nates of the reference point, $(r_1, r_2, r_3, \ldots)$. Paper I retains this but replaces the coordinate increment and rotation keywords CDELTj and CROTAj with the transformation matrix CDj_is. Thus, the transformation of pixel coordinates, $(p_1, p_2, p_3, \ldots)$, to intermediate world coordinate, $(x, y, z, \ldots)$, in step 2 becomes

$$\begin{pmatrix} x \\ y \\ z \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathtt{CD1\_1} & \mathtt{CD1\_2} & \mathtt{CD1\_3} & \ldots \\ \mathtt{CD2\_1} & \mathtt{CD2\_2} & \mathtt{CD2\_3} & \ldots \\ \mathtt{CD3\_1} & \mathtt{CD3\_2} & \mathtt{CD3\_3} & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \times \begin{pmatrix} p_1 - r_1 \\ p_2 - r_2 \\ p_3 - r_3 \\ \vdots \end{pmatrix} . \quad (1)$$

Eq. (1) shows that the origin of the intermediate world coordinates is at the reference point. Without loss of generality we will assume for the rest of this paper that the plane of projection corresponds to the (x, y) coordinate plane. Note that this does not necessarily correspond to the (p_1, p_2) plane since the CDj_is matrix may introduce rotation and/or skew.

Wells et al. (1981) established that all angles in FITS were to be measured in degrees and this has been entrenched by the AIPS convention. Paper I introduced the CUNITjs keyword to define the units of the CRVALjs and CDj_is. Accordingly, we require CUNITjs = 'deg' for the celestial CRVALjs and CDj_is, whether given explicitly or not. Consequently, the (x, y) coordinates in the plane of projection are measured in degrees. For consistency, we use degree measure for native and celestial spherical coordinates and for all other angular measures in this paper.

For linear coordinate systems Wells et al. (1981) prescribed that world coordinates should be computed simply by adding the relative world coordinates, $(x, y, z, \ldots)$, to the coordinate value at the reference point given by CRVALjs. Paper I extends this by providing that particular values of CTYPEjs may be established by convention to denote non-linear transformations involving predefined functions of the $(x, y, z, \ldots)$ parameterized by the CRVALjs and possibly other parameters.

In Sects. 2, 3 and 4 of this paper we will define the functions for the transformation from (x, y) coordinates in the plane of projection to celestial spherical coordinates for all spherical map projections likely to be of use in astronomy.

Section 5 considers the translation of older AIPS convention FITS headers to the new system and provisions that may be made to support older FITS-reading programs. Section 6 discusses the concepts presented here, including worked examples of header interpretation and construction.

The FITS header keywords discussed within the main body of this paper apply to the primary image header and image extension headers. Image fragments within binary tables extensions defined by Cotton et al. (1995) have additional nomenclature difficulties, a solution for which was proposed in Paper I. Coordinate descriptions may also be associated with the random groups data format defined by Greisen & Harten (1981). These issues will be expanded upon in Appendix A.

2. Basic concepts

2.1. Spherical projection

As indicated in Fig. 1, the first step in transforming (x, y) coordinates in the plane of projection to celestial coordinates is to convert them to native longitude and latitude, (ϕ, θ). The equations for the transformation depend on the particular projection and this will be specified by characters 5 through 8 of the CTYPEjs keyword. In keeping with usage established by the AIPS convention, only the last three characters will be used, so that, for example, the stereographic projection will be specified as '-STG'. Some projections require additional parameters which will be specified by the FITS keywords PVj_ms for $m = 0, 1, 2, \ldots$, introduced in Paper I. These parameters may be associated with the longitude and/or latitude coordinate as specified for each projection. However, definition of the three-letter codes for the projections and the equations, their inverses and the parameters which define them, form a large part of this work and will be deferred to Sect. 4. Characters 1 to 4 of the CTYPEjs are used to identify the celestial coordinate system and this will be discussed in Sect. 3.

2.2. Reference point of the projection

The last step in the chain of transformations shown in Fig. 1 is the spherical rotation from native coordinates, (ϕ, θ), to celestial[2] coordinates (α, δ). Since a spherical rotation is completely specified by three Euler angles it remains only to define them.

As we are following the AIPS convention, the CRVALjs must specify (α_0, δ_0), the celestial coordinates, of the reference pixel as defined by the CRPIXis. Furthermore, the AIPS convention associates the reference pixel with a particular point on the projection, which we will refer to simply as the *reference point*, thereby inextricably linking it to the reference pixel. By definition this has $(x, y) = (0, 0)$ and throughout this work we will use (ϕ_0, θ_0) to denote its native coordinates. For example, zenithal projections have their reference point at the native pole i.e. $(\phi_0, \theta_0) = (0, 90°)$, while cylindrical projections have it at the intersection of the equator and prime meridian, i.e. $(\phi_0, \theta_0) = (0, 0)$.

Thus, the CRVALjs indirectly provide two of the Euler angles needed to define the spherical coordinate rotation. Since the reference point differs for different classes of projection it is apparent that the relationship between (α_0, δ_0) and the required Euler angles must differ for different projections.

It is important to understand why the reference point differs for different projection types. There are two main reasons, the first makes it difficult for it to be the same, while the second makes it desirable that it differs. Of the

[2] Usage here of the conventional symbols for right ascension and declination for celestial coordinates is meant only as a mnemonic. It does not preclude other celestial systems.

former, some projections such as Mercator's, diverge at the native pole, therefore they cannot have the reference point there because that would imply infinite values for the `CRPIX`*is*. On the other hand, the gnomonic projection diverges at the equator so it can't have the reference point there for the same reason. Possibly a reference point chosen at some mid-latitude could satisfy all projections, but that leads us to the second reason.

Different projection types are best suited to different purposes. For example, zenithal projections are best for mapping the region in the vicinity of a point, often a pole; cylindrical projections are appropriate for the neighbourhood of a great circle, usually an equator; and the conics are suitable for small circles such as parallels of latitude. Thus, it would be awkward if a cylindrical used to map, say, a few degrees on either side of the galactic plane, had its reference point, and thus `CRPIX`*is* and `CRVAL`*js*, at the native pole, way outside the map boundary. The projection equations themselves reflect a preference for a particular reference point; the native coordinate system is chosen to make them as simple as possible. For the zenithals this is incontrovertibly the pole whereas for the cylindricals it is equally so the equator.

In certain circumstances it is desirable that the reference point of a projection be user-specifiable, typically to force coincidence with the reference pixel of the image. Although such should be considered advanced usage, the FITS keywords `PV`*j*`_0`*s* and `PV`*j*`_1`*s*, attached to *longitude* coordinate j, are here reserved to specify ϕ_0 and θ_0. Normally we expect that these will either be omitted or set to the projection-specific defaults given in Sect. 4.

As discussed above, a third Euler angle must be specified and this will be given by the native longitude of the celestial pole, ϕ_p, specified by the new FITS keyword `LONPOLE`*s*. The default value of `LONPOLE`*s* will be 0 for $\delta_0 \geq \theta_0$ or 180° for $\delta_0 < \theta_0$. This is the condition for the celestial latitude to increase in the same direction as the native latitude at the reference point. Thus, for example, in zenithal projections the default is always 180° (unless $\delta_0 = 90°$) since $\theta_0 = 90°$. In cylindrical projections, where $\theta_0 = 0$, the default value for `LONPOLE`*s* is 0 for $\delta_0 \geq 0$, but it is 180° for $\delta_0 < 0$.

2.3. Spherical coordinate rotation

For zenithal projections, the reference point is at the native pole so the `CRVAL`*js* specify the celestial coordinates of the native pole, i.e. $(\alpha_0, \delta_0) = (\alpha_\mathrm{p}, \delta_\mathrm{p})$. Now there is a simple relationship between the Euler angles for consecutive rotations about the Z-, X-, and Z-axes and α_p, δ_p and ϕ_p; the ZXZ Euler angles are $(\alpha_\mathrm{p}+90°, 90°-\delta_\mathrm{p}, \phi_\mathrm{p}-90°)$. Given this close correspondence it is convenient to write the Euler angle transformation formulæ directly in terms of α_p, δ_p and ϕ_p

$$\begin{aligned} \alpha &= \alpha_\mathrm{p} + \arg\left(\sin\theta\cos\delta_\mathrm{p} - \cos\theta\sin\delta_\mathrm{p}\cos(\phi-\phi_\mathrm{p}),\right. \\ &\qquad \left. -\cos\theta\sin(\phi-\phi_\mathrm{p})\right), \\ \delta &= \sin^{-1}(\sin\theta\sin\delta_\mathrm{p} + \cos\theta\cos\delta_\mathrm{p}\cos(\phi-\phi_\mathrm{p})). \end{aligned} \tag{2}$$

where arg is an inverse tangent function that returns the correct quadrant, i.e. if $(x, y) = (r\cos\beta, r\sin\beta)$ with $r > 0$ then $\arg(x, y) = \beta$. Note that, if $\delta_\mathrm{p} = 90°$, Eqs. (2) become

$$\begin{aligned} \alpha &= \alpha_\mathrm{p} + \phi - \phi_\mathrm{p} - 180°, \\ \delta &= \theta, \end{aligned} \tag{3}$$

which may be used to define a simple change in the origin of longitude. Likewise for $\delta_\mathrm{p} = -90°$

$$\begin{aligned} \alpha &= \alpha_\mathrm{p} - \phi + \phi_\mathrm{p}, \\ \delta &= -\theta. \end{aligned} \tag{4}$$

The inverse equations are

$$\begin{aligned} \phi &= \phi_\mathrm{p} + \arg\left(\sin\delta\cos\delta_\mathrm{p} - \cos\delta\sin\delta_\mathrm{p}\cos(\alpha-\alpha_\mathrm{p}),\right. \\ &\qquad \left. -\cos\delta\sin(\alpha-\alpha_\mathrm{p})\right), \\ \theta &= \sin^{-1}(\sin\delta\sin\delta_\mathrm{p} + \cos\delta\cos\delta_\mathrm{p}\cos(\alpha-\alpha_\mathrm{p})), \end{aligned} \tag{5}$$

Useful relations derived from Eqs. (2) and (5) are

$$\begin{aligned} \cos\delta\cos(\alpha-\alpha_\mathrm{p}) &= \sin\theta\cos\delta_\mathrm{p} - \cos\theta\sin\delta_\mathrm{p}\cos(\phi-\phi_\mathrm{p}), \\ \cos\delta\sin(\alpha-\alpha_\mathrm{p}) &= -\cos\theta\sin(\phi-\phi_\mathrm{p}), \end{aligned} \tag{6}$$

$$\begin{aligned} \cos\theta\cos(\phi-\phi_\mathrm{p}) &= \sin\delta\cos\delta_\mathrm{p} - \cos\delta\sin\delta_\mathrm{p}\cos(\alpha-\alpha_\mathrm{p}), \\ \cos\theta\sin(\phi-\phi_\mathrm{p}) &= -\cos\delta\sin(\alpha-\alpha_\mathrm{p}). \end{aligned} \tag{7}$$

A matrix method of handling the spherical coordinate rotation is described in Appendix B.

2.4. Non-polar reference point

Projections, such as the cylindricals and conics, which do not normally have the reference point at the native pole are handled by computing $(\alpha_\mathrm{p}, \delta_\mathrm{p})$ from (α_0, δ_0) whence the above equations may be used.

Given that the native coordinates of the reference point are (ϕ_0, θ_0) and the celestial coordinates are (α_0, δ_0), then the celestial coordinates of the native pole may be computed using

$$\begin{aligned} \delta_\mathrm{p} &= \arg\left(\cos\theta_0\cos(\phi_\mathrm{p}-\phi_0), \sin\theta_0\right) \pm \\ &\qquad \cos^{-1}\left(\frac{\sin\delta_0}{\sqrt{1-\cos^2\theta_0\sin^2(\phi_\mathrm{p}-\phi_0)}}\right), \end{aligned} \tag{8}$$

$$\sin(\alpha_0-\alpha_\mathrm{p}) = \sin(\phi_\mathrm{p}-\phi_0)\cos\theta_0/\cos\delta_0, \tag{9}$$

$$\cos(\alpha_0-\alpha_\mathrm{p}) = \frac{\sin\theta_0 - \sin\delta_\mathrm{p}\sin\delta_0}{\cos\delta_\mathrm{p}\cos\delta_0}, \tag{10}$$

whence Eqs. (2) may be used to determine the celestial coordinates. Note that Eq. (8) contains an ambiguity in the sign of the inverse cosine and that all three indicate that some combinations of ϕ_0, θ_0, α_0, δ_0, and ϕ_p are not allowed. For these projections, we must therefore adopt additional conventions:

1. Eqs. (9) and (10) indicate that α_p is undefined when $\delta_0 = \pm 90°$. This simply represents the longitude singularity at the pole and forces us to define $\alpha_\mathrm{p} = \alpha_0$ in this case.

2. If $\delta_{\rm p} = \pm 90°$ then the longitude at the native pole is $\alpha_{\rm p} = \alpha_0 - \phi_0 + \phi_{\rm p} - 180°$ for $\delta_{\rm p} = 90°$ and $\alpha_{\rm p} = \alpha_0 + \phi_0 - \phi_{\rm p}$ for $\delta_{\rm p} = -90°$.
3. Some combinations of ϕ_0, θ_0, δ_0, and $\phi_{\rm p}$ produce an invalid argument for the $\cos^{-1}()$ in Eq. (8). This is indicative of an inconsistency for which there is no solution for $\delta_{\rm p}$. Otherwise Eq. (8) produces two solutions for $\delta_{\rm p}$. Valid solutions are ones that lie in the range $-90°$ to $+90°$, and it is possible in some cases that neither solution is valid.
   Note, however, that if $\phi_0 = 0$, as is usually the case, then when `LONPOLE`*s* ($\equiv \phi_{\rm p}$) takes its default value of 0 or 180° (depending on θ_0) then *any* combination of δ_0 and θ_0 produces a valid argument to the $\cos^{-1}()$ in Eq. (8), and at least one of the solutions is valid.
4. Where Eq. (8) has two valid solutions the one closest to the value of the new FITS keyword `LATPOLE`*s* is chosen. It is acceptable to set `LATPOLE`*s* to a number greater than $+90°$ to choose the northerly solution (the default if `LATPOLE`*s* is omitted), or a number less than $-90°$ to select the southern solution.
5. Eq. (8) often only has one valid solution (because the other lies outside the range $-90°$ to $+90°$). In this case `LATPOLE`*s* is ignored.
6. For the special case where $\theta_0 = 0$, $\delta_0 = 0$, and $\phi_{\rm p} - \phi_0 = \pm 90°$ then $\delta_{\rm p}$ is not determined and `LATPOLE`*s* specifies it completely. `LATPOLE`*s* has no default value in this case.

These rules governing the application of Eqs. (8), (9) and (10) are certainly the most complex of this formalism. FITS writers are well advised to check their values of ϕ_0, θ_0, α_0, δ_0, and $\phi_{\rm p}$ against them to ensure their validity.

2.5. Comparison with linear coordinate systems

It must be stressed that the coordinate transformation described here differs from the linear transformation defined by Wells et al. (1981) even for some simple projections where at first glance they may appear to be the same. Consider the plate carrée projection defined in Sect. 4.2.3 with $\phi = x$, $\theta = y$ and illustrated in Fig.15. Figure 1 shows that while the transformation from (x, y) to (ϕ, θ) may be linear (in fact identical), there still remains the non-linear transformation from (ϕ, θ) to (α, δ). Hence the linear coordinate description defined by the unqualified `CTYPE`*js* pair of '`RA`', '`DEC`' which uses the Wells et al. prescription will generally differ from that of '`RA---CAR`', '`DEC---CAR`' with the same `CRVAL`*js*, etc. If `LONPOLE`*s* assumes its default value then they will agree to first order at points near the reference point but gradually diverge at points away from it.

Figure 2 illustrates this point for a plate carrée projection with reference coordinates of $8^{\rm hr}$ right ascension and $+60°$ declination and with $\phi_{\rm p} = 0$. It is evident that since the plate carrée has $(\phi_0, \theta_0) = (0, 0)$, a non-oblique graticule may only be obtained by setting $\delta_0 = 0$ with $\phi_{\rm p} = 0$. It should also be noted that where a larger map is to be composed of tiled submaps the coordinate description of a submap should only differ in the value of its reference pixel coordinate.

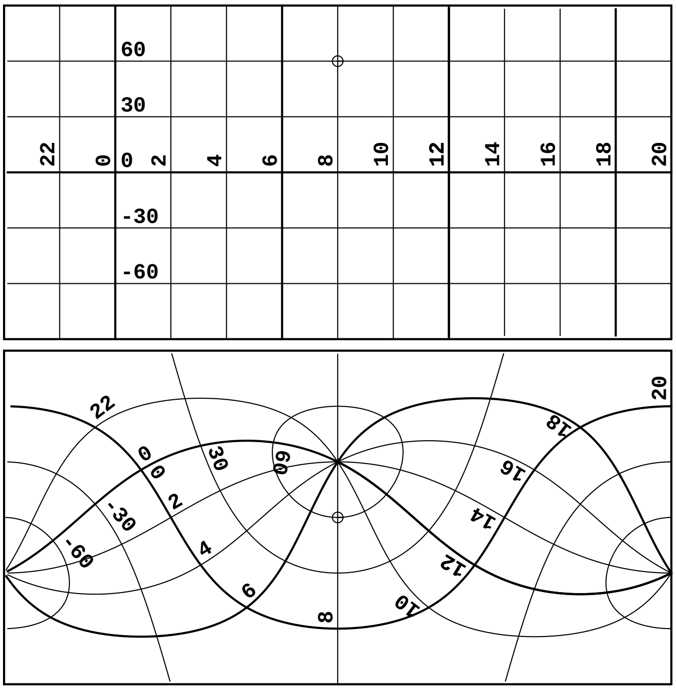

Fig. 2. A linear equatorial coordinate system (top) defined via the methods of Wells et al. (1981), and the corresponding oblique system constructed using the methods of this paper. The reference coordinate for each is at right ascension $8^{\rm hr}$, declination $+60°$ (marked). The two sets of FITS header cards differ only in the `CTYPE`*js*. The non-oblique graticule could be obtained in the current system by setting $(\alpha_0, \delta_0, \phi_{\rm p}) = (120°, 0, 0)$.

2.6. Change of coordinate system

A change of coordinate system may be effected in a straightforward way if the transformation from the original system, (α, δ), to the new system, (α', δ'), and its inverse are known. The new coordinates of the reference point, (α'_0, δ'_0) are obtained simply by transforming (α_0, δ_0). To obtain $\phi'_{\rm p}$, first transform the coordinates of the pole of the new system to the original celestial system and then transform the result to native coordinates via Eq. (5) to obtain $(\phi'_{\rm p}, \theta'_{\rm p})$. As a check, compute $\delta'_{\rm p}$ via Eq. (8) and verify that $\theta'_{\rm p} = \delta'_{\rm p}$.

2.7. Encapsulation

In order to encapsulate all required transformation parameters completely within the recognized world coordinate system (WCS) header cards, the values of `LONPOLE`*s* and `LATPOLE`*s* may be recorded as `PV`*j*`_2`*s* and `PV`*j*`_3`*s*, attached to *longitude* coordinate *j*, and these take precedence where a conflict arises.

We recommend that FITS writers should always include the PVj_0s, PVj_1s, PVj_2s, and PVj_3s cards in the header, even if only to denote the correct use of the default values.

3. Celestial coordinate systems

While the second half of the CTYPEjs specify the transformations to be applied in computing the spherical coordinate pair, (α, δ), the first half of the CTYPEjs identify the celestial system with which (α, δ) are to be associated. In this sense the CTYPEjs contain an active half which drives the transformation and a passive half which simply describes the results.

Following the AIPS convention, an equatorial coordinate pair is denoted by 'RA--' and 'DEC-', and other celestial systems are of the form 'xLON' and 'xLAT' for longitude and latitude pairs, where x = G for galactic[3], E for ecliptic, H for helioecliptic[4], and S for supergalactic coordinates. Since representation of planetary, lunar and solar coordinate systems could exceed the 26 possibilities of the AIPS convention we also here allow 'yzLN' and 'yzLT' pairs. Additional values of x and yz will undoubtedly be defined. A basic requirement, however, is that the coordinate system be right-handed and that the pole be at latitude +90°. A coordinate pair such as azimuth and zenith distance would have to be represented as a negative azimuth and elevation with implied conversion. In some situations negation of the longitude coordinate may be implemented via a sign reversal in the appropriate elements of the CDj_is matrix. It remains the responsibility of the authors of new coordinate system types to define them properly and to gain general recognition for them from the FITS community. However, FITS interpreters should be able to associate coordinate pairs even if the particular coordinate system is not recognized.

Paper I clarified that, while the subscript of the CRPIXis and the i subscript of the CDj_is refer to pixel coordinate elements, the j subscript of the CDj_is and the subscripts of CTYPEjs and CRVALjs refer to world coordinate elements. However we now have three different sets of world coordinates, (x, y), (ϕ, θ) and (α, δ). In line with the AIPS convention this leads us to associate x, ϕ, and α on the one hand, and y, θ, and δ on the other. This simply means, for example, that if CTYPE3 = 'GLON-AIT', then the third element of the intermediate world coordinate calculated via Eq. (1) corresponds to what we have been calling the x-coordinate in the plane of projection, the association being between α and x. In this way pairs of CTYPEjs with complementary left halves and matching right halves define which elements of the intermediate world coordinate vector form the plane of projection.

[3] "New" galactic coordinates are assumed here, Blaauw et al. (1960). Users of the older system or future systems should adopt a different value of x and document its meaning.

[4] Both ecliptic and helioecliptic systems have the ecliptic plane as the zero latitude plane. The reference point for ecliptic longitude is the vernal equinox while that for helioecliptic longitude is the sun vector.

Table 1. Allowed values of RADESYSs.

RADESYSs	Definition
'ICRS'	International Celestial Reference System
'FK5'	mean place, new (IAU 1984) system
'FK4'	mean place, old (Bessell-Newcomb) system
'FK4-NO-E'	mean place, old system but without e-terms
'GAPPT'	geocentric apparent place, IAU 1984 system

3.1. Equatorial and ecliptic coordinates

Several systems of equatorial coordinates (right ascension and declination) are in common use and each is parameterized by time. In particular, mean equatorial coordinates are defined in terms of the epoch (i.e. instant of time) of the mean equator and equinox (i.e. pole and origin of right ascension). The same applies for ecliptic coordinate systems. Several additional keywords are therefore required to specify these systems fully. We introduce the new keyword RADESYSs containing a character string to specify the particular system. Recognized values are given in Table 1. Apart from FK4-NO-E these keywords are applicable to ecliptic as well as equatorial coordinates.

Wells et al. (1981) introduced the keyword EPOCH to mean the epoch of the mean equator and equinox. However we here replace it with EQUINOXs since the word "epoch" is often used in astrometry to refer to the time of observation. The new keyword takes preference over the old one if both are given. Note that EQUINOXs applies to ecliptic as well as to equatorial coordinates.

For RADESYSs values of FK4 and FK4-NO-E, any stated equinox is Besselian and, if neither EQUINOXs nor EPOCH are given, a default of 1950.0 is to be taken. For FK5, any stated equinox is Julian and, if neither keyword is given, it defaults to 2000.0.

If the EQUINOXs keyword is given it should always be accompanied by RADESYSs. However, if it should happen to appear by itself then RADESYSs defaults to FK4 if EQUINOXs $<$ 1984.0, or to FK5 if EQUINOXs $\geq$ 1984.0. Note that these defaults, while probably true of older files using the EPOCH keyword, are not required of them.

Geocentric apparent equatorial and ecliptic coordinates (GAPPT) require the epoch of the equator and equinox of date. This will be taken as the time of observation rather than EQUINOXs. The time of observation may also be required for other astrometric purposes, for example, to specify when the mean place was correct in accounting for proper motion, including "fictitious" proper motions in the conversion between the FK4 and FK5 systems. The old DATE-OBS keyword may be used for this pur-