

# 항공LiDAR 데이터의 관계형 DBMS 저장 및 관리방안 연구

## A Study of Store & Management of Airborne LiDAR Data

김호근\*, 권창희\*\*

Ho-Kun Kim\* and Chang-Hee Kwon\*\*

### 요 약

GIS기술의 발달로 인하여 과거와 같이 MicroStation과 같은 측량 장비를 이용하여 현지 측량 작업을 수행하지 않고도 항공기에 탑재된 LiDAR(Light Detection and Ranging)장비와 GPS를 이용하여 보다 더 효율적이고 정밀한 지도를 제작하게 되었다. 이러한 데이터는 그 양이 방대하기 때문에 관계형 데이터베이스를 이용하여 관리해야 할 필요성이 대두되고 있다. 본 연구에서는 취득한 LiDAR데이터를 데이터베이스에 저장하고 관리하기 위한 방안을 모색하고자 한다.

### Abstract

While in the past map-making process by field survey devices such as MicroStation needs more time relatively, we can make more precise map effectively with airborne LiDAR and GPS devices. Also the data, captured by LiDAR, are very large in size and so it needs to use Relational DBMS to manage and process LiDAR data. In this study we propose how to store and manage LiDAR data using RDBMS.

Key words : LiDAR, GIS, Remote Sensing

### I. 서 론

정부주도로 추진된 1,2차 NGIS사업은 국가기본 지리정보체계를 통하여 전국토를 디지털화된 데이터베이스를 구축하였으며, NGIS사업에 참여한 민간사업체를 통한 GIS응용 기초기술 발전에 괄목할 만한 성장을 이루었다. 이를 기반으로 3차 NGIS사업(2006-2010)은 U-Korea를 지향하는 유비쿼터스 시대를 대비한 기술혁신 및 고도화 기술개발에 역점을 두고 국토해양부에서는 2007년부터 5년간 약1450억원의 R&D예산을 투입하여 『지능형국토정보기술혁신

사업』을 추진하고 있다.[1] 이와 관련된 여러가지 요소기술로는 GIS, RS, LBS, USN, RFID, IT Infra. 등이 있으며, 그 기저에 측량이라는 분야가 핵심으로 자리 잡고 있다. 그동안 수치지도의 제작은 항공사진이나 위성영상을 이용하여 제작되었으며 영상도화라는 전처리 공정을 필수적으로 거쳐야 했다. 이것은 많은 시간과 전문화된 인력을 필요로 하므로 수치지도를 생산하는 기간이 많이 소요되고 있다. GIS를 사용하는 목적에 따라서는 보다 정교한 현실적인 자료를 요구하는 분야가 많으며, 시간에 따라 변화는 지표의 형상에 관한 정밀한 시계열 자료가 없으면 공간

\* 한세대학교 유시티IT산업정책학과 석사과정(IT Division, Hansei University)

\*\* 한세대학교 유시티IT산업정책학과 조교수(IT Division, Hansei University)

· 제1저자 (First Author) : 김호근

· 투고일자 : 2008년 11월 24일

· 심사(수정)일자 : 2008년 11월 25일 (수정일자 : 2008년 12월 18일)

· 게재일자 : 2008년 12월 30일



- PUBLIC HEADER BLOCK : 좌표범위, Point수 저장
- VARIABLE LENGTH RECORDS : 투영법, 메타 데이터, 사용자 응용 데이터 등의 변수형태 저장
- POINT DATA : 측량 좌표 저장

표 2. las에서 사용되는 변수 종류

Table 2. Variables use in las

char	1 byte
unsigned char	1 byte
short	2 bytes
unsigned short	2 bytes
long	4 bytes
unsigned long	4 bytes
double	8 byte IEEE floating point

표 3. Public header block 구조

Table 3. Structure of Public header block

Item	변수형	필수
File Signature ("LASF")	char[4]	○
Reserved	unsigned long	
GUID data 1	unsigned long	
GUID data 2	unsigned short	
GUID data 3	unsigned short	
GUID data 4	unsigned char[8]	
Version Major	unsigned char	○
Version Minor	unsigned char	○
System Identifier	char[32]	○
Generating Software	char[32]	○
Flight Date Julian	unsigned short	
Year	unsigned short	
Header Size	unsigned short	○
Offset to data	unsigned long	○
Number of records	unsigned long	○
Point Data Format ID (0-99 for spec)	unsigned char	○
Point Data Record Length	unsigned short	○

Number of point records	unsigned long	○
Number of points by return	unsigned long[5]	○
X scale factor	double	○
Y scale factor	double	○
Z scale factor	double	○
X offset	double	○
Y offset	double	○
Z offset	double	○
Max X	double	○
Min X	double	○
Max Y	double	○
Min Y	double	○
Max Z	double	○
Min Z	double	○

표 4. 가변길이 Record 구조

Table 4. Structure of variable length record

Item	변수형	필수
Record Signature(0xAABB)	unsigned short	○
User ID	char[16]	○
Record ID	unsigned short	○
Record Length After Header	unsigned short	○
Description	char[32]	

표 5. Point data record format 0 구조

Table 5. Structure of Point data record 0

Item	변수형	필수
X	long	○
Y	long	○
Z	long	○
Intensity	unsigned short	
Return Number	3 bits	○
Number of Returns (given pulse)	3 bits	○
Scan Direction Flag	1 bit	○
Edge of Flight Line	1 bit	○

Classification	unsigned char	
Scan Angle Rank (-90 to +90) - Left side	char	○
File Marker	unsigned char	
User Bit Field	unsigned short	

표 6. Point data record format 1 구조  
Table 6. Structure of Point data record 1

Item	변수형	필수
X	long	○
Y	long	○
Z	long	○
Intensity	unsigned short	
Return Number	3 bits	○
Number of Returns (given pulse)	3 bits	○
Scan Direction Flag	1 bit	○
Edge of Flight Line	1 bit	○
Classification	unsigned char	
Scan Angle Rank (-90 to +90) - Left side	char	○
File Marker	unsigned char	
User Bit Field	unsigned short	
GPS Time	double	○

2-1 가변길이 Record Georeferencing정보

Georeferencing의 ASPRS 표준은 GeoTIFF표준에 따라 사용할 수 있도록 규정한 포맷으로 가변길이 레코드의 section은 TIFF파일의 GeoTIFF key tag를 포함하는 동일한 형태의 구조를 가지고 있다. LAS파일은 Raster포맷이 아니기 때문에 각 point 는 자신의 절대 위치값을 가지고 있다. 그러므로 GeoTIFF tag의 6개 중에서 3개의 tag만을 사용하며 ModelTiePointTag (33922), ModelPixelScaleTag(33550), ModelTransformationTag(34264) record는 제외되고 GeoKeyDirectoryTag(34735), GeoDouble ParamsTag(34736), GeoASCIIParamsTag(34737) record가 사용된다.

GeoKeyDirectoryTag record는 필수이며 GeoDoubleParamsTag와 GeoASCIIParamsTag record는

GeoKeyDirectoryTag에 따라서 사용할 수도 있고 안할 수도 있다.

□ GeoKeyDirectoryTag Record

- User ID: LASF\_Projection
- Record ID: 34735

이 record는 좌표체계를 정의하는 key값을 가진다. GeoTIFF 형식으로 되어 있으며, unsigned short형식의 배열이다.

□ GeoDoubleParamsTag Record

- User ID: LASF\_Projection
- Record ID: 34736

이 record는 GeoKeyDirectoryTag에 있는 tag set이 참조하는 값을 가진 double 배열이다.

□ GeoASCIIParamsTag Record

- User ID: LASF\_Projection
- Record ID: 34737

이 record는 GeoKeyDirectoryTag에 있는 데이터 위치를 참조하는 문자열을 가진 ASCII 문자 배열이다.

□ Classification lookup

- User ID: LASF\_Spec
- Record ID: 0
- Length: 256 recs x 16 byte struct len

□ Header lookup for flight-lines

- User ID: LASF\_Spec
- Record ID: 1
- Length: 256 recs x struct len

□ Histogram

- User ID: LASF\_Spec
- Record ID: 2

□ Text area description

- User ID: LASF\_Spec
- Record ID: 3

III. 데이터베이스 설계 및 구현

3-1 데이터베이스 설계

앞에서 살펴본 바와 같이 2진파일로 저장된 las 파일은 각각의 파일로 존재하므로 관리에 많은 불편이 따른다. 이것을 관계형데이터베이스(RDBMS)에 저장하여 관리함으로써 운용과 관리에 효율성을 기할 수 있다.

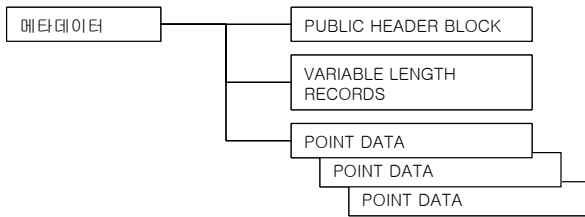


그림 3. LAS의 논리적 구조  
Fig. 3. Logical structure of LAS

3-2 데이터베이스 구현

데이터베이스는 open source license인 MySql5.0을 사용하여 schema를 구성 하였다. bit, unsigned integer, unsigned long 등의 data type은 MySql에서 표현할 수 없으므로 값이 손실되지 않도록 형 변환을 하였다.[5]

표 7 변수 형 변환표

Table 7. Table of variable type conversion

las파일 variable type	DB variable type
unsigned long	Bigint
unsigned short	Mediumint
unsigned char	Char(2)
bit	Char

표 8. 가변길이 Record 테이블

Table 8. Variable length record

Name	Type	Null	Default
Record_Signature	mediumint(9)	Yes	NULL
User_ID	char(16)	Yes	NULL
Record_ID	mediumint(9)	Yes	NULL
Record_Length_After_Hea...	mediumint(9)	Yes	NULL
Description	char(32)	Yes	NULL

표 9. PUBLIC HEADER BLOCK 테이블

Table 9. Table of PUBLIC HEADER BLOCK

Name	Type	Null	Default
File_Signature	char(4)	Yes	'LASF'
Reserved	bigint(20)	Yes	NULL
GUID_data_1	bigint(20)	Yes	NULL
GUID_data_2	mediumint(9)	Yes	NULL
GUID_data_3	mediumint(9)	Yes	NULL
GUID_data_4	char(8)	Yes	NULL
Version_Major	char(1)	Yes	NULL
Version_Minor	char(1)	Yes	NULL
System_Identifier	char(32)	Yes	NULL
Generating_Software	char(32)	Yes	NULL
Flight_Date_Julian	mediumint(9)	Yes	NULL
Year	mediumint(9)	Yes	NULL
Header_Size	mediumint(9)	Yes	NULL
Offset_to_data	bigint(20)	Yes	NULL
Number_of_records	bigint(20)	Yes	NULL
Point_Data_Format_ID	char(1)	Yes	NULL
Point_Data_Record_Length	mediumint(9)	Yes	NULL
Number_of_point_records	bigint(20)	Yes	NULL
Number_of_points_by_return	bigint(20)	Yes	NULL
X_scale_factor	double	Yes	NULL
Y_scale_factor	double	Yes	NULL
Z_scale_factor	double	Yes	NULL
X_offset	double	Yes	NULL
Y_offset	double	Yes	NULL
Z_offset	double	Yes	NULL
Max_X	double	Yes	NULL
Min_X	double	Yes	NULL
Max_Y	double	Yes	NULL
Min_Y	double	Yes	NULL
Max_Z	double	Yes	NULL
Min_Z	double	Yes	NULL

표 10. Point1 테이블

Table 10. Table of Point0

Name	Type	Null	Default
X	mediumtext	Yes	NULL
Y	mediumtext	Yes	NULL
Z	mediumtext	Yes	NULL
Intensity	mediumint(9)	Yes	NULL
Return_Number	char(1)	Yes	NULL
Number_of>Returns	char(1)	Yes	NULL
Scan_Direction_Flag	char(1)	Yes	NULL
Edge_of_Flight_Line	char(1)	Yes	NULL
Classification	char(1)	Yes	NULL
Scan_Angle_Rank	char(1)	Yes	NULL
File_Marker	char(1)	Yes	NULL
User_Bit_Field	mediumint(9)	Yes	NULL

표 11. Point1 테이블  
Table 11. Table of Point1

Name	Type	Null	Default
◆ X	mediumtext	Yes	NULL
◆ Y	mediumtext	Yes	NULL
◆ Z	mediumtext	Yes	NULL
◆ Intensity	mediumint(9)	Yes	NULL
◆ Return_Number	char(1)	Yes	NULL
◆ Number_of>Returns	char(1)	Yes	NULL
◆ Scan_Direction_Flag	char(1)	Yes	NULL
◆ Edge_of_Flight_Line	char(1)	Yes	NULL
◆ Classification	char(1)	Yes	NULL
◆ Scan_Angle_Rank	char(1)	Yes	NULL
◆ File_Marker	char(1)	Yes	NULL
◆ User_Bit_Field	mediumint(9)	Yes	NULL
◆ GPS_Time	double	Yes	NULL

#### IV. 결 론

본 연구에서는 LiDAR 데이터 파일의 구조를 살펴보고 데이터베이스로 전환할 수 있는 방법을 찾고자 하였다. File System과 Database의 변수형이 서로 다른 부분은 값이 손실되지 않도록 변수 형식을 변경 하였다. DBMS는 OpenSource인 MySQL5.0을 사용하여 LiDAR 데이터를 저장하였으며, 촬영일자, 촬영지역 등의 메타데이터를 데이터베이스에서 관리할 수 있도록 함으로써 효율적인 관리가 이루어질 수 있도록 하였다. 앞으로 본 연구의 결과를 범용적으로 사용할 수 있도록 하기 위한 연구가 계속되어야 할 것이다.

#### 참 고 문 헌

- [1] 허 준,김종홍,김진우,이정빈, "지능형국토정보기 술혁신사업을 위한 SWOT 분석", 2006 대한토목 학회 정기학술대회, 김대중컨벤션센터, p.304, 2006.10
- [2] 김대식, "항공레이저 매핑시스템 LiDAR와 GPS 기 술의 응용", 한국관계배수 제10 권1호, p.103, 2003.03
- [3] <http://www.csc.noaa.gov/products/sccoasts/html/tutlid.htm>
- [4] ASPRS, "LIDAR Data Exchange Format Standard

Version 1.0", pp.2~9, 2003.09

[5] <http://dev.mysql.com/doc/refman/5.0/en/data-types.html>

#### 김 호 근 (金鎬根)



1994년: 서울산업대학교 전자계산학과 공학사  
2007년-2008년: 한세대학교 유시티 IT산업정책학과 석사과정  
2008년현재: 인포런스 대표  
관심분야 : U-City, GIS, RS, Semantic Web, GIS감리

#### 권 창 희 (權昌希)



1998년 3월 : 동경도립대학교 도시 과학연구과(도시과학석사)  
2003년 3월 : 동경도립대학교 도시 과학연구과(도시과학박사)  
2003년 3월~현재 : 한세대학교 컴퓨터공학과/U-City IT 산업정책대학원,조교수

관심분야 : U-City,LBS,GIS,모바일컴퓨팅,전자정부,3D