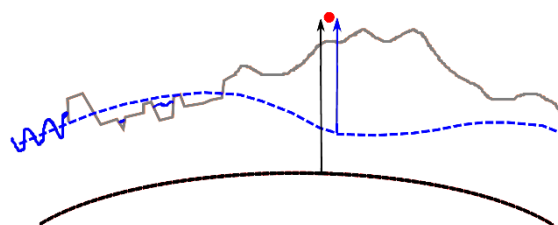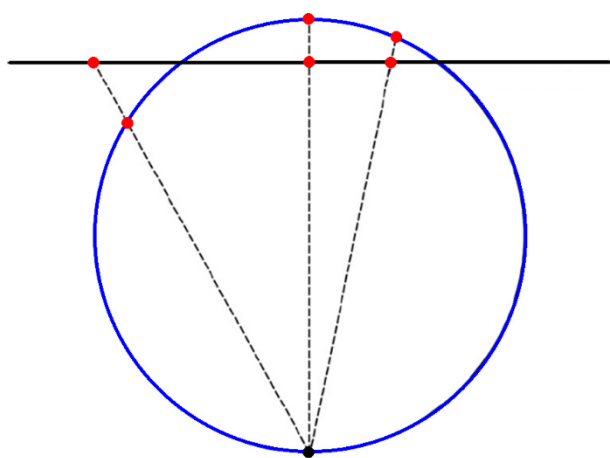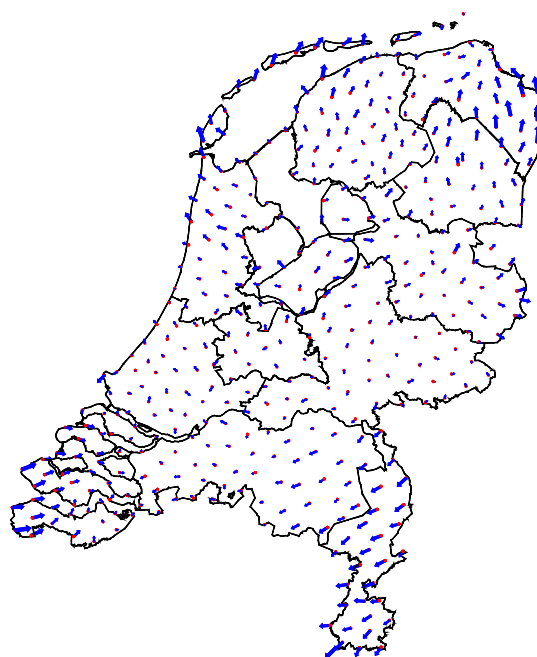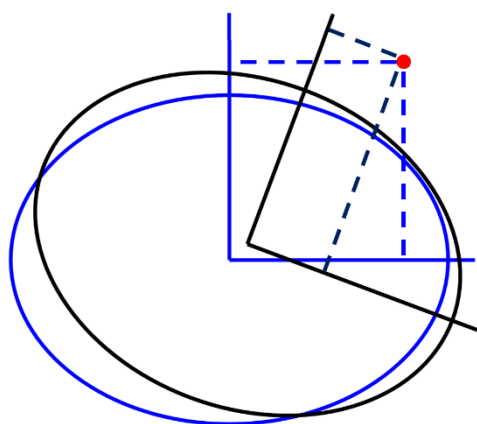# RDNAPTRANS™2018

Coordinate transformation to and from *Stelsel van de Rijksdriehoeksmeting* and *Normaal Amsterdams Peil*

Jochem Lesparre, Lennard Huisman, Bas Alberts and Huib de Ligt
Nederlandse Samenwerking Geodetische Infrastructuur

Content

1 Introduction

1.1 Coordinate systems in the Netherlands

The European Terrestrial Reference System 1989 (ETRS89) is the official 3D coordinate system of the Netherlands and Europe. ETRS89 is linked to the International Terrestrial Reference System (ITRS) by a time-dependant coordinate transformation. National coordinate systems in Europe are linked to ETRS89.

Coordinates in the Dutch Stelsel van de Rijksdriehoeksmeting (RD) are the most-frequently used 2D coordinates on land and internal waters of the European part of the Netherlands for storage and exchange of geo-information (often also including some data outside this area on the North Sea, in Belgium and Germany). RD coordinates are defined by the official transformation from ETRS89 coordinates. Maintaining reference points for ETRS89 in the Netherlands and the transformation to RD coordinates are legal responsibilities of Kadaster.

Heights relative to Normaal Amsterdams Peil (NAP) are the official and the most-frequently used heights on land and internal waters of the European part of the Netherlands for storage and exchange of geo-information. NAP is defined by the published heights of the height benchmarks, which is a legal responsibility of Rijkswaterstaat. Ellipsoidal heights in ETRS89 can be transformed with the quasi-geoid model to NAP with a precision higher than ETRS89 coordinates obtained with most GNSS measurements.

For storage and exchange of geo-information at sea, the International Hydrographic Organisation (IHO) has agreed upon World Geodetic System 1984 (WGS 84), the EU has agreed upon ETRS89. The difference is presently (2019) approximately 0.75 m and increasing by 0.024 m per year. Since ellipsoidal heights in ETRS89 are only geometrical and have no physical meaning, other height references are used too. Dienst der Hydrografie uses ETRS89 coordinates with the maritime reference surface Lowest Astronomical Tide (LAT) as chart datum and publishes the relation of LAT with the quasi-geoid model and the relation of LAT with the ETRS89 ellipsoid for the Dutch part of the North Sea.
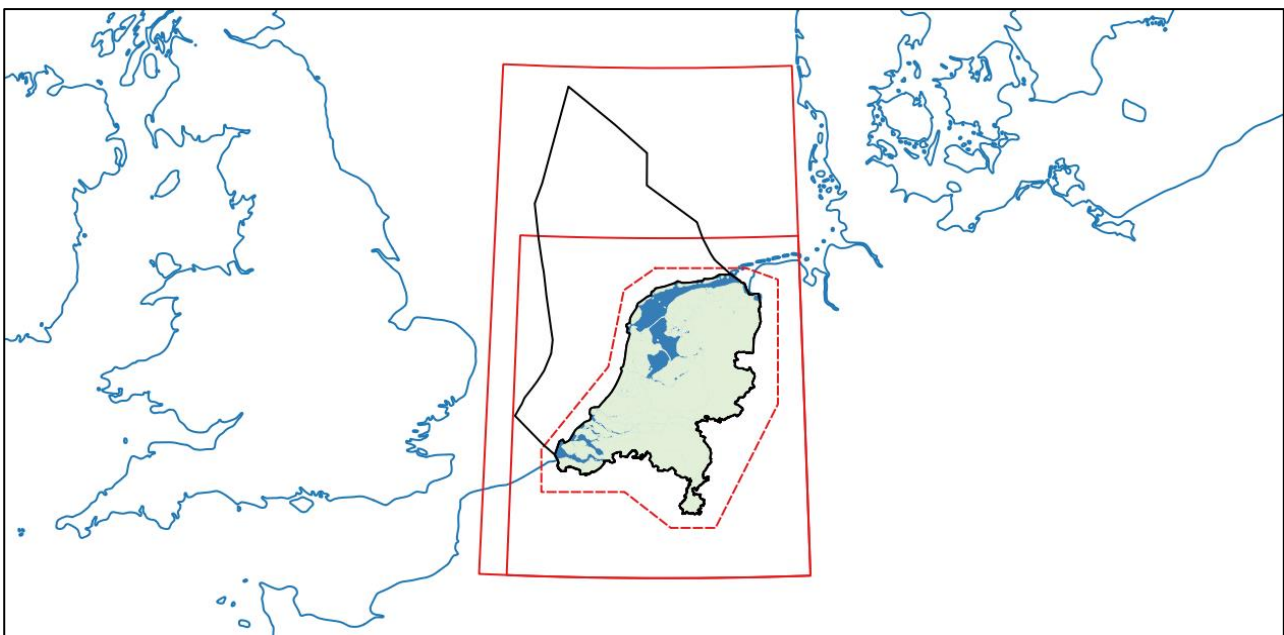
Figure 1.1.1. Validity area of RDNAPTRANS™ versions 2000, 2004 and 2008 (dashed red line) and bounds of the RDNAPTRANS™2018 grid and dense subgrid of the NTV2 and GeoTIFF file formats (solid red lines) relative to the administrative borders and EEZ of the Netherlands (coloured area and black lines).

Geo-information in ETRS89 coordinates cannot be visualised on a paper map or a map on a computer screen without a map projection. RD coordinates are very suitable as map projection for visualisation of geo-information in ETRS89

coordinates for the entire European part of the Netherlands, including the Dutch Exclusive Economic Zone (EEZ) of the North Sea, and its surroundings (Figure 1.1.1).

1.2 Coordinate transformation

The official coordinate transformation between European ETRS89 coordinates and Dutch coordinates in RD and NAP is called RDNAPTRANS™. It is determined by the partnership of Kadaster, Rijkswaterstaat and Dienst der Hydrografie, working together under the name Nederlandse Samenwerking Geodetische Infrastructuur (NSGI). The name RDNAPTRANS is a registered trademark and can only be used after written permission. Permission is granted for applications where the transformation is implemented correctly. The correctness of a implementation of the transformation can be checked with the Validation Service available on the NSGI website. In some software, the transformation method is implemented correctly, but the software needs to be configured or the right options need to be selected to support RDNAPTRANS™. Instructions for configuration of specific software packages are out of the scope of this document.

This document is restricted to the transformation between ETRS89 coordinates, RD coordinates and NAP height. Transformation between RD coordinates and coordinates in worldwide systems like International Terrestrial Reference System (ITRS), World Geodetic Reference System 1984 (WGS 84) and recommended or often used map projections like Lambert Conformal Conic projection (LCC), Lambert Azimuthal Equal-Area projection (LAEA), Universal Transverse Mercator projection (UTM) and Web Mercator projection are not part of RDNAPTRANS™ and therefore out of scope of this document.

Since the introduction of RDNAPTRANS™ in 2000, several new versions have been released. The current version is RDNAPTRANS™2018. This version contains a new datum transformation based on the updated ETRS89 coordinates of realisation ETRF2000 (epoch 2017.50). Next to this, a new and more accurate quasi-geoid grid model is used. This NLGEO2018 quasi-geoid model covers a larger area including a large part of the North Sea (Figure 1.1.1). A change with big impact is the use of a new data format of the grid files and a corresponding transformation procedure that changes the order of the steps of the transformation and uses a fixed height in the datum transformation (Figure 1.2.1). As a result, the transformation is now possible conform a *de facto* standard by including the datum transformation in the correction grid (variant 2). This allows straightforward implementation in software like GIS packages and can resolve current problems due to incorrect implementations of the transformation.
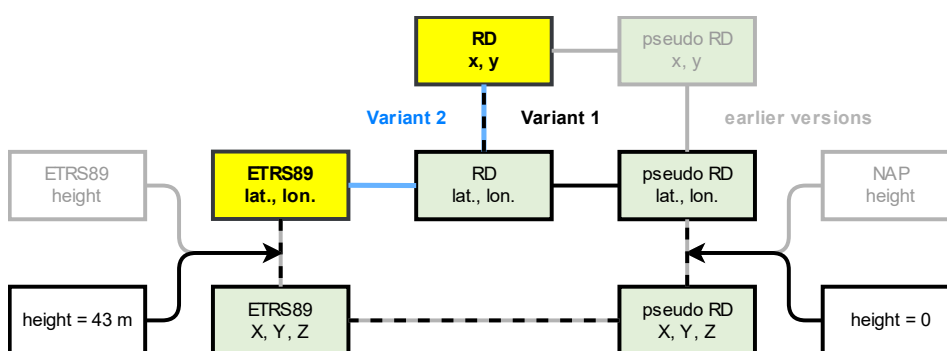
Figure 1.2.1. Transformation procedure between RD and ETRS89 using RDNAPTRANS™2018 implementation variant 1 (black lines) and implementation variant 2 (blue lines) compared to RDNAPTRANS™2008 and earlier versions (grey lines).

Within the administrative borders of the Netherlands (Figure 1.1.1), the differences in the resulting horizontal coordinates due to the changes from version 2008 to 2018 of RDNAPTRANS™ are at maximum 0.010 m at sea level (zero NAP height), and at maximum an additional 0.001 m per 50 m height above or below zero NAP height. The new correction grid has a different sampling in ellipsoidal coordinates. With this resampling, discontinuities in the correction grid are

smoothed, especially outside the administrative borders of the Netherlands, to allow bilinear interpolation. As a result, changes in the resulting horizontal coordinates up to 0.05 m occur occasionally in Belgium, Germany and the Dutch EEZ of the North Sea. To use the same bounding box around the Netherlands including the Dutch EEZ as the quasi-geoid grid model for the correction grid, the correction grid has been faded out to zero correction for a seamless land-sea transition. This introduces latitude and longitude corrections up to 0.25 m in areas outside the former validity area of RDNAPTRANS™ where no corrections were defined before (Figure 1.1.1).

The maximum change in the transformed height coordinates due to the updated quasi-geoid grid model is about 0.03 m within the administrative borders of the Netherlands (Figure 1.1.1) and up to 0.05 m outside these borders.

The recommended ETRS89 realisation is ETRF2000 at epoch 2010.50 (AGRS2010). When using RDNAPTRANS™2018 it is important to use this realisation and epoch, especially for the height. For applications demanding high accuracy, it is recommended to obtain the NAP height of the point of interest by levelling to nearby NAP benchmarks.

This document describes the formulas needed to implement the transformation according to RDNAPTRANS™2018. First, the transformation from ETRS89 to RD and NAP is described (Section 2). Next, the transformation in the other direction, from RD and NAP to ETRS89, is given (Section 3). The transformations in both directions consist of the same steps, but are described separately to explain the differences of the inverse formulas of each step. The transformation between ETRS89 and horizontal RD coordinates is independent from the transformation between ETRS89 and NAP heights.

Additional information is available in the appendices, for instance more detailed diagrams of the transformation procedure are given (Appendix 1). The most recent version of this document and grid files for the implementation of the transformation are available for download (Appendix 2). No C++ source code is provided for RDNAPTRANS™2018, but instructions for implementation with the open source library PROJ are given (Appendix 3). A form is provided to apply for permission to use the trademark RDNAPTRANS by software developers that implemented the official transformation (Appendix 4).
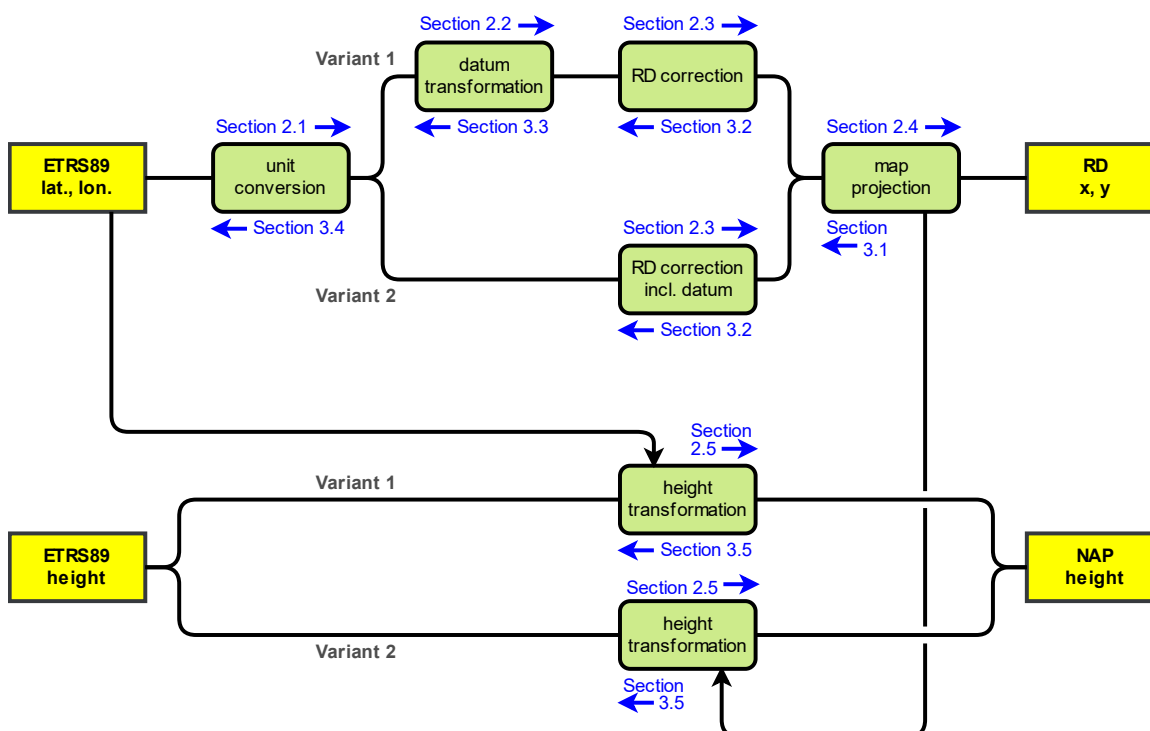
Figure 1.2.2. Guide to the sections (blue) of this document for the different steps of the transformation procedure.

There are two variants for the implementation of the horizontal component of RDNAPTRANS™2018 and two variants for the vertical component (Figure 1.2.2). Implementation variant 1 applies the datum transformation as a separate step using a 3D similarity transformation. Implementation variant 2 includes the datum transformation in the correction grid and uses a different quasi-geoid grid for the height transformation. The advantage of implementation variant 1 is that it has no strict bounds for the area where horizontal coordinates can be transformed correctly. The disadvantage is that many software packages do not support implementation variant 1 for the horizontal component. Implementation variant 2 for the horizontal component is supported by more software but can only be used within the bounds of the correction grid (Figure 1.1.1). The difference in the resulting coordinates between the two variants is well below 0.0010 m within the bounds of the RDNAPTRANS™2018 grids. Although transformation at sea and even outside the grid bounds is possible, the scale factor of the map projection increases rapidly and also the precision of transformation back and forth deteriorates. There are bounds to the recommended use of RD and NAP at sea and outside the Netherlands.

The Transformation API at the NSGI website is available for transformation with RDNAPTRANS™ for data sets of limited size. For larger data sets it is more efficient to implement the transformation formulas of RDNAPTRANS™, e.g. by using the PROJ library. After implementing the transformation, it can be tested using the Validation Service of NSGI.

2 Transformation from ETRS89 to RD and NAP

2.1 Notation in degrees, minutes and seconds

ETRS89 coordinates are commonly expressed in ellipsoidal geographic coordinates latitude, longitude and ellipsoidal height. The latitude and longitude of a point of interest can be given as decimal degrees (D.D), as integer degrees and decimal minutes (D M.M), or as integer degrees, integer minutes and decimal seconds (D M S.S). The hemisphere is indicated with the abbreviation "N" or "S" for latitude north or south of the equator and "E" or "W" for longitude east or west of the Greenwich meridian. The hemisphere can also be indicated by the sign (+/–) of the number in decimal degrees (d.d), where latitudes south of the equator and longitudes west of the Greenwich meridian get a negative sign. The height is given in metres. For 2D applications, the height can be omitted.

The number of needed decimals in degrees, minutes and seconds for a resolution corresponding to approximately 0.01 m, are given in the examples below.

Examples of notation of 2D ellipsoidal geographic coordinates

D M S.S: 52° 5′ 29.870″ N, 5° 7′ 18.141″ E   (Dutch notation: 52° 5′ 29,870″ NB; 5° 7′ 18,141″ OL)
D M.M: 52° 5.49783′ N, 5° 7.30235′ E   (Dutch notation: 52° 5,49783′ NB; 5° 7,30235′ OL)
D.D: 52.0916305° N, 5.1217059° E   (Dutch notation: 52,0916305° NB; 5,1217059° OL)
d.d: +52.0916305°, +5.1217059°   (Dutch notation: +52,0916305°; +5,1217059°)

NB: Variations on these notation formats are used too.

To transform the coordinates of a point of interest, its ETRS89 coordinates must be converted first to decimal degrees or to radians (Formula 2.1), depending on the type of goniometry functions used.

Formula 2.1a. Conversion of notation from coordinates in degrees, minutes and seconds to decimal degrees

$$d = \begin{cases} D + M/60 + S/3600 & \text{if } C = \text{"N" or } C = \text{"E"} \\ -D - M/60 - S/3600 & \text{if } C = \text{"S" or } C = \text{"W"} \end{cases}$$

Formula 2.1b. Unit conversion from decimal degrees to radians

$$r = d\,\pi/180°$$

where:
d	signed decimal degrees latitude or longitude (°)
D	unsigned integer (or decimal) degrees part of latitude or longitude (°)
M	unsigned integer (or decimal) minutes part of latitude or longitude (′)
S	unsigned decimal seconds part of latitude or longitude (″)
C	hemisphere code (character)
r	signed decimal degrees latitude or longitude (rad)
$\pi = 3.14159265358$	mathematical constant (dimensionless)

NB: The equal sign is used for all expressions resulting in a transformation accuracy of 0.0001 m or better.

2.2   Datum transformation

2.2.1   Conversion to geocentric Cartesian coordinates

The ellipsoidal geographic ETRS89 coordinates of a point of interest must be converted to geocentric Cartesian ETRS89 coordinates (Formula 2.2.1) to be able to apply a 3D similarity transformation. This is only needed for implementation variant 1, for variant 2 the datum transformation (Section 2.2.1, 2.2.2 and 2.2.3) is included in the correction grid (Section 2.3).

Formula 2.2.1a, b, c and d. Conversion from ellipsoidal geographic coordinates to geocentric Cartesian coordinates

$$R_N = \frac{a}{\sqrt{1 - e^2 \, (\sin \varphi)^2}}$$

$$X = (R_N + h) \cos \varphi \cos \lambda$$
$$Y = (R_N + h) \cos \varphi \sin \lambda$$
$$Z = \left(R_N \, (1 - e^2) + h \right) \sin \varphi$$

where:

φ, λ	latitude and longitude of point of interest in source datum (rad or °)
$h = h_0$	ellipsoidal height approximately corresponding to 0 m in NAP (m)
X, Y, Z	geocentric Cartesian coordinates of point of interest in source datum (m)
R_N	second (east-west) principal radius of curvature of ellipsoid at point of interest (m)
$e^2 = f(2 - f)$	squared first eccentricity of ellipsoid used in source datum (dimensionless)
a, f	parameters of the ellipsoid used in source datum

NB: Unlike conventional use, including RDNAPTRANS™ versions before RDNAPTRANS™2018, a fixed height is used.

The parameters needed for the conversion to geocentric Cartesian ETRS89 coordinates (Formula 2.2.1) are listed below.

Parameters of GRS80 ellipsoid

$a = 6378137$ m	half major (equator) axis of GRS80 ellipsoid
$f = 1/298.257222101$	flattening of GRS80 ellipsoid (dimensionless)

Unlike conventional use in transformations, including RDNAPTRANS™ versions before RDNAPTRANS™2018, a fixed ellipsoidal height is used instead of the actual height of the point of interest. As a result, points with the same latitude and longitude in ETRS89 that differ in height get exactly the same RD coordinates. This enables 2D transformation between ETRS89 and RD and straightforward implementation in software like GIS packages. However, it introduces small differences between back and forth transformation. These differences are below 0.0010 m up to 500 km outside the bounds of the RDNAPTRANS™2018 grids.

The fixed ellipsoidal height used in the conversion to geocentric Cartesian ETRS89 (Formula 2.2.1) is listed below.

Parameter for fixed ellipsoidal ETRS89 height

$h_0 = 43 \, m$	ellipsoidal ETRS89 height approximately corresponding to 0 m in NAP (m)

Do not use these geocentric Cartesian ETRS89 coordinates for other purposes than RDNAPTRANS™, as the use of a fixed height might be not appropriate and the actual height of the point of interest should be used. Also, do not start the transformation to RD coordinates with geocentric Cartesian ETRS89 coordinates obtained in a different way than by conversion with the fixed height (e.g. geocentric Cartesian ETRS89 coordinates obtained directly from GNSS measurements). Instead, convert such geocentric Cartesian ETRS89 coordinates first to ellipsoidal geographic ETRS89 coordinates (Section 3.3) and then perform the transformation to RD coordinates (Chapter 2).

2.2.2 3D similarity transformation

The formula for a 3D similarity transformation must be applied to the geocentric Cartesian ETRS89 coordinates of the point of interest (Formula 2.2.2). The obtained geocentric Cartesian coordinates are in the geodetic datum of RD. Since the name RD is often used for projected coordinates only, the geodetic datum is often referred to as RD Bessel or just Bessel, even though geocentric Cartesian coordinates do not use the Bessel ellipsoid.

Formula 2.2.2a. Rigorous 3D similarity transformation of geocentric Cartesian coordinates

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = s \begin{bmatrix} R_{1,1} & R_{1,2} & R_{1,3} \\ R_{2,1} & R_{2,2} & R_{2,3} \\ R_{3,1} & R_{3,2} & R_{3,3} \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} + \begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix}$$

Formula 2.2.2b, c and d. Equivalent without matrix notation

$$X_2 = s\left(R_{1,1}\,X_1 + R_{1,2}\,Y_1 + R_{1,3}\,Z_1\right) + t_X$$
$$Y_2 = s\left(R_{2,1}\,X_1 + R_{2,2}\,Y_1 + R_{2,3}\,Z_1\right) + t_Y$$
$$Z_2 = s\left(R_{3,1}\,X_1 + R_{3,2}\,Y_1 + R_{3,3}\,Z_1\right) + t_Z$$

where:

X_1, Y_1, Z_1	geocentric Cartesian coordinates of point of interest in source datum (m)
X_2, Y_2, Z_2	geocentric Cartesian coordinates of point of interest in target datum (m)
$s = 1 + \delta$	scale factor (dimensionless)
$R_{1,1} = \cos\gamma\cos\beta$	element of rigorous 3D rotation matrix (dimensionless)
$R_{1,2} = \cos\gamma\sin\beta\sin\alpha + \sin\gamma\cos\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$R_{1,3} = -\cos\gamma\sin\beta\cos\alpha + \sin\gamma\sin\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$R_{2,1} = -\sin\gamma\cos\beta$	element of rigorous 3D rotation matrix (dimensionless)
$R_{2,2} = -\sin\gamma\sin\beta\sin\alpha + \cos\gamma\cos\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$R_{2,3} = \sin\gamma\sin\beta\cos\alpha + \cos\gamma\sin\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$R_{3,1} = \sin\beta$	element of rigorous 3D rotation matrix (dimensionless)
$R_{3,2} = -\cos\beta\sin\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$R_{3,3} = \cos\beta\cos\alpha$	element of rigorous 3D rotation matrix (dimensionless)
$t_X, t_Y, t_Z, \alpha, \beta, \gamma, \delta$	7 parameters of 3D similarity transformation

NB: A frequently used alternative formula with an approximated rotation matrix is less accurate but acceptable as the difference is well below 0.0010 m for the small rotation angles between ETRS89 and RD.

The 7 parameters for the 3D similarity transformation (Formula 2.2.2) from ETRS89 to RD are listed below. The parameters could be computed from the inverse parameters too (Section 3.3). Often, only the parameters of one transformation direction are set in software.

$t_X = -565.7346$ m   translation in direction of X axis
$t_Y = -50.4058$ m   translation in direction of Y axis
$t_Z = -465.2895$ m   translation in direction of Z axis
$\alpha = -1.91513 \cdot 10^{-6}$ rad $= -0.395023''$ rotation angle around X axis
$\beta = +1.60365 \cdot 10^{-6}$ rad $= +0.330776''$ rotation angle around Y axis
$\gamma = -9.09546 \cdot 10^{-6}$ rad $= -1.876073''$ rotation angle around Z axis
$\delta = -4.07242 \cdot 10^{-6}$ scale difference (dimensionless)

NB: The three rotation parameters (α, β, γ) are for use with the formulas above according to the "coordinate frame" convention, that assumes the positive rotation direction to be anticlockwise. The signs of the three rotation parameters must be inverted for the alternative "position vector" convention.

2.2.3 Conversion from geocentric Cartesian coordinates

After the 3D similarity transformation, the geocentric Cartesian Bessel coordinates of the point of interest must be converted back to ellipsoidal geographic Bessel coordinates (Formula 2.2.3). The ellipsoidal height is not given in the formula, as it is not used. The formula for latitude contains the latitude itself. The initial value for the latitude should be used to obtain an approximation of the latitude. This approximate value is then used to obtain an improved approximation. The latitude is computed iteratively until the difference between subsequent iterations becomes smaller than the precision threshold.

Formula 2.2.3a, b, c, d and e. Conversion from geocentric Cartesian coordinates to ellipsoidal geographic coordinates

$$\varphi_{i=0} = 0$$

$$R_N(\varphi_i) = \frac{a}{\sqrt{1 - e^2 \, (\sin \varphi_i)^2}}$$

$$\varphi_{i+1} = \begin{cases} \arctan\left(\dfrac{Z + e^2 \, R_N(\varphi_i) \sin \varphi_i}{\sqrt{X^2 + Y^2}} \right) & \text{if } X \neq 0 \text{ or } Y \neq 0 \\ +90° & \text{if } X = 0 \text{ and } Y = 0 \text{ and } Z \geq 0 \\ -90° & \text{if } X = 0 \text{ and } Y = 0 \text{ and } Z < 0 \end{cases}$$

$$\varphi = \varphi_{i+1} \quad \text{if } |\varphi_{i+1} - \varphi_i| < \varepsilon$$

$$\lambda = \begin{cases} \arctan(Y/X) & \text{if } X > 0 \\ \arctan(Y/X) + 180° & \text{if } X < 0 \text{ and } Y \geq 0 \\ \arctan(Y/X) - 180° & \text{if } X < 0 \text{ and } Y < 0 \\ +90° & \text{if } X = 0 \text{ and } Y > 0 \\ -90° & \text{if } X = 0 \text{ and } Y < 0 \\ 0° & \text{if } X = 0 \text{ and } Y = 0 \end{cases}$$

where:
X, Y, Z geocentric Cartesian coordinates of point of interest in target datum (m)
$\varphi_{i=0}$ initial approximate latitude of point of interest in target datum (rad or °)
φ_i improved latitude of point of interest in target datum after i iterations (rad or °)
φ, λ latitude and longitude of point of interest in target datum (rad or °)
                         NB: The latitude needs to be computed iteratively until precision (ε) is met.
$\varepsilon = 0.000\,000\,001° = 2 \cdot 10^{-11}$ rad   precision threshold for termination of iteration, corresponding to 0.0001 m
$R_N(\varphi_i)$ second (east-west) principal radius of curvature of ellipsoid at point of interest (m)
                         NB: The radius of curvature needs to be recomputed in each iteration.

$e^2 = f(2 - f)$ first eccentricity squared of ellipsoid used in target datum (dimensionless)

a, f parameters of the ellipsoid used in target datum

$|...|$ mathematical function for absolute value

NB: To prevent arithmetic overflow due to division of a large number by a number close to zero, a tolerance should be used to test if the geocentric Cartesian coordinates are zero. Rounding the coordinates in advance to 0.0001 m suffices.

The parameters needed in the conversion to ellipsoidal geographic Bessel coordinates (Formula 2.2.3) are listed below.

Parameters of Bessel 1841 ellipsoid

$a = 6377397.155$ m   half major (equator) axis of Bessel 1841 ellipsoid

$f = 1/299.1528128$ flattening of Bessel 1841 ellipsoid (dimensionless)

Do not use Bessel coordinates for other purposes than RDNAPTRANS™ to avoid confusion with ETRS89 coordinates.

2.3   RD correction

2.3.1   Bilinear correction grid interpolation

The ellipsoidal geographic coordinates of a point of interest obtained by datum transformation of implementation variant 1 are pseudo Bessel coordinates. Due to the error propagation of measurement noise of the original (1888–1928) measurements of RD, the pseudo Bessel coordinates must be corrected up to 0.25 m to obtain real Bessel coordinates. For implementation variant 2, the datum transformation is included in the correction grid (Section 2.3.4).

The corrections are obtained from a regular grid of values for latitude correction and a regular grid of values for longitude correction, using bilinear interpolation (Formula 2.3.1). Unit conversion (Formula 2.1) might be needed, as the correction values are given in degrees or arcseconds, depending on the data format of the correction grid.

Formula 2.3.1a, b, c, d, and e. Bilinear correction grid interpolation and test for points out of bounds

$$b = \begin{cases} 1 & \text{if } \varphi_{min} \leq \varphi \leq \varphi_{max} \text{ and } \lambda_{min} \leq \lambda \leq \lambda_{max} \\ 0 & \text{if } \varphi < \varphi_{min} \text{ or } \varphi > \varphi_{max} \text{ or } \lambda < \lambda_{min} \text{ or } \lambda > \lambda_{max} \end{cases}$$

$$\bar{\varphi} = \frac{\varphi - \varphi_{min}}{\Delta_\varphi}$$

$$\bar{\lambda} = \frac{\lambda - \lambda_{min}}{\Delta_\lambda}$$

$$c_\varphi(\varphi, \lambda) = \begin{cases} \begin{bmatrix} \bar{\varphi} - \lfloor\bar{\varphi}\rfloor & \lfloor\bar{\varphi}\rfloor + 1 - \bar{\varphi} \end{bmatrix} \begin{bmatrix} g_\varphi(i_{NW}) & g_\varphi(i_{NE}) \\ g_\varphi(i_{SW}) & g_\varphi(i_{SE}) \end{bmatrix} \begin{bmatrix} \lfloor\bar{\lambda}\rfloor + 1 - \bar{\lambda} \\ \bar{\lambda} - \lfloor\bar{\lambda}\rfloor \end{bmatrix} & \text{if } b = 1 \\ c_0 & \text{if } b = 0 \end{cases}$$

$$c_\lambda(\varphi, \lambda) = \begin{cases} \begin{bmatrix} \bar{\varphi} - \lfloor\bar{\varphi}\rfloor & \lfloor\bar{\varphi}\rfloor + 1 - \bar{\varphi} \end{bmatrix} \begin{bmatrix} g_\lambda(i_{NW}) & g_\lambda(i_{NE}) \\ g_\lambda(i_{SW}) & g_\lambda(i_{SE}) \end{bmatrix} \begin{bmatrix} \lfloor\bar{\lambda}\rfloor + 1 - \bar{\lambda} \\ \bar{\lambda} - \lfloor\bar{\lambda}\rfloor \end{bmatrix} & \text{if } b = 1 \\ c_0 & \text{if } b = 0 \end{cases}$$

The correction grid for bilinear interpolation (Formula 2.3.1) is supplied in three data formats, in tab-separated value ASCII text file format (.txt), in binary NTv2 file format (.gsb) and in binary GeoTIFF file format (.tif).

The tab-separated value ASCII text file format has a header line and uses one line per grid point. The columns are latitude of grid point (°), longitude of grid point (°), latitude correction at grid point (°) and longitude correction at grid point (°). The order in which correction values are listed, is from southwest to northeast and starts in west-east direction. The properties of the ASCII text correction grid file are specified by the parameters listed below.

The binary NTv2 file format is supported by many software packages for geo-information. It has some peculiarities however. Correction values, spacing and even grid bounds are given in arcseconds. Longitude values have inverted sign, thus east of the Greenwich meridian is negative. This affects the correction values and the values for the grid bounds. Also the order in which the correction values are given in the NTv2 correction grid file, is from southeast to northwest and starts in east-west direction. When the units are converted (Formula 2.1) and the sign of the values are inverted directly when reading the values from file, the interpolation (Formula 2.3.1) does not need to be adapted. Note that the NTv2 file format contains a precision parameter in metres for each correction value and a header for the parent grid and each subgrid. A more detailed file format description is provided with the list of the associated files of RDNAPTRANS™2018 (Appendix 2).

The binary GeoTIFF file format is supported by many software packages for geo-information. The usage for Geodetic TIFF Grids for datum transformation with a correction grid, as specified by PROJ, is less common. The GeoTIFF files use lossless data compression. This reduces the file size, but this makes reading from the binary file more complicated. Multiple values per grid point are stored in separate bands. The first two bands contain the latitude correction and longitude correction in arcseconds. Band three and four contain the precision in metres. The order in which the correction values are given in the GeoTIFF correction grid file, is from northwest to southeast and starts in west-east direction. When the units are converted (Formula 2.1) directly when reading the values from file, the interpolation (Formula 2.3.1) does not need to be adapted. Note that the GeoTIFF files use the same parent and subgrid as the NTv2 files. Additional file format information is provided with the list of the associated files of RDNAPTRANS™2018 (Appendix 2).

The properties of the NTv2 and GeoTIFF correction grid files are specified by the parameters listed below. In both formats, the file contains two grids, a course parent grid for a large area covering the Netherlands with EEZ and a dense subgrid for a smaller area without EEZ. The course parent grid is not to be used in the area of the dense subgrid.

Parameters of binary NTv2 (.gsb) and GeoTIFF (.tif) correction grid files

Parent grid name: NL_EEZ

$\varphi_{min} = 50° = 180000''$	latitude of southern bound of correction grid
$\varphi_{max} = 56° = 201600''$	latitude of northern bound of correction grid
$\lambda_{min} = 2° = 7200''$	longitude of western bound of correction grid
$\lambda_{max} = 8° = 28800''$	longitude of eastern bound of correction grid
$\Delta_{\varphi} = 0.1° = 360''$	latitude spacing of correction grid, corresponding to about 11 km
$\Delta_{\lambda} = 0.1° = 360''$	longitude spacing of correction grid, corresponding to about 7 km

Subgrid name: NL

$\varphi_{min} = 50° = 180000''$	latitude of southern bound of correction grid
$\varphi_{max} = 54° = 194400''$	latitude of northern bound of correction grid
$\lambda_{min} = 2.5° = 9000''$	longitude of western bound of correction grid
$\lambda_{max} = 8° = 28800''$	longitude of eastern bound of correction grid
$\Delta_{\varphi} = 0.0125° = 45''$	latitude spacing of correction grid, corresponding to about 1.4 km
$\Delta_{\lambda} = 0.02° = 72''$	longitude spacing of correction grid, corresponding to about 1.4 km

NB: These values are expressed in arcseconds (″) and the longitudes have opposite sign in the NTv2 (.gsb) file format. Since GeoTIFF files are often used for raster images and the grid values are valid for the pixel centre in Geodetic TIFF Grids, the bounds of the GeoTIFF may appear sightly larger (half the spacing on all sides of the grid).

RDNAPTRANS™2008 and previous versions of RDNAPTRANS™ used a correction grid in a different binary format and with corrections defined in projected coordinates instead of ellipsoidal geographic coordinates. The NTv2 correction grid file for a precise approximation of RDNAPTRANS™2008 had different bounds and spacing.

The zero correction value to be used in the correction grid interpolation (Formula 2.3.1) outside the bounds of the correction grid, is specified by the parameter listed below.

Parameter of correction grid model (variant 1)

$c_0 = 0.000000000°$ correction value outside bounds of correction grid

Although transformation with implementation variant 1 is possible outside the bounds of the correction grid, results outside the grid bounds are best accompanied with a warning that the RD coordinates are out of bounds.

2.3.2 Determine nearest grid points

To transform the point of interest, the nearest NW, NE, SW and SE grid values are required. Grid values can be read one by one from the binary grid file by direct access or the entire grid of the binary or ASCII text file can be assigned to an array variable first. In both cases, the indices of the required grid values need to be determined. To use the same formulas for computing the indices of an array variable as for the original grid file, the array variable should have the same dimensions and order as the grid file. Thus, using a 1D array for a binary file and a 2D array with one row for each grid point for an ASCII text file. The indices can be computed from the coordinates of the point of interest (Formula 2.3.2). The formulas are different for the three data formats due to the order of the grid values.

Formula 2.3.2a, b, c, d, e, f and g. Determine indices of nearest grid values

$$n_d = \begin{cases} 0 & \text{if reading latitude correction} \\ 1 & \text{if reading longitude correction (or quasi-geoid height)} \end{cases}$$

$$\bar{\varphi} = \frac{\varphi - \varphi_{min}}{\Delta_\varphi}$$

$$\bar{\lambda} = \frac{\lambda - \lambda_{min}}{\Delta_\lambda}$$

$$i_{NW} = \begin{cases} \lceil \bar{\varphi} \rceil \, n_\lambda + \lfloor \bar{\lambda} \rfloor + n_0 & \text{if reading data in order of ASCII text file (or VDatum order)} \\ 4 \lceil \bar{\varphi} + 1 \rceil \, n_\lambda - \lfloor \bar{\lambda} + 1 \rfloor + n_0 + n_d & \text{if reading data in NTv2 order} \\ (n_\varphi - \lceil \bar{\varphi} \rceil - 1) \, n_\lambda + \lfloor \bar{\lambda} \rfloor + n_0 + (1 - n_d) \, n_\varphi \, n_\lambda & \text{if reading data in GeoTIFF order} \end{cases}$$

$$i_{NE} = \begin{cases} \lceil \bar{\varphi} \rceil \, n_\lambda + \lceil \bar{\lambda} \rceil + n_0 & \text{if reading data in order of ASCII text file (or VDatum order)} \\ 4 \lceil \bar{\varphi} + 1 \rceil \, n_\lambda - \lceil \bar{\lambda} + 1 \rceil + n_0 + n_d & \text{if reading data in NTv2 order} \\ (n_\varphi - \lceil \bar{\varphi} \rceil - 1) \, n_\lambda + \lceil \bar{\lambda} \rceil + n_0 + (1 - n_d) \, n_\varphi \, n_\lambda & \text{if reading data in GeoTIFF order} \end{cases}$$

$$i_{SW} = \begin{cases} \lfloor \bar{\varphi} \rfloor \, n_\lambda + \lfloor \bar{\lambda} \rfloor + n_0 & \text{if reading data in order of ASCII text file (or VDatum order)} \\ 4 \lfloor \bar{\varphi} + 1 \rfloor \, n_\lambda - \lfloor \bar{\lambda} + 1 \rfloor + n_0 + n_d & \text{if reading data in NTv2 order} \\ (n_\varphi - \lfloor \bar{\varphi} \rfloor - 1) \, n_\lambda + \lfloor \bar{\lambda} \rfloor + n_0 + (1 - n_d) \, n_\varphi \, n_\lambda & \text{if reading data in GeoTIFF order} \end{cases}$$

$$i_{SE} = \begin{cases} \lfloor \bar{\varphi} \rfloor \, n_\lambda + \lceil \bar{\lambda} \rceil + n_0 & \text{if reading data in order of ASCII text file (or VDatum order)} \\ 4 \lfloor \bar{\varphi} + 1 \rfloor \, n_\lambda - \lceil \bar{\lambda} + 1 \rceil + n_0 + n_d & \text{if reading data in NTv2 order} \\ (n_\varphi - \lfloor \bar{\varphi} \rfloor - 1) \, n_\lambda + \lceil \bar{\lambda} \rceil + n_0 + (1 - n_d) \, n_\varphi \, n_\lambda & \text{if reading data in GeoTIFF order} \end{cases}$$

where:

φ, λ	latitude and longitude of point of interest in real RD Bessel (°)
$\bar{\varphi}, \bar{\lambda}$	normalised latitude and longitude of point of interest (dimensionless)
$i_{NW}, i_{NE}, i_{SW}, i_{SE}$	indices of NW, NE, SW and SE grid values in binary file or 1D array variable or
	line indices of NW, NE, SW and SE grid points in ASCII text file or
	row indices of NW, NE, SW and SE grid points in 2D array variable (dimensionless)
	NB: For a 2D array or ASCII text file, also the right column needs to be selected for reading latitudes (2nd column), longitudes (3rd column) or quasi-geoid heights (2nd column). Note that the first column has index 0 in some code languages.
n_d	constant to set reading of latitude or longitude from NTv2 file
$n_\varphi = 1 + (\varphi_{max} - \varphi_{min})/\Delta_\varphi$	number of grid points in a south-north or north-south grid column (dimensionless)
$n_\lambda = 1 + (\lambda_{max} - \lambda_{min})/\Delta_\lambda$	number of grid points in a west-east or east-west grid row (dimensionless)
n_0	index of first data value after the header information (dimensionless)
	NB: The index of the first data value in a NTv2 file is different for the parent and subgrid.
$\varphi_{min}, \lambda_{min}, \lambda_{max}, \Delta_\varphi, \Delta_\lambda$	parameters of correction grid (parent or subgrid) (Section 2.3.1 or 2.5.1)
$\lfloor ... \rfloor$	mathematical floor function
$\lceil ... \rceil$	mathematical ceiling function

2.3.3 Iterative correction

The horizontal ellipsoidal geographic pseudo Bessel coordinates of the point of interest must be corrected to real Bessel coordinates (Formula 2.3.3) using the interpolated correction grid value of the point of interest. The horizontal ellipsoidal geographic coordinates of the correction grid points are in real Bessel. Therefore, also the coordinates of the point of interest are needed in real Bessel to determine the right correction. This is easy when transforming from RD to ETRS89. However, when transforming from ETRS89 to RD, the real Bessel coordinates are needed to correct the pseudo Bessel coordinates to real Bessel coordinates. To solve this, the real Bessel coordinates are computed iteratively, until the difference between subsequent iterations becomes smaller than the precision threshold. The pseudo Bessel coordinates are used as initial approximate values for the first iteration.

Formula 2.3.3a, b, c, d, e and f. Iterative correction of pseudo RD Bessel coordinates to real RD Bessel coordinates

$$\varphi_{i=0} = \varphi^*$$
$$\lambda_{i=0} = \lambda^*$$

$$\varphi_{i+1} = \varphi^* - c_\varphi(\varphi_i, \lambda_i)$$
$$\lambda_{i+1} = \lambda^* - c_\lambda(\varphi_i, \lambda_i)$$

$$\varphi = \varphi_{i+1} \quad \text{if } |\varphi_{i+1} - \varphi_i| < \varepsilon \text{ and } |\lambda_{i+1} - \lambda_i| < \varepsilon$$
$$\lambda = \lambda_{i+1} \quad \text{if } |\varphi_{i+1} - \varphi_i| < \varepsilon \text{ and } |\lambda_{i+1} - \lambda_i| < \varepsilon$$

where:

φ^*, λ^*	latitude and longitude of point of interest in pseudo RD Bessel (°)
$\varphi_{i=0}$, $\lambda_{i=0}$	initial approximate latitude and longitude of point of interest in real RD Bessel (°)
φ_i, λ_i	improved latitude and longitude of point of interest in real RD Bessel after i iterations (°)
φ, λ	latitude and longitude of point of interest in real RD Bessel (°)
	NB: The latitude and longitude need to be computed iteratively until precision (ε) is met.
$c_\varphi(\varphi, \lambda)$, $c_\lambda(\varphi, \lambda)$	latitude and longitude correction at point of interest (°) (Formula 2.3.1)
$\varepsilon = 0.000\,000\,001°$	precision threshold for termination of iteration, corresponding to 0.0001 m
$|...|$	mathematical function for absolute value

Do not use Bessel coordinates for other purposes than RDNAPTRANS™ to avoid confusion with ETRS89 coordinates.

2.3.4 Datum transformation in the correction grid

It is possible to include the datum transformation in the correction grid. The alternative grid for this implementation variant 2 contains the latitude and longitude corrections up to 0.25 m, but also the datum difference (about 0.1 km in the central part of the Netherlands). In that way the 3D similarity transformation (Section 2.2) is not needed. With this alternative grid a bilinear interpolation of the latitude and longitude corrections (Formula 2.3.1) at the nearest grid points (Formula 2.3.2) and correction to real Bessel coordinates (Formula 2.3.3) can be applied as for a correction grid without the datum transformation, but in this case ETRS89 coordinates of the point of interest are used as input instead of the pseudo Bessel coordinates.

The correction grid including the datum transformation is supplied in three data formats, in tab-separated value ASCII text file format (.txt), in binary NTv2 file format (.gsb) and in binary GeoTIFF file format (.tif). These grids have the same properties as the correction grids that do not include the datum transformation (Section 2.3.1).

When the datum transformation is included in the correction grid, it is not possible to use a zero correction outside the bounds of the grid. A point of interest outside the grid bounds should be transformed with the 3D similarity transformation (Section 2.2) or no value should be given at all for such point. The undefined correction outside the bounds of the correction grid, to be used in the correction grid interpolation (Formula 2.3.1), is specified by the parameter listed below.

Parameter of correction grid model (variant 2)

$c_0 = $ undefined   correction value outside bounds of correction grid

A no result outside the bounds of the correction grid is best accompanied with a warning that the RD coordinates are out of bounds.

2.4   Map projection

2.4.1   Projection from ellipsoid to sphere

The corrected ellipsoidal geographic Bessel coordinates of a point of interest must be projected to obtain RD coordinates. The used RD map projection is a double projection. The first step is a Gauss conformal projection from the ellipsoid to a sphere (Formula 2.4.1).

Formula 2.4.1a, b, c, and d. Gauss conformal projection of coordinates on ellipsoid to coordinates on sphere

$$q = \ln\left(\tan\frac{\varphi + 90°}{2}\right) - \frac{e}{2}\ln\frac{1 + e\sin\varphi}{1 - e\sin\varphi}$$

$$w = n\,q + m$$

$$\Phi = \begin{cases} 2\arctan\left(\exp(w)\right) - 90° & \text{if } -90° < \varphi < 90° \\ \varphi & \text{if } \varphi = -90° \text{ or } \varphi = 90° \end{cases}$$

$$\Lambda = \Lambda_0 + n(\lambda - \lambda_0)$$

where:

φ, λ	latitude and longitude of point of interest on ellipsoid (rad or °)
q	isometric latitude of point of interest on ellipsoid (dimensionless)
w	isometric latitude of point of interest on sphere (dimensionless)
Φ, Λ	latitude and longitude of point of interest on sphere (rad or °)
	NB: The longitude on the sphere is not wrapped to −180° to 180° here.
$e = \sqrt{f(2 - f)}$	first eccentricity of Bessel ellipsoid (dimensionless)
$\exp(...)$	mathematical exponential function with mathematical constant *e* as base
λ_0	parameter of RD map projection
Λ_0, n, m	derived parameters of RD map projection
f	parameter of Bessel 1841 ellipsoid (Section 2.2.3)

NB: To prevent undefined results due to taking the tangent of a number close to 90°, a tolerance should be used to test if the latitude is 90°. Rounding the ellipsoidal coordinates in advance to 0.000 000 001° or $2 \cdot 10^{-11}$ rad suffices.

The parameters of the RD map projection (Formula 2.4.1) are listed below.

Parameters of RD map projection

$\varphi_0 = 52°9'22.178''$	latitude of central point Amersfoort on Bessel ellipsoid
$\lambda_0 = 5°23'15.500''$	longitude of central point Amersfoort on Bessel ellipsoid
$k = 0.9999079$	scale factor (dimensionless)
$x_0 = 155000$ m	false Easting
$y_0 = 463000$ m	false Northing

The derived parameters of the RD map projection (Formula 2.4.1) are listed below.

Derived parameters of RD map projection

$$n = \sqrt{1 + \frac{e^2 (\cos \varphi_0)^4}{1 - e^2}}$$

$$m = w_0 - n \, q_0$$

$$R_N = \frac{a}{\sqrt{1 - e^2 (\sin \varphi_0)^2}}$$

$$R_M = \frac{R_N (1 - e^2)}{1 - e^2 (\sin \varphi_0)^2}$$

$$R = \sqrt{R_M \, R_N}$$

$$q_0 = \ln\left(\tan\frac{\varphi_0 + 90°}{2}\right) - \frac{e}{2} \ln\frac{1 + e \sin \varphi_0}{1 - e \sin \varphi_0}$$

$$w_0 = \ln\left(\tan\frac{\Phi_0 + 90°}{2}\right)$$

$$\Phi_0 = \arctan\left(\frac{\sqrt{R_M}}{\sqrt{R_N}} \, \tan \varphi_0\right)$$

$$\Lambda_0 = \lambda_0$$

where:

n	constant of Gaussian projection (dimensionless)
m	factor of Gaussian projection (dimensionless)
R_N	second (east-west) principal radius of curvature of ellipsoid at central point (m)
R_M	first (north-south) principal radius of curvature of ellipsoid at central point (m)
R	radius of sphere (m)
q_0	isometric latitude of central point on ellipsoid (dimensionless)
w_0	isometric latitude of central point on sphere (dimensionless)
Φ_0	latitude of central point Amersfoort on sphere (rad or °)
Λ_0	longitude of central point Amersfoort on sphere (rad or °)
$e = \sqrt{f(2 - f)}$	first eccentricity of Bessel ellipsoid (dimensionless)
φ_0, λ_0	parameters of RD map projection
a, f	parameters of Bessel 1841 ellipsoid (Section 2.2.3)

Do not use the spherical coordinates for other purposes than RDNAPTRANS™ to avoid confusion with ellipsoidal coordinates.

2.4.2 Projection from sphere to plane

The second step of the RD map projection of the point of interest is an oblique stereographic conformal projection from sphere to a plane to obtain RD coordinates (Formula 2.4.2).

Formula 2.4.2a, b, c, d, e, f, g and h. Oblique stereographic projection of coordinates on sphere to coordinates on plane

$$\sin\tfrac{\psi}{2} = \sqrt{\left(\sin\frac{\Phi - \Phi_0}{2}\right)^2 + \left(\sin\frac{\Lambda - \Lambda_0}{2}\right)^2 \cos\Phi \ \cos\Phi_0}$$

$$\cos\tfrac{\psi}{2} = \sqrt{1 - \left(\sin\tfrac{\psi}{2}\right)^2}$$

$$\tan\tfrac{\psi}{2} = \frac{\sin\tfrac{\psi}{2}}{\cos\tfrac{\psi}{2}}$$

$$\sin\alpha = \frac{\sin(\Lambda - \Lambda_0)\cos\Phi}{2\sin\tfrac{\psi}{2}\cos\tfrac{\psi}{2}}$$

$$\cos\alpha = \frac{\sin\Phi - \sin\Phi_0 + 2\sin\Phi_0\left(\sin\tfrac{\psi}{2}\right)^2}{2\cos\Phi_0\sin\tfrac{\psi}{2}\cos\tfrac{\psi}{2}}$$

$$r = 2kR\tan\tfrac{\psi}{2}$$

$$x = \begin{cases} r\sin\alpha + x_0 & \text{if } (\Phi \neq \Phi_0 \quad \text{or} \quad \Lambda \neq \Lambda_0) \text{ and } (\Phi \neq -\Phi_0 \text{ or } \Lambda \neq 180° - \Lambda_0) \\ x_0 & \text{if } \Phi = \Phi_0 \quad \text{and } \Lambda = \Lambda_0 \\ \text{undefined} & \text{if } \Phi = -\Phi_0 \text{ and } \Lambda = \Lambda_0 - 180° \end{cases}$$

$$y = \begin{cases} r\cos\alpha + y_0 & \text{if } (\Phi \neq \Phi_0 \quad \text{or} \quad \Lambda \neq \Lambda_0) \text{ and } (\Phi \neq -\Phi_0 \text{ or } \Lambda \neq 180° - \Lambda_0) \\ y_0 & \text{if } \Phi = \Phi_0 \quad \text{and } \Lambda = \Lambda_0 \\ \text{undefined} & \text{if } \Phi = -\Phi_0 \text{ and } \Lambda = \Lambda_0 - 180° \end{cases}$$

where:

Φ, Λ	latitude and longitude of point of interest on sphere (rad or °)
ψ	distance angle from central point to point of interest on sphere (rad or °)
r	distance from central point to point of interest in projection plane (m)
α	azimuth from central point to point of interest (m)
x, y	projected RD coordinates of point of interest (m)
k, x_0, y_0	parameters of RD map projection (Section 2.4.1)
Φ_0, Λ_0, R	derived parameters of RD map projection (Section 2.4.1)

NB: To prevent arithmetic overflow due to division by a number close to zero, a tolerance should be used to test if the point of interest is close to the central point itself or the point opposite on the sphere. Rounding the spherical coordinates in advance to 0.000 000 001° or 2 · 10⁻¹¹ rad suffices.

2.5   Height transformation

2.5.1   Bilinear quasi-geoid grid interpolation

The ellipsoidal height is not used with RD coordinates as it is purely geometrical and has no physical meaning. The height transformation from ellipsoidal ETRS89 height of a point of interest to NAP height is based on the quasi-geoid model NLGEO2018. The quasi-geoid height at the point of interest is obtained by bilinear interpolation of a regular grid of quasi-geoid height values (Formula 2.5.1).

Formula 2.5.1a, b, c and d. Bilinear quasi-geoid grid interpolation and test for points out of bounds

$$b = \begin{cases} 1 & \text{if } \varphi_{min} \leq \varphi \leq \varphi_{max} \text{ and } \lambda_{min} \leq \lambda \leq \lambda_{max} \\ 0 & \text{if } \varphi < \varphi_{min} \text{ or } \varphi > \varphi_{max} \text{ or } \lambda < \lambda_{min} \text{ or } \lambda > \lambda_{max} \end{cases}$$

$$\bar{\varphi} = \frac{\varphi - \varphi_{min}}{\Delta_\varphi}$$

$$\bar{\lambda} = \frac{\lambda - \lambda_{min}}{\Delta_\lambda}$$

$$N(\varphi, \lambda) = \begin{cases} \begin{bmatrix} \bar{\varphi} - \lfloor\bar{\varphi}\rfloor & \lfloor\bar{\varphi}\rfloor + 1 - \bar{\varphi} \end{bmatrix} \begin{bmatrix} g_N(i_{NW}) & g_N(i_{NE}) \\ g_N(i_{SW}) & g_N(i_{SE}) \end{bmatrix} \begin{bmatrix} \lfloor\bar{\lambda}\rfloor + 1 - \bar{\lambda} \\ \bar{\lambda} - \lfloor\bar{\lambda}\rfloor \end{bmatrix} & \text{if } b = 1 \\ c_0 & \text{if } b = 0 \end{cases}$$

Formula 2.5.1e. Equivalent without matrix notation

$$N(\varphi, \lambda) = \begin{cases} (\bar{\varphi} - \lfloor\bar{\varphi}\rfloor)\left(g_N(i_{NW})(\lfloor\bar{\lambda}\rfloor + 1 - \bar{\lambda}) + g_N(i_{NE})(\bar{\lambda} - \lfloor\bar{\lambda}\rfloor)\right) + (\lfloor\bar{\varphi}\rfloor + 1 - \bar{\varphi})\left(g_N(i_{SW})(\lfloor\bar{\lambda}\rfloor + 1 - \bar{\lambda}) + g_N(i_{SE})(\bar{\lambda} - \lfloor\bar{\lambda}\rfloor)\right) & \text{if } b = 1 \\ c_0 & \text{if } b = 0 \end{cases}$$

where:

φ, λ	latitude and longitude of point of interest in horizontal datum of grid file (°)
$\bar{\varphi}, \bar{\lambda}$	normalised latitude and longitude of point of interest (dimensionless)
$g_N(i_{NW}), g_N(i_{NE}), g_N(i_{SW}), g_N(i_{SE})$	quasi-geoid height at nearest NW, NE, SW and SE grid points (°) (Formula 2.3.2)
$N(\varphi, \lambda)$	ETRS89 quasi-geoid height at point of interest in horizontal datum of grid file (°)
$\varphi_{min}, \varphi_{max}, \lambda_{min}, \lambda_{max}, \Delta_\varphi, \Delta_\lambda$	parameters of quasi-geoid grid
c_0	additional parameter of quasi-geoid grid model
b	logical value of test if point of interest is inside bounds of quasi-geoid grid (Boolean)
$\lfloor \dots \rfloor$	mathematical floor function

NB: The test if a point is inside or outside the bounds of the quasi-geoid grid is performed in real RD Bessel or ETRS89 coordinates, depending on the horizontal datum of the grid, but the quasi-geoid height is always in ETRS89.

To transform the point of interest, the nearest NW, NE, SW and SE grid values are required. Grid values can be read one by one from the binary grid file by direct access or the entire grid of the binary or ASCII text file can be assigned to an array variable first. In both cases, the indices of the required grid values need to be determined. To use the same formulas for computing the indices for an array variable as for the original grid file, the array variable should have the same dimensions and order as the grid file. Thus, using a 1D array for a binary file and a 2D array with one row for each grid point for an ASCII text file. The indices can be computed from the coordinates of the point of interest (Formula 2.3.2).

The quasi-geoid grid for bilinear interpolation (Formula 2.5.1) is supplied in two horizontal datums. The horizontal coordinates of the grid points for which the quasi-geoid height is given are in ETRS89 (variant 1) or in Bessel (variant 2), but the quasi-geoid height is relative to the ETRS89 ellipsoid in both cases. These grids are also supplied in three data formats, in tab-separated value ASCII text file format (.txt), in binary VDatum file format (.gtx) and in binary GeoTIFF file format (.tif). The supplied quasi-geoid grid files all have the same bounds and spacing.

Implementation variant 1 uses the ETRS89 grid for transformation in both transformation directions, for ETRS89 to RD and NAP as well as RD and NAP to ETRS89. Using a different grid for the transformation back is not recommended, as it can result in too large differences after repeatedly transforming back and forth.

Some software expect the quasi-geoid grid to give ETRS89 quasi-geoid height in a Bessel grid. For these applications implementation variant 2 with the Bessel grid is used in both transformation directions, for ETRS89 to RD and NAP as well as RD and NAP to ETRS89.

The tab-separated value ASCII text file format has a header line and uses one line per grid point. The columns are latitude of grid point (°), longitude of grid point (°) and ETRS89 quasi-geoid height at grid point (m). The order in which correction values are listed is from southwest to northeast and starts in west-east direction. The properties of quasi-geoid grid files are specified by the parameters listed below.

The binary VDatum file format is supported by open source software for geo-information. The quasi-geoid heights are in metres, spacing and coordinates of grid bounds are given in decimal degrees with conventional sign, thus east of the Greenwich meridian is positive. The order in which the correction values are listed in the VDatum grid file, is from southwest to northeast and starts in west-east direction. A unit conversion (Formula 2.1) is only needed when working with radians. The VDatum file format contains a header but no subgrids. Note that the VDatum file is in big-endian representation. A more detailed file format description is provided with the list of the associated files of RDNAPTRANS™2018 (Appendix 2).

The binary GeoTIFF file format is supported by many software packages for geo-information. The usage for Geodetic TIFF Grids for datum transformation with a correction grid as specified by PROJ is less common. The GeoTIFF files use lossless data compression. This reduces the file size, but this makes reading from the binary file more complicated. When the units are converted (Formula 2.1) directly when reading the values from file, the interpolation (Formula 2.3.1) does not need to be adapted. Additional file format information is provided with the list of the associated files of RDNAPTRANS™2018 (Appendix 2).

RDNAPTRANS™2008 and previous versions of RDNAPTRANS™ used a quasi-geoid grid in a different binary file format. The VDatum file for a precise approximation of RDNAPTRANS™2008 used different bounds and spacing.

Parameters of tab-separated value ASCII text (.txt) quasi-geoid grid files and of binary VDatum (.gtx) and GeoTIFF (.tif) quasi-geoid grid files

$\varphi_{min} = 50°$	latitude of southern bound of correction grid
$\varphi_{max} = 56°$	latitude of northern bound of correction grid
$\lambda_{min} = 2°$	longitude of western bound of correction grid
$\lambda_{max} = 8°$	longitude of eastern bound of correction grid
$\Delta_{\varphi} = 0.0125° = 45''$	latitude spacing of correction grid, corresponding to about 1.4 km
$\Delta_{\lambda} = 0.02° = 72''$	longitude spacing of correction grid, corresponding to about 1.4 km

NB: Since GeoTIFF files are often used for raster images and the grid values are valid for the pixel centre in Geodetic TIFF Grids, the bounds of the GeoTIFF may appear sightly larger (half the spacing on all sides of the grid).

The undefined quasi-geoid height outside the bounds of the grid to be used in the quasi-geoid grid interpolation (Formula 2.5.1), is specified by the parameter listed below.

Parameter of quasi-geoid grid model

$c_0 = \text{undefined}$ quasi-geoid height outside bounds of quasi-geoid grid

A no result outside the bounds of the quasi-geoid grid model is best accompanied with a warning that the NAP coordinates are out of bounds.

2.5.2   Transformation to NAP

The ellipsoidal ETRS89 height of the point of interest must be transformed to NAP height (Formula 2.5.2) using the interpolated quasi-geoid height of the point of interest.

Formula 2.5.2. Transformation from ellipsoidal ETRS89 height to NAP height using quasi-geoid model

$$H = h - N(\varphi, \lambda)$$

where:
H NAP height of point of interest (m)
h ellipsoidal ETRS89 height of point of interest (m)
$N(\varphi, \lambda)$ ETRS89 quasi-geoid height at point of interest in horizontal datum of grid file (m) (Formula 2.5.1)

3 Transformation from RD and NAP to ETRS89

3.1 Inverse map projection

3.1.1 Projection from plane to sphere

RD coordinates of a point of interest must be converted to Bessel coordinates before the other steps of the transformation can be performed. The RD map projection is a double projection. The first step of the inverse map projection is an inverse oblique stereographic conformal projection from the RD projection plane to a sphere (Formula 3.1.1).

Formula 3.1.1a, b, c, d, e, f, g, h, i. Inverse oblique stereographic projection of coordinates on plane to coordinates on sphere

$$r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$$

$$\sin \alpha = \frac{x - x_0}{r}$$

$$\cos \alpha = \frac{y - y_0}{r}$$

$$\psi = 2 \arctan \frac{r}{2kR}$$

$$\bar{X} = \begin{cases} \cos \Phi_0 \cos \psi - \cos \alpha \sin \Phi_0 \sin \psi & \text{if } x \neq x_0 \text{ or } y \neq y_0 \\ \cos \Phi_0 & \text{if } x = x_0 \text{ and } y = y_0 \end{cases}$$

$$\bar{Y} = \begin{cases} \sin \alpha \sin \psi & \text{if } x \neq x_0 \text{ or } y \neq y_0 \\ 0 & \text{if } x = x_0 \text{ and } y = y_0 \end{cases}$$

$$\bar{Z} = \begin{cases} \cos \alpha \cos \Phi_0 \sin \psi + \sin \Phi_0 \cos \psi & \text{if } x \neq x_0 \text{ or } y \neq y_0 \\ \sin \Phi_0 & \text{if } x = x_0 \text{ and } y = y_0 \end{cases}$$

$$\Phi = \arcsin(\bar{Z})$$

$$\Lambda = \begin{cases} \Lambda_0 + \arctan(\bar{Y}/\bar{X}) & \text{if } \bar{X} > 0 \\ \Lambda_0 + \arctan(\bar{Y}/\bar{X}) + 180° & \text{if } \bar{X} < 0 \text{ and } x \geq x_0 \\ \Lambda_0 + \arctan(\bar{Y}/\bar{X}) - 180° & \text{if } \bar{X} < 0 \text{ and } x < x_0 \\ \Lambda_0 + 90° & \text{if } \bar{X} = 0 \text{ and } x > x_0 \\ \Lambda_0 - 90° & \text{if } \bar{X} = 0 \text{ and } x < x_0 \\ \Lambda_0 & \text{if } \bar{X} = 0 \text{ and } x = x_0 \end{cases}$$

where:

x, y	projected RD coordinates of point of interest (m)
α	azimuth from central point to point of interest (rad or °)
r	distance from central point to point of interest in projection plane (m)
ψ	distance angle from central point to point of interest on sphere (rad or °)
$\bar{X}, \bar{Y}, \bar{Z}$	normalised geocentric Cartesian coordinates (with prime meridian through the central point) of point of interest on sphere (dimensionless)
Φ, Λ	latitude and longitude of point of interest on sphere (rad or °)
	NB: The longitude on the sphere is not wrapped to −180° to 180° here.
k, x_0, y_0	parameters of RD map projection (Section 2.4.1)
Φ_0, Λ_0, R	derived parameters of RD map projection (Section 2.4.1)

21

3.1.2 Projection from sphere to ellipsoid

The second step of the inverse RD map projection is an inverse Gauss conformal projection from the sphere to the Bessel ellipsoid to obtain Bessel coordinates of the point of interest (Formula 3.1.2).

Formula 3.1.2a, b, c, d, e, f, g. Inverse Gauss conformal projection of coordinates on sphere to coordinates on ellipsoid

$$w = \ln\left(\tan\frac{\Phi + 90°}{2}\right)$$

$$q = \frac{w - m}{n}$$

$$\varphi_{i=0} = 0$$

$$\varphi_{i+1} = \begin{cases} 2\arctan\left(\exp\left(q + \frac{e}{2}\ln\frac{1 + e\sin\varphi_i}{1 - e\sin\varphi_i}\right)\right) - 90° & \text{if } -90° < \Phi < 90° \\ \Phi & \text{if } \Phi = -90° \text{ or } \Phi = 90° \end{cases}$$

$$\varphi = \varphi_{i+1} \quad \text{if } |\varphi_{i+1} - \varphi_i| < \varepsilon$$

$$\lambda_n = \frac{\Lambda - \Lambda_0}{n} + \lambda_0$$

$$\lambda = \lambda_n + 360°\left\lfloor\frac{180° - \lambda_n}{360°}\right\rfloor$$

where:

Φ, Λ	latitude and longitude of point of interest on sphere (rad or °)
w	isometric latitude of point of interest on sphere (dimensionless)
q	isometric latitude of point of interest on ellipsoid (dimensionless)
$\varphi_{i=0}$	initial approximate latitude of point of interest on ellipsoid (rad or °)
φ_i	improved latitude of point of interest on ellipsoid after i iterations (rad or °)
λ_n	longitude of point of interest on ellipsoid, not wrapped to −180° to 180° (rad or °)
φ, λ	latitude and longitude of point of interest on ellipsoid (rad or °)
	NB: The latitude needs to be computed iteratively until precision (ε) is met.
$\varepsilon = 0.000\,000\,001° = 2 \cdot 10^{-11}$ rad	precision threshold for termination of iteration, corresponding to 0.0001 m
$e = \sqrt{f(2 - f)}$	first eccentricity of Bessel ellipsoid (dimensionless)
$\exp(\dots)$	mathematical natural exponential function with mathematical constant e as base
$\lfloor\dots\rfloor$	mathematical floor function
λ_0	parameter of RD map projection (Section 2.4.1)
Λ_0, n, m	derived parameters of RD map projection (Section 2.4.1)

Do not use spherical or ellipsoidal Bessel coordinates for other purposes than RDNAPTRANS™ to avoid confusion with ETRS89 coordinates.

3.2 RD correction

3.2.1 Direct correction

The ellipsoidal geographic coordinates of a point of interest obtained by the inverse map projection, are real Bessel coordinates. Due to the error propagation of measurement noise of the original (1888–1928) measurements of RD, the real Bessel coordinates must be corrected up to 0.25 m to obtain pseudo Bessel coordinates. For implementation variant 2, the datum transformation is included in the correction grid (Section 3.2.2).

The corrections are obtained from a regular grid of values for latitude correction and a regular grid of values for longitude correction, using bilinear interpolation (Formula 2.3.1). A unit conversion (Formula 2.1) might be needed, as the correction values are given in degrees or arcseconds, depending on the data format of the correction grid. The correction grid for bilinear interpolation is supplied in three data formats, in tab-separated value ASCII text file format (.txt), in binary NTv2 file format (.gsb) and in binary GeoTIFF file format (.tif) with different properties (Section 2.3.1).

To transform the point of interest, the nearest NW, NE, SW and SE grid values are required. Grid values can be read one by one from the binary grid file by direct access or the entire grid of the binary or ASCII text file can be assigned to an array variable first. In both cases, the indices of the required grid values need to be determined. To use the same formulas for computing the indices for an array variable as for the original grid file, the array variable should have the same dimensions and order as the grid file. Thus, using a 1D array for a binary file and a 2D array with one row for each grid point for an ASCII text file. The indices can be computed from the coordinates of the point of interest (Formula 2.3.2).

The horizontal ellipsoidal geographic real Bessel coordinates of the point of interest must be corrected to pseudo Bessel coordinates (Formula 3.2.1) using the interpolated correction grid value of the point of interest. No iteration is needed for the transformation from RD to ETRS89 coordinates as the grid is given in real Bessel coordinates.

Formula 3.2.1a, b. Correction of real RD Bessel coordinates to pseudo RD Bessel coordinates

$$\varphi^* = \varphi + c_\varphi(\varphi, \lambda)$$
$$\lambda^* = \lambda + c_\lambda(\varphi, \lambda)$$

where:

$\varphi,\ \lambda$	latitude and longitude of point of interest in real RD Bessel (°)
$\varphi^*,\ \lambda^*$	latitude and longitude of point of interest in pseudo RD Bessel (°)
$c_\varphi(\varphi, \lambda), c_\lambda(\varphi, \lambda)$	latitude and longitude correction at point of interest in real RD Bessel (°) (Formula 2.3.1)

Do not use Bessel coordinates for other purposes than RDNAPTRANS™ to avoid confusion with ETRS89 coordinates.

3.2.2 Datum transformation in the correction grid

It is possible to include the datum transformation in the correction grid. The alternative grid for this implementation variant 2 contains the latitude and longitude corrections up to 0.25 m, but also the datum difference (about 0.1 km in the central part of the Netherlands). In that way the 3D similarity transformation (Section 3.3) is not needed. With this alternative grid a bilinear interpolation of the latitude and longitude corrections (Formula 2.3.1) at the nearest grid points (Formula 2.3.2) and correction of real Bessel coordinates (Formula 3.2.1) can be applied as for a correction grid without the datum transformation, but in this case the output are ETRS89 coordinates of the point of interest instead of pseudo Bessel coordinates.

The correction grid including the datum transformation is supplied in two data formats, in tab-separated value ASCII text file format (.txt), in binary NTv2 file format (.gsb) and in binary GeoTIFF file format (.tif). These grids have the same properties as the correction grids that do not include the datum transformation (Section 2.3.1).

When the datum transformation is included in the correction grid, it is not possible to use a zero correction outside the bounds of the grid. A point of interest outside the grid bounds should be transformed with the 3D similarity transformation (Section 3.3) or no value should be given at all for such point. The undefined correction outside the bounds of the correction grid, to be used in the correction grid interpolation (Formula 2.3.1), is specified by the parameter listed below.

Parameter of correction grid model (variant 2)

$c_0 = \text{undefined}$ correction value outside bounds of correction grid

A no result outside the bounds of the correction grid is best accompanied with a warning that the RD coordinates are out of bounds.

3.3   Datum transformation

The corrected ellipsoidal geographic Bessel coordinates of a point of interest must be transformed to ellipsoidal geographic ETRS89 coordinates. This is only needed for implementation variant 1, for variant 2 the datum transformation (Section 3.3) is included in the correction grid (Section 3.2). First, the ellipsoidal geographic Bessel coordinates of a point of interest must be converted to geocentric Cartesian Bessel coordinates (Formula 2.2.1) to be able to apply a 3D similarity transformation. Note that the parameters of the Bessel ellipsoid (Section 2.2.3) are needed for the conversion to geocentric Cartesian coordinates for the transformation from RD to ETRS89.

Unlike conventional use in transformations, including RDNAPTRANS™2008 and earlier versions of RDNAPTRANS™, a fixed ellipsoidal height is used instead of the actual height of the point of interest. The fixed ellipsoidal height used in the conversion to geocentric Cartesian Bessel (Formula 2.2.1) is listed below.

Points with the same latitude and longitude in RD that differ in height get exactly the same horizontal ETRS89 coordinates. This enables 2D transformation between RD and ETRS89 and straightforward implementation in software like GIS packages. However, it introduces small differences between back and forth transformation. These differences are below 0.0010 m up to 500 km outside the bounds of the RDNAPTRANS™2018 grids.

Parameter for fixed ellipsoidal RD Bessel height

$h_0 = 0$ ellipsoidal RD Bessel height approximately corresponding to 0 m in NAP (m)

The 3D similarity transformation must be applied to the geocentric Cartesian Bessel coordinates of the point of interest to obtain geocentric Cartesian ETRS89 coordinates. The datum transformation uses the same formulas for the 3D similarity transformation from RD to ETRS89 as from ETRS89 to RD (Formula 2.2.2). So the order is first rotate and then translate, instead of inversing the order to translating and rotating. As a result, the inverse transformation parameters do not only have opposite sign, but the parameters also have slightly different numbers (except for the scale difference parameter). Sometimes the differences are only apparent in the last digit. The 7 parameters for the transformation from RD to ETRS89 are listed below. The parameters could be computed from the inverse parameters too (Section 2.2.2). Often, only the parameters of one transformation direction are set in software.

Parameters of 3D similarity transformation from RD to ETRS89

$t_X = +565.7381 \text{ m}$ translation in direction of X axis
$t_Y = +50.4018 \text{ m}$ translation in direction of Y axis
$t_Z = +465.2904 \text{ m}$ translation in direction of Z axis

$$\alpha = +1.91514 \cdot 10^{-6} \text{ rad} = +0.395026'' \quad \text{rotation angle around X axis}$$
$$\beta = -1.60363 \cdot 10^{-6} \text{ rad} = -0.330772'' \quad \text{rotation angle around Y axis}$$
$$\gamma = +9.09546 \cdot 10^{-6} \text{ rad} = +1.876074'' \quad \text{rotation angle around Z axis}$$
$$\delta = +4.07244 \cdot 10^{-6} \qquad\qquad\qquad\ \text{scale difference (dimensionless)}$$

NB: The three rotation parameters (α, β, γ) are for use with the formulas above according to the "coordinate frame" convention, that assume the positive rotation direction to be anticlockwise. The signs of the three rotation parameters must be inverted for the alternative "position vector" convention.

Do not use these geocentric Cartesian ETRS89 coordinates for other purposes than RDNAPTRANS™, as the use of a fixed height might be not appropriate and the actual height of the point of interest should be used. Instead, first convert the coordinates of the point of interest to ellipsoidal ETRS89 geographic coordinates and perform the height transformation (Section 3.5), then convert to geocentric Cartesian ETRS89 coordinates with the actual height of the point of interest to obtain geocentric Cartesian ETRS89 coordinates.

After the 3D similarity transformation, the geocentric Cartesian ETRS89 coordinates of the point of interest must be converted back to ellipsoidal geographic ETRS89 coordinates (Formula 2.2.3). The latitude is computed iteratively. The parameters of the GRS80 ellipsoid (Section 2.2.1) are needed for the conversion to ellipsoidal geographic coordinates for the transformation from RD to ETRS89.

3.4   Notation in degrees, minutes and seconds

The ETRS89 coordinates of a point of interest obtained after transformation are in radians or decimal degrees, depending on the type of goniometry functions used. These coordinates must be converted to decimal degrees and optionally to degrees, minutes and seconds (Formula 3.4). Traditionally, the latitude and longitude of a point of interest are given as integer degrees, integer minutes and decimal seconds with a hemisphere code, but other notation formats and variations on these are used too (Section 2.1).

Formula 3.4a. Unit conversion from radians to decimal degrees

$$d = r\ 180°/\pi$$

Formula 3.4b, c, d, e, f, and g. Conversion of notation from decimal degrees to degrees, minutes and seconds

$$D = \frac{\lfloor 3600 \cdot 10^5\ |d| + 0.5 \rfloor}{3600 \cdot 10^5}$$

$$D_{int} = \lfloor D \rfloor$$

$$M = 60(D - D_{int})$$

$$M_{int} = \lfloor M \rfloor$$

$$S = 60(M - M_{int})$$

$$C = \begin{cases} \text{"N"} & \text{if } \varphi \geq 0 \\ \text{"S"} & \text{if } \varphi < 0 \\ \text{"E"} & \text{if } \lambda \geq 0 \\ \text{"W"} & \text{if } \lambda < 0 \end{cases}$$

where:
r \qquad\qquad\qquad\qquad\qquad\qquad signed decimal degrees latitude or longitude (rad)

d	signed decimal degrees latitude or longitude (°)
D	unsigned decimal degrees part of latitude or longitude (°)
D_{int}	unsigned integer degrees part of latitude or longitude (°)
M	unsigned decimal minutes part of latitude or longitude (′)
M_{int}	unsigned integer minutes part of latitude or longitude (′)
S	unsigned decimal seconds part of latitude or longitude (″)
C	hemisphere code (character)
$\pi = 3.14159265358$	mathematical constant (dimensionless)
$\lvert \ldots \rvert$	mathematical function for absolute value
$\lfloor \ldots \rfloor$	mathematical floor function

NB: Rounding to 5 digits in decimal seconds is performed in formula b. instead of formula f., to prevent $0°0'59.999995''$ to be rounded to $0°0'60.00000''$ instead of $0°1'0.00000''$.

3.5   Height transformation

The physical NAP height of a point of interest can be transformed to the purely geometrical ellipsoidal ETRS89 height. The height transformation from NAP to ETRS89 is based on the quasi-geoid model NLGEO2018. The quasi-geoid height at the point of interest is obtained by bilinear interpolation of a regular grid of quasi-geoid height values (Formula 2.5.1). A unit conversion (Formula 2.1) is only needed when working with radians. The undefined quasi-geoid height outside the bounds of the grid to be used in the quasi-geoid grid interpolation (Formula 2.5.1), is specified by the parameter listed below.

Parameter of quasi-geoid grid model

$c_0 = \text{undefined}$ quasi-geoid height outside bounds of quasi-geoid grid

A no result outside the bounds of the quasi-geoid grid model is best accompanied with a warning that the NAP coordinates are out of bounds.

To transform the point of interest, the nearest NW, NE, SW and SE grid values are required. Grid values can be read one by one from the binary grid file by direct access or the entire grid of the binary or ASCII text file can be assigned to an array variable first. In both cases, the indices of the required grid values need to be determined. To use the same formulas for computing the indices for an array variable as for the original grid file, the array variable should have the same dimensions and order as the grid file. Thus, using a 1D array for a binary file and a 2D array with one row for each grid point for an ASCII text file. The indices can be computed from the coordinates of the point of interest (Formula 2.3.2).

The grid for the bilinear interpolation is supplied in two horizontal datums. The horizontal coordinates of the grid points for which the quasi-geoid height is given are in ETRS89 (variant 1) or in Bessel (variant 2), but the quasi-geoid height is relative to the ETRS89 ellipsoid in both cases. These grids are also supplied in three data formats, in tab-separated value ASCII text file format (.txt), in binary VDatum file format (.gtx) and in binary GeoTIFF file format (.tif) with different properties (Section 2.5.1). The six supplied quasi-geoid grid files all have the same bounds and spacing (Section 2.5.1).

Implementation variant 1 uses the ETRS89 grid for transformation in both transformation directions, for RD and NAP to ETRS89 as well as ETRS89 to RD and NAP. Using a different grid for the transformation back is not recommended, as it can result in too large differences after repeatedly transforming back and forth.

Some software expect the quasi-geoid grid to give ETRS89 quasi-geoid height in a Bessel grid. For these applications implementation variant 2 with the Bessel grid is used in both transformation directions, for RD and NAP to ETRS89 as well as ETRS89 to RD and NAP.

The NAP height of the point of interest must be transformed to ellipsoidal ETRS89 height (Formula 3.5) using the interpolated quasi-geoid height of the point of interest.

Formula 3.5. Transformation from NAP height to ellipsoidal ETRS89 height using quasi-geoid model

$$h = H + N(\varphi, \lambda)$$

where:
h	ellipsoidal ETRS89 height of point of interest (m)
H	NAP height of point of interest (m)
$N(\varphi, \lambda)$	ETRS89 quasi-geoid height at point of interest in horizontal datum of grid file (m) (Formula 2.5.1)

Appendix 1: Detailed diagrams

For an overview of the differences between RDNAPTRANS™ versions and implementation variants, detailed diagrams of the transformation procedures are given for RDNAPTRANS™2018 variant 1 (Figure A1.1), RDNAPTRANS™2018 variant 2 and RDNAPTRANS™2008 NTv2 approximation (Figure A1.2), RDNAPTRANS™2008 and earlier versions (Figure A1.3) and RDNAPTRANS™2018 hybrid variant (Figure A1.4). The differences in the input parameters and grid files between versions with the same transformation procedure are listed below. Transformation outside a grid is discouraged, but allowed for grids marked with an asterisk (*).

Differences in input parameters and grid files between RDNAPTRANS™ versions and variants

Version, variant	7 parameters	RD correction grid	Geoid grid	Height offset
RDNAPTRANS™2000	2000	RD correction 2000	De Min	*none*
RDNAPTRANS™2004	2004	RD correction 2000	NLGEO2004	*none*
RDNAPTRANS™2008	2008	RD correction 2000*	NLGEO2004	0.0088 m
RDNAPTRANS™2008 NTv2 approximation	*none*	rdtrans2008	naptrans2008	*none*
RDNAPTRANS™2018 variant 1	2018	rdcorr2018*	NLGEO2018	*none*
RDNAPTRANS™2018 variant 2	*none*	rdtrans2018	naptrans2018	*none*

NB: The first version was originally called RDNAPTRANS™ without the year 2000. The 7 parameters of this version predate the year 2000 and had been published before. The RDNAPTRANS™2008 NTv2 approximation was published in 2014.

Before RDNAPTRANS™2018 another data format was used for the grid files covering a smaller area excluding a large part of the Dutch EEZ of the North Sea (Figure 1.1.1).
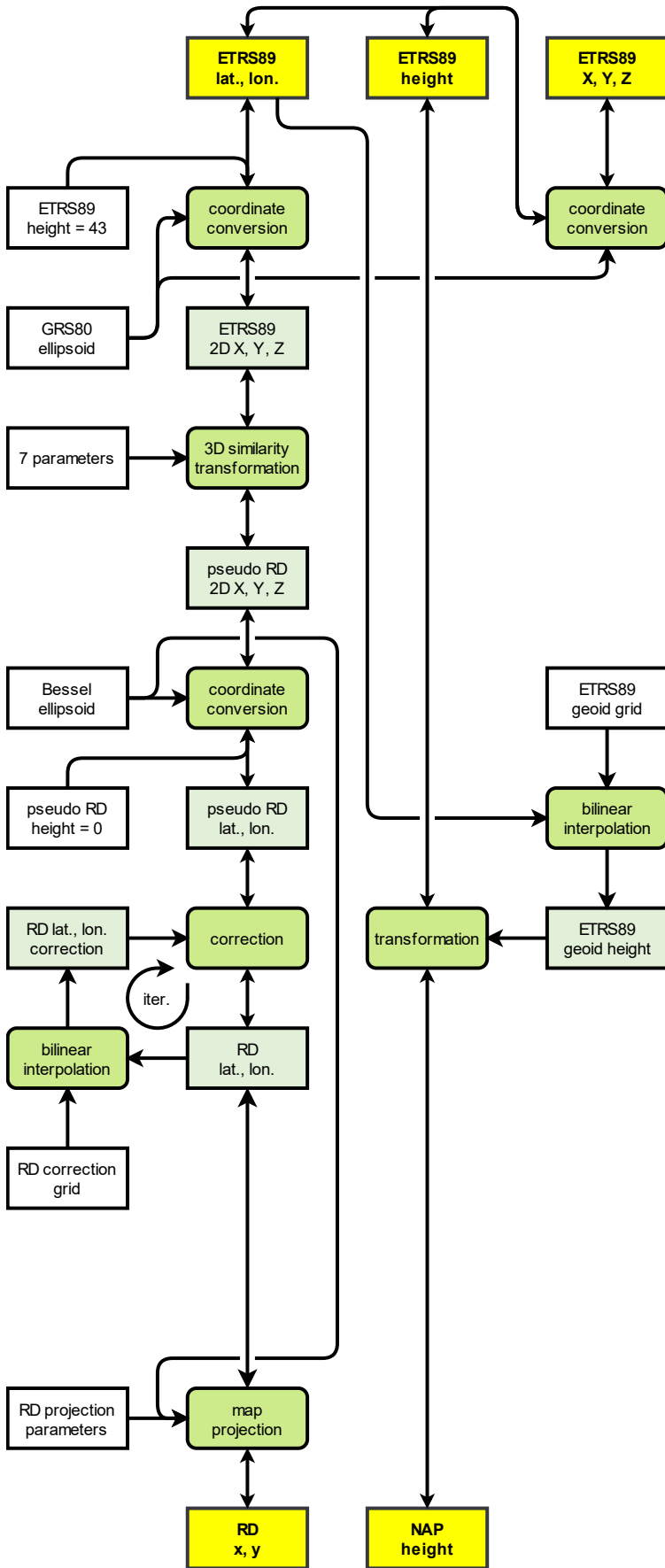
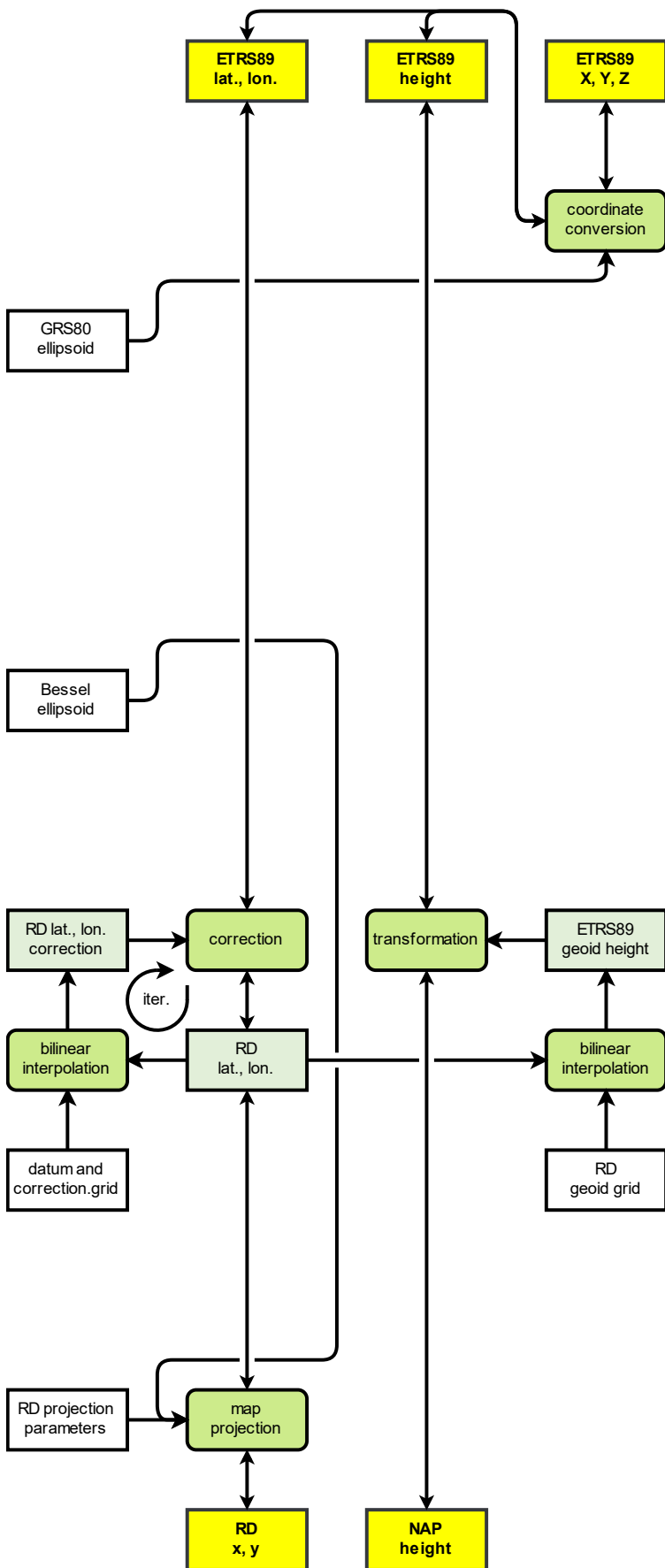Figure A1.1 Transformation procedure of RDNAPTRANS™2018 variant 1 (recommended).

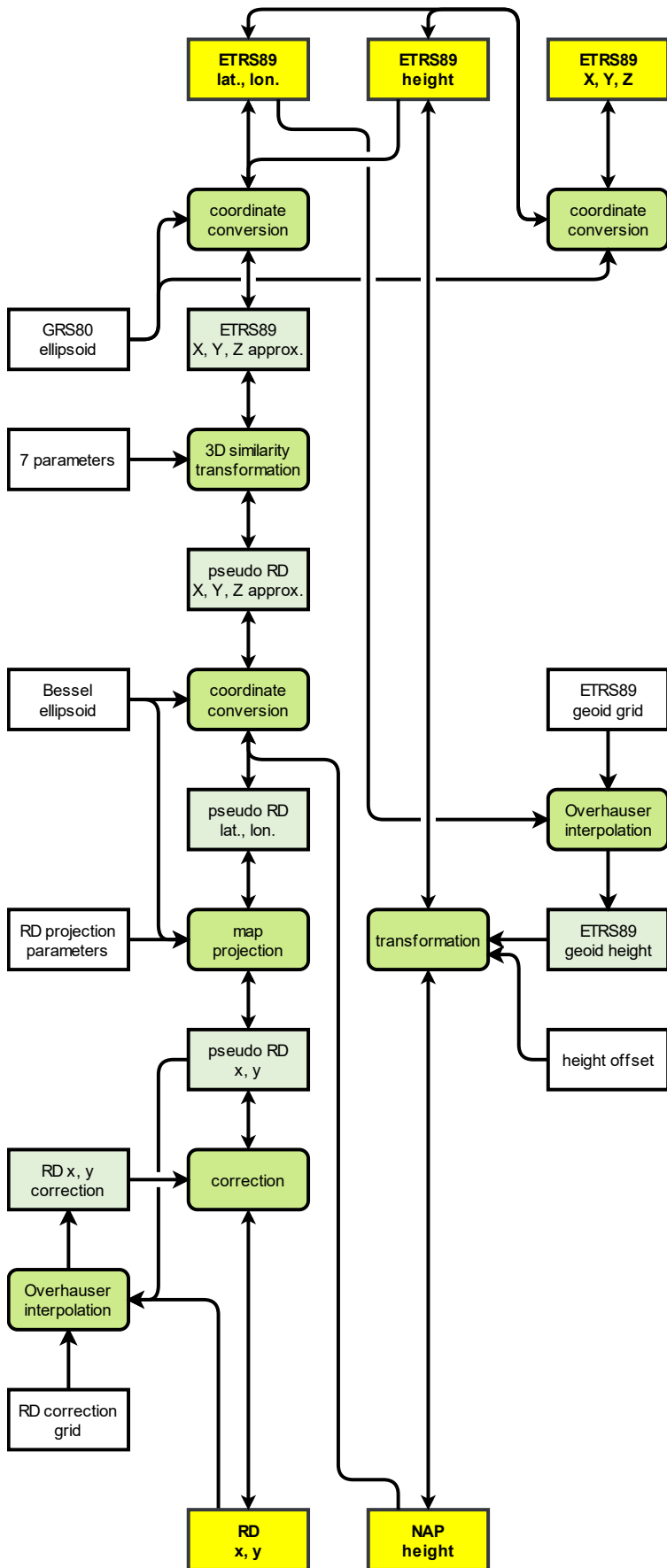Figure A1.2. Transformation procedure of RDNAPTRANS™2018 variant 2 and RDNAPTRANS™2008 NTv2 approximation.

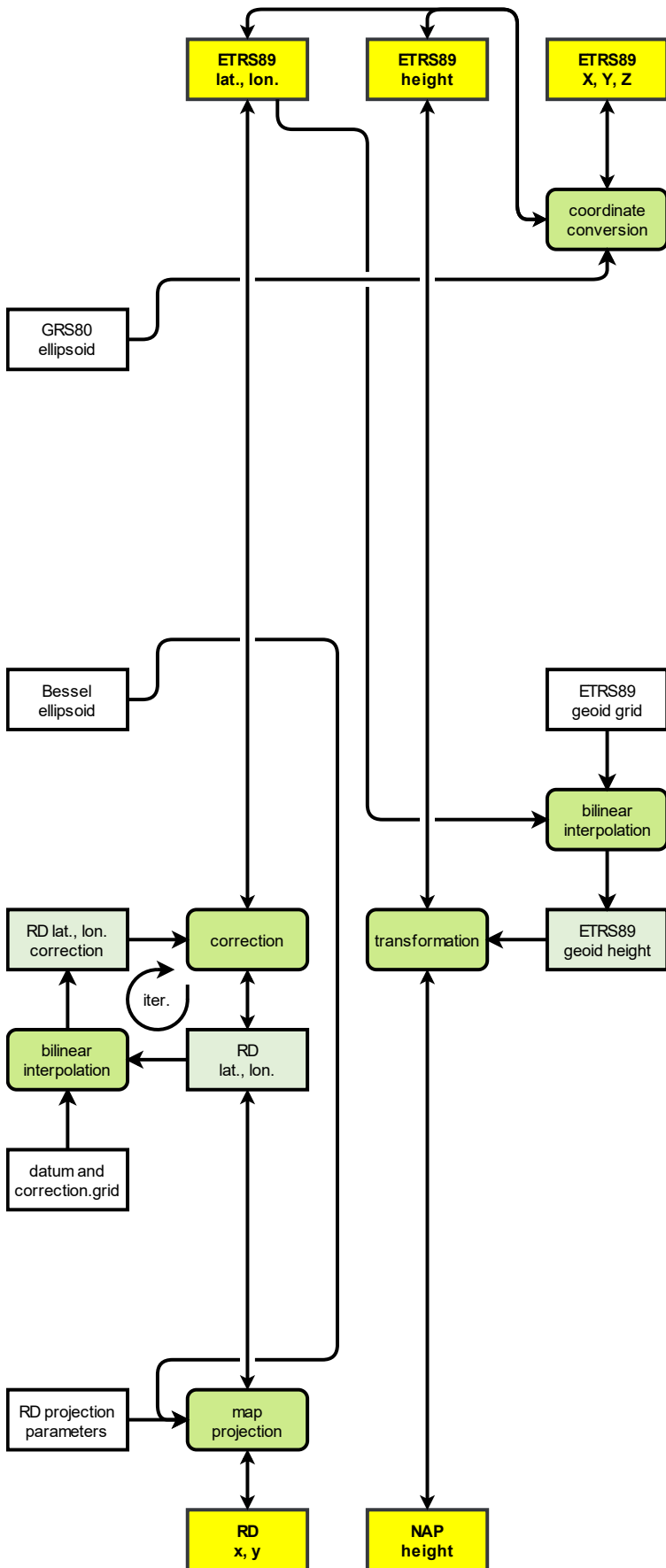Figure A1.3. Transformation procedure of RDNAPTRANS™2008 (official variant) and earlier versions of RDNAPTRANS™.

Figure A1.4. Transformation procedure of RDNAPTRANS™2018 hybrid variant.

Appendix 2: Files in RDNAPTRANS™2018 download

This document and the necessary files for the implementation of RDNAPTRANS™2018 can be downloaded from: https://www.nsgi.nl/rdnaptrans/ (in Dutch).

The download rdnaptrans2018.zip includes files for correction grid and quasi-geoid grid, for two implementation variants and three data formats. These files by NSGI are licensed under a Creative Commons Attribution 4.0 International License. However, changing the grid values is strongly discouraged and would make it impossible to obtain permission to use the trademark RDNAPTRANS (Appendix 4).

Files in download for implementation variant 1 (CC BY 4.0)

File name	Format	Grid	Values
rdcorr2018.txt	ASCII	RD Bessel	RD Bessel corrections
rdcorr2018.gsb	NTv2	RD Bessel	RD Bessel corrections
rdcorr2018.tif	GeoTIFF	RD Bessel	RD Bessel corrections
nlgeo2018.txt	ASCII	ETRS89	ETRS89 quasi-geoid height
nlgeo2018.gtx	VDatum	ETRS89	ETRS89 quasi-geoid height
nlgeo2018.tif	GeoTIFF	ETRS89	ETRS89 quasi-geoid height

Files in download for implementation variant 2 (CC BY 4.0)

File name	Format	Grid	Values
rdtrans2018.txt	ASCII	RD Bessel	RD Bessel corrections including datum transformation
rdtrans2018.gsb	NTv2	RD Bessel	RD Bessel corrections including datum transformation
rdtrans2018.tif	GeoTIFF	RD Bessel	RD Bessel corrections including datum transformation
naptrans2018.txt	ASCII	RD Bessel	ETRS89 quasi-geoid height
naptrans2018.gtx	VDatum	RD Bessel	ETRS89 quasi-geoid height
naptrans2018.txt	GeoTIFF	RD Bessel	ETRS89 quasi-geoid height

A2.1   NTv2 file format
The binary NTv2 file format (.gsb) uses 8-bit ascii characters (char), 32-bit unsigned integer numbers (uint32), 64-bit floating-point numbers (float64) and 32-bit floating-point numbers (float32) in little-endian representation.

The file starts with a main header that among others specifies the number of grids. This is followed by a grid header that among others specifies the size of the grid. Then the grid data is listed. This can be followed by the header and data of one or more additional subgrids. The final part of the file is a footer to indicate the end of file.

Note that besides correction values, also spacing and even grid bounds are given in arcseconds, but the precision values are in metres. Longitude values have inverted sign, thus east of the Greenwich meridian is negative. This affects the values for the grid bounds in the header and the correction values of the grid data. Also the order in which the correction values are listed, is from southeast to northwest. The values in the header are preceded by an attribute name for the value.

Main header

attribute name (8 char); number of main header lines (uint32); zero value (uint32);
attribute name (8 char); number of grid header lines (uint32); zero value (uint32);

attribute name (8 char); number of grids (uint32); zero value (uint32);
attribute name (8 char); units (8 char);
attribute name (8 char); file format version (8 char);
attribute name (8 char); source datum (8 char);
attribute name (8 char); target datum (8 char);
attribute name (8 char); half major axis of ellipsoid of source datum in metres (float64);
attribute name (8 char); half minor axis of ellipsoid of source datum in metres (float64);
attribute name (8 char); half major axis of ellipsoid of target datum in metres (float64);
attribute name (8 char); half minor axis of ellipsoid of target datum in metres (float64)

Grid header

attribute name (8 char); grid name (8 char);
attribute name (8 char); parent name (8 char);
attribute name (8 char); creation date (8 char);
attribute name (8 char); modification date (8 char);
attribute name (8 char); latitude in arcseconds of southern bound of grid (float64);
attribute name (8 char); latitude in arcseconds of northern bound of grid (float64);
attribute name (8 char); longitude in arcseconds of western bound of grid (float64);
attribute name (8 char); longitude in arcseconds of eastern bound of grid (float64);
attribute name (8 char); latitude spacing in arcseconds (float64);
attribute name (8 char); longitude spacing in arcseconds (float64);
attribute name (8 char); number of grid points (uint32); zero value (uint32)

Grid data

latitude correction in arcseconds of grid point in SE corner of grid (float32);
longitude correction in arcseconds of grid point in SE corner of grid (float32);
precision in metres of latitude correction of grid point in SE corner of grid (float32);
precision in metres of longitude correction of grid point in SE corner of grid (float32);

latitude correction in arcseconds of grid point <u>west</u> of SE corner of grid (float32);
longitude correction in arcseconds of grid point <u>west</u> of SE corner of grid (float32);
precision in metres of latitude correction of grid point <u>west</u> of SE corner of grid (float32);
precision in metres of longitude correction of grid point <u>west</u> of SE corner of grid (float32);

[…]

latitude correction in arcseconds of grid point in NW corner of grid (float32);
longitude correction in arcseconds of grid point in NW corner of grid (float32);
precision in metres of latitude correction of grid point in NW corner of grid (float32);
precision in metres of longitude correction of grid point in NW corner of grid (float32)

Footer

attribute name (8 char); value $0.33 \cdot 10^{33}$ (float64)

Some data like the units and axis parameters of the ellipsoids of the source and target datum are ignored by PROJ.

A2.2   VDatum file format

The binary VDatum file format (.gtx) uses 64-bit floating-point numbers (float64), 32-bit unsigned integer numbers (uint32) and 32-bit floating-point numbers (float32) in big-endian representation. The file starts with a header that among others specifies the size of the grid. This is followed the grid data. No footer is used to indicate the end of file. Note that the big-endian representation is different than the little-endian representation of the NTv2 file.

Header

latitude of grid point in SW corner of grid in decimal degrees (float64);
longitude of grid point of SW corner of grid in decimal degrees (float64);
latitude spacing in decimal degrees (float64);
longitude spacing in decimal degrees (float64);
number of grid rows (uint32);
number of grid columns (uint32)

Grid data

quasi-geoid height in metres of grid point in SW corner of grid (float32);
quasi-geoid height in metres of grid point east of SW corner of grid (float32);
[…]
quasi-geoid height in metres of grid point in NE corner of grid (float32)

A2.2   Geodetic TIFF grid file format

The binary GeoTIFF file format (.tif) is well documented and the use for Geodetic TIFF Grid files is specified by PROJ. A Geodetic TIFF Grid file uses floating point numbers in little-endian representation with compression. The file starts with a header that among others specifies the size of the grid and the fact that the grid values are point values for the pixel centres. Then the grid data is listed with band interleave, so first all values of the first band are given and then all values for the next band (latitude, longitude). Vertical grids typically have one single band. This can be followed by one or more additional subgrids. The order in which the grid values are given in GeoTIFF files is from northwest to southeast and starts in west-east direction. Since Geodetic TIFF grid files are mostly created by conversion from NTv2 and VDatum files, often the same units are used. Latitude and longitude values have conventional sign, thus north of the equator and east of the Greenwich meridian are positive.

Appendix 3: Implementation of RDNAPTRANS™2018 using open source library PROJ

The open source library PROJ can be used to transform coordinates with RDNAPTRANS™. The recommended version is PROJ 7 or newer. Commands for the cct program of PROJ are given for 3D transformation, 2D transformation and height transformation for the two implementation variants. Each transformation is one command split over several lines, with each step starting on a new line with comments for readability. Commands for transformation to and from geocentric Cartesian ETRS89 coordinates and commands for the PROJ program cs2cs to convert to and from degrees, minutes and seconds are given too.

Implementation variant 1 is the recommended variant. However, it requires a pipeline of transformation steps, which is not possible in many software packages, including PROJ 4 (which provided the current version for 1994–2018). Horizontal implementation variant 2 is limited to the bounds of the correction grid, but is supported by more software packages. Vertical implementation variant 2 is mainly used for old PROJ versions. For other software, a hybrid implementation using horizontal implementation variant 2 and vertical implementation variant 1 can be preferable.

A3.1   3D transformation with implementation variant 1

When height information is available, 3D ETRS89 coordinates can be transformed to and from RD and NAP by using separate commands for 2D transformation (Section A3.3) and height transformation (Section A3.5). These commands for implementation variant 1 give 2D results outside the bounds of the quasi-geoid grid, without height value.

A3.2   3D transformation implementation with the hybrid variant

When height information is available, 3D ETRS89 coordinates can be transformed to and from RD and NAP with the command below. This command for the hybrid implementation variant (horizontal variant 2 and vertical variant 1) returns an error outside the bounds of the correction and quasi-geoid grids. If no height is present in the input, this command returns an error too.

From RD and NAP to ETRS89

Command	Comments
`cct -I -o output.txt +proj=pipeline`	*Set optional output file;*
`+step +proj=unitconvert +xy_in=deg +xy_out=rad`	*Convert units (Section 2.1 / 3.4);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=vgridshift +grids=nlgeo2018.tif`	*Transformation (Section 2.5 / 3.5);*
`+step +proj=hgridshift +inv +grids=rdtrans2018.tif`	*Correction (Section 2.3 / 3.2);*
`+step +proj=sterea +lat_0=52.156160556 +lon_0=5.387638889`	*Projection (Section 2.4 / 3.1);*
`  +k=0.9999079 +x_0=155000 +y_0=463000 +ellps=bessel`	
`input.txt`	*Set optional input file*

For the inverse transformation from ETRS89 to RD and NAP, the option `-I` should be omitted.

A3.3   2D transformation implementation variant 1

When no height information is available or needed, the horizontal ETRS89 coordinates can be transformed to and from RD with the command below. This command for implementation variant 1 gives results outside the bounds of the correction grid too. Any height information is replaced by zero.

<div style="background-color:#e0ebd0; padding:1em;">

From RD to ETRS89 without height transformation

Command	Comments
`cct -I -o output.txt +proj=pipeline`	*Set height and optional output file;*
`+step +proj=unitconvert +xy_in=deg +xy_out=rad`	*Convert units (Section 2.1 / 3.4);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=push +v_3`	*Save height value*
`+step +proj=set +v_3=43 +omit_inv`	*Set height for Helmert transformation;*
`+step +proj=cart +ellps=GRS80`	*Conversion (Section 2.2 / 3.3);*
`+step +proj=helmert +x=-565.7346 +y=-50.4058 +z=-465.2895`	*Transformation (Section 2.2 / 3.3);*
`  +rx=-0.395023 +ry=0.330776 +rz=-1.876073 +s=-4.07242`	
`  +convention=coordinate_frame +exact`	
`+step +proj=cart +inv +ellps=bessel`	*Conversion (Section 2.2 / 3.3);*
`+step +proj=hgridshift +inv +grids=rdcorr2018.tif,null`	*Correction (Section 2.3 / 3.2);*
`+step +proj=sterea +lat_0=52.156160556 +lon_0=5.387638889`	*Projection (Section 2.4 / 3.1);*
`  +k=0.9999079 +x_0=155000 +y_0=463000 +ellps=bessel`	
`+step +proj=set +v_3=0 +omit_fwd`	*Set height for Helmert transformation;*
`+step +proj=pop +v_3`	*Save height value;*
`input.txt`	*Set optional input file*

</div>

For the inverse transformation from ETRS89 to RD, the option `-I` should be omitted.

A3.4   2D transformation implementation variant 2

When no height information is available or needed, the horizontal ETRS89 coordinates can be transformed to and from RD with the command below. This command for implementation variant 2 returns an error outside the bounds of the correction grid. Any height information is replaced by zero.

<div style="background-color:#e0ebd0; padding:1em;">

From ETRS89 to RD without height transformation

Command	Comments
`cct -I -o output.txt +proj=pipeline`	*Set height and optional output file;*
`+step +proj=unitconvert +xy_in=deg +xy_out=rad`	*Convert units (Section 2.1 / 3.4);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=hgridshift +inv +grids=rdtrans2018.tif`	*Correction (Section 2.3 / 3.2);*
`+step +proj=sterea +lat_0=52.156160556 +lon_0=5.387638889`	*Projection (Section 2.4 / 3.1);*
`  +k=0.9999079 +x_0=155000 +y_0=463000 +ellps=bessel`	
`input.txt`	*Set optional input file*

</div>

For the inverse transformation from ETRS89 to RD, the option `-I` should be omitted.

A3.5   Height transformation implementation variant 1

When using ETRS89 coordinates, it can be useful to transform only the ellipsoidal ETRS89 height to and from NAP height with the command below. The command gives an error outside the bounds of the quasi-geoid grid or if no height is present in the input.

<div style="background-color:#e4ecd4; padding:10px;">

From ETRS89 with NAP height to ETRS89 with ellipsoidal height

Command	Comments
`cct -I -o output.txt +proj=pipeline`	*Set optional output file;*
`+step +proj=unitconvert +xy_in=deg +xy_out=rad`	*Convert units (Section 2.1 / 3.4);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=vgridshift +grids=nlgeo2018.tif`	*Transformation (Section 2.5 / 3.5);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=unitconvert +xy_in=rad +xy_out=deg`	*Convert units (Section 2.1 / 3.4);*
`input.txt`	*Set optional input file*

</div>

For the inverse transformation from ETRS89 with ellipsoidal height to ETRS89 with NAP height, the option `-I` should be omitted.

A3.6 Conversion of geocentric Cartesian ETRS89 coordinates

When geocentric Cartesian ETR89 coordinates are needed as input or output, these can be converted to or from ellipsoidal geographic ETRS89 coordinates with the commands below.

The intermediate results in RDNAPTRANS™ should not be used as geocentric Cartesian ETR89 coordinates for input or output, as these have a fixed height which might not be appropriate for other applications and coordinates with the actual height are not appropriate as intermediate input for RDNAPTRANS™. To transform between RD coordinates with NAP height and geocentric Cartesian ETRS89 coordinates, the conversion to or from geocentric Cartesian ETRS89 coordinates should be an additional step before or after the 3D transformation of RDNAPTRANS™.

<div style="background-color:#e4ecd4; padding:10px;">

Conversion from geocentric Cartesian ETRS89

Command	Comments
`cct -I -o output.txt +proj=pipeline`	*Set optional output file;*
`+step +proj=unitconvert +xy_in=deg +xy_out=rad`	*Convert units (Section 2.1 / 3.4);*
`+step +proj=axisswap +order=2,1`	*Swap for PROJ order longitude, latitude;*
`+step +proj=cart +ellps=GRS80`	*Conversion;*
`input.txt`	*Set optional input file*

</div>

For the inverse transformation to geocentric Cartesian ETRS89, the option `-I` should be omitted.

A3.7 Notation in degrees, minutes and seconds

When coordinates in degrees, minutes and seconds are needed as input or output, the notation can be converted with the separate commands below, using a different program of PROJ named cs2cs.

<div style="background-color:#e4ecd4; padding:10px;">

From decimal degrees to degrees, minutes and seconds

Command	Comment
`cs2cs -r -s -W5 +proj=latlon +to +proj=latlon`	*Notation (Section 3.4);*

</div>

From degrees, minutes and seconds to decimal degrees	
Command	Comment
`cs2cs -r -s -d 9 +proj=latlon +to +proj=latlon`	*Notation (Section 2.1);*

A3.8 PROJ4-style strings

When using software relying on legacy PROJ 4-style strings, the string below can be used for RD and NAP.

PROJ 4-style string of RD and NAP in PROJ ≥6 (hybrid implementation variant)

```
+proj=sterea +lat_0=52.156160556 +lon_0=5.387638889 +k=0.9999079 +x_0=155000 +y_0=463000
+ellps=bessel +nadgrids=rdtrans2018.tif +geoidgrids=nlgeo2018.tif +geoid_crs=WGS84 +type=crs
```

With PROJ versions before PROJ 6.0.0, the file naptrans2018.gtx had to be used instead of nlgeo2018.tif, besides some differences in the syntax below.

PROJ 4-style string of RD and NAP in PROJ 4 (implementation variant 2)

```
+proj=sterea +lat_0=52.156160556 +lon_0=5.387638889 +k=0.9999079 +x_0=155000 +y_0=463000
+ellps=bessel +nadgrids=rdtrans2018.gsb +wktext +geoidgrids=naptrans2018.gtx +no_defs
```

PROJ versions before PROJ 6.3.0 give an error or wrong results on the bounds of the dense subgrid (and the bounds of the course parent grid) of the NTv2 file. However, due to the location over 50 km from the Dutch shore, this will be of little concern for most users.

Appendix 4: Application for use of the trademark RDNAPTRANS

Kadaster and Rijkswaterstaat, working together in the Netherlands Partnership Geodetic Infrastructure (NSGI), developed RDNAPTRANS™2018 as precise and official transformation between RD with NAP height and ETRS89. NSGI makes the parameters and associated data files for RDNAPTRANS™ freely available under the creative commons license CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/). The validity of RDNAPTRANS™2018 starts: 01-09-2019. The validity of RDNAPTRANS™2008 stops: 01-10-2022. RDNAPTRANS is a registered trademark, all rights on the use of this name are in the hands of NSGI.

To receive permission to use the name RDNAPTRANS™2018 in software, brochures, advertisements etc., fill in this sheet and send it with attachments* to NSGI.

Software name and first version number that uses RDNAPTRANS™2018	
Company name and address	
Contact name and e-mail address	

I herewith declare that the above-mentioned software meets the RDNAPTRANS™2018 specifications as tested by the Validation Service of RDNAPTRANS™2018 (http://www.nsgi.nl/, in Dutch).* Users of the software can recognise this correct transformation by "transformation according to RDNAPTRANS™2018". I approve that a notice will be sent to the contact e-mail address for important changes in RDNAPTRANS™2018 or the specifications of the Validation Service.

Name *The signatory must be authorized to sign on behalf of the company*	
Date	
Signature	

Software permitted to use the name RDNAPTRANS™2018 may be listed on an NSGI website. This permission can be stopped by NSGI at any moment when the software does not meet the specifications as used by the Validation Service of RDNAPTRANS™2018. The NSGI and/or the agencies of NSGI are not responsible for the correctness of a program using the name RDNAPTRANS™2018. The NSGI and agencies of NSGI are not liable for any damage, loss or injury caused by use or incorrectness of the supplied information and files.

Please send this form and attachments* to: rd@kadaster.nl

*   **Required attachments:** Screen shots of the output of the Validation Service in both transformation directions (RD and NAP to ETRS89 as well as ETRS89 to RD and NAP), showing 100% within the bounds of the grids. Any results outside the grids, are best accompanied with a warning that the RD / NAP coordinates are out of bounds.