

Table of Contents

1 File types.....	2
1.1 Waypoint file.....	2
1.2 Set file.....	2
1.3 Route file.....	2
1.4 MultiArea file.....	2
1.5 MultiTrack file.....	3
1.6 Primitives.....	3
1.7 Structures.....	3
1.7.1 Structure {Location}.....	3
1.7.2 Structure {LocationValue}.....	3
1.7.2.1 Types of values.....	3
1.7.3 Structure {Locations}.....	4
1.7.4 Structure {Waypoint}.....	4
1.7.5 Structure {Waypoints}.....	4
1.7.6 Structure {AreaPolygons}.....	4
1.7.7 Structure {AreaPolygon}.....	4
1.7.8 Structure {TrackSegments}.....	4
1.7.9 Structure {TrackSegment}.....	4
1.7.10 Structure {Metadata}.....	5
1.7.11 Structure {MetadataContent}.....	5
1.7.12 Structure {MetadataEntry}.....	5
1.7.12.1 Types of data.....	5
1.7.13 Structure {MetadataExtensions}.....	5
1.7.14 Structure {MetadataExtension}.....	5

1 File types

	File extension	File version (supported since)
Waypoint	.wpt	1 (OM 3.7 / AQ 2.2.8)
Set	.set	1 (OM 3.7 / AQ 2.2.8)
Route	.rte	1 (OM 3.7 / AQ 2.2.8)
Area	.are	1 (OM 3.7 / AQ 2.2.8)
Track	.trk	1 (OM 3.7 / AQ 2.2.8)

1.1 Waypoint file

Type	Description
int	file magic (3 bytes, <code>0x50500A</code>) + version (1 byte)
int	header size (bytes before {Waypoint})
{Waypoint}	waypoint

1.2 Set file

Type	Description
int	file magic (3 bytes, <code>0x50500B</code>) + version (1 byte)
int	header size (bytes before {Waypoints})
{Metadata}	technical metadata
{Metadata}	user metadata
{Waypoints}	waypoints

1.3 Route file

Type	Description
int	file magic (3 bytes, <code>0x50500C</code>) + version (1 byte)
int	header size (bytes before {Waypoints})
{Metadata}	technical metadata
{Metadata}	user metadata
{Waypoints}	waypoints

1.4 MultiArea file

Type	Description
int	file magic (3 bytes, <code>0x50500D</code>) + version (1 byte)
int	header size (bytes before {AreaPolygons})
{Metadata}	technical metadata
{Metadata}	user metadata
{AreaPolygons}	polygons

1.5 MultiTrack file

Type	Description
int	file magic (3 bytes, 0x50500E) + version (1 byte)
int	header size (bytes before {Waypoints})
{Metadata}	technical metadata
{Metadata}	user metadata
{Waypoints}	waypoints
{TrackSegments}	segments

1.6 Primitives

Type	Size (byte)	Format
byte	1	signed
int	4	big endian, signed
long	8	big endian, signed
double	8	IEE 754 encoding
string	variable	UTF8 encoding, starts with an int giving the string length in bytes

1.7 Structures

1.7.1 Structure {Location}

Type	Description
int	structure size (bytes)
int	WGS84 longitude (degrees*1e7)
int	WGS84 latitude (degrees*1e7)
{LocationValue} ⁿ	additional values, until structure size is reached

1.7.2 Structure {LocationValue}

Type	Description
byte	value type (see table below)
variable	value data, length depends on the value type

1.7.2.1 Types of values

Value type	Data type	Description
0x61	int	accuracy (meters*1e2)
0x62	byte	battery level (0-100%) (added in OM 3.8b / AQ 2.2.9b)
0x65	int	WGS84 elevation (meters*1e3)
0x6e	2*byte	cell network info: cell type in byte 1 (generation in tens, protocol in units), signal strength in byte 2 (from 1=BAD to 127=GOOD) (added in OM 3.8b / AQ 2.2.9b)
0x70	int	pressure (hpa*1e3)
0x73	8*byte	satellites in use per constellation (UNKNOWN, GPS, SBAS, GLONASS, QZSS, BEIDOU, GALILEO, IRNSS) (added in OM 3.8b / AQ 2.2.9b)
0x74	long	UTC time (millisecond)
0x76	int	vertical accuracy (meters*1e2) (added in OM 3.10b / AQ 2.3.2c)

1.7.3 Structure {Locations}

Type	Description
int	number of locations
{Location} ⁿ	locations

1.7.4 Structure {Waypoint}

Type	Description
{Metadata}	user metadata
{Location}	location

1.7.5 Structure {Waypoints}

Type	Description
int	number of waypoints
{Waypoint} ⁿ	waypoints

1.7.6 Structure {AreaPolygons}

Type	Description
int	number of polygons
{AreaPolygon} ⁿ	polygons

1.7.7 Structure {AreaPolygon}

Type	Description
{Metadata}	user metadata
{Locations}	locations
int	number of holes
{Locations} ⁿ	holes

1.7.8 Structure {TrackSegments}

Type	Description
int	number of segments
{TrackSegment} ⁿ	segments

1.7.9 Structure {TrackSegment}

Type	Description
{Metadata}	user metadata
{Locations}	locations

1.7.10 Structure {Metadata}

Type	Description
{MetadataContent}	main metadata content
[MetadataExtensions]	optional metadata extensions, only if number of main metadata content entries is not -1

1.7.11 Structure {MetadataContent}

Type	Description
int	number of metadata entries, or -1 if none
{MetadataContentEntry} ⁿ	entries
[int]	metadata version (automatically increased on any entry changes), only if number of metadata entries is not -1

1.7.12 Structure {MetadataEntry}

Type	Description
string	entry name
int	type of entry
[int]	size of data (optional, only if type of entry is -4)
[variable]	data (size depends on the type of entry)

1.7.12.1 Types of data

Type of entry	Type of data	Size of data (byte)
-1	boolean	1
-2	long	8
-3	double	8
-4	raw data	[see size of data]
>=0	string	[see type of entry]

1.7.13 Structure {MetadataExtensions}

Type	Description
int	number of metadata extensions, or -1 if none
{MetadataExtension} ⁿ	metadata extensions

1.7.14 Structure {MetadataExtension}

Type	Description
string	name of extension
{MetadataContent}	extension