wavecall
See your net-work

wavesight™

Input / Output File Formats

Command line mode

Company Information

Address

Lausanne	Wavecall SA Science Park of the Swiss Institute of Technology PSE-B / EPFL 1015 Lausanne Phone +41 21 693 84 05 Fax +41 21 693 84 06
Amsterdam	Wavecall BV NZ Voorburgwal 66/III 1012 SC Amsterdam / NL Phone +31 20 528 7362 Fax +31 20 528 7363
Contact	Dr. Karim Rizk, CEO
Email	info@wavecall.com
Web	http://www.wavecall.com

Document History

Version	Revision	Date
1.0	K. Rizk	November 2, 1998
2.0	P. Bauermeister	October 25, 1999
3.0	O. Ghawi	March 18 2000
4.0	K. Rizk	March 22 2000
5.0	K. Rizk / R. Schweikert	November 20, 2000
6.0 Added description on Logica Odyssey buildings	R. Schweikert	January 21, 2001
6.1 minor modifications on Odyssey buildings format	R. Schweikert	February 11, 2001
7.0 Documentation on "Path" keyword in infiles.txt	R. Schweikert	March 11, 2001
8.0	O. Ghawi	March 31 2004

Contents

1. Running WaveSight

Wavesight runs as command line with the following synopsis:

wavesight [options]

OPTIONS:

-ctxt STR where STR is a string that specifies the context in which Wavesight is running. Typically this could be useful to run Wavesight in parallel, thus the STR would be process ID of the calling process. To use this option the STR must pre pended to the “infiles.txt” and “comp.txt” (see section XX). Then all the out files (see section XX) will be prepended by STR.

2. Input Files

WaveSight uses the input file formats described below.

2.1 Building Vector Data

Building vector data is a specialized version of normal vector data, describing the shapes and heights of buildings in great detail. The application of such data will typically be to provide input for ray-tracing type propagation models, used for the generation microcell coverage within the planning tool.

Building vector data is stored in ASCII format and requires three types of input file - a vector data file, an attributes file and an index file. The index file is similar to a normal vector index file. Each line describes: a vector data file name, an attribute file name, Easting Northing co-ordinates representing a bounding box around that feature and the feature name itself. Since each segment within the building vector data file represents a building wall, the building vector index file usually only refers to one 'feature'.

The data file contains segments of discrete vector points - each segment being a different building wall perimeter. The attributes file contains floating point height value for each segment in the data file. The heights can be either given with respect to local ground or to sea level. The flag Is2Ground in the comp.txt indicates whether these heights are relative to the ground or to the sea level. (By default building is assumed to be relative local ground)

Important consistency of the building database:

1. Identical polygons are not accepted
2. Polygons must be closed. The first and the last point in each polygon must me identical.
3. At least 3 vertices per polygon
4. One vertex must belong to exactly two walls
5. Each two walls can only intersect if they are successive walls belonging to the same polygon. In such a case the intersection forms the vertex.
6. All heights must be above local ground

2.1.1 Supported Formats

The infiles.txt must contain a reference to one of the following formats:

Format	Entry in infiles.txt
SIM File	BldgFile C:\Wavecall\Data\Demo\lisbon.sim
Asset / Planet style vector files	IndexBldgDir C:\Wavecall\Data\Demo\
Odyssey Genamap19 format	IndexBldgDirLO /odyssey/maps/vectors/ BldgLOFeatureName Building_outlines The IndexBldgDirLO points to the Odyssey directory where the "List" file is located. The BldgLOFeatureName is the name of the Odyssey vector class, which contains the buildings.

2.1.2 SIM Files

This is a simple proprietary format:

- The first line indicates if building elevation is given with respect to local ground (1) or with respect to sea level (0). Each following line corresponds to a building.
- The number after the keyword Id is an integer number representing the building id (can be set to 0 if not available).
- The field after the keyword FloorElev is a real number representing the building floor elevation with respect to sea level. If building elevations are given with respect to local ground then FloorElev becomes irrelevant and can be set to 0.
- The field after the keyword TopElev is a real number representing the building top elevation with respect to local ground or sea level, depending on the setting of the parameter on the first line.
- After the keyword Floor follows the x and y coordinates for the floor vertices. Building floor is a closed polygon. Note that the first and the last coordinates are identical

EXAMPLE

Is2Ground 0

Id 1 FloorElev 508 TopElev 521 Floor 590 3338 594 3381 582 3383 578 3340 590 3338

Id 2 FloorElev 511 TopElev 533 Floor 1759 3245 1791 3247 1817 3252 1759 3245

2.1.3 Asset / Planet style vector files

Header Record

Easting Northing

Easting Northing

Easting Northing

Easting Northing

Easting Northing

Header Record

Easting Northing

Easting Northing

Easting Northing

The final row is terminated by a carriage return.

The format of the header record is as follows:

Field	Position	Description
1	1-5	Record Identifier (used to identify building segment in attributes file)
2	6-15	Blanks
3	16-47	32 Character description (not used, same as feature name field in index file entry)
4	48-50	Blanks
5	51-55	Record Count

For example:

1 buildings 5
1629899.00 6582457.30
1629886.40 6582464.70
1629951.50 6582573.70
1629963.20 6582566.90
1629899.00 6582457.30
2 buildings 19
1628823.70 6582523.30
1628821.50 6582527.20

The final row is terminated by a carriage return.

Vector Index File

An ASCII text file called index.txt contains positional information about vector file. This file must be in the same directory as the vector data.

Each row contains the following variables separated by a space:

Field	Description
Data filename	Filename of vector data file
Attribute filename	Filename of building attributes (heights) file
Eastmin	Minimum Easting value (metres)
Eastmax	Maximum Easting value (metres)
Northmin	Minimum Northing value (metres)
Northmax	Maximum Northing value (metres)
Feature Name	Name of the feature stored in the vector data file (Motorway, Coast, etc.)

For example:

buildings_vec.txt buildings_atr.txt 1627764 1630022 6579401 6582574 buildings

The final row is terminated by a carriage return.

Vector Attributes File

An ASCII text file, named in the index file contains height information about the building segments contained in the vector data file.

Each row contains a record, as follows :

Field	Position	Description
1	1-5	Record Identifier (used to identify building segment in vector data file)
2	6	Delimiter <space>
3	7-19	12 Character vector segment description (not used)
4	20	Delimiter <space>
5	21-26	Vector segment height (floating point, two decimal places)

For example:

1 Terminal_1 25.10

2 Terminal_2 25.40

3 Terminal_3 24.20

4 ATC_Tower 35.30

4 Shopping_Ctr 25.30

2.1.4 Odyssey Genamap19 Format

Note: This section is listed here only for completeness, as access to the data is usually provided through the Odyssey External Predictor OEP API (which is used by the Odyssey/WaveSight interface).

Odyssey uses the same "Genamap19" file format for building vector files as used in Asset or Planet (described above), however, to access this data, a "List" file, which references the vectors data, must be set up inside Odyssey.

To access the data, the infiles.txt must contain the two parameters IndexBldgDirLO , which points to the directory, where the "List" file is located, and BldgLOFeatureName, which contains the feature class name for the to be used building vectors.

The building file name in Odyssey (for instance wbuild.vec) must contain at least but not more than a dot '.', otherwise it will be impossible to recover the name. WaveSight checks for this condition and generates an error message in case it is not fulfilled.

The polygons must be closed.

2.2 Input parameter file

Located in the same folder as Wavesight executable. The name of the file must be STRinfiles.txt, where STR is the argument of -ctxt option as described in section one. In case no argument is specified the name of the file will be infiles.txt

This file contains the path for the topographic files used in WaveSight.

EXAMPLE (Each parameter is described by preceding comments, which can be omitted in an actual file, the key words are highlighted):

path to the directory in of the terrain index file. This filed is mandatory

IndexTerrDir C:\Wavecall\~Data\Demo\

path to the directory of the building index file. This filed is mandatory

IndexBldgDir C:\Wavecall\~Data\Demo\

path to the Tx file. This filed is mandatory

TxFile /u1/DataBase/munich/tx.tx

path to the frame of computation file. This filed is mandatory

FrameFile /u1/DataBase/munich/frame.frm

path to measurement route file, this field is optional

RteFile /u1/DataBase/munich/meas.obs

path to the Antenna file. If an antenna pattern is specified in the transmitter file,

this field is mandatory if an antenna pattern is specified in the transmitter file

AntFile u1/DataBase/munich/ant_db.dat

On NT, path to the algorithm DLL

Path C:\Program%20Files\WaveCall\WaveSight\

2.3 Computation parameter file

Located in the same folder as WaveSight executable. The name of the file must be STRcomp.txt, where STR is the argument of -ctxt option as described in section ZZ. In case no argument is specified the name of the file will be comp.txt

It contains the computation parameters. None of the field is mandatory.

EXAMPLE (Each parameter is described by preceding comments, which can be omitted in an actual file, the key word are highlited):

Freq: The Frequency **[GHz],** the default value is 0.9

Freq 0.947

Receiver height above ground **[m]**

RxHeight 1.5

Output file format: 1 for low res ASCII, 2 for high res ASCII, 3 binary, 4 for the three formats
(to reconsidered, not yet implemented).

OutFileFormat 3

Is2Ground: Flag to indicate whether buildings data is relative to sea level or to the ground (DEM).
Is2Ground 1 == Relative to ground (DEM), Is2Ground 0 == Relative to sea level
Note: The formerly used parameter in the SIM file is obsolete and is now ignored.
The default value is Is2Ground 1

Is2Ground 1

IsTx2Ground: Flag to indicate whether the z coordinate of the Transmitter is relative to sea level
or relative to ground (DEM).
IsTx2Ground 1 == Relative to ground (DEM), IsTx2Ground 0 == Relative to sea level
The default value follows the definition of Is2Ground

IsTx2Ground 1

Res: The resolution in meters for the computation of the result.
Recommended values are 5 or 10
Note: The formerly redundant parameter in the frame file is now removed.

Res 5

Parameters for the indoor computation:
IsOnlyIndoor, IsOnlyOutdoor, IsIndoorOutdoor.
These mutually exclusive parameters are used to determine the WaveSight mode for indoor or
outdoor calculations.
The default value is IsOnlyOutdoor 1

IsOnlyIndoor 1

IndoorAttenuation: This parameter is used when the WaveSight mode is either IsOnlyIndoor or
IsIndoorOutdoor. It indicates the penetration loss from the outside to the inside ot a building in dB

IndoorAttenuation 20

2.4 Transmitter file (ASCII *.tx)

Consists of at least two lines. One contains the name of the transmitter. The second line contains the coordinates of the transmitter (x, y, z).

In addition the following lines are optional:

EIRP power in dBm, for path loss computation (default mode) power must be 0

Power 36

The Tilt in degree, considered *positive if the angle above the horizon* line and negative if it is lower

than the horizon line

Tilt -6

The Azimuth in degree, 0 points to the north, 90 to the east.

Azimuth 270

Antenna pattern name as listed in the antenna file given in the infiles.txt. If the antenna

pattern not mentioned an isotropic pattern is the default one.

AntPtrn K735141

EXAMPLE:

GDB11

599135.0 2427470.0 76.0

Power 53.0

Tilt 0

Azimuth 240

AntPtrn PCN_S_085_19_5

2.5 Frame file (ASCII *.frm)

Contains the rectangular frame in which propagation is computed. The first line contains the coordinates of the lowest left limit and right upper limit. It must contain the transmitter.

The format is as follows:

EastMin NorthMin EastMax NorthMax

EXAMPLE:

-1002 -800 1700 2100

2.6 Terrain file

WaveSight supports the following terrain formats:

2.6.1 Digital Terrain Height Data (DTM)

The digital terrain height data (DTM) is stored in a binary format where each element of the data represents the height above sea level in meters for a square area of, for example, 50m x 50m. Each element is two bytes in size and the most significant byte is stored first.

The elements are stored in one continuous array such that the size of the array in the following example would be 500 (wide) x 500 (high) x 2 (bytes per element) = 500,000 bytes.

If there are pixels within the file that are outside the limits of the map, the value -9999 is stored at that location.

An ASCII text file called index.txt contains positional information about each binary height file. This file must be in the same directory as the height data.

The file contains one row describing each height file. Each row contains the following variables separated by a space:

Field	Description
Filename	Filename of DTM Height file
Eastmin	Minimum Easting value (metres)
Eastmax	Maximum Easting value (metres)
Northmin	Minimum Northing value (metres)
Northmax	Maximum Northing value (metres)
Square Size	Size of each element of the height data (metres)

For example:

file1.bin 100000 125000 50000 75000 50

file2.bin 125000 150000 50000 75000 50

The final row is terminated by a carriage return.

2.6.2 TRN format (Wavecall proprietary format)

Required data. Contains one area per file.

Terrain files are in are in the following format:

- The first line contains the rectangular frame in which terrain data is given.
- The second contains one number, which represents terrain-sampling resolution.
- The remaining part of the file is a two dimensional matrix of real (but usually integer) values representing the elevation in meters of the terrain contained in the rectangular frame defined in the first line. The matrix first values correspond to the lowest row, starting at the lower left corner of the frame.
- The values in the file are with respect to the center of each cell given by the frame and the sampling resolution. For example, given a frame of (0,0) (15,15) and a resolution of 5 giving nine cells, nine values need to be written into the terrain file.
- The values are stored as follows:

Values are blank-separated.

EXAMPLE:

592425.000000□2421075.000000□596475.000000□2425125.000000

5.000000

169 169 169 169 169 169 168 168 168 167 167 167 166 ...

2.7 Antenna pattern file (ASCII *.dat)

Required data. Contains one or more antenna patterns per file.

Antenna pattern files are in the following format:

- The first line contains the identifier of the antenna (may be arbitrarily chosen).
- The second line indicates the number of gain points (361 for 1 degree increments, 181 for 2-degree increments, etc.).
- Horizontal gains are listed, separated by a blank. The diagram is given clock wise, the 1st point being for the front direction.
- Then there is a line with the "*V1" marker that indicates the start of the vertical part of the antenna pattern
- Vertical gains are listed, separated by a blank. The vertical part has the same number of points as the horizontal part. The diagram is given clock wise, the 1st point being for the front direction, the next ones gradually pointing towards the ground.
- A negative gain value means an attenuation of the transmitted power in the specified direction. If the power given in the transmitter file (sect. 2.4) indicates EiRP then all the gains are negative and given relative to the main lobe.
- Other antennas can be listed after each other, following the same syntax.

EXAMPLE:

K735141
361
-0.01
-0.01
-0.02
-0.03
…
-0.02
-0.01
*V1
-0.45
-0.1
-0.01
….
-1.96
-1.04
-0.45
…

2.8 Measurement route file (ASCII *.obs)

Optional data. Contains one route per file.

Measurement files are in the following format:

The first line indicates the name of the route.

Each of the following lines consist of:

A receiver identifier

A location (x, y, z), note that the z coordinate is not used. Instead a one constant height given in the comp.txt file is used for all receivers

A measurement value

All five columns in the file must exist. If the receiver height or the measurement value does not exist, a value of zero must be written in the file.

Values are blank-separated.

EXAMPLE:

metro200

1	1431.94	2619.23	1.5	-139.2
2	1431.94	2630.23	1.5	-138.7
3	1431.94	2642.23	1.5	-139.5
4	1431.94	2656.23	1.5	-139.2
5	1429.94	2670.23	1.5	-137.5
6	1427.94	2682.23	1.5	-135.2
7	1426.94	2691.23	1.5	-135.6

...

3. Output files

3.1 Propagation map file

Located in the same folder as Wavesight executable. The prediction file contains prediction for the rectangular area defined in the frame file.

Three types of format are available according to the flag OutFileFormat in the comp.txt file

3.1.1 Low resolution ASCII

It is named mapSTR.txt, where STR is the argument of -ctxt option as described in section SS. In case no argument is specified the name of the file will be map.txt. The number.

Propagation map files are in the following format:

- The first line is X,Y,Power
- Each of the following lines consists of a location on the propagation frame and power level in dBm

Note that only one location out of four is stored to reduce the amount of disk storage

EXAMPLE

X,Y,Power

601282.950000,200411.800000,-116.456726

601282.950000,200421.800000,-122.996704

3.1.2 High resolution ASCII

It is named mapallSTR.txt, where STR is the argument of -ctxt option as described in section YY. In case no argument is specified the name of the file will be mapall.txt.

Propagation map files are in the following format:

It is named mapSTR.txt, where STR is the argument of -ctxt option as described in section YY. In case no argument is specified the name of the file will be map.txt. The number.

Propagation map files are in the following format:

- The first line is X,Y,Power
- Each of the following lines consist of a location on the propagation frame and power level in dBm

EXAMPLE

X,Y,Power

601282.950000,200411.800000,-116.456726

601282.950000,200421.800000,-122.996704

3.1.3 Binary

It is named mapSTR.bin, where STR is the argument of -ctxt option as described in section VV. In case no argument is specified the name of the file will be map.bin. The number.

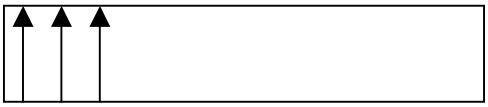
Propagation map files are in the following format:

The prediction is stored in a binary format as on array where each element of the represents the prediction in dBm. Each element is one byte (-127,+127 dBm) in size. The first element represents the predictions at the (EastMin, NorthMin) corner of the computation area

The number of bytes in the file is

((EastMax - EastMin) x (NorthMax - NorthMin)) /(res^2)

The values are stored as follows:

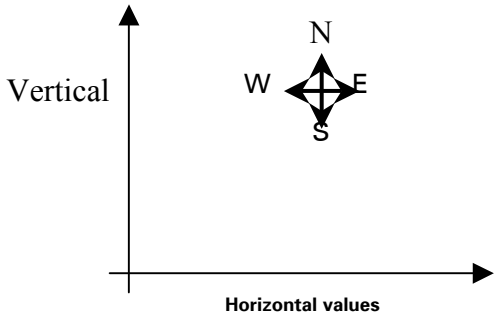


3.2 Route prediction file (route.txt)

Located in the same folder as Wavesight executable. It is generated only if a measurement route is indicated in the infiles.txt file (sect. 2.2). It is named routeSTR.out, where STR is the argument of -ctxt option as described in section TT. In case no argument is specified the name of the file will be route.out. The number. Field strength is computed on the receiver locations listed in the measurement route file.

4. Remarks

- The geographical co-ordinates used in buildings and terrain files are:



- Co-ordinates values are in meters, ASCII-encoded as integers or floating point numbers.
- The ASCII files are plain text files, with DOS or UNIX newlines.
- Separating blanks consist of one or more of the following characters: whitespace, tab, DOS newline or UNIX newline.