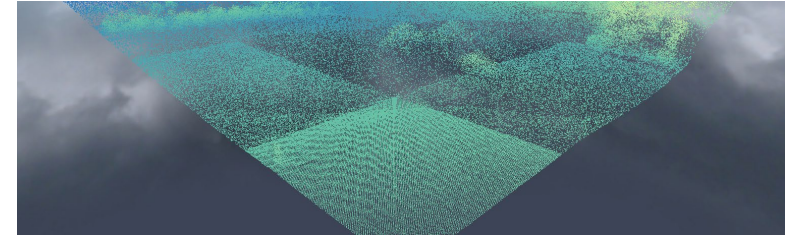


# Conversion of LAS/LAZ to VPC: json & COPC

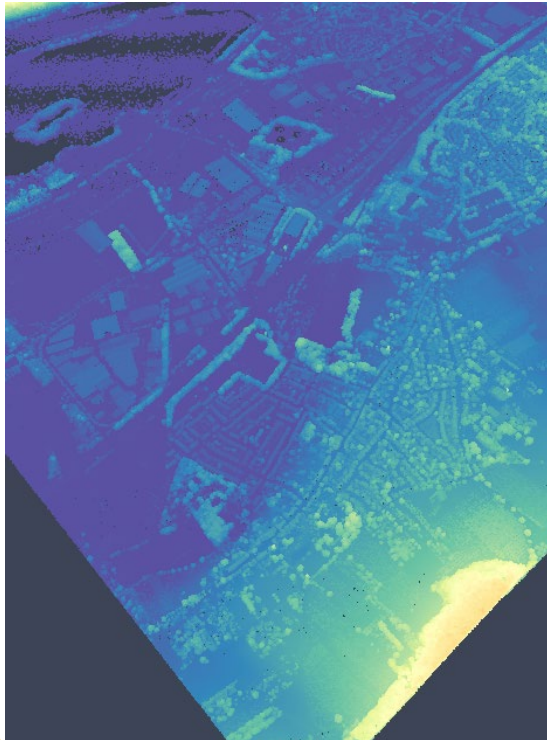
nDPointClouds project fall 2023 update

Thijs van Lankveld

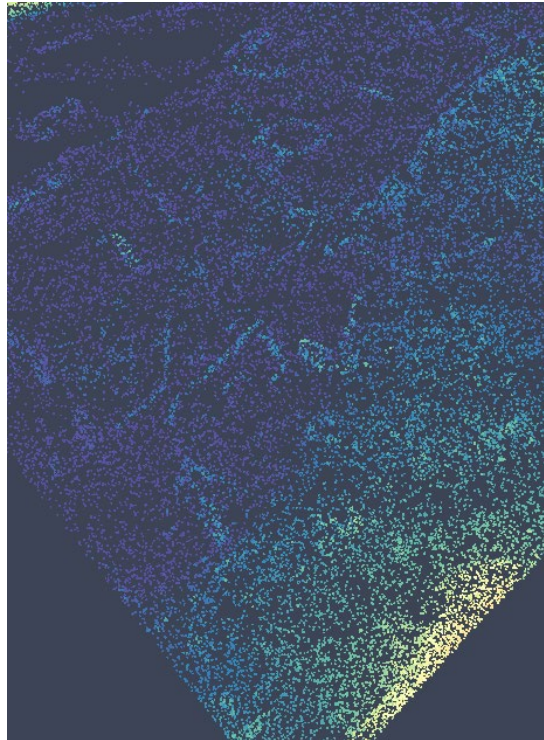
# Previously: cLoI in potree



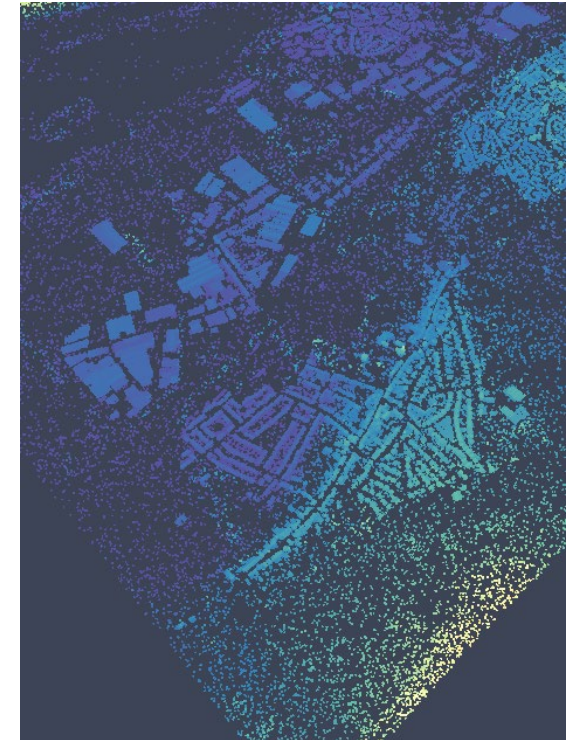
Base



cLoI



cLoI (wgt)



# File storage options

- LAS/LAZ
- EPT
- COPC
- EPT-COPC hybrid
- VPC



# LAS/LAZ

- Metadata in header
- Various point formats
- Point coordinates and attributes
  - Good compression
- Variable Length Records

Table 21: Point Data Record Format 10

Item	Format	Size	Required
X	long	4 bytes	*
Y	long	4 bytes	*
Z	long	4 bytes	*
Intensity	unsigned short	2 bytes	
Return Number	4 bits (bit 0 – 3)	4 bits	*
Number of Returns (given pulse)	4 bits (bit 4 – 7)	4 bits	*
Classification Flags	4 bits (bits 0 – 3)	4 bits	
Scanner Channel	2 bits (bits 4-5)	2 bits	*
Scan Direction Flag	1 bit (bit 6)	1 bit	*
Edge of Flight Line	1 bit (bit 7)	1 bit	*
Classification	unsigned char	1 byte	*
User Data	unsigned char	1 byte	
Scan Angle	short	2 bytes	*
Point Source ID	unsigned short	2 bytes	*
GPS Time	double	8 bytes	*
Red	unsigned short	2 bytes	*
Green	unsigned short	2 bytes	*
Blue	unsigned short	2 bytes	*
NIR	unsigned short	2 bytes	*
Wave Packet Descriptor Index	unsigned char	1 byte	*
Byte offset to waveform data	unsigned long long	8 bytes	*
Waveform packet size in bytes	unsigned long	4 bytes	*
Return Point Waveform Location	float	4 bytes	*
X(t)	float	4 bytes	*
Y(t)	float	4 bytes	*
Z(t)	float	4 bytes	*



# EPT

## EPT metadata

```
|— ept.json
|— ept-data
|   └─ 0-0-0-0.laz
|— ept-hierarchy
|   └─ 0-0-0-0.json
└─ ept-sources
    └─ list.json
        └─ 0.json
```

## EPT point cloud tiles (laz)

```
{
  "0-0-0-0": 65341,
  "1-0-0-0": 438,
  "2-0-1-0": 322,
  "1-0-0-1": 56209,
  "2-0-1-2": 4332,
  "2-1-1-2": 20300,
  "2-1-1-3": 64020,
  "3-2-3-6": 32004,
  "4-4-6-12": 1500,
  "4-5-6-13": 2400,
  "3-3-3-7": 542,
  "1-0-1-0": 30390,
  "2-1-2-0": 2300,
  "1-1-1-1": 2303
}
```



<https://entwine.io/en/latest/entwine-point-tile.html>

# COPC

## EPT metadata

```
├─ ept.json
├─ ept-data
│   └─ 0-0-0-0.laz
├─ ept-hierarchy
│   └─ 0-0-0-0.json
└─ ept-sources
    ├── list.json
    └─ 0.json
```

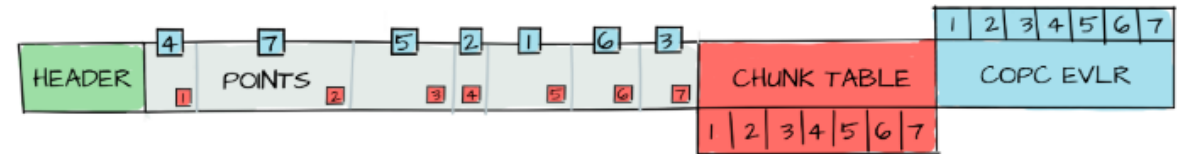
## EPT point cloud tiles (laz)

```
{
  "0-0-0-0": 65341,
  "1-0-0-0": 438,
  "2-0-1-0": 322,
  "1-0-0-1": 56209,
  "2-0-1-2": 4332,
  "2-1-1-2": 20300,
  "2-1-1-3": 64020,
  "3-2-3-6": 32004,
  "4-4-6-12": 1500,
  "4-5-6-13": 2400,
  "3-3-3-7": 542,
  "1-0-1-0": 30390,
  "2-1-2-0": 2300,
  "1-1-1-1": 2303
}
```

Loader for potree in development

## COPC file (laz)

point clouds + octree metadata  
accessible chunks



**COPC.io**  
Cloud Optimized Point Cloud

<https://copc.io/>

# EPT-COPC hybrid

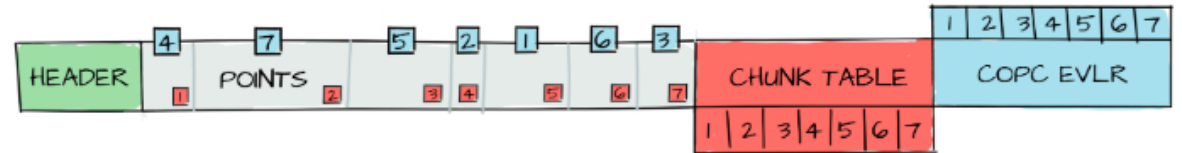
## EPT metadata

```
├─ ept.json
├─ ept-data
│   └─ 0-0-0-0.laz
├─ ept-hierarchy
│   └─ 0-0-0-0.json
└─ ept-sources
    ├── list.json
    └─ 0.json
```

## COPC point cloud clusters (laz)

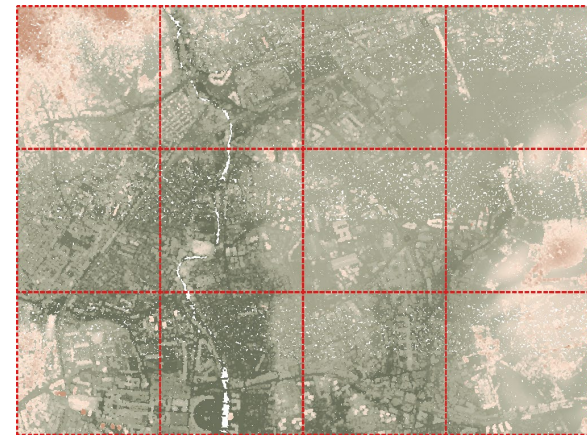
```
{
  "0-0-0-0": 65341,
  "1-0-0-0": 438,
  "2-0-1-0": 322,
  "1-0-0-1": 56209,
  "2-0-1-2": 4332,
  "2-1-1-2": 20300,
  "2-1-1-3": 64020,
  "3-2-3-6": 32004,
  "4-4-6-12": 1500,
  "4-5-6-13": 2400,
  "3-3-3-7": 542,
  "1-0-1-0": 30390,
  "2-1-2-0": 2300,
  "1-1-1-1": 2303
}
```

Costly to transfer many files  
Intractable to have high-TB file



# VPC

- Metadata for multiple LAS/LAZ/COPC files
  - Similar to EPT metadata
- STAC API ItemCollection
- Referenced by URL: remote data enabled
- Implemented in PDAL wrench
  - Module in QGIS 3.32\*
  - Depends on PDAL dev, GDAL dev



\*in Ubuntu: use flatpak install

Loader for potree in development



# Conclusions

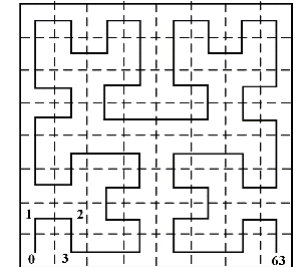
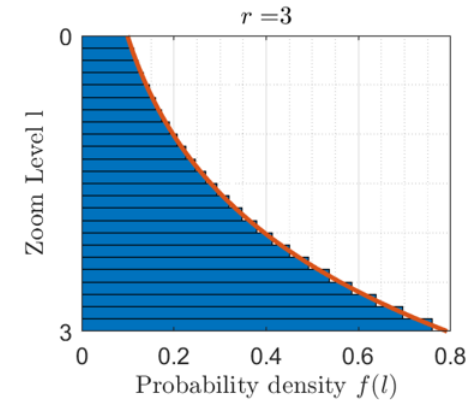
## File storage options for huge data

- LAS/LAZ
- EPT
- COPC
- EPT-COPC hybrid
- VPC



# Considerations

- File size vs. transfer cost
- Local vs. remote
- Additional point features
  - cLoI
  - Changes forward & backward.
- Octree vs. space-filling curve
- File storage vs. other storage options



“Empowering  
researchers across  
all disciplines  
through innovative  
research software”

## Contact Person



T.vanLankveld@esciencecenter.nl