



Grid InQuest DLL Guidance Note

Version 7

This guidance note is Crown Copyright (c) 2009, Secretary of State for Communities and Local Government, acting through Ordnance Survey.

All rights reserved.

Contents

1	Introduction	3
1.1	Overview	3
1.2	System Requirements.....	3
1.3	Installation of the Grid InQuest dll and dat files	4
2	DLL reference	5
2.1	Overview	5
2.2	Distributable Files	5
2.3	Set the path for the data files	5
2.4	Set the path for the data files from a registry key	6
2.5	Set the area to work in	6
2.6	Get the current working area.....	7
2.7	Initialise the DLL and load the data files	7
2.8	Set an ETRS89 Cartesian coordinate	8
2.9	Get an ETRS89 Cartesian coordinate	8
2.10	Set an ETRS89 Geodetic coordinate	9
2.11	Get an ETRS89 Geodetic coordinate	9
2.12	Set an ETRS89 UTM coordinate	10
2.13	Get an ETRS89 UTM coordinate.....	10
2.14	Set an Irish Grid coordinate	11
2.15	Get an Irish Grid coordinate	11
2.16	Set an Irish Transverse Mercator coordinate.....	12
2.17	Get an Irish Transverse Mercator coordinate	13
2.18	Set an OSGB36 coordinate	13
2.19	Get an OSGB36 coordinate	14
2.20	DLL Constant Definitions	15

1 Introduction

This document provides a reference for using the Grid InQuest dll. It is intended to help you get started using the dll in order to carry out accurate coordinate transformation in Great Britain, Northern Ireland and the Republic of Ireland.

1.1 Overview

The Grid InQuest ATL COM DLL provides a means for transforming coordinates between ETRS89 (WGS84) and the National coordinate systems of Great Britain, Northern Ireland and the Republic of Ireland.

A coordinate may be set and retrieved as any of the following:

- ETRS89 Cartesian
- ETRS89 Geodetic
- ETRS89 UTM
- OSGB36
- Irish Grid
- Irish Transverse Mercator

In addition, depending on where the coordinates are located, it will transform between ETRS89 ellipsoidal heights and the following orthometric height datums:

- Newlyn
- St Marys (Scilly Isles)
- Douglas02 (Isle of Man)
- Stornoway (Outer Hebrides)
- St Kilda
- Lerwick (Shetland Isles)
- Newlyn (Orkney Isles only)
- Fair Isle
- Flannan Isles
- North Rona
- Sule Skerry
- Foula
- Malin Head
- Belfast Lough

1.2 System Requirements

The Grid InQuest DLL will run on Windows XP, Vista, Windows 7.

1.3 Installation of the Grid InQuest dll and dat files

Upon installation of the Grid InQuest software, the dll and dat files are automatically made available on disk.

By default, the dll file can be found in the following location:-

C:\Program Files\Common Files\GridInQuest\Geodetics\GIQ60.dll

The installation process places GIQ60.DAT into [AllUsersAppData]\GridInQuest\GIQ60. On Windows XP this is...

C:\Documents and Settings\All Users\Application Data\GridInQuest\GIQ60.DAT

NB. The folder **Application Data** is by default a hidden folder. To see this folder and its contents, go to **Tools > Folder Options > View** to *Show Hidden Files and Folders*

On other Windows operating systems the .DAT file may be in a different location so perform a search of the system disk to locate it.

2 DLL reference

2.1 Overview

As part of the installation procedure, the dll should be registered on your computer. If it is not, it can be registered manually from the command prompt using the `regsvr32` command.

Essentially developers need to carry out the following steps in order to perform a coordinate transformation:

1. Load the dll into the software application
2. If the data file (GIQ60.dat) has been installed under a different location than the default, set the path of where the data file has been installed, either directly or through a registry key.
3. Set the area of interest (ie Great Britain, Northern Ireland or the Republic of Ireland)
4. Initialise the dll. Use the unlock string “**GIQ.6.0**” to pass into the function call. This will unlock the dll and load the transformation shifts and geoid model into memory.
5. Call one of the Set functions to set a coordinate in a particular format (eg ETRS89 Latitude, Longitude and ellipsoidal height, or OSGB36 Eastings, Northings and orthometric height).
6. Call one of the Get functions to retrieve the converted coordinate in a particular format (eg OSGB36 Eastings, Northings and orthometric height, or ETRS89 Latitude, Longitude and ellipsoidal height).
7. Repeat 5 and 6 as necessary.

2.2 Distributable Files

Once the application has been completed, the following files must be distributed as part of the Grid InQuest DLL:

1. GIQ60.dll
2. GIQ60.dat

Make sure the data file (GIQ60.dat) is installed in the same path as your software is expecting.

The remainder of the section gives a list of all the dll functionality that is available.

2.3 Set the path for the data files

Function Call:

Inputs:	BSTR	Full path of the location of the data files
---------	------	---

a) eSuccess	All ok
b) eInvalidDataFilePath	Data files not found under the path supplied

Function Call:

Inputs:	BSTR	Full registry key containing the path of the location of the data files
---------	------	---

a) eSuccess	All ok
b) eInvalidRegKey	Registry key not found
c) eInvalidDataFilePath	Data files not found under the path supplied by the registry key

Function Call:

Inputs:	eArea	Area to work in
a)	eAreaGreatBritain	Use the transformations for work in Great Britain
b)	eAreaNI	Use the transformations for work in Northern Ireland.
c)	eAreaRoI	Use the transformations for work in Republic of Ireland.

Return Values: `eErrorCode`

- a) eSuccess
- b) eFailure

All ok
Invalid area entered

2.6 Get the current working area

Function Call:

```
eErrorCode GetArea (eArea* pnArea)
```

Inputs: eArea*

Pointer to the area

Outputs: a) eAreaGreatBritain

Using the transformations for work in Great Britain

b) eAreaNI

Using the transformations for work in Northern Ireland

c) eAreaRoI

Using the transformations for work in Republic of Ireland.

Return Values: `eErrorCode`

- a) eSuccess

All ok

2.7 Initialise the DLL and load the data files

Function Call:

```
eErrorCode Initialise(BSTR bstrUnlockCode)
```

Inputs: BSTR

DLL unlock string ("GIQ.6.0")

Return Values: `eErrorCode`

- a) eSuccess
b) eInvalidUnlockCode
c) eInvalidDataFile
d) eAreaNotSet

All ok

The DLL unlock string supplied is invalid

The data file can not be opened

No valid area has been set for the DLL to work with

2.8 Set an ETRS89 Cartesian coordinate

Function Call:

```
eErrorCode SetETRS89Cartesian(double dX, double dY, double dZ)
```

Inputs:	double dX	X Cartesian coordinate
	double dY	Y Cartesian coordinate
	double dZ	Z Cartesian coordinate

Return Values: eErrorCode

a) eSuccess	All ok
b) eDLLNotInitialised	The DLL has not been initialised
c) eOutsideArea	The supplied coordinate is outside the working area

2.9 Get an ETRS89 Cartesian coordinate

Function Call:

```
eErrorCode GetETRS89Cartesian(double* pdX, double* pdY, double*  
                                pdZ)
```

Outputs:	double* pdX	pointer to the X Cartesian coordinate
	double* pdY	pointer to the Y Cartesian coordinate
	double* pdZ	pointer to the Z Cartesian coordinate

Return Values: eErrorCode

a) eSuccess	All ok
b) eDLLNotInitialised	The DLL has not been initialised

2.10 Set an ETRS89 Geodetic coordinate

Function Call:

```
eErrorCode SetETRS89Geodetic(double dLatDeps, double dLonDeps,  
                             double dEllipHt)
```

Inputs:	double dLatDeps	ETRS89 Latitude in decimal degrees
	double dLonDeps	ETRS89 Longitude in decimal degrees
	double dEllipHt	ETRS89 Ellipsoidal height in metres

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eDLLNotInitialised	The DLL has not been initialised
d) eOutsideArea	The supplied coordinate is outside the working area

2.11 Get an ETRS89 Geodetic coordinate

Function Call:

```
eErrorCode GetETRS89Geodetic(double* pdLatDeps,  
                             double* pdLonDeps,  
                             double* pdEllipHt)
```

Inputs:	double* pdLatDeps	ETRS89 Latitude in decimal degrees
	double* pdLonDeps	ETRS89 Longitude in decimal degrees
	double* pdEllipHt	ETRS89 Ellipsoidal height in metres

Return Values: eErrorCode

a) eSuccess	All ok
b) eDLLNotInitialised	The DLL has not been initialised

2.12 Set an ETRS89 UTM coordinate

Function Call:

```
eErrorCode SetETRS89UTM( double dE, double dN, short nZone,  
                        double dOrthoHt, eVertDatum* pnVertDatum)
```

Inputs:	double dE	UTM Eastings in metres on the ETRS89 datum
	double dN	UTM Northings in metres on the ETRS89 datum
	short nZone	UTM zone
	double dOrthoHt	Orthometric height in metres
	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eDLLNotInitialised	The DLL has not been initialised
d) eOutsideArea	The supplied coordinate is outside the working area

2.13 Get an ETRS89 UTM coordinate

Function Call:

```
eErrorCode GetETRS89UTM(double* pdE, double* pdN, short* pnZone  
                        double* pdOrthoHt, eVertDatum* pnVertDatum)
```

Outputs:	double* pdE	UTM Eastings in metres on the ETRS89 datum
	double* pdN	UTM Northings in metres on the ETRS89 datum
	short* pnZone	UTM zone
	double* pdOrthoHt	Orthometric height in metres
	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eDLLNotInitialised	The DLL has not been initialised
d) eAreaNotSet	No valid area has been set for the DLL to work with

2.14 Set an Irish Grid coordinate

Function Call:

```
eErrorCode SetIrishGrid(double dE, double dN, double dOrthoHt,  
                        eVertDatum nVertDatum)
```

Inputs:	double dE	Irish Grid Eastings in metres
	double dN	Irish Grid Northings in metres
	double dOrthoHt	Orthometric height in metres
	eVertDatum nVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eInvalidVertDatum	An invalid vertical datum has been supplied
d) eDLLNotInitialised	The DLL has not been initialised
e) eOutsideArea	The supplied coordinate is outside the working area

2.15 Get an Irish Grid coordinate

Function Call:

```
eErrorCode GetIrishGrid(double* pdE, double* pdN,  
                        double* pdOrthoHt,  
                        eVertDatum* pnVertDatum)
```

Inputs:	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is to be referred.
---------	-------------------------	---

Outputs:	double* pdE	Irish Grid Eastings in metres
	double* pdN	Irish Grid Northings in metres
	double* pdOrthoHt	Orthometric height in metres
	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eInvalidVertDatum	An invalid vertical datum has been supplied
d) eDLLNotInitialised	The DLL has not been initialised
e) eAreaNotSet	No valid area has been set for the DLL to work with

2.16 Set an Irish Transverse Mercator coordinate

Function Call:

```
eErrorCode SetIrishTransMerc(double dE, double dN,
                             double dOrthoHt,
                             eVertDatum nVertDatum)
```

Inputs:	double dE	Irish Transverse Mercator Eastings in metres
	double dN	Irish Transverse Mercator Northings in metres
	double dOrthoHt	Orthometric height in metres
	eVertDatum nVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eInvalidVertDatum	An invalid vertical datum has been supplied
d) eDLLNotInitialised	The DLL has not been initialised
e) eOutsideArea	The supplied coordinate is outside the working area

2.17 Get an Irish Transverse Mercator coordinate

Function Call:

```
eErrorCode GetIrishTransMerc(double* pdE, double* pdN,  
                             double* pdOrthoHt,  
                             eVertDatum* pnVertDatum)
```

Inputs:	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is to be referred.
Outputs:	double* pdE	Irish Transverse Mercator Eastings in metres
	double* pdN	Irish Transverse Mercator Northings in metres
	double* pdOrthoHt	Orthometric height in metres
	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

a) eSuccess	All ok
b) eInvalidCoordinate	An invalid coordinate has been supplied
c) eInvalidVertDatum	An invalid vertical datum has been supplied
d) eDLLNotInitialised	The DLL has not been initialised
e) eAreaNotSet	No valid area has been set for the DLL to work with

2.18 Set an OSGB36 coordinate

Function Call:

```
eErrorCode SetOSGB36(double dE, double dN, double dOrthoHt,  
                     eVertDatum* pnVertDatum)
```

Inputs:	double dE	OSGB36 Eastings in metres
	double dN	OSGB36 Northings in metres
	double dOrthoHt	Orthometric height in metres
	eVertDatum* pnVertDatum	Vertical datum to which the orthometric height is referred. This variable will be changed to the correct

vertical datum for the coordinate supplied, if the incorrect vertical datum is entered.

Return Values: eErrorCode

- | | |
|-----------------------|---|
| a) eSuccess | All ok |
| b) eInvalidVertDatum | An invalid vertical datum has been supplied |
| c) eDLLNotInitialised | The DLL has not been initialised |
| d) eOutsideArea | The supplied coordinate is outside the working area |

2.19 Get an OSGB36 coordinate

Function Call:

```
eErrorCode GetOSGB36(double* pdE, double* pdN, double* pdOrthoHt,
                    eVertDatum* pnVertDatum)
```

Inputs: eVertDatum* pnVertDatum Vertical datum to which the orthometric height is to be referred.

Outputs: double* pdE OSGB36 Eastings in metres
 double* pdN OSGB36 Northings in metres
 double* pdOrthoHt Orthometric height in metres
 eVertDatum* pnVertDatum Vertical datum to which the orthometric height is referred.

Return Values: eErrorCode

- | | |
|-----------------------|---|
| a) eSuccess | All ok |
| b) eInvalidVertDatum | An invalid vertical datum has been supplied |
| c) eDLLNotInitialised | The DLL has not been initialised |
| d) eAreaNotSet | No valid area has been set for the DLL to work with |

2.20 DLL Constant Definitions

Areas

eArea

Constant	Value
eAreaGreatBritain	0
eAreaNI	1
eAreaRoI	2
eAreaUnknown	-1

Error Codes

eErrorCode

Constant	Value
eSuccess	0
eFailure	-1
eInvalidDataFilePath	-2
eInvalidRegKey	-3
eInvalidUnlockCode	-4
eInvalidDataFile	-5
eInvalidPolygonFile	-6
eAreaNotSet	-7
eDLLNotInitialised	-8
eInvalidCoordinate	-9
eInvalidVertDatum	-10
eOutsideArea	-11

Vertical Datums

eVertDatum

Constant	Value
eUnknownVertDatum	0
eNewlynDatum	1
eScillyIslesDatum	2
eIsleManDatum	3
eOuterHebridesDatum	4
eStKildaDatum	5
eShetlandDatum	6
eOrkneyDatum	7
eFairIsleDatum	8
eFlannansDatum	9
eNorthRonaDatum	10
eSuleSkerryDatum	11
eFoulaDatum	12
eMalinHeadDatum	13
eBelfastDatum	14