory keywords for primary header.

Position	Keyword
1	`SIMPLE` = `T`
2	`BITPIX`
3	`NAXIS`
4	`NAXIS`n, $n = 1, \ldots,$ `NAXIS`
	⋮
	(other keywords)
	⋮
last	`END`

BITPIX (Bits/Pixel)

Value	Data represented
8	Character or unsigned binary integer
16	16-bit two's complement binary integer
32	32-bit two's complement binary integer
64	64-bit two's complement binary integer
−32	IEEE single-precision floating point
−64	IEEE double-precision floating point

NAXIS

- 數據軸
- NAXIS 軸數目
- NAXIS1 第一個軸的數量
- ⋯
- NAXISn 第n個軸的數量

$$N_{\text{bits}} = |\texttt{BITPIX}| \times (\texttt{NAXIS1} \times \texttt{NAXIS2} \times \cdots \times \texttt{NAXIS}m)$$

```
SIMPLE  =                    T /FITS STANDARD
BITPIX  =                   16 /FITS BITS/PIXEL
NAXIS   =                    2 /NUMBER OF AXES
NAXIS1  =                14400 /
NAXIS2  =                14400 /
```

观测类关键字

- DATE-OBS
- TELESCOP
- INSTRUME
- OBSERVER
- OBJECT
- EQUINOX

- ORIGIN （机构、单位）
- AUTHOR
- REFERENC (bibcode / doi)

注释类关键字

- HISTORY

```
COMMENT   FITS (Flexible Image Transport System) format is defined in 'Astronomy
COMMENT   and Astrophysics', volume 376, page 359; bibcode: 2001A&A...376..359H
```

- COMMENT


```
COMMENT   FITS (Flexible Image Transport System) format is defined in 'Astronomy
COMMENT   and Astrophysics', volume 376, page 359; bibcode: 2001A&A...376..359H
```


- 0-10字段留空

数据类关键字（IMAGE）

- BSCALE

$$\text{physical\_value} = \text{BZERO} + \text{BSCALE} \times \text{array\_value}.$$

BITPIX	Native data type	Physical data type	BZERO	
8	unsigned	signed byte	-128	(-2^7)
16	signed	unsigned 16-bit	32768	(2^{15})
32	signed	unsigned 32-bit	2147483648	(2^{31})
64	signed	unsigned 64-bit	9223372036854775808	(2^{63})

- BZERO
- BUNIT
- BLANK
- DATAMAX

扩展类关键字

- EXTEND (T/F)
- EXTNAME
- EXTVER 同名的扩展可以以此整数值进行标注
- EXTLEVEL 层次性数据的标注 (整数)

扩展（XTENSION）

- IMAGE　图像
- TABLE 文本表格
- BINTABLE 二进制表格

XTENSION-IMAGE

Position	Keyword
1	XTENSION=␣'IMAGE␣␣␣'
2	BITPIX
3	NAXIS
4	NAXISn, $n = 1, \ldots,$ NAXIS
5	PCOUNT = 0
6	GCOUNT = 1
	⋮
	(other keywords ...)
	⋮
last	END

IMAGE—关键字

- BIXPIX
- NAXIS
- NAXIS*n*
- PCOUNT 0
- GCOUNT 1
- END

XTENSION-TABLE

Position	Keyword
1	XTENSION=␣'TABLE␣␣␣'
2	BITPIX = 8
3	NAXIS = 2
4	NAXIS1
5	NAXIS2
6	PCOUNT = 0
7	GCOUNT = 1
8	TFIELDS
	⋮
	(other keywords, including (if TFIELDS is not zero) . . .)
	TTYPEn, $n = 1, 2, \ldots, k$, where k is the value of TFIELDS (*recommended*)
	TBCOLn, $n = 1, 2, \ldots, k$, where k is the value of TFIELDS (*required*)
	TFORMn, $n = 1, 2, \ldots, k$, where k is the value of TFIELDS (*required*)
	⋮
last	END

TABLE-KEYWORDS

- BITPIX 8
- NAXIS1 表格的列长度，每行记录的字节长度
- NAXIS2 表格的行数
- PCOUNT 0
- GCOUNT 1 单表
- TFIELDS 列数 0-999 支持最多999列

TABLE-字段KEYWORDS

- TTYPE*n* 字段名 （推荐），注意，此值大小写不敏感，推荐大写

- TBCOL*n* 列序号，从1～TFIELDS （必选）

- TFORM*n* 格式 （必选）

Field value	Data type
Aw	Character
Iw	Decimal integer
Fw.d	Floating-point, fixed decimal notation
Ew.d	Floating-point, exponential notation
Dw.d	Floating-point, exponential notation

TABLE-字段KEYWORDS（续）

- TUNIT*n* 单位
- TSCAL*n* 比例因子

$$\text{physical\_value} = \text{TZERO}n + \text{TSCAL}n \times \text{field\_value}$$

- TZERO*n* 零值
- TNULL*n* 空值
- TDISP*n* 显示格式（详见下页）

表 16: Valid `TDISP`*n* format values in `TABLE` extensions.

Field value	Data type
`Aw`	Character
`Iw.m`	Integer
`Bw.m`	Binary, integers only
`Ow.m`	Octal, integers only
`Zw.m`	Hexadecimal, integers only
`Fw.d`	Floating-point, fixed decimal notation
`Ew.dEe`	Floating-point, exponential notation
`ENw.d`	Engineering; E format with exponent multiple of three
`ESw.d`	Scientific; same as EN but non-zero leading digit if not zero
`Gw.dEe`	General; appears as F if significance not lost, else E.
`Dw.dEe`	Floating-point, exponential notation

TABLE 数据顺序

- 二维表格的行列是严格确定的，每行的字符数和行数

- 按行存储

XTENSION-BINTABLE

Position	Keyword
1	XTENSION=␣'BINTABLE'
2	BITPIX = 8
3	NAXIS = 2
4	NAXIS1
5	NAXIS2
6	PCOUNT
7	GCOUNT = 1
8	TFIELDS
	⋮
	(other keywords, including (if TFIELDS is not zero) . . .)
	TTYPEn, $n = 1, 2, \ldots, k$, where k is the value of TFIELDS (*recommended*)
	TFORMn, $n = 1, 2, \ldots, k$, where k is the value of TFIELDS (*required*)
	⋮
last	END

BINTABLE-KEYWORDS

- BITPIX 8
- NAXIS 2
- NAXIS1 每行的字节数（8位字节）
- NAXIS2 行数
- PCOUNT heap
- GCOUNT 1
- TFIELDS 列数

BINTABLE-KEYWORDS

- TTYPE*n* 字段名 （推荐），注意，此值大小写不敏感，推荐大写

- TFORM*n* 格式 （必选）

TFORM*n* value	Description	Eight-bit Byte
'L'	Logical	1
'X'	Bit	†
'B'	Unsigned byte	1
'I'	16-bit integer	2
'J'	32-bit integer	4
'K'	64-bit integer	8
'A'	Character	1
'E'	Single-precision floating point	4
'D'	Double-precision floating point	8
'C'	Single-precision complex	8
'M'	Double-precision complex	16
'P'	Array Descriptor (32-bit)	8
'Q'	Array Descriptor (64-bit)	16

$$n_{\text{row}} = \sum_{i=1}^{\texttt{TFIELDS}} r_i b_i$$

BINTABLE-字段KEYWORDS（续）

- TUNIT*n* 单位
- TSCAL*n* 比例因子
- TZERO*n* 零值
- TNULL*n* 空值
- TDISP*n* 显示格式（详见下页）

$$\text{physical\_value} = \text{TZERO}n + \text{TSCAL}n \times \text{field\_value}$$

TFORM*n*	Native data type	Physical data type	TZERO*n*	
'B'	unsigned	signed byte	-128	(-2^7)
'I'	signed	unsigned 16-bit	32768	(2^{15})
'J'	signed	unsigned 32-bit	2147483648	(2^{31})
'K'	signed	unsigned 64-bit	9223372036854775808	(2^{63})

Field Value	Data type
`Aw`	Character
`Lw`	Logical
`Iw.m`	Integer
`Bw.m`	Binary, integers only
`Ow.m`	Octal, integers only
`Zw.m`	Hexadecimal, integers only
`Fw.d`	Floating-point, fixed decimal notation
`Ew.dEe`	Floating-point, exponential notation
`ENw.d`	Engineering; E format with exponent multiple of three
`ESw.d`	Scientific; same as EN but non-zero leading digit if not zero
`Gw.dEe`	General; appears as F if significance not lost, else E.
`Dw.dEe`	Floating-point, exponential notation

时间关键字

- ISO8601 格式
- DATE-xxxx　CCYY-MM-DD[Thh:mm:ss[.s…]]　字符格式 [±C]CCYY-MM-DD[Thh:mm:ss[.s…]]

时间关键字— TIMESYS

- TIMESYS 时标，默认 UTC

Value	Meaning
'TAI'	(International Atomic Time): atomic-time standard maintained on the rotating geoid
'TT'	(Terrestrial Time; IAU standard): defined on the rotating geoid, usually derived as TAI + 32.184 s
'TDT'	(Terrestrial Dynamical Time): synonym for TT (deprecated)
'ET'	(Ephemeris Time): continuous with TT; *should not* be used for data taken after 1984-01-01
'IAT'	synonym for TAI (deprecated)
'UT1'	(Universal Time): Earth rotation time
'UTC'	(Universal Time, Coordinated; default): runs synchronously with TAI, except for the occasional insertion of leap seconds intended to keep UTC within 0.9 s of UT1; as of 2015-07-01 UTC = TAI − 36 s
'GMT'	(Greenwich Mean Time): continuous with UTC; its use is deprecated for dates after 1972-01-01
'UT()'[1]	(Universal Time, with qualifier): for high-precision use of radio-signal distributions between 1955 and 1972; see Rots et al. (2015), Appendix A
'GPS'	(Global Positioning System): runs (approximately) synchronously with TAI; GPS ≈ TAI − 19 s
'TCG'	(Geocentric Coordinate Time): TT reduced to the geocenter, corrected for the relativistic effects of the Earth's rotation and gravitational potential; TCG runs faster than TT at a constant rate
'TCB'	(Barycentric Coordinate Time): derived from TCG by a four-dimensional transformation, taking into account the relativistic effects of the gravitational potential at the barycenter (relative to that on the rotating geoid) as well as velocity time-dilation variations due to the eccentricity of the Earth's orbit, thus ensuring consistency with fundamental physical constants; Irwin & Fukushima (1999) provide a time ephemeris
'TDB'	(Barycentric Dynamical Time): runs slower than TCB at a constant rate so as to remain approximately in step with TT; runs therefore quasi-synchronously with TT, except for the relativistic effects introduced by variations in the Earth's velocity relative to the barycenter. When referring to celestial observations, a pathlength correction to the barycenter may be needed, which requires the Time Reference Direction used in calculating the pathlength correction.
'LOCAL'	for simulation data and for free-running clocks.

时间关键字—时间参考值

- MJDREF 默认0.0
  - MJDREFI 0
  - MJDREFF 0.0
- JDREF 默认none
  - JDREFI 0
  - JDREFF 0.0
- DATEREF 默认none

时间关键字—TREFPOS

- TREFPOS 默认 TOPOCENTER

表 31: Standard Time Reference Position Values

Value[1]	Meaning
'TOPOCENTER'	Topocenter: the location from where the observation was made (default)
'GEOCENTER'	Geocenter
'BARYCENTER'	Barycenter of the Solar System
'RELOCATABLE'	Relocatable: to be used for simulation data only
'CUSTOM'	A position specified by coordinates that is not the observatory location
Less-common, but allowed standard values	
'HELIOCENTER'	Heliocenter
'GALACTIC'	Galactic center
'EMBARYCENTER'	Earth-Moon barycenter
'MERCURY'	Center of Mercury
'VENUS'	Center of Venus
'MARS'	Center of Mars
'JUPITER'	Barycenter of the Jupiter system
'SATURN'	Barycenter of the Saturn system
'URANUS'	Barycenter of the Uranus system
'NEPTUNE'	Barycenter of the Neptune system

时间单位

Value	Definition
's'	second (default)
'd'	day (= 86,400 s)
'a'	(Julian) year (= 365.25 d)
'cy'	(Julian) century (= 100 a)
The following values are also acceptable.	
'min'	minute (= 60 s)
'h'	day (= 86,400 s)
'yr'	(Julian) year (= 'a' = 365.25 d)
'ta'	tropical year
'Ba'	Besselian year

全局时间关键字总结

- DATE HDU产生日期[时间]
  - YYYY-MM-DD
  - YYYY-MMDDThh:mm:ss[.ss]
- DATE-OBS 观测的日期[时间] YYYY-MMDDThh:mm:ss[.ss]
- DATE-BEG 观测开始日期时间 YYYY-MMDDThh:mm:ss[.ss]
- DATE-AVG 观测中点时间 YYYY-MMDDThh:mm:ss[.ss]
- DATE-END 观测结束时间 YYYY-MMDDThh:mm:ss[.ss]

- MJD-OBS 观测时间
- MJD-BEG 观测开始时间
- MJD-AVG 观测中点时间
- MJD-END 观测结束时间

数据校验

- DATASUM （推荐）HDU的数据体的校验码，值为一个32位的无符号整数，左对齐
  - 2880 = 720 x 4
- CHECKSUM （推荐）HDU的校验码，值为16位字符串，左对齐

```
DATASUM = '2503531142'          /  2015-06-28T18:30:45
CHECKSUM= 'hcHjjc9ghcEghc9g'    /  2015-06-28T18:30:45
```

```
void checksum (
   unsigned char *buf,  /* Input array of bytes to be checksummed */
                        /*  (interpret as 4-byte unsigned ints)   */
   int length,          /* Length of buf array, in bytes          */
                        /*  (must be multiple of 4)               */
   unsigned int *sum32) /* 32-bit checksum                        */
{
/*
   Increment the input value of sum32 with the 1's complement sum
   accumulated over the input buf array.
*/
   unsigned int hi, lo, hicarry, locarry, i;

   /*  Accumulate the sum of the high-order 16 bits and the */
   /*  low-order 16 bits of each 32-bit word, separately.  */
   /*  The first byte in each pair is the most significant. */
   /*  This algorithm works on both big and little endian machines.*/
   hi = (*sum32 >> 16);
   lo = *sum32 & 0xFFFF;
   for (i=0; i < length; i+=4) {
       hi += ((buf[i]   << 8) + buf[i+1]);
       lo += ((buf[i+2] << 8) + buf[i+3]);
   }

   /* fold carry bits from each 16 bit sum into the other sum */
   hicarry = hi >> 16;
   locarry = lo >> 16;
   while (hicarry || locarry) {
       hi = (hi & 0xFFFF) + locarry;
       lo = (lo & 0xFFFF) + hicarry;
       hicarry = hi >> 16;
       locarry = lo >> 16;
   }

   /* concatenate the full 32-bit value from the 2 halves */
   *sum32 = (hi << 16) + lo;
}
```

```
unsigned int exclude[13] = {0x3a, 0x3b, 0x3c, 0x3d, 0x3e, 0x3f, 0x40,
                            0x5b, 0x5c, 0x5d, 0x5e, 0x5f, 0x60 };

int offset = 0x30;                          /* ASCII 0 (zero) */
unsigned long mask[4] = { 0xff000000, 0xff0000, 0xff00, 0xff };

void char_encode (
  unsigned int value, /* 1's complement of the checksum     */
                      /*     value to be encoded            */
  char *ascii)        /* Output 16-character encoded string */
{
   int byte, quotient, remainder, ch[4], check, i, j, k;
   char asc[32];

   for (i=0; i < 4; i++) {
       /* each byte becomes four */
       byte = (value & mask[i]) >> ((3 - i) * 8);
       quotient = byte / 4 + offset;
       remainder = byte % 4;
       for (j=0; j < 4; j++)
           ch[j] = quotient;

       ch[0] += remainder;

       for (check=1; check;)     /* avoid ASCII punctuation */
          for (check=0, k=0; k < 13; k++)
             for (j=0; j < 4; j+=2)
                 if (ch[j]==exclude[k] || ch[j+1]==exclude[k]) {
                     ch[j]++;
                     ch[j+1]--;
                     check++;
                  }

        for (j=0; j < 4; j++)     /* assign the bytes */
            asc[4*j+i] = ch[j];
    }

    for (i=0; i < 16; i++)       /* permute the bytes for FITS */
       ascii[i] = asc[(i+15)%16];

    ascii[16] = 0;               /* terminate the string */
}
```

关键字总结

表 C.1: Mandatory *FITS* keywords for the structures described in this document.

Primary HDU	Conforming extension	Image extension	ASCII-table extension	Binary-table extension	Compressed images[6]	Compressed tables[6]	Random-groups records
SIMPLE	XTENSION	XTENSION[1]	XTENSION[2]	XTENSION[3]	ZIMAGE =T	ZTABLE =T	SIMPLE
BITPIX	BITPIX	BITPIX	BITPIX = 8	BITPIX = 8	ZBITPIX	ZNAXIS1	BITPIX
NAXIS	NAXIS	NAXIS	NAXIS = 2	NAXIS = 2	ZNAXIS	ZNAXIS2	NAXIS
NAXISn[4]	NAXISn[4]	NAXISn[4]	NAXIS1	NAXIS1	ZNAXISn	ZPCOUNT	NAXIS1 = 0
END	PCOUNT	PCOUNT = 0	NAXIS2	NAXIS2	ZCMPTYPE	ZFORMn	NAXISn[4]
GCOUNT	GCOUNT = 1	PCOUNT = 0	PCOUNT = 0	PCOUNT	ZCTYPn	ZTILELEN	GROUPS = T
END	END	GCOUNT = 1	GCOUNT = 1	GCOUNT = 1	ZTILELEN		PCOUNT
		END	TFIELDS	TFIELDS			GCOUNT
			TFORMn[5]	TFORMn[5]			END
			TBCOLn[5]	END			
			END				

(1) XTENSION=␣'IMAGE␣␣␣' for the image extension. (2) XTENSION=␣'TABLE␣␣␣' for the ASCII-table extension. (3) XTENSION=␣'BINTABLE' for the binary-table extension. (4) Runs from 1 through the value of NAXIS. (5) Runs from 1 through the value of TFIELDS. (6) Required in addition to the mandatory keywords for binary tables.

http://heasarc.gsfc.nasa.gov/docs/fcg/standard_dict.html

表 C.2: Reserved *FITS* keywords for the structures described in this document.

All[1] HDUs		Array[2] HDUs	ASCII-table extension	Binary-table extension	Compressed images	Compressed tables	Random-groups records
DATE	EXTNAME	BSCALE	TSCAL*n*	TSCAL*n*	ZTILE*n*	FZTILELN	PTYPE*n*
DATE-OBS	EXTVER	BZERO	TZERO*n*	TZERO*n*	ZNAME*i*	FZALGOR	PSCAL*n*
ORIGIN	EXTLEVEL	BUNIT	TNULL*n*	TNULL*n*	ZVAL*i*	FZALG*n*	PZERO*n*
AUTHOR	EQUINOX	BLANK	TTYPE*n*	TTYPE*n*	ZMASKCMP		
REFERENC	EPOCH[3]	DATAMAX	TUNIT*n*	TUNIT*n*	ZQUANTIZ		
COMMENT	BLOCKED[3]	DATAMIN	TDISP*n*	TDISP*n*	ZDITHER0		
HISTORY	EXTEND[4]		TDMAX*n*	TDIM*n*	ZSIMPLE	ZTHEAP	
␣␣␣␣␣␣␣␣	TELESCOP		TDMIN*n*	THEAP	ZEXTEND		
OBJECT	INSTRUME		TLMAX*n*	TDMAX*n*	ZBLOCKED		
OBSERVER			TLMIN*n*	TDMIN*n*	ZTENSION		
CONTINUE				TLMAX*n*	ZPCOUNT		
INHERIT [5]				TLMIN*n*	ZGCOUNT		
CHECKSUM					ZHECKSUM	ZHECKSUM	
DATASUM					ZDATASUM	ZDATASUM	

表 C.3: General reserved *FITS* keywords described in this document.

Production	Bibliographic	Commentary	Observation
DATE	AUTHOR	COMMENT	DATE-OBS
ORIGIN	REFERENC	HISTORY	TELESCOP
BLOCKED[1]		␣␣␣␣␣␣␣␣	INSTRUME
			OBSERVER
			OBJECT
			EQUINOX
			EPOCH[1]

FITS程序库

Library	Language[s]	Level	Images	Groups	ASCII Table	Binary Table	Var. Len. Arrays
CFITSIO	C/Fortran	Low	rw	rw	rw	rw	rw
WCS FITS library	C (Fortran callable)	High	rw	-	r	r	-
fitsy/funtools	C	High	rw	-	rw	rw	-
qfits	C/Python	Low	rw	-	-	rw	-
CCfits	C++	Medium	rw	-	rw	rw	rw
C++ FITS	C++	Medium	rw	rw	rw	rw	rw
CSharpFITS	C-Sharp (.Net)	Medium	rw	rw	rw	rw	rw
FitsLib	C-Sharp (.Net)	Medium	rw	rw	rw	rw	rw
MIIPS FITS	Fortran	High	rw	-	-	-	-
DeLaFits	Pascal (Delphi & Lazarus)	High	rw	-	-	-	-
MRDFITS/ MWRFITS	IDL	High	rw	rw	rw	rw	rw
FX library	IDL	Low	rw	rw	-	rw	rw
READFITS/ WRITEFITS	IDL	High	rw	rw	rw*	r*	-
FITS_library	IDL	Medium	rw	rw	rw	r	-
IUEDAC	IDL	Medium	rw	-	rw	rw	-
nom.tam.fits	Java	Medium	rw	rw	rw	rw	rw
eap.fits	Java	Medium	rw	-	rw	rw	rw
jfits	Java	Medium	rw	-	rw	rw	-
STIL	Java	Medium	-	-	r	rw	rw
fitsjs	JavaScript	Medium	r	-	r	r	-
CFITSIO.pm	Perl	Low	rw	rw	rw	rw	rw
FitsTcl	TCL	Medium	rw	rw	rw	rw	rw
PDL FITS	Perl	High	rw	-	-	rw	-
MFITSIO	MatLab	Medium	rw	-	-	-	-
MatLab	MatLab	High/Low	rw	rw	rw	rw	rw
FITS static class	MatLab	High	rw	-	rw	rw	-
Canterbury	MatLab	High	rw	-	-	-	-
GFITSIO	LabVIEW	High	rw	-	rw	rw	-
PyFITS	Python	Low	rw	rw	rw	rw	rw
Python/fitsio	Python	Low	rw	rw	rw	rw	rw
PFITS	Python	Medium	r	-	r	r	r
Mathematica	Mathematica	High	rw	-	rw	-	-
IGOR Pro	IGOR Pro	High	r	-	r	r	-
R language	R	High	rw	-	-	r	-
Liberator	Photoshop plug-in	High	rw	-	-	-	-
FitsPlug	Photoshop plug-in	High	rw	-	-	-	-
golang FITS reader	golang (google go)	High	r	-	r	r	-
golang wrappers for CFITSIO	golang (google go)	High	rw	-	rw	rw	rw
pure-Go FITS package	golang (google go)	High	rw	-	rw	rw	rw

* using ancillary libraries.

CFITSIO/FITSIO

https://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html

- 官方版本
- C/Fortran
- 多平台支持（Unix、Linux、Windows、macOS、Android等）

一个例子

```
#include <string.h>
#include <stdio.h>
#include "fitsio.h"

int main(int argc, char *argv[])
{
    fitsfile *fptr;
    char card[FLEN_CARD];
    int status = 0,  nkeys, ii;  /* MUST initialize status */

    fits_open_file(&fptr, argv[1], READONLY, &status);
    fits_get_hdrspace(fptr, &nkeys, NULL, &status);

    for (ii = 1; ii <= nkeys; ii++)  {
      fits_read_record(fptr, ii, card, &status); /* read keyword */
      printf("%s\n", card);
    }
    printf("END\n\n");  /* terminate listing with END */
    fits_close_file(fptr, &status);

    if (status)          /* print any error messages */
        fits_report_error(stderr, status);
    return(status);
}
```

程序

- 打开FITS文件
- 读写HDU头信息
- 读写HDU数据体
- 关闭文件

短名称函数

```
int fits_open_file / ffopen
    (fitsfile **fptr, char *filename, int iomode, > int *status)
```

```
int fits_write_key_str / ffpkys
    (fitsfile *fptr, char *keyname, char *value, char *comment,
     > int *status)

int fits_write_key_[log, lng] /  ffpky[lj]
    (fitsfile *fptr, char *keyname, DTYPE numval, char *comment,
     > int *status)
```

短名称函数

Long Names	Short Names	Data Type
_bit	x	bit
_byt	b	unsigned byte
_sbyt	sb	signed byte
_sht	i	short integer
_lng	j	long integer
_lnglng	jj	8-byte LONGLONG integer (see note below)
_usht	ui	unsigned short integer
_ulng	uj	unsigned long integer
_ulnglng	ujj	unsigned long long integer
_uint	uk	unsigned int integer
_int	k	int integer
_flt	e	real exponential floating point (float)
_fixflt	f	real fixed-decimal format floating point (float)
_dbl	d	double precision real floating-point (double)
_fixdbl	g	double precision fixed-format floating point (double)
_cmp	c	complex reals (pairs of float values)
_fixcmp	fc	complex reals, fixed-format floating point
_dblcmp	m	double precision complex (pairs of double values)
_fixdblcmp	fm	double precision complex, fixed-format floating point
_log	l	logical (int)
_str	s	character string

longnam.h

FITS Tools: Handy FITS Utilities that illustrate how to use CFITSIO

https://heasarc.gsfc.nasa.gov/docs/software/fitsio/cexamples.html

- **fitscopy** – copy a file with optional filtering
- **listhead** – list header keywords
- **liststruc** – list the structure of a FITS file.
- **modhead** – write or modify a header keyword
- **imcopy** – copy an image with optional compression/decompression
- **imarith** – add, subtract, multiply, or divide an image by a constant or another image
- **imlist** – list pixel values in an image
- **imstat** – compute statistics of image pixels
- **tablist** – display the contents of a FITS table
- **tabcalc** – general table calculator
- **tabmerge** – merge rows from one table with another table.
- **cookbook** – additional example routines.

DEMO

FITSVERIFY - A FITS File Format-Verification Tool

https://heasarc.gsfc.nasa.gov/docs/software/ftools/fitsverify/

- **fitsverify -l filename.fits** (list all header keywords when verifying each file)
- **fitsverify -q *.fits** (Verify all *.fits files in the current directory in 'quiet' mode. This suppresses the detailed verification report and only writes a 1-line pass/fail message for each file)
- **fitsverify -e filename.fits** (only check for errors, ignore warnings)
- **fitsverify @textfile** (verify the FITS files listed in the text file, 1 file name per line)
- **fitsverify -h** (for complete help listing)

DEMO

FUNTOOLS

https://github.com/ericmandel/funtools

- **funcalc** [-n] [-a argstr] [-e expr] [-f file] [-l link] [-p prog] [-u] <iname> [oname [columns]]
- **funcen** [-i] [-n iter] [-t tol] [-v lev] <iname> <region>
- **funcnts** [switches] <source_file> [source_region] [bkgd_file] [bkgd_region|bkgd_cnts]
- **funcone** [-n] [-x|-X|-j|-J] [[-l|-L] list] [-r ra_col] [-d dec_col] <iname> <oname> <ra[hdr]> <dec[hdr]> <radius[dr'"]> [columns] fundisp [-f format] [-l] [-n] [-T] <iname> [columns|bitpix=n]
- **funhead** [-a] [-l] [-s] [-t] [-L] <iname> [oname ename]
- **funhist** [-n|-w|-T] <iname> [column] [[lo_edge:hi_edge:]bins] funimage [-a] [-l] [-p x|y] <iname> <oname> [bitpix=n] funindex <iname> <key> [oname]
- **funjoin** [switches] <ifile1> <ifile2> ... <ifilen> <ofile> funmerge <iname1> <iname2> ... <oname>
- **funsky** [switches] <iname1> [<lname2> <col1> <col2>]
- **funtable** [-a] [-i|-z] [-m] [-s cols] <iname> <oname> [columns] funt
- funtbl [-c cols] [-h] [-n table] [-p prog] [-s sep] [-T] <iname>f

DEMO

AstroPY—astropy.io.fits

- Python FITS读写库（底层基于cfitsio）
- DEMO

虚拟天文台表格（VOTable）

$$
\begin{aligned}
\textbf{VOTable} &= \text{hierarchy of } \textbf{Metadata} + \text{associated } \textbf{TableData}, \text{ arranged as a set of } \textbf{Tables} \\
\textbf{Metadata} &= \textbf{Parameters} + \textbf{Infos} + \textbf{Descriptions} + \textbf{Links} + \textbf{Fields} + \textbf{Groups} \\
\textbf{Table} &= \text{list of } \textbf{Fields} + \textbf{TableData} \\
\textbf{TableData} &= \text{stream of } \textbf{Rows} \\
\textbf{Row} &= \text{list of } \textbf{Cells} \\
\textbf{Cell} &= \begin{cases} \textbf{Primitive} \\ \text{or variable-length list of } \textbf{Primitives} \\ \text{or multidimensional array of } \textbf{Primitives} \end{cases} \\
\textbf{Primitive} &= \text{integer, character, float, floatComplex, etc (see Table 1 below).}
\end{aligned}
$$

<VOTABLE>
⊕ <*DESCRIPTION*>
† <*COOSYS*>···
○ <*INFO*>···
○ <*PARAM*>···
○ <*GROUP*>···
⊕ <RESOURCE>···
⊕ <*INFO*>···
</VOTABLE>

<RESOURCE>
⊕ <*DESCRIPTION*>
○ <*INFO*>···
† <*COOSYS*>···
○ <*GROUP*>···
○ <*PARAM*>···
⊕ <*LINK*>···
⊕ <*TABLE*>···
⊕ <*RESOURCE*>···
⊕ <*INFO*>···
</RESOURCE>

<TABLE>
⊕ <*DESCRIPTION*>
○ <*FIELD*>···
○ <*PARAM*>···
○ <*GROUP*>···
⊕ <*LINK*>···
⊕ <*DATA*>
⊕ <*INFO*>···
</TABLE>

<FIELD>
⊕ <*DESCRIPTION*>
⊕ <*VALUES*>
⊕ <*LINK*>···
</FIELD>

<PARAM>
⊕ <*DESCRIPTION*>
⊕ <*VALUES*>
⊕ <*LINK*>···
</PARAM>

<DATA>
↦ <TABLEDATA>
⊕ <*TR*>···
⊕ <TD>···
↦ <BINARY>
⊕ <STREAM>
↦ <BINARY2>
⊕ <STREAM>
↦ <FITS>
⊕ <STREAM>
</DATA>
⊕ <*INFO*>···

<GROUP>
⊕ <*DESCRIPTION*>
○ <*FIELDref*>···$^{(t)}$
○ <*PARAM*>···
○ <*PARAMref*>···
○ <*GROUP*>···
</GROUP>

(t) only within <TABLE>

<VALUES>
⊕ <*MIN*>
⊕ <*MAX*>
⊕ <*OPTION*>···
○ <*OPTION*>···
</VALUES>

VOTABLE *(section 3)*
- ID
- version

RESOURCE *(section 3.4)*
- ID
- name
- type
- utype

TABLE *(section 3.6)*
- ID
- name
- ucd
- utype
- ref
- nrows

INFO *(section 4.8)*
- ID
- **name**
- **value**
- xtype
- ref
- unit
- ucd
- utype

STREAM *(section 5.6)*
- type
- href
- actuate
- encoding
- expires
- rights

FITS *(section 5.2)*
- extnum

TR *(section 5.1)*
- *ID*

TD *(section 5.1)*
- *encoding*

GROUP *(section 4.9)*
- ID
- name
- ref
- ucd
- utype

PARAM *(section 4.1)*
- ID
- unit
- **datatype**
- precision
- width
- xtype
- ref
- **name**
- ucd
- utype
- arraysize
- **value**

FIELD *(section 4.1)*
- ID
- unit
- **datatype**
- precision
- width
- xtype
- ref
- **name**
- ucd
- utype
- arraysize
- *type*

FIELDref *(section 4.9)*
- **ref**
- ucd
- utype

PARAMref *(section 4.9)*
- **ref**
- ucd
- utype

MIN *(section 4.7)*
- **value**
- inclusive

MAX *(section 4.7)*
- **value**
- inclusive

OPTION *(section 4.7)*
- name
- **value**

VALUES *(section 4.7)*
- ID
- type
- null
- ref

LINK *(section 3.5)*
- ID
- content-role
- content-type
- title
- value
- href
- *action*

datatype	Meaning	FITS	Bytes
"boolean"	Logical	"L"	1
"bit"	Bit	"X"	*
"unsignedByte"	Byte (0 to 255)	"B"	1
"short"	Short Integer	"I"	2
"int"	Integer	"J"	4
"long"	Long integer	"K"	8
"char"	ASCII Character	"A"	1
"unicodeChar"	Unicode Character		2
"float"	Floating point	"E"	4
"double"	Double	"D"	8
"floatComplex"	Float Complex	"C"	8
"doubleComplex"	Double Complex	"M"	16

AstroPY-astropy.io.votable

- Python
- DEMO

文本型数据

- CSV
- ECSV
- CDS
- 其他
- 参考： http://docs.astropy.org/en/stable/io/ascii/index.html

CSV

```
num,ra,dec,radius,mag
1,32.23222,10.1211,0.8,18.1
2,38.12321,-88.1321,2.2,17.0
```

ECSV

Enhanced Character Separated Values

```
# %ECSV 0.9
# ---
# datatype:
# - {name: a, unit: m / s, datatype: int64, format: '%03d'}
# - {name: b, unit: km, datatype: int64, description: This is column b}
a b
001 2
004 3
```

http://vizier.china-vo.org/ftp/cats/V/153/ReadMe

http://vizier.china-vo.org/ftp/cats/V/53A/ReadMe

IPAC

```
\keyword = value                                                    # Keywords (optional)
\ Comment                                                           # Comments (optional)
|  name1     |  name2     | name3   |  name4   |      name5         | # Column Names (required)
|   double   |   double   |   int   |   real   |      char          | # Data Types (standard)
|   unit     |   unit     |   unit  |   unit   |      unit          | # Data Units (optional)
|   null     |   null     |   null  |   null   |      null          | # Null Values (optional)
  165.466279  -34.704730       5       11.27       A string value    # Data Rows (1 required)
  160.1231     -4.7            3        2.3            value         # Data Rows (1 required)
```

DEMO

謝　謝