nD-PointCloud

Netherlands eScience Center contributions

Thijs van Lankveld

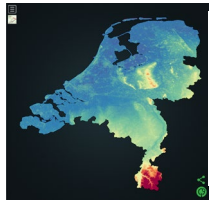
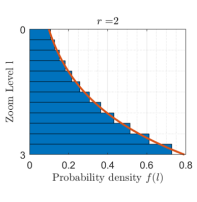
netherlands eScience center

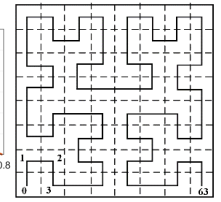
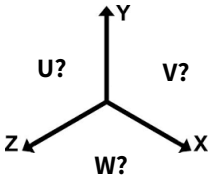
End-of-Project Meeting 2024/11/12

nD-PointCloud

netherlands eScience center

Project Goals


Point clouds as primary representation of spatio-temporal features

Scientific questions

- Will adding a continuous “importance” value improve processes?
- Will organizing points along space-filling curves help efficient processing?
- How can high-dimensional data be handled without performance drops?

eScience challenges

- Enabling HPC to handle innovations and scale of PC data (presented next)
- Updating PC viewer to modern standards
- Processing progressive AHN data for viewing

nD-PointCloud

netherlands eScience center

People involved




TU Delft

- Peter van Oosterom, Martijn Meijers, Vitali Diaz Mercado
- Edward Verbree

NLeSC

- Thijs van Lankveld, Maarten van Meersbergen, Nauman Ahmed
- Peter Kok, Ewan Cahen, Dusan Mijatovic, Sarah Alidoost

External

- Markus Schütz (Potree)
- Fedor Baart, Maarten Pronk (Deltares)



nD-PointCloud

netherlands eScience center

Data Storage Options

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

nD-PointCloud

netherlands eScience center

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	*

asprs THE IMAGING & GEOSPATIAL INFORMATION SOCIETY

https://www.asprs.org/wp-content/uploads/2010/12/LAS_1_4_r13.pdf

nD-PointCloud

netherlands eScience center

Entwine Point Tile

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
}
```


entwine

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

nD-PointCloud

netherlands eScience center

Cloud Optimized Point Cloud

EPT metadata

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

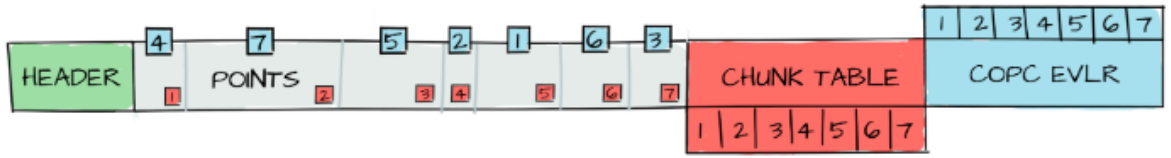
EPT point cloud tiles (laz)

```
{
   "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
}
```

COPC file (laz)

point clouds + octree metadata

accessible chunks



COPC.io
Cloud Optimized Point Cloud

https://copc.io/

nD-PointCloud

netherlands eScience center

EPT-COPC hybrid

EPT metadata

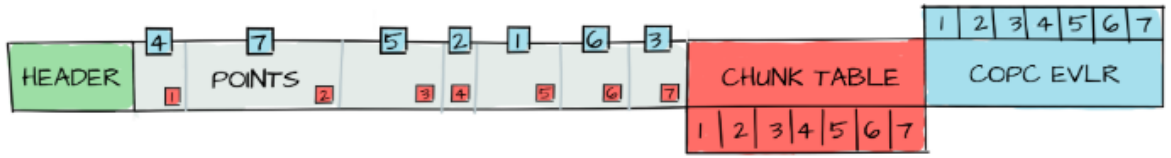
```
├── ept.json
├── ept-data
│   └── 0-0-0-0.laz
├── ept-hierarchy
│   └── 0-0-0-0.json
└── ept-sources
    ├── list.json
    └── 0.json
```

COPC point cloud chunks

```
{
  "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 massive-TB file



entwine

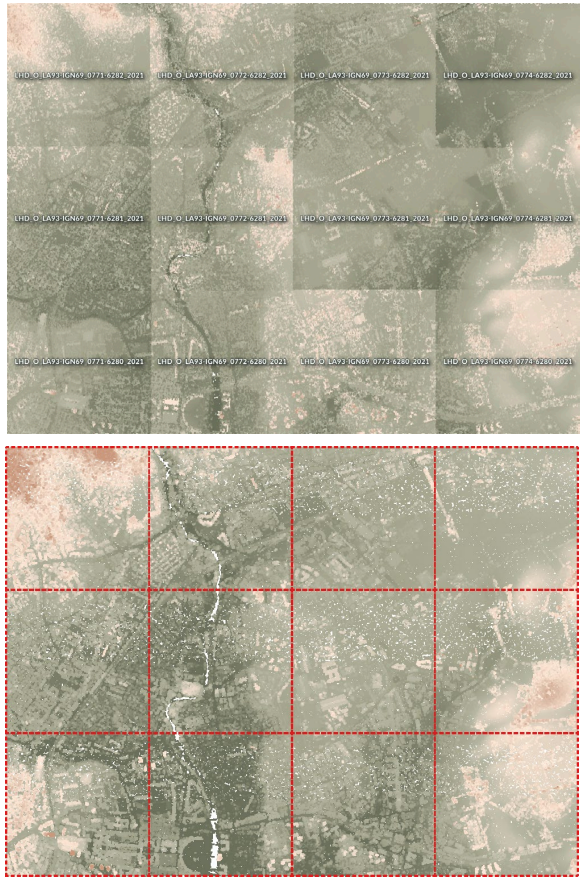
COPC.io
Cloud Optimized Point Cloud

nD-PointCloud

netherlands eScience center

Virtual Point Cloud

- 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

Lutra CONSULTING

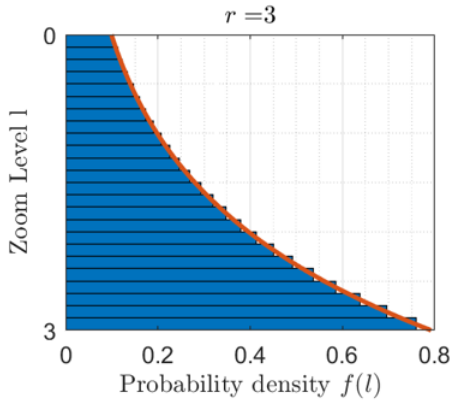
https://www.lutraconsulting.co.uk/blog/2023/06/08/virtual-point-clouds/

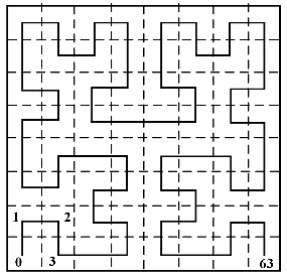
nD-PointCloud

netherlands eScience center

Data Storage 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



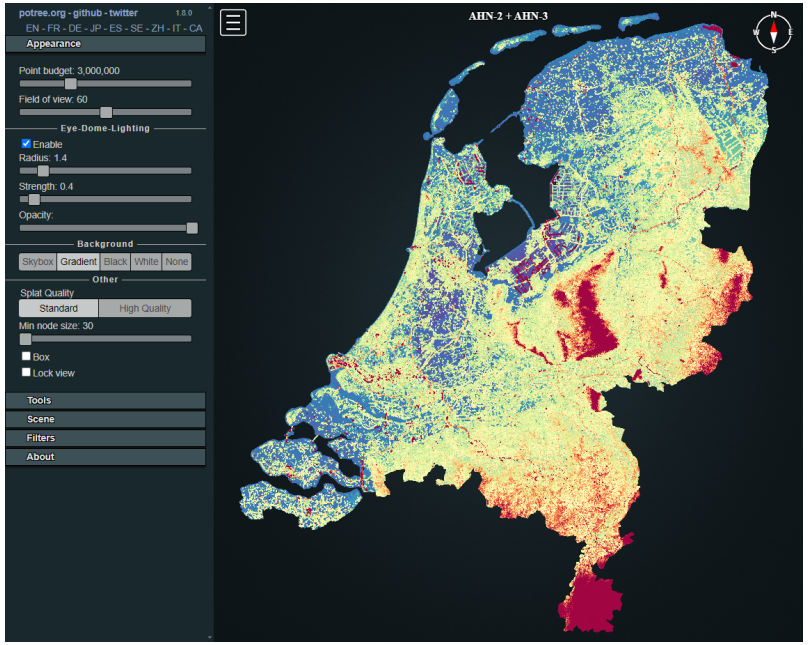


nD-PointCloud

netherlands eScience center

Potree

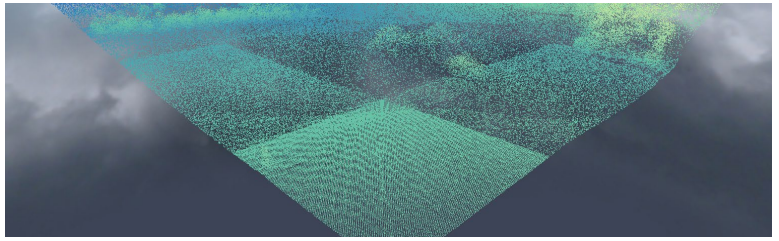
- Developed by Markus Schütz (TU Wien)
- Point cloud viewer
- Open source (2-clause BSD)
- Over 1k forks
- Implemented in JavaScript
  - WebGL renderer (to become WebGPU)
  - Depends on Node.js, Gulp



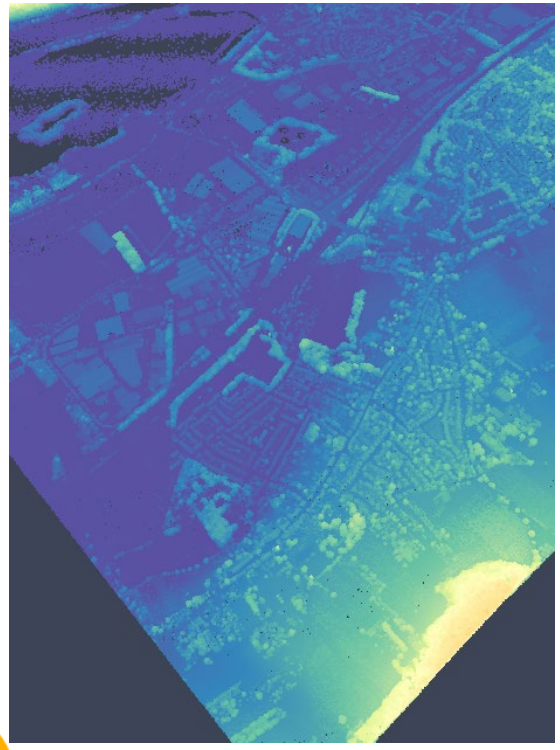
https://github.com/potree/potree

netherlands eScience center

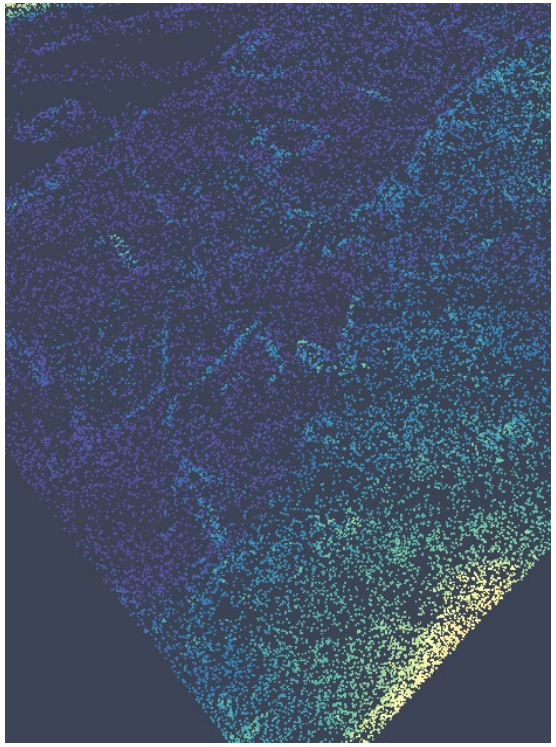
Potree: cLoI Rendering



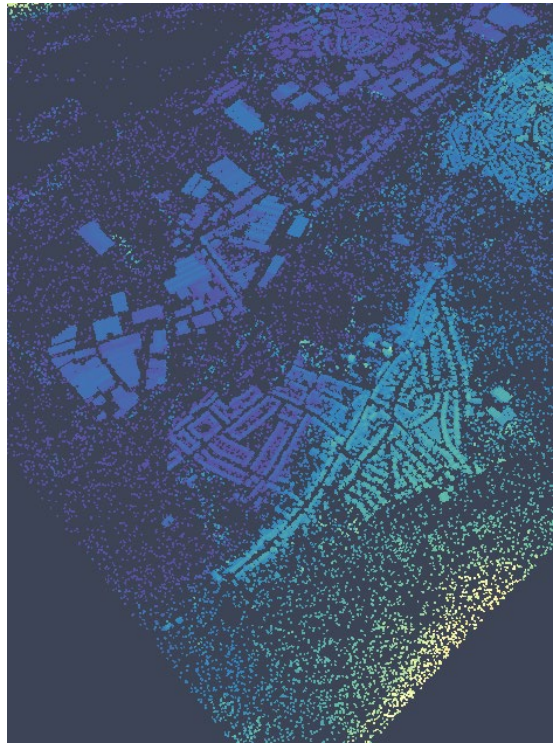
Base



cLoI



Weighted cLoI



netherlands eScience center

Potree: VPC loader

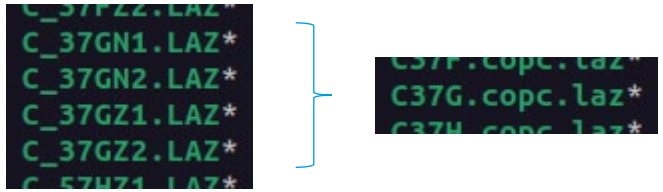
- Developed by Netherlands eScience Center
- Potree loader module for VPC
- Challenges
  - Limited to COPC data (raw LAS/LAZ not natively supported)
  - Potree has many internal assumptions on 8-child structure
  - File extension conflict with Veesus PC Arena 4D

netherlands eScience center

Converting AHN to COPC / VPC

ronna.tudelft.nl: Dell PowerEdge R7525

- Processors: 2x AMD EPYC 7443
- Cores: 2x 24
- Threads: 96*
- Speed: 2.85GHz
- RAM: 512GB (RDIMM)
- Scratch storage (input): 19.2TB SSD SATA
- Bulk storage (output): 60TB SAS, 7.2k rpm, RAID-5



*only 1-4 threads used: memory is bottleneck

nD-PointCloud

netherlands eScience center

Conversion Results

AHN 3

- Input files: 1374
- Input size: 2.4 TB
- Processing time (wc): 14d, 5h
- Output files: 382
- Output size: 3.1 TB

AHN 4

- Input files: 1381
- Input size: 6.1 TB
- Processing time (wc): 26d*
- Output files: 381
- Output size: 5.1 TB

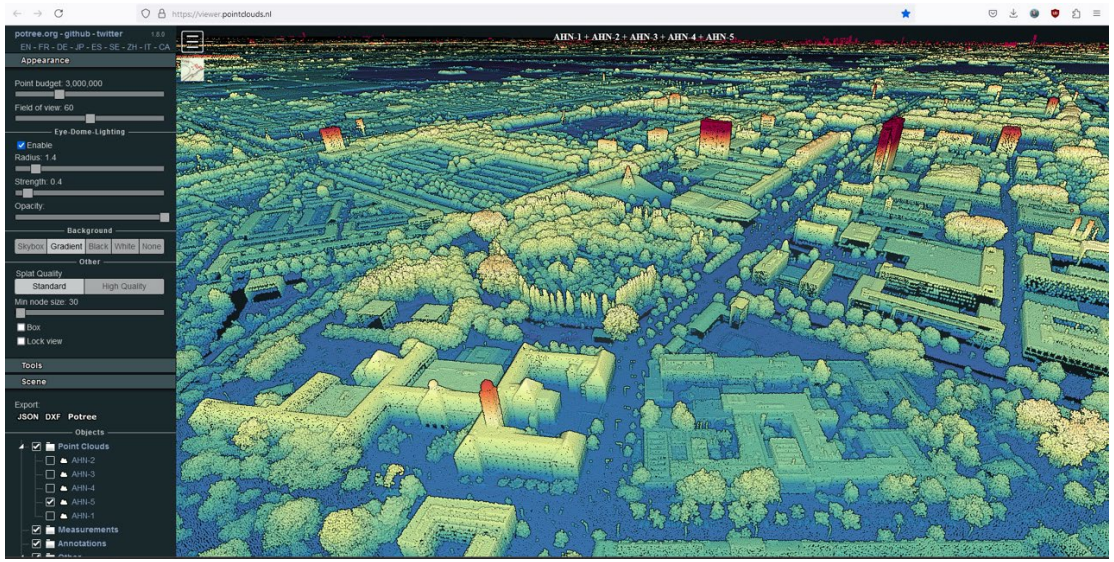
*estimated due to several large interruptions

nD-PointCloud

netherlands eScience center

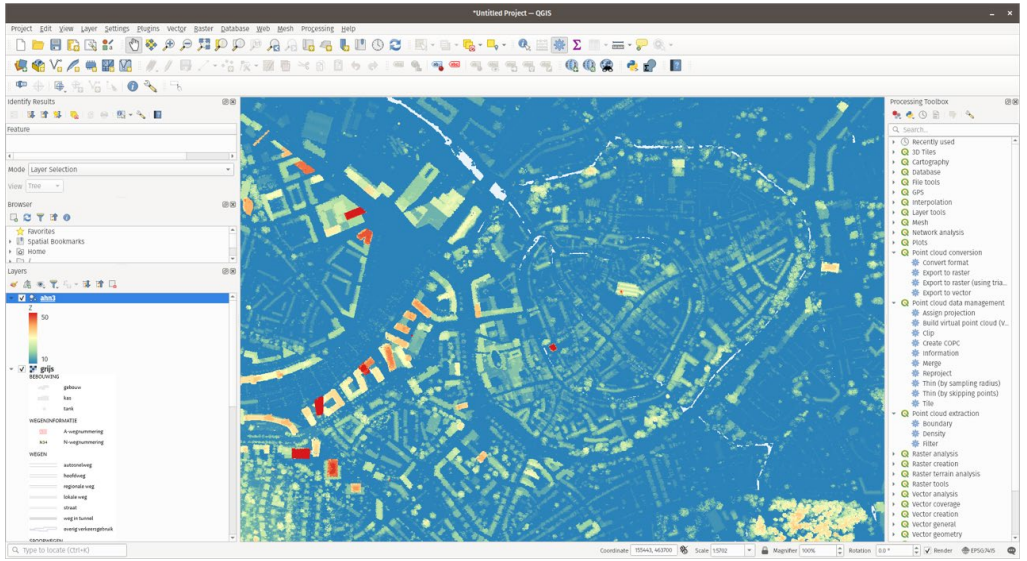
Viewing AHN

Potree



https://viewer.pointclouds.nl/

QGIS



to be released…

netherlands eScience center

Conclusions

eScience contributions

- Exploring options for data storage
- Adapting Potree with cLoI rendering and VPC loader
- Converting AHN 3-4 to COPC and VPC
- Converting AHN 1-5 to .potree files (presented next)

netherlands eScience center

Let’s keep in touch

T.vanLankveld@esciencecenter.nl

nD-PointCloud