ISCLOC: The Location Algorithm Used by the ISC

István Bondár
International Seismological Centre
Pipers Lane, Thatcham, RG19 4NS, Berkshire,
United Kingdom
www.isc.ac.uk

Version 2.2.6 May, 2015

Bondár, I., and D. Storchak, 2011, Improved location procedures at the International Seismological Centre, *Geophys. J. Int.*, doi: 10.1111/j.1365-246X.2011.05107.x.

Release Notes

Dependencies

The LAPACK and LBLAS libraries are required. They are part of the MacOS X 10.6 and later Accelerate framework. On linux/unix machines install the corresponding rpm or download the libraries from www.netlib.org.

The PostgreSQL client is required if the database interface is to be used. On MacOS X 10.6 – 10.8 it was part of the OS. Unfortunately, in 10.9.0 (Mavericks) Apple decided to remove the client. However, the postgresql 9.2 libraries are still there. The necessary include files (libpq-fe.h and postgres_ext.h) are provided in the postgresql_include directory. On linux/unix machines download the postgresql development (to get the include files) package.

The RSTT package is required for the RSTT-enabled ISC locator. Download it from www.sandia.gov/rstt. On MacOS X 10.6 and later no compilation is required, on linux/unix machines compile the package by issuing

```
make clean_objs
make geotess
make cc
make c
```

The ISC locator uses Apple's Grand Central Dispatch for optional multicore processing; obviously it is part of the MacOS X. On linux/unix libdispatch and its dependencies have to be manually installed (see http://mark.heily.com/project/libblocksruntime for further information).

RSTT support

The ISC locator can be enabled to take travel-time predictions for crustal and mantle phases from a 3D velocity model compliant with the RSTT parameterization (Myers et al., 2010). To do this, compile the code with the Makefile in the rstt_src directory. See also the README_RSTT and INSTALL files.

Changes in 2.2.6

Added support for ISF2.1 I/O files. When ISF output is required, it is produced in ISF2. Added a new instruction, *fix_depth_median* that allows to fix the depth to the median of the reported depths. Very minor bug fixes in main.c, rstt_main.c, magnitude.c, locator_parallel, phaseids_parallel.c and travel_times_parallel.c.

Changes in 2.2.5

Very minor bug fixes in phaseids_parallel.c and depth_phases.c.

Changes in 2.2.4

Minor bug fixes in isf.c and read_textfiles.c. Extra spaces are removed from the instruction line before interpreting it. Removed dependency on time zone in write_db.c. Added support for the Murphy-Barker (2003) mb attenuation curves. Introduced a new RSTT-specific configuration parameter, *rstt_model*, to specify the pathname of the RSTT model. The *rstt_model* and *use_RSTT_PgLg* parameters can be specified in both the *config.txt* file or in the instruction line. The RSTT-specific parameters only affect the behaviour of the RSTT-enabled version of the ISC locator.

Changes in 2.2.3 and before

The performance of the *iscloc* is boosted by using LAPACK routines to calculate the eigenvalue decomposition of the symmetric, positive semi-definite data covariance matrix. For large events with thousands of phases this represents a significant increase in speed. It is assumed that the LAPACK libraries are installed in unix/linux systems; for Mac OS users LAPACK is already included in the Accelerate framework.

We also added support for multicore processing via Apple's Grand Central Dispatch. Several aspects of the locator beg for parallel processing, such as phase identification, calculation of time residuals, the block-by-block inversion of the data covariance matrix, and most trivially the evaluation of the trial hypocentres in the NA search. Grand Central Dispatch allowed us to exploit multicore CPUs without going through a major redesign of the location software. The multicore version of the locator runs three times faster on a MacBookPro i7 quadcore than just using a single core. Grand Central Dispatch obviously works on Mac OS 10.6 and later; for implementing GCD on linux boxes, please consult the README and INSTALL files, as well as the Makefile.

We have slightly changed the logic of how the Neighbourhood Algorithm (NA) is applied. If enabled (*do_gridsearch*=1), the NA search is performed in four dimensions around the initial hypocentre. The phases are reidentified with respect to each trial hypocentre, and the misfit of the solution is calculated. The misfit (the Lp-norm of time-defining residuals) is penalized against using too few phases. This allows for a more thorough exploration of the search space. In order to speed up the process, accounting for correlated error structure is disabled for events with a reasonable network geometry and with more than 30 seismic (P or S-type) phases when calculating the misfit of a trial hypocentre. Finally, we identify the phases according to the best solution from the NA search, and then test for depth resolution. If there is depth resolution we continue with a free-depth solution; otherwise we fix the depth to a region-dependent default depth. If the discrepancy between the default depth and the depth from the NA search is larger than 20 km, we repeat the NA search, with the depth fixed.

We introduced a new configuration parameter, *use_RSTT_PgLg* for the RSTT-enabled version of the locator. If set (*use_RSTT_PgLg* =1) RSTT Pg/Lg predictions are used for first-arriving crustal P (Pg or Pb) and S (Sg or Sb) phases. By default this option is turned off.

This release also includes minor bug fixes, as well as improvements in the phase identification procedures, in the handling of readings (phase and amplitude data reported for an event from a single station by a single agency), and duplicates (phases reported from the same station by several agencies), as well as fine tuning of the parameters in the *config.txt* and *ak135_model.txt* files.

Note that MS values are not calculated when the input is read from ISF files. It is due to the shortcomings of the ISF/IMS format that does not provide information on which channel the surface wave amplitude was measured. Channel information is needed for MS computation.

1 Getting Started

1.1 Usage

iscloc [isf|isf2|yyyymmv|-]

- If the argument is 'isf' then *iscloc* will expect input from an ISF text file. Instructions will be taken from stdin and the log will be written to the file specified in the *config.txt* file. The *isf_infile* parameter is expected among the instructions, as well as the *isf_stafile* parameter, either specified in the *config.txt* file or given as an instruction. Only the first instruction line is read, and all events in the input ISF text file are processed according to these instructions.
- If the argument is 'isf2' then *iscloc* will expect input from an ISF2 text file. Instructions will be taken from stdin and the log will be written to the file specified in the *config.txt* file. The *isf_infile* parameter is expected among the instructions, as well as the *isf_stafile* parameter, either specified in the *config.txt* file or given as an instruction. Only the first instruction line is read, and all events in the input ISF text file are processed according to these instructions.
- If the argument is '-' then instructions will be taken from *stdin* and the log will be written to the file specified in the *config.txt* file.
- Otherwise the argument will be used to create the filenames for the instruction file and the log file. This reflects the ISC-specific operational batch mode.

Examples:

```
echo "update_db=0 isf_infile=isf.dat isf_outfile=isf.out" | iscloc isf
iscloc - < instruction_file > logfile
echo "11526867" | iscloc -
echo "11526867 indb=edit do_gridsearch=0 depth=NEIC" | iscloc -
echo "11526867 isf_outfile=11526867.isf.out depth=NEIC" | iscloc -
iscloc 200905b
```

1.2 Environment variables

QETC - Directory pathname for iscloc parameter data files. If it does not exist, it defaults to current working directory.

QEDIT - Directory pathname for the instruction file (ISC specific). If it does not exist, it reads instructions from *stdin*.

QREVISE - Directory pathname for the log file (ISC specific).

PGHOSTADDR, PGPORT, PGDATABASE, PGUSER, PGPASSWORD – PostgreSQL specific environment variables.

SLBMROOT – Directory pathname for the RSTT root directory. Required if compiled with the Makefile in rstt_src.

1.3 Instructions

Instructions can either be given on the command line or read from an instruction file. An instruction line is expected for every event in the form:

isc_evid [par=value [par=value [par [par=value...]]]

where *isc_evid* is the event identifier (required), and the par=value pairs denote the optional instruction name and value pairs.

Note:

For ISF input only the first line of the instruction file is interpreted, and the instructions are applied to *all* events in the input ISF file. For ISF input the *isc_evid* parameter is not required.

Tables 1 and 2 provide the list of instructions accepted by *iscloc*.

Table 1. Instructions

Name	Type	Description
indb	string	database account to read from.
outdb	string	database account to write to.
verbose	integer	Level of verbosity (0..5).
isf_infile	string	Pathname to the ISF input text file.
isf_outfile	string	Pathname to the ISF output text file.
fix_hypo	string	An agency code to fix the hypocentre to. If no code is given, the hypocentre will be fixed to the median reported hypocentre (lat, lon, depth, origin time) parameters.
fix_time	YYYY-MM-DD_HH:MI:SS.SSS or string	A date-time value or an agency code to fix the origin time to. If no value or code is given, the origin time will be fixed to the median reported origin time.
fix_location	string	An agency code to fix the epicentre to. If no code is given, the epicentre will be fixed to the median reported latitude, longitude values.
fix_depth	double or string	A depth value or an agency code to fix the depth to. If no value or code is given, the depth will be fixed to the median reported depth.
fix_depth_default	integer	Fix the depth to the region-dependent default depth (from default depth grid or from Flinn-Engdahl region number).
fix_depth_median	integer	Fix the depth to the median of the reported depths.
lat	double or string	A latitude value or an agency code to start the locator from. If no value or code is given, the initial latitude will be set to the median reported latitude.
lon	double or string	A longitude value or an agency code to start the locator from. If no value or code is given, the initial longitude will be set to the median reported longitude.
depth	double or string	A depth value or an agency code to start the locator from. If no value or code is given, the initial depth will be set to the median reported depth.
time	YYYY-MM-DD_HH:MI:SS.SSS or string	A date-time value or an agency code to start the locator from. If no value or code is given, the initial origin time will be set to the median reported origin time.
write_gridsearch_results	integer	Write Neighbourhood Algorithm results to file [0/1].

The instructions below can temporarily (for one event only) override the parameter values set in the *config.txt* file.

Table 2. Instructions that can temporarily override configuration parameters

Name	Type	Description
update_db	integer	Write results to database [0/1].
isf_stafile	string	Pathname to the comma-separated station file.
ndepagency	integer	Minimum number of agencies reporting depth phases or PcP/ScS for depth resolution.
mindepthpha	integer	Minimum number of time-defining first arriving P and depth phase pairs for depth-phase depth resolution.
mincorepha	integer	Minimum number of time-defining first arriving P and core phase (PcP, ScS) pairs for depth resolution.
localdist	double	Maximum epicentral distance for local stations (degrees).
minlocalsta	integer	Minimum number of time-defining first arriving P within localdist for depth resolution.
spdist	double	Maximum epicentral distance for near-regional stations (degrees).
min_s_p	integer	Minimum number of time-defining first arriving P and S pairs for depth resolution.
do_correlated_errors	integer	Account for correlated travel-time prediction errors [0/1].
do_gridsearch	integer	Perform Neighbourhood Algorithm search [0/1].
iseed	integer	Random number seed.
na_radius	double	Search radius around starting epicentre (degrees).
na_deptol	double	Search radius around starting depth (km).
na_ottol	double	Search radius around starting origin time (s).
na_nsamplei	integer	Number of initial samples in NA search.
na_nsample	integer	Number of subsequent samples in NA search.
na_ncells	integer	Number of cells to be resampled in NA search.
na_itermax	integer	Maximum number of iterations in Neighbourhood Algorithm search.
pertol	double	Period tolerance around MS_Z when searching for horizontal MS amplitudes (s)
use_RSTT_PgLg	integer	Use RSTT crustal phase predictions [0/1]. Valid only for the RSTT-enabled version of the ISC locator.
rstt_model	string	Full pathname for the RSTT model.

1.4 Input/output

The locator can read bulletin data either from an SQL database, or from ISF text files. With the 'isf' and 'isf2' command line argument, *iscloc* will read the input bulletin from the *isf_infile* file (in this case the *isf_stafile* parameter must also be specified). With the command line arguments '-' or 'yyyymmv' *iscloc* will read the data from the database. The results can be written to an ISF output file (described by the *isf_outfile* instruction), to the database, or both. The log and error files are by default written to the *stdout* and *stderr*, respectively. We recommend that the users download the file of station parameters (isc_stalist and IR2_stalist) from the ISC website (www.isc.ac.uk), where it is regularly updated. These files are required for ISF and ISF2 input, respectively.

For the ISC database schema description see **Appendix A** or www.isc.ac.uk. The locator also reads the data files below (see also **Appendix B**).

Table 3. Data files

Pathname	Description
$QETC/iscloc/config.txt	Configuration parameters.
$QETC/iscloc/<*ttime_table*>_model.txt	Velocity model and phase specific parameters. Currently the ak135 and jb travel-time tables are supported.
$QETC/<*ttime_table*>/*.tab	Travel-time tables. The ak135 travel-time tables were generated by *libtau*; the Jeffreys-Bullen travel-time tables were generated using the old ISC locator code.
$QETC/ak135/ELCOR.dat	Ellipticity correction coefficients for ak135.
$QETC/topo/etopo5_bed_g_i2.bin	Topography/bathymetry data to calculate bounce point corrections. The ETOPO1 bedrock file was resampled to 5'x5' resolution (binary file of etoponlat * etoponlon 2-byte integers).
$QETC/FlinnEngdahl/FE.dat	Flinn-Engdahl regionalization.
$QETC/FlinnEngdahl/default.depth0.5.grid	Default depth grid defined on a 0.5 x 0.5 degree grid.
$QETC/FlinnEngdahl/grn_default_depth.<*ttime_table*>.dat	Default depths for Flinn-Engdahl regions for locations where no default depth grid point exists.
$QETC/variogram/variogram.model	Generic variogram model to calculate *a priori* data covariance matrix.
$QETC/magnitude/<*mbQ_table*>mbQ.dat	Magnitude attenuation Q(d,h) curves. Currently the GR (Gutenberg-Richter) , VC (Veith-Clawson) and MB (Murphy-Barker) curves are supported.
$QETC/Stations/isc_stalist	Station parameters. Required for ISF input.
$QETC/Stations/IR2_stalist	Station parameters. Required for ISF2 input.

1.5 Database interface

By default, *iscloc* reads event data from and writes the results to an SQL database, described by the ISC database schema (see **Appendix A** and www.isc.ac.uk). The database platform implemented at the ISC is PostgreSQL 8.4.

In PostgreSQL the database account is identified by the environment variables $PGUSER and $PGPASSWORD, the database instance by $PGDATABASE and the database server host by $PGHOSTADDR and $PGPORT. *iscloc* connects to the database using these environment variables.

By default, it reads and writes the account identified by $PGUSER. This can be overridden by the optional instruction line parameters *indb* (the database account to read from) and *outdb* (the database account to write to) parameters. The user must have read permission to *indb* and read/write permission to *outdb*. New unique identifiers (*hypid*, *magid*, *stamagid*, *rdmagid*, *mszhid* and *ampmagid*) are obtained from a central database account, *nextid_db*, if necessary. Writing the results to the database can be disabled by setting the *update_db* parameter to 0 in the *config.txt* file, or temporarily by setting it in an event instruction line.

Note that if the database platform is anything other than PostgreSQL, the source code (*iscloc.h, main.c, pgsql_funcs.c, read_db.c* and *write_db.c*) should be modified accordingly and the corresponding SQL library interface should be linked to the code.

1.6 Configuration parameters

General *iscloc* configuration parameters are read from the *config.txt* file.

Table 4. Configuration parameters

Name	Type	Description	Default
logfile	string	Pathname for the iscloc logfile.	stdout
errfile	string	Pathname for the iscloc error file.	stderr
isf_stafile	string	Pathname for the station file.	isc_stalist
update_db	integer	Write results to DB [0/1].	1
nextid_db	string	Get new unique ids from this account.	isc
repid	integer	Reporter id for new hypocentres and associations. ISC-specific.	100
out_agency	string	Author for new hypocentres and associations.	ISC
in_agency	string	Author for input associations.	ISC
ttime_table	string	TT table name [ak135|jb].	ak135
default_depth	double	Used if seed hypocentre depth is null.	0.
maxdeperror_shallow	double	Maximum depth error for crustal events to accept a free-depth solution.	30.
maxdeperror_deep	double	Maximum depth error for deep events to accept a free-depth solution.	60.
etopofile	string	Filename for ETOPO file.	etopo5_bed_g_i2.bin
etoponlon	integer	Number of longitude samples in ETOPO.	4321
etoponlat	integer	Number of latitude samples in ETOPO.	2161
etopores	double	Cellsize in ETOPO.	0.0833333
ndepagency	integer	Minimum number of agencies reporting depth phases or PcP/ScS for depth resolution.	2
mindepthpha	integer	Minimum number of time-defining first arriving P and depth phase pairs for depth-phase depth resolution.	5
mincorepha	integer	Minimum number of time-defining first arriving P and core phase (PcP, ScS) pairs for depth resolution.	5
localdist	double	Maximum epicentral distance for local stations (degrees).	0.2
minlocalsta	integer	Minimum number of time-defining first arriving P within localdist for depth resolution.	1
spdist	double	Maximum epicentral distance for near-regional stations (degrees).	5
min_s_p	integer	Minimum number of time-defining first arriving P and S pairs for depth resolution.	5
min_iter	integer	Minimum number of iterations.	4
max_iter	integer	Maximum number of iterations.	20
min_phases	integer	Minimum number of defining phases.	4
allow_damping	integer	Allow damping of solution [0/1].	1
sigmathres	double	Residuals beyond *sigmathres * prior measurement error* will be made defining.	4.
confidence	double	Confidence level for formal uncertainties [90|95|98].	90.
do_correlated_errors	integer	Account for correlated travel-time prediction	1

		errors [0/1].	
do_gridsearch	integer	Perform Neighbourhood Algorithm (NA) search [0/1].	1
iseed	integer	Random number seed.	5590
na_radius	double	Search radius around starting epicentre (degrees).	5.
na_deptol	double	Search radius around starting depth (km).	300.
na_ottol	double	Search radius around starting origin time (s).	30.
na_itermax	integer	Maximum number of iterations in NA search.	5
na_lpnorm	double	p-value for norm to compute misfit in NA search [1 ≤ p ≤ 2].	1
na_nsamplei	integer	Number of initial samples in NA search.	700
na_nsample	integer	Number of subsequent samples in NA search.	100
na_ncells	integer	Number of cells to be resampled in NA search.	25
no_hypo_agencies	string	Comma separated list of agencies not to be used in setting the initial hypocentre.	UNK,NIED,HFS,HFS1, HFS2,NAO
body_mag_min_dist	double	Minimum mb distance (degrees).	21.
body_mag_max_dist	double	Maximum mb distance (degrees).	100.
surf_mag_min_dist	double	Minimum MS distance (degrees).	20.
body_mag_max_dist	double	Maximum MS distance (degrees).	160.
body_mag_min_per	double	Minimum mb period (s).	0.
body_mag_max_per	double	Maximum mb period (s).	3.
surf_mag_min_per	double	Minimum MS period (s).	10.
surf_mag_max_per	double	Maximum MS period (s).	60.
surf_mag_max_depth	double	Maximum MS depth (km).	60.
pertol	double	Period tolerance around MS_Z when searching for horizontal MS amplitudes (s)	5.
mag_range_warn_thres	double	Generate warning if station magnitude range exceeds this.	2.2
mbQ_table	string	Magnitude attenuation correction table for mb [GR|VC|MB|none].	GR
use_RSTT_PgLg	integer	Use RSTT crustal phase predictions [0/1]. Valid only for the RSTT-enabled version of the ISC locator.	0
rstt_model	string	Full pathname for the RSTT model.	

Configuration parameters specific to phase names and/or velocity model are read from the <*ttime_table*>_model.txt file.

Table 5. Configuration parameters specific to phases and/or velocity models

Name	Type	Description
moho	double	Depth of Moho discontinuity (km).
conrad	double	Depth of Conrad discontinuity (km).
moho_radius	double	Distance of Moho discontinuity from centre of the Earth (km).
max_depth_km	double	Maximum allowable event depth (km).
ssurfvel	double	Sg velocity for elevation corrections (km/s).
psurfvel	double	Pg velocity for elevation corrections (km/s).
phase_map	table	Table of reported and corresponding ISC phase names to map reported phases to IASPEI standard phase names.

allowable_phases	table	List of phase names to which reported phases can be renamed.
allowable_first_P	table	List of phase names to which reported first-arriving P phases can be renamed.
optional_first_P	table	Additional list of phase names to which reported first-arriving P phases can be renamed.
allowable_first_S	table	List of phase names to which reported first-arriving S phases can be renamed.
optional_first_S	table	Additional list of phase names to which reported first-arriving S phases can be renamed.
no_resid_phase	table	List of phase names for which residuals will not be calculated (and not used in the location, e.g. amplitude phases).
phase_weight	table	List of *a priori* measurement errors within specified delta ranges for IASPEI phases. (phase, mindelta, maxdelta, measurement error values)
mb_phase	table	List of phase names that contribute to mb calculation.
ms_phase	table	List of phase names that contribute to MS calculation.

2 Event location

2.1 Initializations

The initialization step sets the starting hypocentre according to the instructions given for the event. Since the reported hypocentres may exhibit a large scatter, and it is not guaranteed that any of them is close to the global optimum, the initial hypocentre parameters (latitude, longitude, origin time, depth) are set to the median of the corresponding reported hypocentre parameters. However, if the author of the prime (preferred) hypocentre is ISC, EHB, or IASPEI, the initial hypocentre will be set to the prime hypocentre. Furthermore, if instructions (**Table 1**) are given to fix or provide starting values for any of the hypocentre parameters, they will be set according to the instructions.

The event type is read from the **event** table (or taken from the prime hypocentre if the input is ISF). If the event type indicates an anthropogenic event, the depth will be fixed to the surface, unless an instruction is given to fix the depth to some other specific value.

2.2 Phase identification

Reported phases are mapped to IASPEI standard phase names (Storchak et al., 2003) using the map provided in *phase_map* (**Table 5**). The *phase_map* table lists all possible variations of reported phase names and the corresponding IASPEI phase name. Note that if the *phase_fixed* flag is set in the **association** table, the phase will not be renamed at all. Unrecognised phase names will be mapped to a null value.

Once the reported phase names are mapped to the IASPEI standards, they are reidentified with respect to the initial hypocentre. If the Neighbourhood Algorithm (NA) search is turned on, phases are identified with respect to each trial hypocentre. Once NA is finished, the phases will be identified with respect to the refined initial hypocentre guess obtained from the NA search. In order to avoid oscillating solutions, phase names are kept the same during the iterations of the linearised least-squares algorithm unless the depth crosses the Moho or Conrad discontinuities between two iterations. In that case local and regional phases are reidentified.

Phases in a reading (phases reported by a single agency for an event at a single station) are treated as a group in *iscloc*. In each reading the phases are identified with respect to the initial hypocentre.

Phases that are listed in *no_resid_phase* (**Table 5**) are skipped, thus preventing them to be used in the location.

For a phase in a reading the phase type is determined by the first leg of the ray path indicated by the first letter in the phase name; for depth phases the phase type is determined by the second letter in the phase name (e.g. Pn, sP and pwP are P-type phases, Sn, ScP and pS are S-type phases). Currently only P-type and S-type phases are considered. In other words, Lg, Rg, hydroacoustic or infrasound phases are not used in the location, even if they had *ak135* travel-time predictions.

If the *phase_fixed* flag is set in the **association** table for a phase, the time residual is calculated using the fixed phase name and no further attempt is made to rename the phase. Otherwise *iscloc* checks if the phase is in the list of *allowable_phases* (**Table 5**). The list of allowable phases was introduced to prevent the locator renaming phases to unlikely 'exotic' phases, just because a travel-time prediction fits the observed travel-time better. For instance, we do not want to reidentify a reported Sn as SgSg, SPn or *horribile dictu*, sSn. Recall that phases may suffer from picking errors or the observed travel-times may reflect true 3D structures not modelled by the velocity model. Introducing the list of allowable phases helps to maintain the sanity of the bulletin and mitigates the risk of misidentifying phases. However, if a reported phase is not in the list of allowable phases, it is temporarily added to the list accepting the fact that station operators may confidently pick later phases. In other words, exotic phase names can appear in the final bulletin only if they were reported as such.

For each phase in a reading *iscloc* loops through the (possibly amended) list of allowable phases and calculates the time residual with respect to the initial hypocentre. Certain rules apply:

- Cannot rename a P-type phase to S-type, and vice versa.
- A phase name can appear only once in a reading.

Further restrictions apply to first-arriving P and S phases. First-arriving P and S phases can be identified as those in the list of allowable first-arriving P and S phases (**Table 5**). Occasionally a station operator may not report the true first-arriving phase due to high noise conditions. To account for this situation the list of optional first-arriving P and S phases (**Table 5**) is also checked.

Finally, having observed the rules above, the phase is identified as the phase in the allowable phase list that has the smallest residual. If no eligible phase has been found, i.e. if the smallest residual is beyond ±60 seconds, the phase is treated as unidentified.

Once the phases are identified with respect to the initial hypocentre, *iscloc* sets the time defining flag and the *a priori* estimate of the measurement error for each phase. The prior measurement errors are given in *phase_weight* (**Table 5**). The time defining flag is set to true if a valid entry is found in the *phase_weight* table. However, the phase is made non-defining if its residual is larger than a *sigmathres* times the prior measurement error, or it was explicitly set to non-defining by an analyst (i.e. the *non_def* flag is set in the **association** table). Note that only time-defining phases are used in the location.

Since phases can be picked and reported by several agencies for the same station, there can be multiple entries for the same phase arrival. At this stage *iscloc* considers time-defining phases only. Arrival picks are considered duplicates if they are reported at the same site for the same event and if they arrival times is within 0.1 seconds. To account for alternative station codes, the primary stations codes are used. For duplicates the arrival time is taken as the mean of the arrival times, and the phase name is forced to be the one with the smallest residual. If accounting for correlated errors is turned off, duplicates are explicitly down-weighted. However, if correlated errors are accounted for, down-weighting is not necessary as duplicates are simply projected to the null space.

2.3 Travel-time prediction

Currently *iscloc* supports the JB (Jeffreys and Bullen, 1940) and the *ak135* (Kennett et al., 1995) travel-time tables. One of the major advantages of using the *ak135* travel-time predictions is that they do not suffer from the baseline difference between P, S and PKP phases compared to the Jeffreys-Bullen tables. Furthermore, *ak135* offers an abundance of phases from the IASPEI standard phase list (Storchak et al., 2003) that can be used in the location, most notably the PKP branches and depth-sensitive phases.

The *ak135* travel-time tables were generated by *libtau* (Buland and Chapman, 1983); the Jeffreys-Bullen travel-time tables were generated using the old ISC locator code. Composite tables for first-arriving P and S phases were also constructed and they are used to get a valid travel-time table value at local/regional crossover distances without reidentifying the phase during the subsequent iterations of the location algorithm.

The travel-time table values for a given phase, delta and depth are calculated by bicubic spline interpolation. Vertical slowness table values are calculated only if vertical partial derivatives are requested. For depth phases bounce point distances are also calculated.

The P and S velocities in the surface layer to calculate elevation corrections are given by *psurfvel* and *ssurfvel* (**Table 5**). Ellipticity corrections (Dziewonski and Gilbert, 1976; Kennett and Gudmundsson, 1996), using the WG84 ellipsoid parameters, are added to the *ak135* predictions. For JB an approximate geoid correction is calculated to account for the ellipticity effects.

Bounce point (elevation correction at the surface reflection point) corrections are calculated by the EHB algorithm (Engdahl et al., 1998) for depth phases. Water depth corrections for pwP are calculated for water columns exceeding 1.5 km. We use the ETOPO1 global relief model (Amante and Eakins, 2009), resampled to 5’ x 5’ resolution, to obtain the elevation or the water depth at the bounce point. The ETOPO parameters are specified in the *config.txt* file (**Table 4**).

2.4 Data covariance matrix

If the observations are assumed to be independent, the data covariance matrix is a diagonal matrix where the diagonal elements are the *a priori* estimates of the measurement error variances. The phase and distance dependent measurement errors are specified in *phase_weight* (**Table 5**). The inverse of the data covariance matrix provides the diagonal weight matrix.

When correlated travel-time prediction errors are present, the data covariance matrix is no longer diagonal, and the redundancy in the observations reduces the effective number of degrees of freedom (Bondár and McLaughlin, 2009). Assuming that the similarity between ray paths is well approximated by the station separation, the covariances between station pairs can be estimated from a generic P variogram model derived from ground truth residuals. The estimates for the elements of the data covariance matrix are obtained as

$$C_D(i,j) = \sigma_{sill}^2 - \gamma(h_{ij}) + \delta_{ij}\sigma_{phase}^2$$

where σ_{sill}^2 denotes the background variance where the variogram levels off (i.e. where the pairs become independent), $\gamma(h_{ij})$ is the variogram value for the distance, h_{ij}, between the i^{th} and j^{th} stations, δ_{ij} is the Kronecker delta and σ_{phase}^2 is the *a priori* estimate of the measurement error covariance for an observed phase. The last term indicates that the measurement error variances add to the diagonal of the covariance matrix.

Because in this representation the covariances depend only on station separations, the covariance matrix (and its inverse) needs to be calculated only once. We assume that different phases owing to the different ray paths they travel along, as well as station pairs with a separation larger than 1,000 km are uncorrelated. Hence, the data covariance matrix is a sparse, block-diagonal matrix.

Furthermore, if the stations in each phase block are ordered by their nearest neighbour distance, the phase blocks themselves become block-diagonal. We determine the nearest neighbour ordering of the stations by performing a single-linkage hierarchical cluster analysis (de Hoon et al., 2004; Sibson, 1973) using the distance matrix constructed from the station separations.

To reduce the computational time of inverting large matrices we exploit the inherent block-diagonal structure by inverting the covariance matrix block-by-block. The singular value decomposition of the data covariance matrix is written as

$$\mathbf{C}_D = \mathbf{U}_D \mathbf{\Lambda}_D \mathbf{V}_D^T$$

where $\mathbf{\Lambda}_D$ is the diagonal matrix of eigenvalues and the columns of $\mathbf{U}_D$ contain the eigenvectors of $\mathbf{C}_D$. Let $\mathbf{C}_D = \mathbf{B}\mathbf{B}^T$, with $\mathbf{B} = \mathbf{U}_D \mathbf{\Lambda}_D^{1/2}$, then the projection matrix

$$\mathbf{W} = \mathbf{B}^{-1} = \mathbf{\Lambda}_D^{-1/2} \mathbf{U}_D^T$$

will orthogonalise the data set and project redundant observations into the null space.

2.5 Neighbourhood Algorithm (NA) search

Linearised inversion algorithms are quite sensitive to the initial guess. However, it is not guaranteed that any of the reported hypocentres would provide a suitable starting point for the linearised inversion. In order to find an initial hypocentre guess for the linearised inversion we run the Neigbourhood Algorithm (Sambridge, 1999; Sambridge and Kennett, 2001; Kennett, 2006) around the starting hypocentre.

The parameters governing the NA search are read from the *config.txt* file (**Table 4**) or can be given as instructions (**Table 2**). As with the linearised location algorithm, the forward calculations in NA use all P and S-type phases and can account for correlated travel-time prediction errors. Note that for events that have a reasonable network geometry (networks that are not heavily unbalanced) we temporarily disable the accounting for correlated errors in order to improve performance. During the NA search, we identify the phases with respect to each trial hypocentre and calculate the misfit of the trial hypocentre. The misfit is defined as

$$misfit = \frac{\sum_{i=0}^{N} |tres_i|^p}{Nrank - M} + \alpha \frac{Nass - Ndef}{Nass}$$

where *Nass* is the total number of P and S-type phases, *Ndef* is the number of defining phases, *Nrank* is the number of independent defining phases, *M* is the number of degrees of freedom (the number of model parameters), α (=4.0) is a penalty factor, and *p* is defined by the parameter *na_lpnorm*. Recall that the phase identification process makes a phase non-defining if its residual exceeds the threshold defined by *sigmathres* times the prior measurement error. Thus, each trial hypocentre can have different number of defining phases. The second term in the misfit expression is introduced to penalize against freakish local minima provided by just a few phases.

The NA parameters, *na_nsamplei*, *na_nsample* and *na_ncells*, are tuned to achieve a reasonable compromise between speed and the exhaustive exploration of the search space. An initial sample size (*na_nsamplei*) of 3000 or larger provides a very thorough initial sampling of the search space, but for large events with thousands of phases the NA search would be very slow. An initial sample size somewhere between 500 and 1000 typically provides good results. The parameter *na_ncells* determines the number of the cells with the best misfits whose neighbourhood is explored in subsequent NA iterations; *na_nsample* is the number of samples generated in subsequent iterations. Thus, for *na_ncells*=25 and *na_nsample*=100, each candidate cell is resampled by four new samples. Keeping the number of cells to be resampled relatively high prevents NA prematurely falling into a local minima.

Local and near-regional events with lots of stations around the Pg/Pb/Pn crossover distance range could generate misfit surfaces with quite a number of local minima. In such cases, it is advisable to run the locator with several parameter settings for NA. On the other hand, for events where the user may have a good initial guess for the location (especially for anthropogenic events), the NA search could be safely disabled.

If the instruction *write_gridsearch_results* is set to 1, the trial hypocentres together with their misfit and the number of defining phases, as well as the best models after every iteration are written to a file.

2.6 Depth determination

2.6.1 Depth resolution

Solving for depth is attempted by *iscloc* only if the data provide sufficient depth resolution. Depth resolution can be provided by a local network, depth phases, core reflections and to a lesser extent near-regional secondary phases. We have developed a number of criteria to test whether the reported data for an event have sufficient depth resolution.

1. Local network: at least *minlocalsta* stations with time-defining phases within *localdist* epicentral distance.
2. Depth phases: at least *mindepthpha* time-defining first-arriving P and depth phase pairs reported by at least *ndepagency* agencies (to reduce the chance of misinterpretation by a single inexperienced analyst).
3. Core reflections: at least *mincorepha* time-defining first-arriving P and core reflection pairs (PcP, ScS) reported by at least *ndepagency* agencies.
4. Local/near regional S: at least *min_s_p* time-defining P and S pairs within *spdist* epicentral distance.

The parameters governing the depth resolution criteria are read from the *config.txt* file (**Table 4**) or can be given as instructions (**Table 2**). We attempt a free-depth solution if any of the above criteria are satisfied; otherwise we fix the depth to a default depth, dependent on the epicentre location.

2.6.2 Default depth for fixed-depth events

If there is insufficient depth resolution provided by the data, or the depth uncertainty for a free-depth solution exceeds a threshold (**Table 4**), the hypocentre depth is set to the depth from the default depth grid if a grid point for the epicentre location exists; otherwise it is set to the median of reported depths if the initial depth is larger than 100 km; else it is set to a depth (Bolton et al., 2006) assigned to the corresponding Flinn-Engdahl (Young et al., 1996) geographic region.

The default depth grid (Bondár and Storchak, 2011) is defined on a 0.5º x 0.5º grid and was derived from the free-depth solutions obtained from the relocation of the entire ISC bulletin using the new locator as well as the EHB (Engdahl et al., 1998) free-depth solutions, including the fixed-depth EHB earthquakes that were flagged as having reliable depth estimate (Bob Engdahl, personal communication, 2010). In total, 815,000 events with reliable depths were used to produce the default depth grid.

The default depth in a grid cell is defined as the median of all depths in the cell, provided that there were at least five events in the cell, and the 75–25 percent quartile range was less than 100 km. The latter constraint is imposed to avoid regions with both shallow and deep seismicity.

The default depth grid follows gridline registration, i.e. the nodes are centred on the grid line intersections and the data points represent the median value in a cell of dimensions (*gres* x *gres*) centred on the nodes. Therefore, a point (*lat,lon*) falls in a grid cell if the expression below is true.

$$(|lat - grid\_lat_i| \le gres/2 \text{ and } |lon - grid\_lon_j| \le gres/2)$$

2.6.3 Depth-phase depth

If the reported depth phases provide sufficient depth resolution, iscloc determines the hypocentre depth using the depth-phase stacking method (Murphy and Barker, 2006). The depth-phase stack provides a depth estimate independent from the one obtained by the location algorithm.

The depth-phase stack is always performed if there is depth resolution provided by the depth phases and the linearised least-squares location algorithm has reached a convergent solution, even if the depth was fixed either by instructions or by the location algorithm.

The depth-phase stacking method can be described in three steps.

1. The predicted moveout curves (depth phase – first arriving P arrival time) are generated for each station as a function of depth (they also depend on the epicentral distance).
2. For each observed moveout a depth trace is generated by putting a boxcar at the corresponding depth. The width of the boxcar is defined by the *a priori* measurement error for the depth phase; it is centred on the observed moveout and then projected to the *x*-axis (depth).
3. The depth traces are stacked and the depth-phase depth and its uncertainty are defined as the median and the standard median absolute deviation (SMAD) of the stack, respectively.

2.6.4 Reporting how the hypocentre depth was obtained

The *depfix* field in the **association** table describes how the depth was obtained for an event. An explanation is also given in the *iscloc* log file as well as in the ISC bulletin when searched on the ISC website, www.isc.ac.uk.

Table 6. Depth determination

depfix	Description
null	Free-depth solution.
A	Depth fixed by analyst.
B	Beyond depth limits; depth fixed to 0/600 km.
D	Depth fixed to depth-phase depth.
G	Depth fixed to default depth grid value.
H	Depth fixed to depth of a reported hypocentre.
M	Depth fixed to median of reported hypocentre depths.
R	Depth fixed to default Flinn-Engdahl region depth value.
S	Anthropogenic event; depth fixed to the surface.

2.7 Linearised least-squares location algorithm

2.7.1 Iterative inversion scheme

An iterative linearised least-squares inversion of travel-times (Bondár and McLaughlin, 2009) is performed by *iscloc* (Bondár and Storchak, 2011) to obtain a solution for the hypocentre. The algorithm solves the matrix equation

$$\mathbf{G}_W\mathbf{m} = \mathbf{WGm} = \mathbf{Wd} = \mathbf{d}_W$$

where **G** is the (*NxM*) design matrix containing the partial derivatives of *N* data by *M* model parameters, **m** is the (*Mx1*) model adjustment vector $[\Delta T, \Delta x, \Delta y, \Delta z]^T$, **d** is the (*Nx1*) vector of time residuals. If *do_correlated_errors* (**Table 4**) is false, **W** represents the diagonal weight matrix.

If correlated travel-time errors are accounted for, **W** is the (*NxN*) projection matrix that orthogonalises the data set. In other words, we solve the inversion problem in the eigen-coordinate system in which the transformed observations are independent, that is, $\mathbf{d}_W$ represents linear combinations of the observed residuals, the "eigen residuals". The equation above is solved by singular value decomposition, which yields the general inverse

$$\mathbf{G}_W^{-1} = \mathbf{V}_W\mathbf{\Lambda}_W^{-1}\mathbf{U}_W^T$$

and the model adjustment of

$$\mathbf{m}_{est} = \mathbf{G}_W^{-1}\mathbf{d}_W$$

After the j^{th} iteration, the model vector is adjusted such that $\mathbf{m}_{j+1} = \mathbf{m}_j + \mathbf{m}_{est}$. Damping can be applied for large condition numbers.

For free-depth solutions the depth is kept fixed for the first *min_iter*-1 (**Table 4**) iterations, to mitigate the trade-off between depth and epicentre. After each iteration the depth is checked against negative or excessively large values and fixed to 0 or 600 km. The depth remains fixed for the rest of the iterations if the number of occurrences of airquakes/deepquakes exceeds 2.

The $\mathbf{G_w}$ matrix and the $\mathbf{d_w}$ vector are recalculated with respect to the new solution. A phase is made non-defining if its residual exceeds *sigmathres* times the *a priori* measurement error (see **Tables 4** and **5**). Phases are reidentified if the depth crosses the Moho or Conrad discontinuity. Normally the projection matrix needs to be calculated only once. However, if defining phases were renamed or defining phases were made non-defining, the projection matrix is recalculated.

The convergence test is based on the Paige-Saunders convergence test value (Paige and Saunders, 1982) and on the history of model and data norms. The convergence test is applied after every iteration.

2.7.2 Formal uncertainties

Once a convergent solution is obtained, the location uncertainty is defined by the *a posteriori* model covariance matrix

$$\mathbf{C}_M = \mathbf{G}^{-1}\mathbf{C}_D\mathbf{G}^{-1^T} = \mathbf{V}_W\mathbf{\Lambda}_W^{-2}\mathbf{V}_W^T$$

The model covariance matrix yields the four-dimensional error ellipsoid whose projections provide the two-dimensional error ellipse and one-dimensional errors for depth and origin time. The error ellipse encompassing the confidence region at a given *α* percentile level is defined by

$$(\mathbf{r} - \mathbf{r}_{loc})^T\mathbf{C}_M(\mathbf{r} - \mathbf{r}_{loc}) = \kappa_\alpha^2$$

where $\mathbf{r}_{loc}$ denotes the location vector of the epicentre. We follow Jordan and Sverdrup (1981) to define κ_α^2 as

$$\kappa_\alpha^2 = Ms^2 F_\alpha (M, K + N - M)$$

where the variance scaling factor s^2 is defined as

$$s^2 = \frac{K + \frac{1}{N}\sum d_w^2}{K + N - M}$$

and F_α is an F statistic (Zwillinger and Kokoska, 2000) with M and K+N-M degrees of freedom at the critical level α = *confidence*% (**Table 4**) with $M = 2$ and with N *independent* observations, that is, the total number of observations less the number of observations projected to the null space. K is set to a large value (99999) so that the formal uncertainty estimates approximate "coverage" error ellipses.

2.7.3 Location quality metrics

Besides calculating the formal uncertainties, *iscloc* also computes various location quality metrics (Bondár et al., 2004; Bondár and McLaughlin, 2009) based on the network geometry. The maximal azimuthal gap, the maximal secondary azimuthal gap and the deviation from a uniformly distributed station network are calculated for local, near-regional, teleseismic distance ranges as well as for the entire network.

- Local network: 0 - 150 km
- Near regional: 3 - 10 degrees
- Teleseismic: 28 - 180 degrees
- Entire network: 0 - 180 degrees

The location quality metrics help to assess the accuracy of the hypocentre solution and identify ground truth candidate events.

2.8 Magnitude calculation

Currently *iscloc* calculates body and surface wave magnitudes. MS is calculated for shallow events (depth < *surf_mag_max_depth*) only. At least three station magnitudes are required for a network (mb or MS) magnitude. The network magnitude is defined as the median of the station magnitudes, and its uncertainty is defined as the standard median absolute deviation (SMAD) of the alpha-trimmed (*alpha* = 20%) station magnitudes.

The station magnitude is defined as the median of reading magnitudes for a station. The reading magnitude is defined as the magnitude computed from the maximal log(A/T) in a reading. Amplitude magnitudes are calculated for each reported amplitude - period pairs.

Note that station, reading, amplitude and MS horizontal and vertical components are always stored in the database, even if there were insufficient number of station magnitudes to calculate a network magnitude.

2.8.1 Body-wave magnitudes

Body-wave magnitudes are calculated for each reported amplitude-period pairs, provided that the phase is in the list of phases that can contribute to mb (*mb_phase*), the station is between the epicentral distances *body_mag_min_dist* and *body_mag_max_dist*, and the period is between *body_mag_min_per* and *body_mag_max_per*.

A reading contains all parametric data reported by a single agency for an event at a station, and it may have several reported amplitude and periods. For each pair an amplitude mb is calculated.

$$\text{Amplitude mb} = \log(A/T) + Q(\Delta,h)$$

If no amplitude-period pairs are reported for a reading, the body-wave magnitude is calculated using the reported logat values.

$$\text{Amplitude mb} = logat + Q(\Delta,h)$$

The magnitude attenuation Q(*d,h*) value is calculated using the Gutenberg-Richter (Gutenberg and Richter, 1956), or optionally, using the Veith-Clawson (Veith and Clawson, 1972) or the Murphy-Barker (Murphy and Barker, 2003) tables. For the Gutenberg-Richter tables amplitudes are measured in micrometers, while for the Veith-Clawson and Murphy-Barker tables amplitudes are measured in nanometers.

$$Q(\Delta,h) = Q_{GR}(\Delta,h) - 3$$

$$Q(\Delta,h) = Q_{VC}(\Delta,h)$$

$$Q(\Delta,h) = Q_{MB}(\Delta,h)$$

For each reading *iscloc* finds the reported amplitude-period pair for which *A/T* is maximal:

$$\text{Reading mb} = \log(\max(A/T)) + Q(\Delta,h)$$

Or, if no amplitude-period pairs were reported for the reading:

$$\text{Reading mb} = \max(logat) + Q(\Delta,h)$$

Several agencies may report data from the same station. The station magnitude is defined as the median of the reading magnitudes for a station.

$$\text{Station mb} = \text{median}(\text{Reading mb})$$

Once all station mb values are determined, the station magnitudes are sorted and the lower and upper α percentiles are made non-defining. The network mb and its uncertainty are then calculated as the median and the standard median absolute deviation (SMAD) of the alpha-trimmed station magnitudes, respectively.

2.8.2 Surface-wave magnitudes

Surface-wave magnitudes are calculated for each reported amplitude-period pairs, provided that the phase is in the list of phases that can contribute to MS (*ms_phase*), the station is between the epicentral distances *surf_mag_min_dist* and *surf_mag_max_dist*, and the period is between *surf_mag_min_per* and *surf_mag_max_per*.

For each reported amplitude-period pairs MS is calculated using the Prague formula (Vanek et al., 1962). Amplitude MS is calculated for each component (Z, E, N) separately.

$$\text{Amplitude MS} = \log(A/T) + 1.66 * \log(\Delta) + 0.3$$

To calculate the reading MS, *iscloc* first finds the reported amplitude-period pair for which *A/T* is maximal on the vertical component.

$$MS_Z = \log(\max(A_Z/T_Z)) + 1.66 * \log(\Delta) + 0.3$$

Then it finds the max(*A/T*) for the E and N components for which the period measured on the horizontal components is within ±pertol seconds from the period measured on the vertical component. The horizontal MS is calculated as

$$\max(A/T)_H = \begin{cases} \sqrt{2(\max(A_E/T_E))^2} & if\ MS_N\ does\ not\ exist \\ \sqrt{(\max(A_E/T_E))^2 + (\max(A_N/T_N))^2} & if\ MS_E\ and\ MS_N\ exist \\ \sqrt{2(\max(A_N/T_N))^2} & if\ MS_E\ does\ not\ exist \end{cases}$$

$$\mathrm{MS_H} = \log(\max(A/T)_H) + 1.66 * \log(\Delta) + 0.3$$

The reading MS is defined as

$$\text{Reading } MS = \begin{cases} (MS_Z + MS_H)/2 & if\ MS_Z\ and\ MS_H\ exist \\ MS_H & if\ MS_Z\ does\ not\ exist \\ MS_Z & if\ MS_H\ does\ not\ exist \end{cases}$$

Several agencies may report data from the same station. The station magnitude is defined as the median of the reading magnitudes for a station.

Station MS = median(Reading MS)

Once all station MS values are determined, the station magnitudes are sorted and the lower and upper α percentiles are made non-defining. The network MS and its uncertainty are then calculated as the median and the standard median absolute deviation (SMAD) of the alpha-trimmed station magnitudes, respectively.

3 Input/output

3.1 Database

The locator reads the **event**, **hypocenter**, **hypoc_err**, **hypoc_acc**, **association**, **phase**, **amplitude**, **report** and **site** tables. Note that some of the fields are read from the database only for the sake of producing an optional ISF output file.

Note that several important assumptions are made:

- There is a prime (preferred) hypocentre identified by the *prime_hyp* in the **event** table;
- There is an author identified by *in_agency* (**Table 4**) in the **event** table for every physical event.
- The *etype* field in the **event** table describes the preferred event type (if it is null, 'ke' is taken by default) when the author is *in_agency*.
- The author of the phases associated to the prime hypocentre is identified by *in_agency*.

Once the data are read from the database, the reported hypocentres are sorted by their score, the prime hypocentre being the first. A reported phase name may be replaced if it was reidentified by NEIC, CSEM, EHB or IASPEI. The phases are sorted by *delta*, *prista*, *rdid*, and *time*.

Table 7. Data read from database tables

Table	Field	Description
event	prime_hyp	hypocentre id of the prime hypocentre
event	evid	physical event id for the prime hypocentre

event	etype	event type
hypocenter	hypid	hypocentre id
hypocenter	day	origin time to the second
hypocenter	msec	origin time milliseconds
hypocenter	lat	event latitude
hypocenter	lon	event longitude
hypocenter	depth	event depth (depdp if depth is null)
hypocenter	depfix	fixed depth flag (for ISF output only)
hypocenter	epifix	fixed epicentre flag (for ISF output only)
hypocenter	timfix	fixed origin time flag (for ISF output only)
hypocenter	mindist	closest station distance (for ISF output only)
hypocenter	maxdist	farthest station distance (for ISF output only)
hypocenter	azimgap	azimuthal gap (for ISF output only)
hypocenter	author	author of hypocentre report
hypocenter	nsta	number of readings (for ISF output only)
hypocenter	ndefsta	number of defining stations (for ISF output only)
hypocenter	nass	number of associated phases (for ISF output only)
hypocenter	mdef	number of defining phases (for ISF output only)
hypoc_err	sminax	semi-minor axis (for ISF output only)
hypoc_err	smajax	semi-major axis (for ISF output only)
hypoc_err	strike	azimuth of semi-major axis (for ISF output only)
hypoc_err	stime	origin time uncertainty (for ISF output only)
hypoc_err	sdepth	depth uncertainty (for ISF output only)
hypoc_err	sdobs	standard error of residuals (for ISF output only)
hypoc_acc	score	event score (to sort hypocentres)
site	prista	primary station code (sta if prista is null)
site	lat	station latitude
site	lon	station longitude
site	elev	station elevation
reporter	reporter	reporting agency (phase data)
association	reporter	data report id
association	phase	phase associated to the prime hypocentre, and/or phase associated to NEIC, EHB, IASPEI, CSEM hypocentres
association	delta	epicentral distance
association	phase_fixed	phase name fixed by analyst
association	nondef	phase made non-defining by analyst
phase	rdid	reading id
phase	phid	phase id
phase	sta	station code
phase	phase	reported phase name
phase	day	reported arrival time to the second
phase	msec	reported arrival time milliseconds
phase	slow	reported slowness (for ISF output only)
phase	azim	reported station-to-event azimuth (for ISF output only)
phase	chan	reported channel (for ISF output only)
phase	sp_fm	reported short period first motion (for ISF output only)
phase	emergent	reported emergent flag (for ISF output only)

phase	impulsive	reported impulsive flag (for ISF output only)
amplitude	ampid	amplitude id
amplitude	amp	reported amplitude
amplitude	per	reported period
amplitude	logat	reported log(A/T)
amplitude	chan	reported channel of amplitude measurement
amplitude	amptype	reported amplitude type

WARNING: on output, *iscloc* overwrites the associations that belong to the prime solution!

If a convergent solution is reached, the previous *in_agency* solution, if any, is removed from the database. A new *hypid* is obtained if the previous prime was not *in_agency* and the **event** table is updated accordingly. The **hypocenter**, **hypoc_err**, **hypoc_acc**, **network_quality**, **association**, **netmag**, **stamag**, **readingmag**, **ampmag** and **ms_zh** tables are populated with the new solution.

If the locator failed to get a convergent solution, the **event**, **hypocenter** and **association** tables are reset to the previous prime.

3.2 ISF files

A detailed description of the ISF/ISF2 formats can be found at the ISC website, www.isc.ac.uk. The ISF2 format is an extension of the ISF format that incorporates the new International Seismograph Station Registry station naming conventions, *agency.network.station.location*.

When *iscloc* reads its input from an ISF/ISF2 file, it is assumed that the prime hypocentre is the last one among the hypocentres listed in the origin block of an event. The station coordinates are provided in a separate file, *isf_stafile* (**Table 4**), and for the new International Station Registry, in *IR2_stalist*. The output is written to an ISF2 format file.

4 References

Amante, C. and B.W. Eakins, 2009, ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis. *NOAA Technical Memorandum NESDIS NGDC-24*, 19pp, March 2009.

Bolton, M.K., D.A. Storchak, and J. Harris, 2006, Updating default depth in the ISC bulletin, *Phys. Earth Planet. Int.*, **158**, 27-45.

Bondár, I., and D. Storchak, 2011, Improved location procedures at the International Seismological Centre, *Geophys. J. Int.*, doi: 10.1111/j.1365-246X.2011.05107.x.

Bondár, I., and K. McLaughlin, 2009, Seismic location bias and uncertainty in the presence of correlated and non-Gaussian travel-time errors, *Bull. Seism. Soc. Am.*, **99**, 172-193.

Bondár, I. and K. McLaughlin, 2009, A new ground truth data set for seismic studies, *Seism. Res. Let.*, **80**, 465-472.

Bondár, I., S.C. Myers, E.R. Engdahl and E.A. Bergman, 2004, Epicenter accuracy based on seismic network criteria, *Geophys. J. Int.*, **156**, 483-496, doi: 10.1111/j.1365-246X.2004.02070.x.

Buland, R. and C.H. Chapman, 1983. The computation of seismic travel times, *Bull. Seism. Soc. Am.*, **73**, 1271-1302.

de Hoon, M.J.L., S. Imoto, J. Nolan and S. Miyano, 2004, Open source clustering software, *Bioinformatics*, **20**, 1453-1454.

Sibson, R., 1973, SLINK: An optimally efficient algorithm for the single-link cluster method. *Comp. J.*, **16**, 30-34.

Dziewonski, A.M. and F. Gilbert, 1976, The effect of small, aspherical perturbations on travel times and a re-examination of the correction for ellipticity, *Geophys., J. R. Astr. Soc.*, **44**, 7-17.

Engdahl, E.R., R. van der Hilst, and R. Buland, 1998. Global teleseismic earthquake relocation with improved travel times and procedures for depth determination, *Bull. Seism. Soc. Am.,* **88**, 722-743.

Gutenberg, B. and C.F. Richter, 1956, Magnitude and energy of earthquakes, *Ann. Geof.*, **9**, 1-5.

Jeffreys, H., and K.E. Bullen, 1940. *Seismological Tables*, Brit. Assoc. Adv. Sci., Gray-Milne Trust, London.

Jordan T.H. and K.A. Sverdrup, 1981, Teleseismic location techniques and their application to earthquake clusters in the South-Central Pacific, *Bull. Seism. Soc. Am.*, **71**, 1105-1130.

Kennett, B.L.N., 2006. Non-linear methods for event location in a global context, *Phys. Earth Planet. Int.*, **158**, 45-64.

Kennett, B.L.N., E.R. Engdahl, and R. Buland, 1995. Constraints on seismic velocities in the Earth from traveltimes, *Geophys. J. Int.*, **122**, 108-124.

Kennett, B.L.N. and O. Gudmundsson, 1996, Ellipticity corrections for seismic phases, *Geophys. J. Int.*, **127**, 40-48.

Murphy J.R. and B.W. Barker, 2003, Revised B(d,h) correction factors for use in estimation of mb magnitudes, *Bull. Seism. Soc. Am.*, **93**, 1746-1764.

Murphy J.R. and B.W. Barker, 2006, Improved focal-depth determination through automated identification of the seismic depth phases pP and sP, *Bull. Seism. Soc. Am.*, **96**, 1213-1229.

Myers, S.C, M.L. Begnaud, S. Ballard, M.E. Pasyanos, W.S. Phillips, A.L. Ramirez, M.S. Antolik, K.D. Hutchenson, J. Dwyer, C. A. Rowe, and G. S. Wagner, 2010, A crust and upper mantle model of Eurasia and North Africa for Pn travel time calculation, *Bull. Seism. Soc. Am.*, **100**, 640-656.

Paige, C. and M. Saunders, 1982, LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares, *ACM Trans. Math. Soft.*, **8**, 43-71.

Sambridge, M., 1999, Geophysical inversion with a neighbourhood algorithm. I. Searching the parameter space, *Geophys. J. Int.*, **138**, 479-494.

Sambridge, M. and B.L.N. Kennett, 2001, Seismic event location: non-linear inversion using a neighbourhood algorithm, *Pageoph*, **158**, 241-257.

Storchak, D.A., J. Schweitzer and P. Bormann, 2003. The IASPEI Standard Seismic Phase List, *Seism. Res. Let.*, **74**, 761-772.

Vanek, J., A. Zatopek, V. Karnik, N.V. Kondorskaya, Y.V. Riznichenko, Y.F. Savarensky, S.L. Solovev and N.V. Shebalin, 1962, Standardization of magnitude scales, *Bull. (Izvest.) Acad. Sci. USSR Geophys. Ser.*, **2**, 108-111.

Veith, K.F. and G.E. Clawson, 1972, Magnitude from short-period P-wave data, *Bull. Seism. Soc. Am.*, **62**, 435-452.

Young, J.B., B.W. Presgrave, H. Aichele, D.A. Wiens and E.A. Flinn, 1996. The Flinn-Engdahl regionalization scheme: the 1995 revision, *Phys. Earth Planet. Int.*, **96**, 223-297.

Zwillinger, D. and S. Kokoska, 2000, *CRC Standard Probability and Statistics Tables and Formulae*, Chapman and Hall/CRC.

Appendix A: ISC Database Schema

A detailed description of the entire ISC database schema can be found at the ISC website, www.isc.ac.uk. This document describes only the database tables the ISC locator reads and writes.

With the introduction of the new ISC locator the ISC database schema has undergone some changes:

- new tables (**ampmag**, **readingmag**, **ms_zh**, **magnitude_type**, **network_quality**) were added to the schema;
- new fields were added to some tables (**hypocenter**, **hypoc_err**, **hypoc_acc**, **netmag**, **stamag**, **association**);
- some fields became obsolete (marked as '*not used*').

In order to maintain the consistency of the ISC database, the unique identifiers assigned to each measurement or computed entities are governed by the sequences implemented on the main ISC account. The main account is specified by the *nextid_db* configuration parameter. The locator obtains unique identifiers from the sequences below.

- **hypid** – hypocentre identifier;
- **magid** – network magnitude identifier;
- **stamag_stamagid_seq** – station magnitude identifier;
- **rdmagid** – reading magnitude identifier;
- **ampmagid** – amplitude magnitude identifier;
- **mszhid** – MS vertical/horizontal component magnitude identifier;

event table

Column	Type	Description	Comment
author	varchar(16)	author	
banished	varchar(1)	banished	[B|null]
evid	integer	reported event id	**primary key**
lddate	timestamp	load date	
moddate	timestamp	modification date	
prime_hyp	integer	prime hypocentre id	not null
ready	varchar(1)	ready to publish	[R|T|null]
remid	integer	remark id	
reporter	integer	data report id	
etype	varchar(4)	event type	[k|s|d|f][e|h|i|m|n|r|x] [ls] [uk]

Remarks

- There should be an *in_agency* (**Table 4**) author for every physical event
- The *etype* field contains the preferred event type when author = ‘<*in_agency*>’.
- The *etype* field is a two letter code where the first letter stands for
  k – known
  s – suspected
  f – felt
  d – damaging
  and the second letter stands for
  e – earthquake
  h – chemical explosion
  i – induced event, mine collapse
  m – mine explosion
  n – nuclear explosion
  q – quarry blast
  r – rockburst, coalbump
  x – explosion
  exceptions:
  uk – unknown
  ls – landslide
- The *ready* field indicates the status of the analyst review process
  T – waiting for review
  R – reviewed
  null – not reviewed by the ISC

hypocenter table

Column	Type	Description	Comment
accuracy	varchar(10)	location accuracy	(km)
author	varchar(16)	author	
azimgap	numeric(6,3)	max azimuthal gap	
centroid	varchar(1)	centroid hypo	[C|null]
coll_evid	integer	reported event id	
day	timestamp	origin time to the second	
depdp	numeric(8,5)	depth-phase depth	(km)
depfix	varchar(6)	fixed-depth flag	[null|A|B|D|G|H|M|R|S]
deprecated	varchar(1)	deprecated flag	[D|M|P|null]
depth	numeric(8,5)	depth	(km)
epifix	varchar(6)	fixed-epicentre flag	[F|null]
etype	varchar(4)	event type	
grn	integer	geographic region number	
hypid	integer	hypocentre id	**primary key**
isc_evid	integer	physical event id	
lat	numeric(8,5)	latitude	(decimal degrees)
lddate	timestamp	load date	
lon	numeric(8,5)	longitude	(decimal degrees)
magid	integer	magnitude id	(*not used*)
magnitude	mumeric(4,2)	magnitude	(*not used*)
magtype	varchar(2)	magnitude type	(*not used*)
maxdist	numeric(6,3)	max station distance	(decimal degrees)
mindist	numeric(6,3)	min station distance	(decimal degrees)
moddate	timestamp	modification date	
msec	integer	origin time milliseconds	
nass	integer	# of associated phases	
ndef	integer	# of defining phases	
ndefsta	integer	# of defining stations	
ndp	integer	# of depth phases	
nsta	integer	# of readings(!)	NOT # of stations
pref_hypid	integer	preferred hypid for an author	
prime	varchar(1)	prime hypocentre	[P|null]
remid	integer	remark id	
reporter	integer	data report id	
srn	integer	seismic region number	
timfix	varchar(6)	fixed-origin time flag	[F|null]
velo_model	varchar(6)	velocity model	
nrank	integer	# of independent defining phases	

Remarks

- *nsta* is the number of readings (a reading contains all phases reported by a single agency for an event at a single station), not the number of distinct stations.
- The *depfix* field can be

  null - free-depth solution
  A - depth fixed by the analyst
  S - anthropogenic event; depth fixed to surface
  G - depth fixed to ISC default depth grid
  R - depth fixed to ISC default region depth
  M - depth fixed to median depth of reported hypocentres
  B - beyond depth limits; depth fixed to 0/*max_depth_km*
  H - depth fixed to depth of a reported hypocentre

D - depth fixed to depth-phase depth

- *pref_hypid* stands for the preferred hypocentre for multiple reports from the same agency
- The *deprecated* field can be
  D – deprecated, *M* – modified, *P* – preliminary, or null

hypoc_err table

Column	Type	Description	Comment
author	varchar(16)	author	
confidence	varchar(6)	confidence level %	
deprecated	varchar(1)	deprecated flag	[D|M|P|null]
hypid	integer	hypocentre id	**primary key**
lddate	timestamp	load date	
moddate	timestamp	modification date	
remid	integer	remark id	
reporter	integer	data report id	
sdepth	real	depth uncertainty	(km)
sdobs	numeric(8,4)	unweighted RMS	(s)
slat	numeric(8,5)	latitude error	(decimal degree)
slon	numeric(8,5)	longitude error	(decimal degree)
smajax	real	semi-major axis	(km)
sminax	real	semi-minor axis	(km)
stime	numeric(6,3)	origin time uncertainty	(s)
strike	numeric(4,1)	semi-major axis azimuth	(decimal degree)
stt	numeric(12,4)	model covariance matrix	
stx	numeric(12,4)	model covariance matrix	
sty	numeric(12,4)	model covariance matrix	
stz	numeric(12,4)	model covariance matrix	
sxx	numeric(12,4)	model covariance matrix	
sxy	numeric(12,4)	model covariance matrix	
syy	numeric(12,4)	model covariance matrix	
syz	numeric(12,4)	model covariance matrix	
szx	numeric(12,4)	model covariance matrix	
szz	numeric(12,4)	model covariance matrix	
sdepdp	real	depth-phase depth uncertainty	(km)

Remarks

- *smajax, sminax, stime and sdepth* are scaled to the *confidence* percent level

hypoc_acc table

Column	Type	Description	Comment
hypid	integer	hypocentre id	**primary key**
reporter	integer	data report id	
score	real	event score to set prime	
nstaloc	integer	# of defining local stations	local: 0-150 km
gtcand	integer	GT candidate	presumed location accuracy
nsta10	integer	# of stations within 10 km	

network_quality table

Column	Type	Description	Comment
hypid	integer	hypocentre id	not null
reporter	integer	data report id	
type	varchar(10)	distance range	not null
du	numeric(8,5)	deviation from uniformly distributed stations	
gap	numeric(8,5)	largest azimuthal gap	
secondary_gap	numeric(8,5)	largest secondary azimuthal gap	
nsta	integer	# of defining stations	
mindist	numeric(7,3)	min station distance	
maxdist	numeric(7,3)	max station distance	

primary key: (hypid, type)

Remarks

- the network quality metrics are given for various distance ranges.
- the *type* field describes the distance ranges:
  local: 0 - 150 km
  near: 3 - 10 degrees
  tele: 28 - 180 degrees
  whole: 0 - 180 degrees

association table

Column	Type	Description	Comment
author	varchar(16)	author	not null
azimdef	varchar(1)	defining azimuth	[A|null]
azimres	numeric(6,3)	azimuth residual	
delta	numeric(6,3)	epicentral distance	(decimal degrees)
deprecated	varchar(1)	deprecated flag	[D|null]
esaz	numeric(6,3)	event-station azimuth	(decimal degrees)
hypid	integer	hypocentre id	not null
lddate	timestamp	load date	
moddate	timestamp	modification date	
net	varchar(6)	network code	
phase	varchar(8)	IASPEI phase code	
phase_fixed	varchar(1)	fixed phase code	{F|null]; phase cannot be renamed
phid	integer	phase id	not null
remid	integer	remark id	
reporter	integer	data report id	
seaz	numeric(6,3)	station-event azimuth	(decimal degrees)
slowdef	varchar(1)	defining slowness	[S|null]
slowres	numeric(6,3)	slowness residual	(s / degree)
sta	varchar(6)	station code	
timedef	varchar(1)	defining time	[T|null]
timeres	numeric(9,3)	time residual	(s)
weight	numeric(4,3)	weight by locator	1/prior measurement error
nondef	varchar(1)	do not use in location	[U|null]

primary key: (phid, author)

phase table

Column	Type	Description	Comment
ampid	integer	amplitude id	
author	varchar(16)	author	
azim	numeric(5,2)	azimuth	(decimal degrees)
chan	varchar(3)	channel	
day	timestamp	arrival time to the second	
delazim	numeric(5,2)	azimuth picking error	(decimal degrees)
delslow	numeric(5,2)	slowness picking error	(s/degree)
deltime	numeric(5,2)	time picking error	(s)
deprecated	varchar(1)	deprecated flag	[D|M|null]
emergent	varchar(1)	emergent signal	[e|E|null]
impulsive	varchar(1)	impulsive signal	[i|I|null]
init	varchar(1)	first phase in a reading	
lddate	timestamp	load date	
lp_fm	varchar(1)	long period first motion	[c|C|+|d|D|-|null]
moddate	timestamp	modification date	
msec	integer	arrival time milliseconds	(ms)
net	varchar(6)	network code	
phase	varchar(8)	reported phase code	
phid	integer	phase id	**primary key**
pref_rd	integer	preferred reading id	deals with duplicates
rdid	integer	reading id	
remid	integer	remark id	
reporter	integer	data report id	
slow	numeric(9,2)	slowness	(s/degree)
sp_fm	varchar(1)	short period first motion	[c|C|+|d|D|-|null]
sta	varchar(6)	station code	

amplitude table

Column	Type	Description	Comment
amp	numeric(12,1)	amplitude	(nm)
ampid	integer	amplitude id	**primary key**
amptype	varchar(8)	amplitude type	[0-to-p|p-to-p]
author	varchar(16)	author	
chan	varchar(3)	channel	
day	timestamp	amplitude time to the second	
delamp	numeric(10,1)	amplitude picking error	(nm)
delper	integer	period picking error	(s)
deprecated	varchar(1)	deprecated flag	[D|M|null]
factor	integer	exponent for moment	
lddate	timestamp	load date	
logat	numeric(9,2)	log(A/T)	
moddate	timestamp	modification date	
moment	numeric(8,5)	scalar moment mantissa	
msec	integer	amplitude time milliseconds	(ms)
per	numeric(6,2)	period	(s)
phid	integer	phase id	not null
reporter	integer	data report id	

netmag table

Column	Type	Description	Comment
author	varchar(16)	author	
deprecated	varchar(1)	deprecated flag	[D|M|P|null]
hypid	integer	hypocentre id	not null
lddate	timestamp	load date	
magid	integer	magnitude id	**primary key**
magnitude	numeric(4,2)	magnitude value	
magtype	varchar(6)	magnitude type	
moddate	timestamp	modification date	
nsta	integer	# of stations	
remid	integer	remark id	
reporter	integer	data report id	
uncertainty	numeric(4,2)	magnitude error	
nagency	integer	# number of agencies	

stamag table

Column	Type	Description	Comment
ampid	integer	amplitude id	(*not used*)
author	varchar(16)	author	
deprecated	varchar(1)	deprecated flag	[D|M|null]
hypid	integer	hypocentre id	not null
lddate	timestamp	load date	
magid	integer	magnitude id	
magnitude	numeric(4,2)	magnitude value	
magtype	varchar(6)	magnitude type	
moddate	timestamp	modification date	
phase	varchar(8)	phase code	(*not used*)
phid	integer	phase id	(*not used*)
remid	integer	remark id	
reporter	integer	data report id	
uncertainty	numeric(4,2)	magnitude error	(*not used*)
stamagid	integer	station magnitude id	**primary key**
magdef	varchar(1)	defining magnitude	[D|null]
rdid	integer	reading id	
sta	varchar(6)	station code	

Remarks

- **stamag** stores the station magnitudes calculated from the reading magnitudes.
- the *magdef* field indicates whether the station magnitude contributed to the calculation of network magnitude uncertainty. It is set to 1 if the *magnitude* is between the 20th and 80th percentile ranges, otherwise to 0.
- **stamag** is populated even if there are an insufficient number of stations to calculate a network magnitude. In that case *magid* is null.

magnitude_type table

Column	Type	Description	Comment
mtypeid	integer	magnitude type id	**primary key**
mtype	varchar(20)	magnitude type	
description	varchar(50)	description	

Remarks

- **magnitude_type** lists the IASPEI and reported magnitude types.

readingmag table

Column	Type	Description	Comment
rdmagid	integer	reading magnitude id	**primary key**
hypid	integer	hypocentre id	not null
magid	integer	magnitude id	
repid	integer	data report id	
rdid	integer	reading id	not null
sta	varchar(6)	station code	
mtypeid	integer	magnitude type id	not null
magnitude	numeric(4,2)	magnitude value	
magdef	integer	defining magnitude	[0|1]
author	varchar(16)	author	

Remarks

- **readingmag** stores the magnitudes calculated for a reading from individual amplitude reports.
- the *magdef* field indicates whether the reading magnitude contributed to the station magnitude. It is set to 1 for the median magnitude, otherwise to 0.
- **readingmag** is populated even if there is an insufficient number of stations to calculate a network magnitude. In that case *magid* is null.

ampmag table

Column	Type	Description	Comment
ampmagid	integer	amplitude id	**primary key**
ampid	integer	amplitude id	not null
hypid	integer	hypocentre id	not null
repid	integer	data report id	
rdid	integer	reading id	
mtypeid	integer	magnitude type id	not null
magnitude	numeric(4,2)	magnitude value	
ampdef	integer	defining amplitude	[0|1]
author	varchar(16)	author	

Remarks

- **ampmag** stores the magnitudes calculated from individual amplitude reports.
- the *ampdef* field indicates whether the amplitude contributed to the reading magnitude. It is set to 1 for the magnitude with max(A/T), otherwise to 0.
- **ampmag** is populated even if there is an insufficient number of stations to calculate a network magnitude. In that case *magid* is null.

ms_zh table

Column	Type	Description	Comment
mszhid	integer	mszh id	**primary key**
repid	integer	data report id	
rdid	integer	reading id	not null
msz	numeric(4,2)	magnitude	MS on vertical component
mszdef	integer	defining MS(z)	[0|1]
msh	numeric(4,2)	magnitude	MS on horizontal component
mshdef	integer	defining MS(h)	[0|1]
hypid	integer	hypocentre id	not null
author	varchar(16)	author	

Remarks

- **ms_zh** stores the MS values calculated on horizontal and vertical components.
- the *mszdef* and *mshdef* fields indicate whether the component magnitude reading magnitude contributed to the reading MS magnitude.
- **ms_zh** is populated even if there is an insufficient number of stations to calculate a network magnitude. In that case *magid* is null.

report table

Column	Type	Description	Comment
collector	varchar(8)	ISC personnel who parsed the report	
dirname	varchar(40)	location of data report file	
filename	varchar(20)	data report filename	
first_hyp	timestamp	time of first hypocentre in report	
first_phase	timestamp	time of first phase arrival in report	
hyp_count	integer	# of hypocentres in report	
hyp_max_lat	numeric(8,5)	extremes of hypocentres in report	
hyp_max_lon	numeric(8,5)	extremes of hypocentres in report	
hyp_min_lat	numeric(8,5)	extremes of hypocentres in report	
hyp_min_lon	numeric(8,5)	extremes of hypocentres in report	
last_hyp	timestamp	time of last hypocentre in report	
last_phase	timestamp	time of last phase in report	
lddate	timestamp	load date	
moddate	timestamp	modification date	
phase_time_precision	integer	precision code	[0|1|2|3|null]
product	varchar(20)	data report type (bulletin, catalog, &c)	
repid	integer	data report id	**primary key**
replaceable	varchar(1)		*not used*
reporter	varchar(8)	agency	
reporter_id	varchar(24)	additional notes on reporter	
status	integer	status of data	
sta_count	integer	# of stations in report	
sta_max_lat	numeric(8,5)	extremes of station locations in report	
sta_max_lon	numeric(8,5)	extremes of station locations in report	
sta_min_lat	numeric(8,5)	extremes of station locations in report	
sta_min_lon	numeric(8,5)	extremes of station locations in report	
issue_date	timestamp	time data were parsed	
parser	varchar(30)	program used to parse data to DB	

Remarks

- *repid* called *reporter* in the other tables

site table

Column	Type	Description	Comment
author	varchar(16)	author	
close_date	timestamp	station close date	
country	varchar(40)		
depth	numeric(5,1)		
digital	varchar(1)	digital station	[0/1]
elev	numeric(8,1)	elevation above seal level	(m)
lat	numeric(8,5)	latitude	(decimal degrees)
lddate	timestamp	load date	
lon	numeric(8,5)	longitude	(decimal degrees)
moddate	timestamp	modification date	
net	varchar(6)	network	
open_date	timestamp	station open date	
prinet	varchar(6)	primary network code	
prista	varchar(6)	primary station code	to deal with alternate codes
region	varchar(80)	station specific region info	
remid	integer	remark id	
reporter	integer	data report id	
sta	varchar(6)	station code	
staname	varchar(80)	station name	
statype	varchar(8)		*not used*
zone	varchar(30)	station specific zone info	
siteid	integer	station id	**primary key**

Appendix B: Data files

B1. General configuration parameters

The general *iscloc* configuration parameters are read from *$QETC/iscloc/config.txt* file. The format is expected as **name = value** pairs. The description of the configuration parameters is given in **Table 4**.

B2. Phase and velocity model specific configuration parameters

These *iscloc* configuration parameters are read from *$QETC/iscloc/<ttime_table>_model.txt* file. Currently *ttime_table* = ak135 or jb is supported. The description of the configuration parameters is given in **Table 5**.

B3. ak135 ellipticity correction coefficients

The *ak135* ellipticity correction parameters (Kennett and Gudmundsson, 1996) are specified in the *$QETC/ak135/ELCOR.dat* file. For each phase a block of tau coefficients (Dziewonski and Gilbert, 1976) is given in a format:

phase name, number of distance samples, min dist, max dist
distance (deg)
tau0 (at 0, 100, 200, 300, 500, 700 km)
tau1 (at 0, 100, 200, 300, 500, 700 km)
tau2 (at 0, 100, 200, 300, 500, 700 km)

B4. ak135 travel-time tables

The *ak135* travel-time tables (Kennett et al., 1995) are listed in the *$QETC/ak135/ak135.<phase>.tab* files. Note that because the Mac OS X file system (HFS+) is case-insensitive, that is, it cannot distinguish between ak135.pP.tab and ak135.PP.tab, the depth phases are prefixed with 'little' so that ak135.pP.tab becomes ak135.littlepP.tab. The travel-time tables follow the format:

number of distance and depth samples
delta samples (max 25 in a line)
depth samples (one line)
TT table (rows - delta, columns - depth)
dtdd table (rows - delta, columns - depth)
dtdh table (rows - delta, columns - depth)
bounce point distance table (rows - delta, columns - depth) if depth phase

B5. JB travel-time tables

The *jb* travel-time tables (Jeffreys and Bullen 1940) are listed in the *$QETC/jb/jb.<phase>.tab* files. The travel-time tables follow the same format and naming conventions as the ak135 travel-time table files.

B6. Flinn-Engdahl regionalisation

The Flinn-Engdahl regionalization scheme, 1995 version (Young et al., 1996) is given in the *$QETC/Flinn_Engdahl/FE.dat* file. The file follows the format:

> For each latitude (from 90N to 90S) a set of longitude ranges is given (first part of the file). The second part of the file lists the geographic region numbers for each latitude within the corresponding longitude ranges.

B7. Default depth grid

The 0.5º x 0.5º default depth grid (Bondár and Storchak, 2011) is given in the *$QETC/Flinn_Engdahl/default_depth0.5.grid* file.

Columns:

- lat, lon: center of the grid cell
- depth: median depth in the cell
- min: minimum depth in the cell
- 25Q: 25th percentile depth in the cell
- 75Q: 75th percentile depth in the cell
- max: maximum depth in the cell
- N: number of observations in the cell
- range: quartile range (75Q - 25Q)

Rows are ordered by descending latitude and increasing longitude.

B8. Default depths for Flinn-Engdahl regions

The default depths for each Flinn-Engdahl region (Bolton et al., 2006) are given in the *$QETC/Flinn_Engdahl/grn_default_depth.ak135.dat* and the *$QETC/Flinn_Engdahl/grn_default_depth.jb.dat* files.

Columns:

- grn: Flinn-Engdahl geographic region number
- depth: depth for GRN

B9. ETOPO topography/bathymetry

The ETOPO1 bedrock topography (Amante and Eakins, 2009) resampled to 5' x 5' resolution is given in the *$QETC/topo/etopo5_bed_g.i2.bin* binary file.

```
grdfilter -I5m etopo1_bed.grd -Fg15 -D4 -Getopo5_bed.grd
```

> Gridline node registration used
> x_min: -180 x_max: 180 x_inc: 0.0833333 nx: 4321
> y_min: -90 y_max: 90 y_inc: 0.0833333 ny: 2161
> z_min: -10515.5 z_max: 6917.75 name: m
> scale_factor: 1 add_offset: 0

The ETOPO parameters are specified in the *config.txt* (**Table 4**) file.
Format:

> etoponlon * etoponlat 2-byte integers.

B10. Generic variogram model

The generic P variogram model (Bondár and McLaughlin, 2009) is given in the *$QETC/variogram/variogram.model* file. The file follows the format:

number of samples
sill
max station separation in km
station separation (km), variance (s^2)

B11. Magnitude attenuation correction tables

The Gutenberg-Richter (Gutenberg and Richter, 1956), the Veith-Clawson (Veith and Clawson, 1972) and the Murphy-Barker (Murphy and Barker, 2003) magnitude attenuation tables are given in the *$QETC/magnitude/GRmbQ.dat,* the *$QETC/magnitude/VCmbQ.dat* and the *$QETC/magnitude/MBmbQ.dat* files. The files follow the format:

number of distance samples
delta samples
number of depth samples
depth samples
number of distance samples, number of depth samples
magnitude attenuation table Q(d, h)

B12. Station lists

The comma-separated list of registered stations is given in the *$QETC/Stations/isc_stalist* file. This file is regularly updated at the ISC website (www.isc.ac.uk).

Columns:
sta, altsta, lat, lon, elevation
where sta and alsta are the station code and the alternative station code; lat and lon are the station coordinates in decimal degrees; elevation is the station eleavtion above sea level in meters.

The station list that complies with the new standards of the International Registry of Seismographic Stations is given in the *$QETC/Stations/IR2_stalist* file.

Columns:
fdsn.network.sta.location:latitude longitude elevation