New BGL File structure

This is an update of my first attempt to understand the file structure of the new FS2004 scenery files. It is still incomplete, since I do not understand all the features.

© Winfried Orthmann
eMail: winfriedorthmann@yahoo.com

BGL Files Overview

FS 2004 BGL-files in the new format

File Name	Contents	Sections
AP*.BGL	Airports	including objects coded within the airport records thru VisualModel and TaxiwaySign subrecords
AT*.BGL	Waypoints and boundaries	
NV*.BGL	Navaids	
OB*.BGL	Airport objects	including .mdl data
[city name].BGL	city objects	including .mdl data

FS 2004 BGL-files in the old format

File Name	Contents
AB*.BGL	Terrain Data : Airport Background
BR*.BGL	Terrain Data : Bridges
FL*.BGL	Terrain Data : Airport flattens
HL*.BGL	Terrain Data : Coastlines
HP*.BGL	Terrain Data : Land/Water masks
PK*.BGL	Terrain Data : parks
RD*.BGL	Terrain Data : roads
RR*.BGL	Terrain Data : railroads
ST*.BGL	Terrain Data
UT*.BGL	Terrain Data : utilities (poles etc)

Data types

Latitude and longitude are no longer represented as before. Each location on the earth is fixed in the LOD grid. Longitude and latitude are each represented by a 4 byte value (DWORD). The formula for obtaining the decimal values is as follows:

```
(double) Lon = (DWORD) Lon * (360.0 / (3 * 0x10000000) – 180.0
(double) Lat = 90.0 - (DWORD) Lat * (180.0 / (2 * 0x10000000)
```

Altitude is given in 1/1000 m as DWORD.

Pitch, bank and heading: is given as ANGLE16 in form of a DWORD. The formula for obtaining the decimal value is as follows:

```
(double) Pitch = (DWORD) Pitch * 360.0 / 0x10000
```

ICAO Identifiers and region codes are coded in a special format. Each number and letter has a value from 0 .. 37:

blank	00
digits 0 .. 9	02 .. 11
letters A .. Z	12 .. 37

The code is calculated by starting from left: the value of the first digit/letter is multiplied by 38, then the value of the next digit/letter to the right is added, the sum s multiplied by 38, and as long as there are more digits/letters this process is repeated.
The region codes have only 2 digits/letters and the result is used as such; for the ICAO identifiers for airports, ILS, VOR, NDB and waypoints there are up to 5 digits/letters, and the result is shifted left by 5 positions, i.e. multiplied by 0x20. Bits 0 .. 4 of the resulting DWORD are frequently used for other purposes.
The ICAO identifiers for primary and secondary ILS in a runway record are not shifted.

BGL file header

The new BGL file header consists of a fixed part with the length of 0x38 (54) bytes and a variable number of section pointers.

The fixed part of the header has the following structure:

offset	length	format	description	contents
0	2	WORD	New bgl ID	0x0201
2	2	WORD	Probably version	0x1992
4	4	DWORD	size of header	0x0034
8	12	DWORD[3]	Unknown, possibly connected to compilation time	
20	4	DWORD	number of section pointers in header	
			rest unknown	

Each section pointer is 20 bytes long and has the following structure

offset	length	format	description	contents
0	4	DWORD	type of section. The following types have been identified: 0x0003: airport data 0x0013: VOR / ILS data 0x0017: NDB data 0x0018: markers 0x0020: boundary data 0x0022: waypoint data 0x0023: geopol data 0x0025: scenery objects 0x0027: namelist 0x002b: mdl data 0x002c: additional airport data 0x002e: exclusionRectangle	
4	4	DWORD	unknown	
8	4	DWORD	number of subsection pointers in section header	
12	4	DWORD	offset from file start to section header	
16	4	DWORD	size of section header	

BGL section header

The section pointer records in the header point to the section header which consist of 1..n subsection pointer records. The number of subsection pointer records present is given in the section pointer record.
Each subsection pointer record is 16 bytes long and has the following structure:

offset	length	format	description	contents
0	4	DWORD	ID. Since some of the sections are apparently subdivided into subsections according to the location of the objects in the LOD system, this ID seems to be an index giving the location of the object (not yet understood)	
4	4	DWORD	number of records in the subsection	
8	4	DWORD	offset from file start to start of object records in this subsection	
12	4	DWORD	size of subsection	

The section header for records of Boundary and Geopol type have a different structure. They consist of a 16 bytes long record for every subsection with the following structure:

offset	length	format	description	contents
0	4	DWORD	ID. Since some of the sections are apparently subdivided into subsections according to the location of the objects in the LOD system, this ID seems to be an index giving the location of the object (not yet understood)	
4	4	DWORD	Number of records in the subsection	
8	4	DWORD	Index into the list following these records	
12	4	DWORD	unknown, seems always to contain	0x00000000

after this list follows a record for every subsection with the following structure, which is repeated for the number of records, i.e. if there are 2 records in the subsection, there will be two records of the following structure:

offset	length	format	description	contents
0	4	DWORD	offset from start of file to start of records	
4	4	DWORD	length of subsection	

BGL subsections

The subsections for each kind of objects (airports, sceneryObjects, ILS etc) consist of a list with the individual records following each other. Each record has at offset 2 a DWORD giving the total size of this record. Thus it is easy to find the start of the next record. Each section and thus each subsection contains records of the same general type. A number of records can contain subrecords, which in turn have a size field at offset 2 after a WORD identifying the type of subrecord.

Airport

Each airport record consists of a fixed part with the length of 52 bytes, followed by a variable part with 0..n subrecords of different types. The structure of fixed part is as follows:

offset	length	format	description	contents
0	2	WORD	ID	0x0003
2	4	DWORD	size of airport record	
6	1	BYTE	number of runways subrecords	
7	1	BYTE	number of com subrecords	
8	1	BYTE	number of start subrecords	
9	1	BYTE	number of approach subrecords (?)	
10	1	BYTE	Bit 0-6: number of aprons (?) Bit 7: flag for deleteAirport record	
11	1	BYTE	number of helipad subrecords	
12	4	DWORD	longitude	
16	4	DWORD	latitude	
20	4	DWORD	altitude in m	
24	4	DWORD	longitude of tower (if present)	
28	4	DWORD	latitude of tower (if present)	
32	4	DWORD	altitude of tower (if different from airport)	
36	4	float	magnetic variation	
40	4	DWORD	ICAO ident (special format)	
44	4	DWORD	unknown	
48	4	DWORD	unknown	

The following subrecords can be present after the main **airport** record:

NAME

offset	length	format	description	contents
0	2	WORD	ID	0x0019
2	4	DWORD	Size of name subrecord	
6		STRING	airport name	

RUNWAY

The runway subrecord consists of a fixed part with a length of 52 byte and a variable number of sub-subrecords. The fixed part has the following structure;

offset	length	format	description	contents
0	2	WORD	ID	0x0004
2	4	DWORD	size of runway subrecord	
6	2	WORD	type of surface. The following numbers have been found: 0x0000 CONCRETE; 0x0001 GRASS; 0x0002 WATER; 0x0004 ASPHALT; 0x0007 CLAY; 0x0008 SNOW;	

			0x0009 ICE; 0x000c DIRT; 0x000d CORAL; 0x000e GRAVEL ; 0x000f OIL_TREATED;0x0010 STEEL_MATS; 0x0011 BITUMINOUS; 0x0012 BRICK; 0x0013 MACADAM; 0x0014 PLANKS; 0x0015 SAND; 0x0016 SHALE; 0x0017 TARMAC; 0x00fe UNKNOWN;	
8	1	BYTE	primary runway number (01 – 36, then 37ss. for NORTH, NORTHEAST, EAST, …	
9	1	BYTE	primary runway designator 0 = NONE, 1 = LEFT, 2 = RIGHT, 3 = CENTER, 4 = WATER	
10	1	BYTE	secondary runway number	
11	1	BYTE	secondary runway designator	
12	4	DWORD	ICAO ident. for primary ILS (special format), 0x0000 if none	
16	4	DWORD	ICAO ident. for secondary ILS	
20	4	DWORD	longitude	
24	4	DWORD	latitude	
28	4	DWORD	elevation	
32	4	float	length in m	
36	4	float	width in m	
40	4	float	heading	
44	4	float	pattern altitude	
48	2	WORD	marking flags: BIT 0: edges; BIT 1: threshold BIT 2: fixedDistance BIT 3: touchdown BIT 4: dashes BIT 5: ident BIT 6: precision BIT 7: edgePavement BIT 8: singleEnd Bit 9: primaryClosed BIT 10: second.Closed BIT 11: primaryStol BIT 12: secondaryStol Bit 13-15 unused (?)	
50	1	BYTE	light flags: BIT 0-1: edge (00 none, 01 low, 10 medium, 11 high) BIT 2-3: center (as with edge) BIT 5: flag for centerRed BIT 5-7: unused (?)	
51	1	BYTE	pattern flags: BIT 0: primaryTakeoff (0 = YES) BIT 1: primaryLanding (0 = YES) BIT 2: primaryPattern (0 = LEFT) BIT 3: secondaryTakeoff BIT 4: secondaryLanding BIT 5: secondaryPattern BIT 6-7: unused (?)	

The following sub-subreports con be present within a runway subrecord:

OffsetThreshold

offset	**length**	**format**	**description**	**contents**
0	2	WORD	ID primary: secondary	0x0005 0x0006
2	4	DWORD	Size of sub-subrecord	0x0010
6	2	WORD	surface (same as in runway)	
8	4	float	length in m	
12	4	float	width in m	

BlastPad

offset	length	format	description	contents
0	2	WORD	ID primary: secondary	0x0007 0x0008
2	4	DWORD	Size of sub-subrecord	0x0010
6	2	WORD	surface (same as in runway)	
8	4	float	length in m	
12	4	float	width in m	

Overrun

offset	length	format	description	contents
0	2	WORD	ID primary: secondary	0x0009 0x000a
2	4	DWORD	Size of sub-subrecord	0x0010
6	2	WORD	surface (same as in runway)	
8	4	float	length in m	
12	4	float	width in m	

VASI

offset	length	format	description	contents
0	2	WORD	ID primary left : primary right: secondary left: secondary right:	0x000b 0x000c 0x000d 0x000e
2	4	DWORD	Size of sub-subrecord	0x0018
6	2	WORD	type 0x01 = VASI21 0x02 = VASI31 0x03 = VASI22 0x04 = VASI32 0x05 = VASI23 0x06 = VASI33 0x07 = PAPI2 0x08 = PAPI4 0x09 = TRICOLOR 0x0a = PVASI 0x0b = TVASI 0x0c = BALL 0x0d = APAP/PANELS	
8	4	float	biasX	
12	4	float	biasZ	
16	4	float	spacing	
20	4	float	pitch	

ApproachLights

offset	length	format	description	contents
0	2	WORD	ID primary: secondary	0x000f 0x0010
2	4	DWORD	Size of sub-subrecord	0x0008
6	1	BYTE	system 0x00 = NONE 0x01 = ODALS 0x02 = MALSF 0x03 = MALSR 0x04 = SSALF 0x05 = SSALR 0x06 = ALSF1 0x07 = ALSF2 0x08 = RAIL 0x09 = CALVERT 0x0a = CALVERT2 0x0b = MALS 0x0c = SALS 0x0e = SSALS	
7	1	BYTE	number of strobes	

(end of runway)

HELIPAD

offset	length	format	description	contents
0	2	WORD	ID	0x0026
2	4	DWORD	Size of helipad subrecord	0x0024
6	1	BYTE	surface (as with runway)	
7	1	BYTE	bit 0-3: type 0 = NONE 1 = H 2 = SQUARE 3 = CIRCLE 4 = MEDICAL bit 4: transparent bit 5: closed bit 6-7: unused	
8	4	BYTE[4]	color (cannot be set with bglcomp because of error kin compiler)	
12	4	DWORD	longitude	
16	4	DWORD	latitude	
20	4	DWORD	altitude * 1000	
24	4	float	length	
28	4	float	width	
32	4	float	heading	

START

(the keywords “Start” and “RunwayStart” produce identical subrecords)

offset	length	format	description	contents
0	2	WORD	ID	0x0011
2	4	DWORD	Size of start subrecord	0x0018
6	1	BYTE	runway number	
7	1	BYTE	bit 0-3:runway designator (as with runway subrecord) bit 4-7: start type 1 = RUNWAY 2 = WATER 3 = HELIPAD	
8	4	DWORD	longitude	
12	4	DWORD	latitude	
16	4	DWORD	elevation	
20	4	float	heading	

COM

offset	length	format	description	contents
0	2	WORD	ID	0x0012
2	4	DWORD	Size of subrecord: variable	
6	2	WORD	type. The following numbers have been identified: 0x0001 ATIS 0x0002 MULTICOM 0x0003 UNICOM 0x0004 CTAF 0x0005 GROUND 0x0006 TOWER 0x0007 CLEARANCE 0x0008 APPROACH 0x0009 DEPARTURE 0x000a CENTER 0x000b FSS 0x000c ASOS	
8	4	DWORD	frequency	
12	variable	STRINGZ	name	

DELETEAIRPORT

The DeleteAirport subrecord has a fixed and a variable part. The fixed part has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0033
2	4	DWORD	Size of subrecord: variable	
6	2	WORD	delete flags BIT 0: allApproaches BIT 1: allApronLights (Note: in the bglcomp.xsd this keyword is written allApronlights, but the compiler accepts only allApronLights. You have to edit bglcomp.xsd, if you want to use this feature) BIT 2: allAprons BIT 3: allFrequencies BIT 4: allHelipads BIT 5: allRunways BIT 6: allStarts BIT 7: allTaxiways	
8	1	BYTE	number of individual runways to delete	
9	1	BYTE	number of individual starts to delete	
10	1	BYTE	number of frequencies to delete	
11	1	BYTE	unused (?)	

according to the number of individual features to delete there are the following parts of the record added:
for runways:

offset	length	format	description	contents
0	1	BYTE	surface (as in runway subrecord)	
1	1	BYTE	runway number primary	
2	1	BYTE	runway number secondary	
3	1	BYTE	bit 0-3: runway designator primary bit 4-7: runway designator secondary	

for starts:

offset	length	format	description	contents
0	1	BYTE	runway number	
1	1	BYTE	runway designator	
2	1	BYTE	type of start 1 = RUNWAY, 2 = WATER, 3 = HELIPAD	
3	1	BYTE	unused (?)	0x00

for frequencies

offset	length	format	description	contents
0	4	DWORD	bit 28-31: type (as with COM records) bit 0-27: frequency * 1000000	

APRON

There are 2 subrecords for each apron which follow each other. Both have variable length.
First record:

offset	length	format	description	contents
0	2	WORD	ID	0x0037
2	4	DWORD	size	

6	1	BYTE	surface (as with runway subrecord)	
7	2	WORD	number of vertices	
			and then for each vertex:	
	4	DWORD	longitude	
	4	DWORD	latitude	
			and then	
			zero-fill to next DWORD boundary	

second record:

offset	length	format	description	contents
0	2	WORD	ID	0x0030
2	4	DWORD	size	
6	1	BYTE	surface (as in first record)	
7	1	BYTE	flags: bit 0: drawSurface bit 1: drawDetail	
8	2	WORD	number of vertices	
10	2	WORD	number of triangles to draw	
			and then for each vertex	
	4	DWORD	longitude	
	4	DWORD	latitude	
			and then for each triangle to draw	
	2	WORD	index of first point	
	2	WORD	index of second point	
	2	WORD	index of third point	

APRONEDGELIGHTS

offset	length	format	description	contents
0	2	WORD	ID	0x0031
2	4	DWORD	size	
6	2	WORD	unknown	
8	2	WORD	number of vertices	
10	2	WORD	number of edges	
12	4	DWORD	unknown, probably color of lights	0xff0000ff
16	4	float	unknown (value 1)	0x3f800000
20	4	float	unknown (value 800)	0x44480000
			and then for each vertex	
	4	DWORD	longitude	
	4	DWORD	latitude	
			end then for each edge	
	4	float	unknown (value 60.96)	
	2	WORD	index of start vertex	
	2	WORD	index of end vertex	

TAXIWAYPOINT

All taxiway points are joined in one record, which has a fixed part of 8 bytes and a variable part with 12 bytes for each point. Structure of the fixed part:

offset	length	format	description	contents
0	2	WORD	ID	0x001A
2	4	DWORD	size : variable	
6	2	WORD	number of taxiway points present	

and for each taxipoint:

0	1	BYTE	type: 1 = NORMAL, 2 = HOLD_SHORT 4 = ILS_HOLD_SHORT	

1	1	BYTE	flag: 0 = FORWARD, 1 = REVERSE	
2	1	WORD	unknown	0x0000
4	4	DWORD	longitude	
8	4	DWORD	latitude	

TAXIWAYPARKING

This record type has a short fixed part for all TaxiwayParking records together and a longer variable part with sections for each TaxiwayParking. The fixed part is 8 bytes long:

offset	length	format	description	contents
0	2	WORD	ID	0x001B
2	4	DWORD	size : variable	
6	2	WORD	number of taxiway parking records present	

The record sections for each TaxiwayParking are again of variable length, depending on the number of airlineCodes present:.

0	4	DWORD	bit 31-24: count of airlineCodes present bit 23-12: number bit 11-8: type 0x1 = RAMP_GA 0x2 = RAMP_GA_SMALL 0x3 = RAMP_GA_MEDIUM 0x4 = RAMP_GA_LARGE 0x5 = RAMP_CARGO 0x6 = RAMP_MIL_CARGO 0x7 = RAMP_MIL_COMBAT 0x8 = GATE_SMALL 0x9 = GATE_MEDIUM 0xa = GATE_HEAVY 0xb = DOCK_GA bit 7-6: pushback (00 = none, 01 = left, 10 = right, 11 = both) bit 5-0: name 0x00 = NONE, 0x01 = PARKING, 0x02 = N_PARKING, 0x03 = NE_PARKING 0x04 = E_PARKING, 0x05 = SE_PARKING 0x06 = S_PARKING, 0x07 = SW_PARKING 0x08 = W_PARKING, 0x09 = NW_PARKING 0x0a = GATE, 0x0b = DOCK, 0x0c = GATE_A, 0x0d = GATE_B, 0x0e = GATE_C 0x0f = GATE_D 0x10 = GATE_E .. .. 0x25 = GATE_Z
4	4	float	radius
8	4	float	heading
12	4	DWORD	longitude
16	4	DWORD	latitude
..	4	STRING	airline designator (0..n times repeated)

TAXIWAYPATH

This record has a fixed length of 8 byte and a variable part with records for each path. It has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x001C

2	4	DWORD	size	
6	2	WORD	number of paths defined	
			and then for each path:	
0	2	WORD	index of start point NB: for type TAXI, the index of the start and of the end must both refer to a TaxiPoint. For type PARKING the start index must refer to a TaxiPoint, the end index must refer to a TaxiwayParking.	
2	2	WORD	Bit 0-11: index of end point	
			Bit 12-15: runway designator	
4	1	BYTE	type 61 = TAXI 62 = RUNWAY 63 = PARKING 04 = PATH 65 = CLOSED	
5	1	BYTE	runway number / index into TaxiName	
6	1	BYTE	bitfield BIT 0: centerline BIT 1: centerLineLighted BIT 2-3: leftEdge (00 = NONE, 01 = SOLID, 10 = DASHED, 11 = SOLID_DASHED) BIT 4: leftEdgeLighted BIT 5-6: rightEdge BIT 7: rightEdgeLighted	
7	1	BYTE	surface	
8	4	float	width	
12	4	DWORD	weightLimit	
16	4	DWORD	??	

TAXINAME

This record has variable length, it consist of 8 bytes as a fixed part and then 8 bytes for each Name

offset	length	format	description	contents
0	2	WORD	ID	0x001D
2	4	DWORD	size : variable	
6	2	WORD	number of name entries	
			and then for each name	
	8	STRING	taxiName	

TAXIWAYSIGN

These record are coded in the section for scenery objects (0x25) with a separate type of entry. Apparently all Taxiway signs for one airport are coded together on one record. There seems to be no coordination of this record with the airport record to which it belongs!

offset	length	format	description	contents
0	2	WORD	ID	0x0005
2	2	WORD	size : variable	
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude (?) cannot be coded with the compiler	
16	2	WORD	altitudeIsAGL cannot be coded	0x0001
18	2	WORD	pitch (?) cannot be coded	
20	2	WORD	bank (?) cannot be coded	
22	2	WORD	(heading) (?) cannot be coded	
24	2	WORD	imageComplexity (?) cannot be coded	
26	2	WORD	unknown	

28	4	DWORD	number of taxiway signs for this airport	
			and then for each sign	
0	4	float	longitude offset from value in main record	
4	4	float	latitude offset from value in main record	
8	2	WORD	heading as coded	
10	1	BYTE	Size (SIZE1 .. SIZE5)	
11	1	BYTE	justification (1 = right, 2 = left)	
12	var	STRINGZ	label (zero filled to next WORD address)	

APPROACH

offset	length	format	description	contents
0	2	WORD	ID for Approach	0x0024
2	4	DWORD	size : variable	
6	1	BYTE	suffix	
7	1	BYTE	runway number	
8	1	BYTE	bit 0-3: type 0x01 = GPS 0x02 = VOR 0x03 = NDB 0x04 = ILS 0x05 = LOCALIZER 0x06 = SDF 0x07 = LDA 0x08 = VORDME 0x09 = NDBDME 0x0a = RNAV 0x0b = LOCALIZER_BACKCOURSE bit 4-6: runway designator bit 7: gpsOverlay flag	
9	1	BYTE	number of transitions ?	
10	1	BYTE	number of approach legs	
11	1	BYTE	number of missedApproach legs ?	
12	4	DWORD	fixIdent BIT 0-4: fixType 02 = VOR 03 = NDB 04 = TERMINAL_NDB 05 = WAYPOINT 06 = TERMINAL_WAYPOINT 09 = RUNWAY BIT 5-31 fixIdent	
16	4	DWORD	bit 0-10: fixRegion bit 11-31: ICAO Id of relevant airport	
20	4	float	altitude	
24	4	float	heading	
28	4	float	missedAltitude	

after this the following record can occur

offset	length	format	description	contents
0	2	WORD	ID for ApproachLegs	0x002D
2	4	DWORD	size : variable	
6	2	WORD	number of legs to follow	

each leg is a structure with a fixed length of 44 bytes

offset	length	format	description	contents
0	1	BYTE	ID of the leg types found: 0x01 = AF 0x02 = CA 0x03 = CD 0x04 = CF 0x05 = CI 0x06 = CR 0x07 = DF 0x08 = FA 0x09 = FC 0x0a = FD 0x0b = FM 0x0c = HA 0x0d = HF 0x0e = HM 0x0f = IF 0x10 = PI 0x11 = RF 0x12 = TF 0x13 = VA 0x14 = VD 0x15 = VI	

			0x16 = VM 0x17 = VR	
1	1	BYTE	altitudeDescriptor 01 = A 02 = + 03 = - 04 = B	
2	2	WORD	flags: bit 0: turnDirection = L bit 1: turnDirection = R bit 8: magneticCourse (0) trueCourse (1) bit 9: distance (0) or time (1) bit 10: flyover false (0) true (1)	
4	4	DWORD	bit 5-31: fixIdent bit 0-4: fixType	
8	4	DWORD	bit 0-10: fixRegion bit 11-32: ICAO Id of relevant airport	
12	4	DWORD	bit 5-31: recommendedIdent bit 0-4: recommendedType	
16	4	DWORD	recommendedRegion	
20	4	float	theta	
24	4	float	rho	
28	4	float	trueCourse / magneticCourse (depending on flag)	
32	4	float	distance / time	
36	4	float	Altitude1	
40	4	float	Altitude2	

offset	length	format	description	contents
0	2	WORD	ID for missedApproachLegs	0x002E
2	4	DWORD	size : variable	
6	2	WORD	number of legs to follow	

offset	length	format	description	contents
0	2	WORD	ID for Transition	0x002C
2	4	DWORD	size : variable	
6	1	BYTE	transitionType 1 = FULL, 2 = DME	
7	1	BYTE	number of TransitionLegs (?)	
8	4	DWORD	bit 0-4: fixType 2 = VOR 3 = NDB 4 = TERMINAL_NDB 5 = WAYPOINT 6 = TERMINAL_WAYPOINT bit 5-31: fixIdent (spezial format)	
12	4	DWORD	bit 0-10: fixRegion bit 11-31 : airportID of relevant airport	
16	4	float	altitude	
			if transitionType = DME and DmeArc record exists, then the following 16 bytes are present	
20	4	DWORD	dmeIdent	
24	4	DWORD	bit 0-10: dmeRegion bit 11-31: airportID of relevant airport	
28	4	DWORD	radial	
32	4	float	distance	

offset	length	format	description	contents
0	2	WORD	ID for TransitionLegs (can follow only after transition	0x002F

2	4	DWORD	size : variable	
6	2	WORD	number of legs to follow	

WAYPOINT

The waypoint record can be part of the Airport group or can be entered independently. In both cases the output for the BGL is the same.

offset	length	format	description	contents
0	2	WORD	ID for Waypoint	0x0022
2	4	DWORD	size : variable	
6	1	BYTE	type 1 = NAMED, 2 = UNNAMED, 3 = VOR 4 = NDB, 5 = OFF_ROUTE, 6 = IAF 7 = FAF	
7	1	BYTE	number of Route entries to follow	
8	4	DWORD	longitude	
12	4	DWORD	latitude	
16	4	float	magvar	
20	4	DWORD	waypointIdent (special format)	
24	4	DWORD	bit 0-10: waypointRegion (special format) bi1 11-31: ICAO ident of the relevant airport, if it is a terminal waypoint, defined within an airport record	
			optional, if Route is given:	
28	1	BYTE	routeType (1 = VICTOR, 2 = JET, 3 = BOTH	
29	8	char[8]	name (zero padded), name cannot be longer than 8 characters	
			for Next:	
37	4	DWORD	BIT 0-2: type 2 = VOR, 3 = NDB, 5 = all other BIT 5-31: waypointIdent (special format)	
41	4	DWORD	Bit 0-10 waypointRegion (special format) BIT 11-31 airportId if terminal waypoint	
45	4	float	altitudeMinimum	
			for Previous:	
49	4	DWORD	type + waypointIdent (as for Next)	
51	4	DWORD	Bit 0-10 waypointRegion (special format) BIT 11-31 airportId if terminal waypoint	
55	4	float	altitudeMinimum	
			Note: it is not necessary for any route to have both previous and next defined, in that case the fields for this part of the record are all zero	

ILS / VOR

The records for ILS and VOR are in the same section and they are identical for the fixed section.
ILS records can have an additional subrecord
The fixed part is 40 bytes long and has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0013
2	4	DWORD	size	
6	1	BYTE	type. The following numbers have been found: 0x0001 VOR TERMINAL 0x0002 VOR LOW 0x0003 VOR HIGH 0x0004 ILS 0x0005 VOR VOT	
7	1	BYTE	flags. The following bits have been recognized: bit 0: if 0 then DME only bit 2: backcourse bit 3: glideslope present bit 4: DME present bit 5: NAV true	
8	4	DWORD	longitude	
12	4	DWORD	latitude	
16	4	DWORD	elevation	
20	4	DWORD	frequency	
24	4	float	range in m	
28	4	float	magnetic variation	
32	4	DWORD	ICAO ident (special format)	
36	4	DWORD	bit 0-10 regionId bit 11-31 airportId (for ILS)	

The following subrecords can follow:
(for ILS)

offset	length	format	description	contents
0	2	WORD	ID	0x0014
2	4	DWORD	size	0x0010
6	2	WORD	unknown	
8	4	float	heading	
12	4	float	width	

:
(for ILS)

offset	length	format	description	contents
0	2	WORD	ID glideslope	0x0015
2	4	DWORD	size	0x001c
6	2	word	unknown	
8	4	DWORD	longitude	
12	4	DWORD	latitude	
16	4	DWORD	elevation	
20	4	float	range	
24	4	float	pitch	

(for ILS/VOR)

offset	length	format	description	contents
0	2	WORD	ID DME	0x0016
2	4	DWORD	size	0x0018
6	2	WORD	unknown	
8	4	DWORD	longitude	

12	4	DWORD	latitude	
16	4	DWORD	elevation	
20	4	float	range	

After these subsections, a name subsection is added:

offset	length	format	description	contents
0	2	WORD	ID	0x0019
2	4	DWORD	size	
6		STRING	Name (max. 48 characters)	

if VisualModel is added in the source file, the compiler adds another section to the file with a record of type 0x0025 (SceneryxObject) with the GUID for the object referenced. The coordinates for this objects are taken from the ILS/VOR and adjusted, if BiasXYZ is added to the VisualModel.

NDB

The NDB records are stored in a separate section. The have a 40 bytes long fixed section and a name section of variable length. The fixed section has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0017
2	4	DWORD	size variable	
6	2	WORD	Type 0 = COMPASS_POINT 1 = MH 2 = H 3 = HH	
8	4	DWORD	frequency	
12	4	DWORD	longitude	
16	4	DWORD	latitude	
20	4	DWORD	elevation	
24	4	float	range	
28	4	float	magnetic variation	
32	4	DWORD	ICAO ident (special format)	
36	4	DWORD	bit 0-10: region bit 11-31: ICAO id of airport, if it was defined with an airport (terminal NDB)	

The name subsection has the following structure

offset	length	format	description	contents
0	2	WORD	ID	0x0019
2	4	DWORD	size	
6		STRING	name	

SceneryObject

LIBRARYOBJECT

The record has a fixed length of 48 byte with the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0002
2	2	WORD	size	0x0030
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude	
16	2	WORD	flag: 1 = isAboveAGL	
18	2	WORD	pitch	
20	2	WORD	bank	
22	2	WORD	heading	
24	2	WORD	imageComplexity 0 = VERYSPARSE 1 = SPARSE 2 = NORMAL 3 = DENSE 4 = VERYDENSE	
26	2	WORD	unknown	
28	16	GUID	name	
44	4	float	scale	

if an AttachedObject exists, there are three other records following:

offset	length	format	description	contents
0 = 48	2	WORD	ID	0x0010
2 = 50	2	WORD	unknown, maybe size	0x0004
			and then 2nd record	
0 = 52	2	WORD	ID	0x0008
2 = 54	2	WORD	size	
4 = 56	2	WORD	unknown (maybe offset of attach point string	0x001c
6 = 58	2	WORD	pitch	
8 = 60	2	WORD	bank	
10 = 62	2	WORD	heading	
12 = 64	12	DWORD[3]	unknown, possibly longitude, latitude, altitude if there is a bias? (Note: against the xml-scheme published, the compiler does not accept a bias for attachements, and in the MS scenery files I did not find a case where these fields were not zero)	
24 = 76	1	BYTE	type 0xf5 = CIVILIAN AIRPORT 0xf6 = CIVILIAN HELIPORT 0xf7 = CIVILIAN SEA_BASE 0xf8 = MILITARY AIRPORT 0xf9 = MILITARY HELIPORT 0xfa = MILITARY SEA_BASE	
25 = 77	1	BYTE	unknown always	0x01 (?)
26 = 78	2	WORD	unknown , always	0x0000
28		STRINGZ	name of attachment point	
			and then 3rd record	
0	2	WORD	ID	0x1001
2	2	WORD	size (?)	0x0004

EFFECT

The record has a fixed part of 108 byte and a variable part. The fixed part has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0004
2	2	WORD	size : variable	
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude	
16	2	WORD	flag: 1 = isAboveAGL	
18	2	WORD	pitch	
20	2	WORD	bank	
22	2	WORD	heading	
24	2	WORD	imageComplexity 0 = VERYSPARSE 1 = SPARSE 2 = NORMAL 3 = DENSE 4 = VERYDENSE	
26	2	WORD	unknown	
28	80	STRINGZ	effectName	
108	variable	STRINGZ	effectParams	

GENERICBUILDING

NB.: BuildingBias is not implemented in the compiler.

offset	length	format	description	contents
0	2	WORD	ID	0x0001
2	2	WORD	size : variable	
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude	
16	2	WORD	flag: 1 = isAboveAGL	
18	2	WORD	pitch	
20	2	WORD	bank	
22	2	WORD	heading	
24	2	WORD	imageComplexity 0 = VERYSPARSE 1 = SPARSE 2 = NORMAL 3 = DENSE 4 = VERYDENSE	
26	2	WORD	unknown	
28	4	float	scale	
32	2	WORD	type: 0x00a0 generic building	
34	2	WORD	size of record	
36	2	WORD	subtype. The following numbers have been identified: 0x0004 rectangular with roofType FLAT 0x0006 rectangular with roofType RIDGE 0x0007 rectangular with roofType PEAKED 0x0008 rectangular with roofType SLANT 0x0009 pyramidal building 0x000a multisidedBuilding	

for all rectangular buildings:

38	2	WORD	sizeX	0
40	2	WORD	sizeZ	1
42	2	WORD	bottomTexture	2

44	2	WORD	sizeBottomY	3
46	2	WORD	textureIndexBottomX	4
48	2	WORD	textureIndexBottomZ	5
50	2	WORD	windowTexture	6
52	2	WORD	sizeWindowY	7
54	2	WORD	textureIndexWindowX	8
56	2	WORD	textureIndexWindowY	9
58	2	WORD	textureIndexWindowZ	10
60	2	WORD	topTexture	11
62	2	WORD	sizeTopY	12
64	2	WORD	textureIndexTopX	13
66	2	WORD	textureIndexTopZ	14
68	2	WORD	roofTexture	15
70	2	WORD	textureIndexRoofX	16
72	2	WORD	textureIndexRoofZ	17

end for rectangular buildings with rooftype FLAT

for rectangular buildings with roofType RIDGE or SLANTED

74	2	WORD	sizeRoofY	18
76	2	WORD	textureIndexGableY	19
78	2	WORD	gableTexture	20
80	2	WORD	textureIndexGableZ	21

for roofType SLANTED only

82	2	WORD	faceTexture	22
84	2	WORD	textureIndexFaceX	23
86	2	WORD	textureIndexFaceY	24

for rectangular buildings with roofType PEAKED

74	2	WORD	sizeRoofY	18
76	2	WORD	textureIndexRoofY	19

for multisided buildings:

38	2	WORD	buildingSides. Note: The Argument for smoothing is required by the compiler, but it has no effect on the BGL-file	
40	2	WORD	sizeX	
42	2	WORD	sizeZ	
44	2	WORD	bottomTexture	
46	2	WORD	sizeBottomY	
48	2	WORD	textureIndexBottomX	
50	2	WORD	windowTexture	
52	2	WORD	sizeWindowY	
54	2	WORD	textureIndexWindoxX	
56	2	WORD	textureIndexWindowY	
58	2	WORD	topTexture	
60	2	WORD	sizeTopY	
62	2	WORD	textureIndexTopX	
64	2	WORD	roofTexture	
66	2	WORD	sizeRoofY	
68	2	WORD	textureIndexRoofX	
70	2	WORD	textureIndexRoofZ	
			Note: textureIndexRoofY is required by the compiler, but it has no effect on the bgl file !	

for pyramidal buildings

38	2	WORD	sizeX	0
40	2	WORD	sizeZ	1

42	2	WORD	sizeTopX	2
44	2	WORD	sizeTopZ	3
46	2	WORD	bottomTexture	4
48	2	WORD	sizeBottomY	5
50	2	WORD	textureIndexBottomX	6
52	2	WORD	textureIndexBottomZ	7
54	2	WORD	windowTexture	8
56	2	WORD	sizeWindowY	9
58	2	WORD	textureIndexWindowX	10
60	2	WORD	textureIndexWindowY	11
62	2	WORD	textureIndexWindowZ	12
64	2	WORD	topTexture	13
66	2	WORD	sizeTopY	14
68	2	WORD	textureIndexTopX	15
70	2	WORD	textureIndexTopZ	16
72	2	WORD	roofTexture	17
74	2	WORD	textureIndexRoofX	18
76	2	WORD	textureIndexRoofZ	19

WINDSOCK

Record with fixed length of 46 byte

offset	length	format	description	contents
0	2	WORD	ID	0x0003
2	2	WORD	size	0x002e
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude	
16	2	WORD	altitudeIsAGL (0x0001 = TRUE)	
18	2	WORD	pitch	
20	2	WORD	bank	
22	2	WORD	heading	
24	2	WORD	imageComplexity	
26	2	WORD	unknown	
28	4	float	poleHeight	
32	4	float	sockLength	
36	1	BYTE	PoleColor: blue	
37	1	BYTE	PoleColor:green	
38	1	BYTE	PoleColor:red	
39	1	BYTE	PoleColor ?	0xff
40	4	BYTE[4]	SockColor	
44	2	WORD	flag: lighted (TRUE = 0x0001)	

TRIGGER

The record consists of a fixed part and a variable part. The fixed part is 34 byte long and has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0007
2	2	WORD	size : variable	
4	4	DWORD	longitude	
8	4	DWORD	latitude	
12	4	DWORD	altitude	
16	2	WORD	altitudeIsAGL (0x00001 = TRUE)	
18	2	WORD	pitch	
20	2	WORD	bank	
22	2	WORD	heading	

24	2	WORD	imageComplexity	
26	2	WORD	unknown	
28	2	WORD	type (0x0000 = REFUEL_REPAIR, 0x0001 = WEATHER	
30	4	float	triggerHeight	
in case of WEATHER the variable part has the following structure				
34	2	WORD	type 0x0001 = RIDGE_LIFT 0x0002 = UNIDIRECTIONAL_TURBULENCE note: in bglcomp.xsd this keyword is spelled NONDIRECTIONAL_TURBULENCE, but the compiles does not understand it. If you change the keyword in bglcomp.xsd compilation is ok. 0x0003 = DIRECTIONAL_TURBULENCE 0x0004 = THERMAL	
36	4	float	heading	
40	4	float	scalar	
44	4	DWORD	number of vertices	
			and then for each vertex:	
	4	float	BiasX	
	4	float	BiasZ	
in case of FUEL_REPAIR the variable part has the following structure				
34	4	DWORD	fuel type and availability BITFIELD: bit 0-1: type 73 bit 2-3: type 87 bit 4-5: type 100 bit 6-7: type 130 bit 8-9: type 145 bit 10-11: type MOGAS bit 12-13: type JET bit 14-15: type JETA bit 16-17: type JETA1 bit 18-19: type JETAP bit 20-21: type JETB bit 22-23: type JET4 bit 24-25: type JET5 bit 26-29 : unused bit 30 : piston type bit 31 : jet type for all except last two : 0 = NO; 1 = UNKNOWN; 2 = PRIOR_REQUEST; 3 = YES when type=UNKNOWN and availability = YES then type=100 and type = JETA both are set to availability=YES	
38	4	DWORD	number of vertices	
			and then for each vertex	
	4	float	BiasX	
	4	float	BiasZ	

Marker

The marker record has a fixed length of 28 byte with the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0018
2	4	DWORD	size	0x0000001c
6	1	BYTE	heading	
7	1	BYTE	Type 0 = INNER; 1 = MIDDLE; 2 = OUTER 3 = BACKCOURSE	
8	4	DWORD	longitude	
12	4	DWORD	latitude	
16	4	DWORD	altitude	
20	4	DWORD	ident (special format)	
24	2	WORD	region (special format)	
26	2	word	unknown	0x0000

Boundary

offset	length	format	description	contents
0	2	WORD	ID	0x0020
2	4	DWORD	size : varying	
6	1	BYTE	type 00 = NONE 01 = CENTER 02 = CLASS_A 03 = CLASS_B 04 = CLASS_C 05 = CLASS_D 06 = CLASS_E 07 = CLASS_F 08 = CLASS_G 09 = TOWER 0a = CLEARANCE 0b = GROUND 0c = DEPARTURE 0d = APPROACH 0e = MOA 0f = RESTRICTED 10 = PROHIBITED 11 = WARNING 12 = ALERT 13 = DANGER 14 = NATIONAL_PARK 15 = MODEC 16 = RADAR 17 = TRAINING	
7	1	BYTE	BIT 0-3: maximumAltitudeType BIT 4-7: minimumAltitudeType 1 = MEAN_SEA_LEVEL (= UNKNOWN) 2 = ABOVE_GROUND_LEVEL 3 = UNLIMITED	
8	4	DWORD	minimum longitude of area covered	
12	4	DWORD	minimum latitude of area covered	
16	4	DWORD	minimumAltitude * 1000	
20	4	DWORD	maximum longitude of area covered	
24	4	DWORD	maximum latitude of area covered	
28	4	DWORD	maximumAltitude	
32	2	WORD	type field of name record	0x19
34	4	DWORD	size of name record	
36	size-6	STRING	name	

on this follows a record describing the drawing of the lines

offset	length	format	description	contents
0	2	WORD	ID	0x0021
2	4	DWORD	size : varying	
6	2	WORD	number of points to follow	
			for each point 10 bytes	
0	2	WORD	type of point 1 = START 2 = LINE 3 = ORIGIN 4 = ARC clockwise 5 = arc counter-clockwise 6 = circle NB: in case of circle, the entries for minimumAltitude and maximumAltitude over-ride the values in start if both are given. the start entry is in case of circle not needed at all Note: there is a bug in the new version of bglcomp.xsd: the word BoundaryStart in grpBoundaryChildren has to be replaced by Start, otherwise the compiler does not accept it!	
2	4	DWORD	latitude of point (in case of circle: unknown, = 0x0000)	
6	4	DWORD	longitude of point (in case of circle:	

			float: radius	

Geopol

fixed part:

offset	length	format	description	contents
0	2	WORD	ID	0x0023
2	4	DWORD	size : varying	
6	2	WORD	Bit 0-13: number of vertices number of vertices BIT 14-15: type (0x40 = BOUNDARY, 0x80 = COASTLINE)	
8	4	DWORD	minimum longitude	
12	4	DWORD	minimum latitude	
16	4	DWORD	maximum longitude	
20	4	DWORD	maximum latitude	

variable part: for each vertex

0	4	DWORD	longitude	
4	4	DWORD	latitude	

Model data

The model data structure has a fixed length of 24 bytes

offset	length	format	description	contents
0	16	GUID	name	
16	4	DWORD	mdl file offset from the start of this subsection	
20	4	DWORD	mdl file length	

ExclusionRectangle

This record has a fixed length record of 20 bytes

offset	length	format	description	contents
0	2	WORD	exclusion type 0x0008 = excludeAll otherwise: bit 8 = BeaconObjects bit 9 = EffectObjects bit 10 = GenericBuildingObjects bit 11 = LibraryObjects bit 12 = TaxiwaySignObjects bit 13 = TriggerObjects bit 14 = WindsockObjects	
2	2	WORD	size (unused)	0x0000
4	4	DWORD	longitude of NW corner	
8	4	DWORD	latitude of NW corner	
12	4	DWORD	longitude of SE corner	
16	4	DWORD	latitude of SE corner	

Namelist

The namelist contains only one record of variable length. It consists of a fixed part and a variable part. The fixed part is 42 bytes long and has the following structure:

offset	length	format	description	contents
0	2	WORD	ID	0x0027
2	4	DWORD	size (?) seems always to be 0x00000000	
6	2	WORD	number of region names	
8	2	WORD	number of country names	
10	2	WORD	number of state names	
12	2	WORD	number of city names	
14	2	WORD	number of airport names	
16	2	WORD	number of ICAO ident.	
18	4	DWORD	offset of region list (from start of record)	
22	4	DWORD	offset of country list	
26	4	DWORD	offset of state list	
30	4	DWORD	offset of city list	
34	4	DWORD	offset of airport list	
38	4	DWORD	offset of ICAO ident list	

The lists for region, country, state, city and airport names have all the same structure:

an index with 1 DWORD for each entry in the list, containing the offset of the nth name from the beginning of the names part (i.e. after the index) followed by the names in form of zero-terminated strings

The ICAO list has a different structure. It contains n entries (one for each ICAO name), each of them 20 bytes long, with the following structure;:

offset	length	format	description
0	1	BYTE	region name index (all indexes start with 0 for the first name in the relevant list)
1	1	BYTE	country name index
2	2	WORD	bit 4-15 : state name index bit 0-3 : unknown
4	2	WORD	city name index
6	2	WORD	airport name index
8	4	DWORD	ICAO identifier (special format)
12	4	DWORD	unknown
16	4	DWORD	unknown