Geodetic Development Kit GeoDLL

Language / Sprache
If you prefer to read in German language, please use the **Liesmich.pdf** file!
Wenn Sie lieber in Deutscher Sprache lesen möchten, benutzen Sie bitte die Datei **Liesmich.pdf**!

Helpful links

https://www.killetsoft.de/bestell/gdl_be_e.htm	Online order form with current prices
https://www.killetsoft.de/s_prei_e.htm	Printable order form and current price list
https://www.killetsoft.de/h_prod_e.htm	Overview of the software distributed by KilletSoft

Content

Dynamic Link Library for software developer
GeoDLL supports the development of geodetic software on many platforms by providing geodetic functions. GeoDLL contains precise calculations on the themes 2D and 3D Coordinate Transformation, Reference System Transitions, meridian strip changing, user defined Coordinate Reference Systems, distance calculations, Digital Elevation Model, NTv2 and HARN support, Analyzing and manipulation of NTv2 files, Polygonal Validity Scopes, Direct / Inverse solutions, map function, Time Zone calculations and geodetic converting functions.

GeoDLL performs Coordinate Transformations fast and with high accuracy. For this purpose the DLL supports thousands of worldwide Coordinate Systems, geodetic Reference Systems, Reference System Transitions (geodetic datum shifts), user-defined systems, 2D/3D transformations, Extra Parameters like Meridian Convergence and Origin Center Meridian, INSPIRE, NTv2, HARN, EPSG, GPS, Continental Drift and more.

The operating system WINDOWS provides software developers the opportunity to use prepared functions of third-parties in their own software applications. Thus, geodetic functions of GeoDLL can be linked to programs written in C, C++, C#, Java, Delphi, MS-Access, Visual Basic, CA-Visual Objects or which are written in other programming languages. To support the GeoDLL, examples and interfaces with source code for many commonly used programming languages are provided.

GeoDLL comes with extensive documentation and is supplied as DLL for 32bit and 64bit architectures file or as C / C++ source code. GeoDLL works with most programming languages and it can be used with Microsoft Office programs. GeoDLL is written in C / C ++ and is developed using Microsoft Visual Studio. Thus very fast performance, compact code and high stability are assured. The geodesic functions of GeoDLL are arranged in function groups that can be licensed separately.

C++ Source Code
It is frequently asked whether the geodetic functions are also available for other operating systems such as LINUX or UNIX. Also some developers would not like to build in geodetic solutions without knowledge of the source code into their programs. For these reasons, the complete GeoDLL can also be purchased as C++ source code.

The source is extensively written in ANSI-C++, so that the migration to other operating systems and hardware platforms is possible with manageable effort. GeoDLL has been developed with Microsoft Visual Studio under the operating system WINDOWS. All source codes of GeoDLL are directly compilable and executable under WINDOWS with Microsoft Visual Studio, Versions 10 to 17. For the conversion to other platforms or other development environments adjustments in the source code may be necessary.

Coordinate and Reference systems (datum shifts) supported by GeoDLL
An actual list of all coordinate and reference systems supported by GeoDLL in English language you find on KilletSoft's website https://www.killetsoft.de/p_gdll_e.htm, in the file crs.pdf or after installation in the GeoDLL help file.

Worldwide Coordinate Transformations
The most important application of GeoDLL is to include professional coordinate transformations in own programs. These coordinate and reference systems are supported:

- The current and historical Systems of all countries of the European Union (EU)
- The Coordinate Reference Systems of the European non-EU countries

- The European ETRS89 systems forced by INSPIRE
- The US and Canadian NAD, NTv2, HARN and SPCS Coordinate Reference Systems
- The Coordinate Reference Systems of most countries of all continents
- The Coordinate Reference Systems of Austria and Switzerland incl. NTv2
- The Coordinate Reference Systems of the old and new German Federal States
- The German 'Lagestatus' Coordinate Reference Systems
- The high-precision Reference Systems of the German Federal States incl. NTv2
- The 40 Soldner Coordinate Systems of the Prussian Land Registers
- The Geographic coordinates in different notations and Cartesian coordinates
- Worldwide NTv2 grid file supported Coordinate Transformations
- Worldwide used numeric and alphanumeric Coordinate Systems
- Most of EPSG supported Coordinate Reference Systems
- ITRS annual realizations or WGS84 epochs for GPS measurements
- WGS84 Coordinate Transformations in consideration of the continental drift
- User defined Coordinate Systems, Reference Systems and earth ellipsoids
- Meridian Convergence, Origin Center Meridian, Geographical Point Coordinates

Scope of services of the function groups
The services of GeoDLL are grouped in function groups, which can be separately licensed and purchased.

- Function group "Coordinate Transformations"
  - Coordinate Transformation
  - Reference System Transition
  - Meridian Strip Transition
  - 2D and 3D Coordinate Transformation
  - Helmert and Molodensky Reference System Transition
  - NTv2 and HARN grid file support
  - Forward and backward transformations
  - Numeric and alphanumeric coordinates
  - Geographic and Cartesian coordinates
  - Many projections (also rarely used ones)
  - Thousands of predefined systems (see below)
  - Use of EPSG codes
  - Use of Measurement Units
  - Calculation of Meridian Convergence and Origin Center Meridian
  - Calculation of the Geographical Point Coordinate
  - Calculation of more extra parameters
  - Coordinate System range validity check
- Additional function group "NTv2 Transformations"
  - Predefined NTv2 Reference Systems
  - Embedding of any NTv2 grid files
  - Support of Polygonal Validity Scopes in NTv2 files
  - Use of ASCII grid files (.gsa) and binary grid files (.gsb)
  - Determining the parameters of a NTv2 grid file
  - Automatically allocation of NTv2 files from a common directory
  - Download of many NTv2 files from the KilletSoft website
  - Links to NTv2 providers on the KilletSoft website
  - HARN grid files of the U.S. supported as equivalent NTv2 grid files
  - Free use of some NTv2 files, which otherwise are available for a fee
  - Access to NTv2 files, which are specially licensed for KilletSoft
- Function group "NTv2 Tools"
  - ASCII grid files (.gsa) and binary grid files (.gsb)
  - Conversion of ASCII grid files to binary grid files
  - Conversion of binary grid files to ASCII grid files
  - Copy of NTv2 grid file areas to new binary files
  - Supplement of a NTv2 binary file with Polygonal Validity Scopes
  - Determining the parameters of a NTv2 grid file
  - Export and import of NTv2 grids
  - Download of many NTv2 files from the KilletSoft website
  - Links to NTv2 providers on the KilletSoft website
  - HARN grid files of the U.S. supported as equivalent NTv2 grid files
  - Free use of some NTv2 files, which otherwise are available for a fee
  - Access to NTv2 files, which are specially licensed for KilletSoft

- Function group "User definitions"
  - Custom coordinate systems for many projections
    - 16 possible projection types
    - Orthogonal Output Device Projection (Pixel calculation)
  - Custom reference systems
    - Coordinate Frame Rotation (Seven parameters, Helmert)
    - Position Vector Transformation (Seven parameters, Bursa-Wolf)
    - European Standard (Seven parameters, ISO 19111)
    - Molodensky (Three parameters)
  - Custom earth ellipsoids
    - Semi major and minor axes
    - Flattening
- Function group "Parameter determination"
  - Determination of GeoDLL codes equivalent to EPSG codes
  - Parameter, notation and range validity of Coordinate Systems
  - Parameters of Reference Systems and Reference System Transitions
  - Semi-axes and flattening of Earth Ellipsoids
  - Designation and parameter of Measurement Units
  - Formatted text representation of a Coordinate Reference System
  - Determination of many other parameters used by GeoDLL
- Function group "Distance calculations"
  - Distance between coordinates on the ellipsoid
  - Distance between coordinates on the sphere
  - Distance between UTM coordinates
  - Destination point from start point, bearing, distance on ellipsoid
  - Destination point from start point, bearing, distance on sphere
  - UTM destination point from UTM start point, bearing and distance
  - Vincenty's Direct and Inverse Position Computation
    - Distance, forward bearing, backward bearing
- Function group "Notation calculations"
  - Conversions of Geographic Coordinates
    - Decimal notation (degree)
    - Gradual notation (DMS)
    - Nautical notation (DM)
    - Second notation
    - Gonal notation
  - Exact rounding of Geographic Coordinates
- Function group "Map calculations"
  - Parameters of the Topographic Maps 1:25000 to 1:200000
    - Determination of TK25 numbers from ordinates
    - Determination of map corner coordinates of a TK
    - Determine TK50, TK100 und TKU200 numbers from a TK25 number
- Function group "Elevation calculations"
  - Calculation of elevations from the 3 sec. elevation model CGIAR
  - Calculation of elevations from the 30 sec. elevation model GLOBE
  - Conversion of CGIAR ASCII files to binary files
  - Detailed Information about the Digital Elevation Models in the help file
  - Links to CGIAR and GLOBE providers on the KilletSoft website
- Function group "Transformation parameter"
  - Calculation of Seven Helmert parameters from identical points
  - Calculation of Three Molodensky parameters from identical points
  - Calculation of outliers in a group of identical points
  - Calculation of the residuals from a group of identical points
  - Correction with Residual Gaps Distribution by Natural Neighbour Interpolation
  - Calculation of the "Maximum Spatial Residual" from a group of residuals
  - Calculation of the "Average Spatial Residual" from a group of residuals
  - Calculation of the "Root Mean Square Residual" (RMS) from residuals
- Function group "Time Zone Calculations"
  - List of Time Zones with designation, UTC and time zone index
  - Exact Time Zone calculation from coordinates using a shape file
  - Fast Time Zone calculation from coordinates using a 0,1 degree grid file
  - Optional inclusion or 3, 12, 24 and 200 mile territorial limits

    - Determination of a time zone index from a coordinate
    - Determination of Time Zone designation from time zone index
    - Calculation of UTC, DST und Daylight Saving start and end dates
- Group of other (free) functions
    - Input of the unlock parameter
    - Information about GeoDLL, provider, author and licensee
    - Latest error code and error description
    - Language selection (English, German) for all text returns
    - Switch for use or non-use
        - of Coordinate System range validity check
        - of internal error handling
        - of multithreading environment
        - of the output of messages to the EventLog
        - of fast Static Variables
        - of the automatic memory management
        - of the event handling in time-intensive functions

Source and target Coordinate Reference Systems
- Worldwide and country-specific Coordinate Reference Systems
- Current and historical Coordinate Reference Systems
- Numeric and alphanumeric Coordinate Systems
- UTMRef, GEOREF, QTH, BNG,NAC und ING with different grid mesh sizes
- Plus Code, Google World / Pixel / Tile coordinates with different grid sizes
- INSPIRE systems, ITRS annual realizations, WGS84 epochs, GPS coordinates
- 2D and 3D Coordinate Transformations
- Use of EPSG codes of the Coordinate Reference Systems
- Selection of the meridian strip with UTM and Gauss-Krueger coordinates
- UTM and Gauss-Krueger coordinates with and without meridian strip number
- Use of measurement units
- Monitoring of range limits
- Option for the automatic assignment of a Reference System to the Coordinate System
- Calculation of Helmert and Molodensky parameter sets from identical points
- Correction with Residual Gaps Distribution by Natural Neighbour Interpolation
- NTv2 tools to manipulate and transform NTv2 ASCII and binary files

Quality
- Strict formulas of Schatz, Schuhr, Klotz and Hooijberg
- Transformation parameters of the Surveying Authorities of the respective countries
- Consideration of the EPSG specifications
- Helmert Seven Parameter Bursa-Wolf and Molodensky Reference System Transitions
- Helmert transitions with forward and backward transformations
- Helmert rotation matrix also for larger rotation angles
- Exact NTv2 transformations for many countries
- Support of Polygonal Validity Scopes in NTv2 files
- High-precision NTv2 transformations for German Federal States and other states provinces
- HARN grid files of the U.S. supported as equivalent NTv2 grid files

Special features
- 32bit and 64bit architecture
- Network capability
- Multithreading capability
- Server capacity
- CITRIX support
- EventLog handling

Help System
- Detailed electronic manual GeoDLL_e.chm included
- Online manual at https://www.killetsoft.de/h_geodll_e/handbuch_e.htm
- Detailed list with predefined Coordinate Reference Systems
- Coordinate Systems and Reference Systems in the list have numerical GeoDLL keys
- Hierarchical structure of the list by continent, country, Coordinate System, Reference System
- Additional alphabetic list

- Online FAQ section for common questions
- Uniformly geodetic terms in all text outputs and in the electronic manual
- Explanation of geodetic terms in the glossary

Multilingualism
- Text outputs in English and German
- User manual in English and German

Application Program Interfaces and Sample Programs
- Template for a C++ interface
- Template for a Visual Basic interface
- Template for a Delphi interface
- Template for a CA-Visual Objects interface
- Template for a C#-interface (NET Framework)
- Sample program in C++ of the Visual Studio Project GeoTestCpp32
- Sample program in C++ of the Visual Studio Project GeoTestCpp64
- Sample program in C++
- Sample program in Delphi
- Sample program in CA-Visual Objects
- Sample program in Python / Spyder
- Sample of function call in C++ syntax
- Sample of function call in Visual Basic syntax

More possibilities
- Possibility to download NTv2 files from the KilletSoft website
- Configuration of user defined Coordinate Systems
- Configuration of user defined Reference Systems and ellipsoids
- Possibility of a service contract for phone and email support
- Possibility to use the automated information service via email
- Transfer of transformation parameter sets from the SEVENPAR program

Before installing...
Favor of the actuality KilletSoft abstains from expensive digital signatures. A signature is only valid for one specific program version. But KilletSoft uploads several times a month improved programs immediately as new versions to the Internet. KilletSoft guarantees the integrity and virus-check of all programs, which are downloaded from the KilletSoft-website. The message "Unknown Publisher" can thus be ignored confidently.

Installation
The files of the Dynamic Link Library GeoDLL are stored in a directory GeoDLL of the data carrier or in a ZIP file which was downloaded from the Internet. In order to be able to use the library, it must be installed first. For the installation of the library please execute the install program geodll_setup.exe.

The installation can be executed under Microsoft WINDOWS 2000, NT, XP, VISTA, 7, 8, 10, 11 and future compatible operating systems.

It is important to close all open applications exept the windows explorer before starting the installation. Still open applications could use files, to which the installation program must have access during the installation.

After the installation all necessary files and the detailed documentation are available in the GeoDLL installation directory and in the subdirectories created by the install program.

To be able to use functions of GeoDLL in your application, the files geodll32.dll or geodll64.dll and geodllbn.bin must be available in the start directory of your application.

The provided interface files help you to integrate the geodetic functions of GeoDLL into the programming language of your choice. Further exemplary interfaces and programming examples for different programming languages are contained in the provided Help File GeoDLL_e.chm in the chapter "Definition and Interface Files".

Unlocking
After the installation GeoDLL is present as a limited test version. After the start of your application nearly all functions from the DLL can be called without reservation for a few times. For testing the executability of the DLL and testing the operability of the DLL functions it should be enough. The test version also displays a small message box. For further tests the application must be started again.

To be able to use the functions of the DLL without any reservation you must buy the unlock codes of the function groups you needed. The unlock code is implemented inside your application by the call of the DLL function setunlockcode(<unlock code>, <licensee designation>). Afterwards all functions of the unlocked function groups can be called without any temporal restriction. A new installation of GeoDLL is not necessary!

Examples program
In the GeoDLL start menu folder you find a small executable program named geotest.exe, which demonstrates some geodetic functions. You find the source code of the program in the file geotest.prg, which is written in the programming language CA-VO.

Price list
Prices and a purchase order form for the order of the unrestricted full version of the GeoDLL functions you will find in the GeoDLL start menu folder or in the GeoDLL installation directory. For fast shipping you can order the unrestricted full version of the GeoDLL with our internet online order form.

Geodetic functions contained in GeoDLL
"Coordinate Transformations" function group
- Function coordtrans() - 2D Coordinate transform. / Reference System Transition, num. / alphanum. (char**)
- Function coordtrans2() - 2D Coordinate transform. / Reference System Transition, num. / alphanum. (char*)
- Function coordtrans3() - 2D Coordinate Transformation / Reference System Transition, only numeric, notations
- Function coordtrans4() - 2D Coordinate Transformation / Reference System Transition, only numeric, no notations
- Function coordtransex() - 2D Coordinate transform. / Reference System Transition, reduced eastings
- Function coordtrans3d() - 3D Coordinate transf. / Reference System Transition, num. / alphanum. (char**)
- Function coordtrans3d2() - 3D Coordinate transf. / Reference System Transition, num. / alphanum. (char*)
- Function coordtrans3d3() - 3D Coordinate Transformation / Reference System Transition, only numeric
- Function coordtrans3d4() - 3D Coordinate Transformation / Reference System Transition, only numeric, no notations
- Function coordtrans3dex() - 3D Coordinate transform. / Reference System Transition, reduced eastings
- Function coordtransepsg() - EPSG code Coordinate Transformation / Datum shift
- Function meritrans() - Meridian strip change with GK und UTM coordinates
- Function setcoordarea() - Coordinate System range validity check on / off
- Function setcalcextra() - Calculation of extra parameters on / off
- Function coordcalcextra() - Calculation of extra parameters during coordinate transformations
- Function coordcalcextraind() - Calculation of extra parameters independent from coordinate transformations

"NTv2 Transformations" function group
(additional requires the unlocked function group "Coordinate Transformations")
- Function getntvbinaryfile() - Determination of NTv2 file names matching a reference system
- Function getntvdirmatch() - Determination of a matching NTv2 file in a common directory
- Function getntvrefbelong() - Determination of the second NTv2 Reference System
- Function getntvrefequiv() - Determination of an NTv2 equivalent Reference System
- Function getntvrefstatus() - Determination of the NTv2 membership of a Reference System
- Function setntvbinaryfile() - Initializes a NTv2 grid data binary file for Reference System Transitions
- Function setntvbinautodir() - Sets a directory for the automatically use of NTv2 binary files
- Function setntvbinautofile() - Sets a NTv2 grid binary file for automatically call in a wait position
- Function setntvpolyvalid() - Sets Polygonal Validity Check in NTv2 files
- Function closentvbinaryfile() - Close NTv2 binary file and free NTv2 instance

"NTv2 Tools" function group
- Function convntvascii2bin() - Converts a NTv2 ASCII file to a binary file
- Function convntvbin2ascii() - Converts a NTv2 binary file to an ASCII file
- Function convntvbin2area() - Copies an area of a NTv2 grid file to a new binary file
- Function convntvbin2gridexport() - Exports a NTv2 grid from a NTv2 binary to a new binary file
- Function convntvbin2gridimport() - Imports a NTv2 grid from a NTv2 binary into an existing binary file
- Function convntvbin2polyvalid() - Adds Polygonal Validity Scopes to NTv2 binary files
- Function getntvgridarray() - Writes the parameters of NTv2 subgrids in arrays
- Function getntvgridcount() - Gets count of subgrids contained in a NTv2 file
- Function getntvgridheader() - Gets header parameter of the NTv2 subgrids
- Function getntvheader() - Gets header parameter of a NTv2 file
- Function getntvminmaxshift() - Gets the min / max shift values of the NTv2 subgrids

"Parameter determination" function group

- Function getepsg2geodll() - Determining GeoDLL codes equivalent to EPSG code
- Function getepsgcrsname() - Determining the designation of an EPSG CRS
- Function getcoordname() - Coordinate System name
- Function getcoordsys() - Formatted Coordinate System parameters
- Function getcoordform() - 2D Coordinate System coordinates notation
- Function getcoordform3d() - 3D Coordinate System coordinates notation
- Function getcoordaxis() - 2D Coordinate System axes names
- Function getcoordaxis3d() - 3D Coordinate System axes names
- Function getcoordarea() - Range validity of a Coordinate System
- Function getcoordfixref() - Determining Coordinate System has fixed Reference System
- Function getcoordstdrefsys() - Default Reference System of a Coordinate System
- Function getcoordstdunitpar() - Default Measurement Unit of a Coordinate System
- Function getcoordproj() - Projection method number of a Coordinate System
- Function getcoordstrstatus() - Determining Coordinate System is a strip system
- Function getrefname() - Reference System name
- Function getrefsys() - Formatted Reference System parameters
- Function getellname() - Ellipsoid name
- Function getellsys() - Formatted ellipsoid parameters
- Function getellsource() - Source ellipsoid demi-axes
- Function getelltarget() - Target ellipsoid demi-axes
- Function getunitname() - Designation of a Measurement Unit
- Function getunitpar() - Calculation constant of a Measurement Unit

"Distance calculations" function group

- Function distancegeo() - Distance between geographic coordinates on the ellipsoid
- Function distancesphere() - Distance between geographic coordinates on the sphere
- Function distanceutm() - Distance between UTM coordinates on the ellipsoid
- Function point2pointgeo() - Dest. point on ellipsoid from start point, bearing, distance
- Function point2pointsphere() - Dest. point on sphere from start point, bearing, distance
- Function point2pointutm() - UTM dest. point from UTM start point, bearing and distance
- Function vincentydirect() - Vincentys Direct Position Computation
- Function vincentyinverse() - Vincentys Inverse Position Computation

"Notation calculations" function group

- Function umfd2g() - Translates decimal notation to degrees notation
- Function umfd2gn() - Translates decimal notation to gonal notation
- Function umfd2n() - Translates decimal notation to nautical notation
- Function umfd2s() - Translates decimal notation to seconds notation
- Function umfg2d() - Translates degrees notation to decimal notation
- Function umfgn2d() - Translates gonal notation to decimal notation
- Function umfn2d() - Translates nautical notation to decimal notation
- Function umfs2d() - Translates seconds notation to decimal notation
- Function umfroundg() - Accurate rounding of degrees coordinates
- Function umfroundn() - Accurate rounding of nautical coordinates

"Map calculations" function group

- Function kartgeo2tk() - Determine TK25 number from geographic coordinates
- Function karttk2geo() - Determine corner map coordinates from TK25 number
- Function karttknum() - Determine TK50, TK100 und TKÜ200 from a TK25 number

"Elevation calculations" function group

- Function getelevation03() - Elevation from the 3 sec. elevation model CGIAR
- Function getelevation30() - Elevation from the 30 sec. elevation model GLOBE
- Function getelevation33() - Elevation from 3/30 sec. elevation model CGIAR/GLOBE
- Function setelev03datapath() - Sets the data path for the elevation model CGIAR
- Function setelev30datapath() - Sets the data path for the elevation model GLOBE
- Function convelev03ascii2bin() - Convert a CGIAR ASCII file to a binary file

"Transformation Parameter" function group

- Function gettranshelmert() - Calculation of Seven Helmert Parameter

- Function gettransmolodensky() - Calculation of Three Molodensky Parameter
- Function gettransoutliers() - Calculation of outliers
- Function gettransresiduals() - Calculation of the residuals
- Function gettransresidualcoord() - Correction with Residual Gaps Distribution
- Function gettransresidualmax() - Calculation of the "Maximum Spatial Residual"
- Function gettransresidualaverage() - Calculation of the "Average Spatial Residual"
- Function gettransresidualrms() - Calculation of the "Root Mean Square Residual" (RMS)

"Time Zone Calculations" function group
- Function settzshapefile() - Initialize and test a shapefile for time zone calculations
- Function gettzcurrentbynum() - Determ. of current time zone param. from GeoDLL index
- Function gettznamebynum() - Determ. of a time zone designation from a GeoDLL index
- Function gettznumbycoordexact() - Exact determination of GeoDLL index from a coordinate
- Function gettznumbycoordfast() - Fast determination of GeoDLL index from a coordinate
- Function gettzparbynum() - Determination of common time zone param. from GeoDLL index

Not locked other functions
- Function getauthor() - Copyright and program author's address
- Function getdisclaimer() - Disclaimer reference for GeoDLL
- Function getdllversion() - GeoDLL version number
- Function geterrorcode() - Latest error description
- Function getlicensee() - Licensee identification
- Function setcoordarea() - Coordinate System range validity check on / off
- Function setdllinit() - Initializing and functional readiness test of the GeoDLL
- Function setdllworkdir() - Directory for geodllbn.bin and other files
- Function seteventloop() - Event handling in time-intensive functions on / off
- Function setinternerrsys() - Internal error handler on / off
- Function setlanguage() - Language selection for all text returns
- Function setmultithreading() - Usage in a multithreading environment on / off
- Function setsilence() - Output of messages to the EventLog on / off
- Function setstaticuse() - Usage of fast Static Variables on / off
- Function setstringallocate() - Automatic memory management on / off
- Function setunlockcode() - Input of the unlock parameter

Killet GeoSoftware Ing.-GbR
Escheln 28a
47906 Kempen
Germany

Phone: +49 (0)2152 961127
Telefax: +49 (0)2152 961128

Email: https://www.killetsoft.de/email.htm
Site: https://www.killetsoft.de