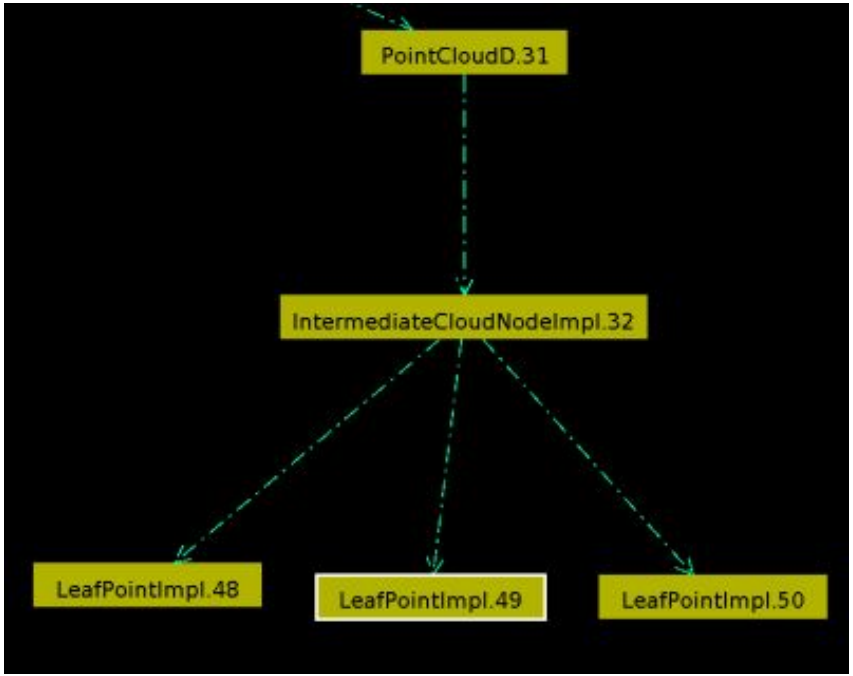
Point clouds as graph in GroIMP

Point cloud design in GroIMP

- PointCloud
  - A node, i.e. part of the graph
  - Manage the display/ picking
- Cloud
  - The data structure
  - Can be array, list, or graph based
- Point
  - Can be a node
  - Can be an object (mesh/ line)

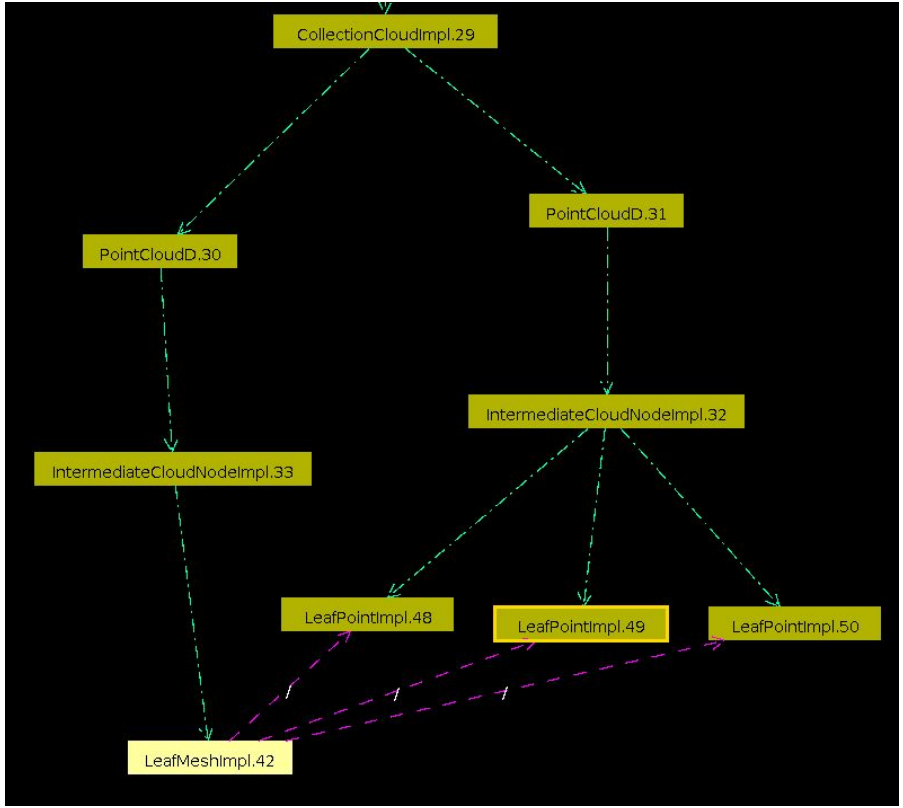


Nodes from PC objects are linked with branch edges

Importing point clouds

- Two data structure
  - List
  - Graph
- Two format:
  - XYZ: only points
  - PLY: can include edges/ faces

Points & Meshes/Lines are linked with Refinement edges



Export point clouds

- Global positions
  - All point clouds as one file
- Local positions
  - One by one
  - Selected
- XYZ or PLY

Use point clouds in RGG

- Create and add to graph
  - As array

```
float[] points = new float[]{1,2,3};
Cloud c = new CloudArray(points);
PointCloud pc = new PointCloudImpl(c);


[ Axiom ==> c; ]  // A Cloud can be directly added to the graph
[ Axiom ==> pc; ]
```

  - As Graph

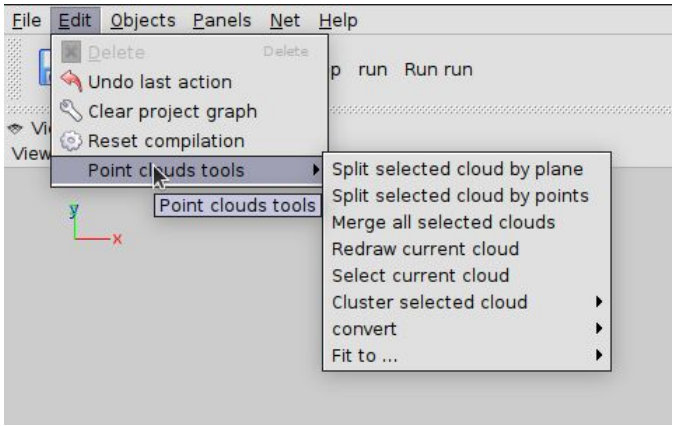
```
double[] coordinates = new double[]{1,2,3};
Cloud c = new CloudGraph();
c.addPoint(new LeafPointImpl(coordinates));
PointCloud pc = new PointCloudImpl(c);
```

- Use in code

```
FluxLightModel flm = new FluxLightModel(500000000,3);
flm.compute();
flm.getAbsorbedPower3d((*mn:LeafMeshImpl*)).getMax()/mn.getSurfaceArea();
```

Basic tools

- From both GUI and RGG
  - Convert
    - Convert the Cloud object and keep the Node
  - Merge
    - Convert if needed
  - Split
    - By nodes
    - By plane




```
Cloud c = first((* pc:PointCloud*)).getCloud();
Cloud[] clouds = Tools.split(slice((*c.getNode() (-->)*LeafPointImpl*),0,3), c);
```

Warning: Point clouds can have local coordinates

Use in queries

- In context

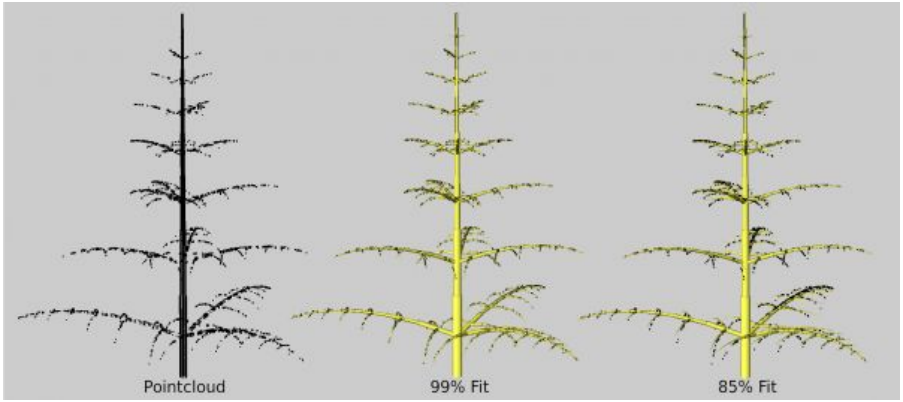
```
// count the number of points insde the generate volume
Volume v = volume(first((*F*)));
long inside = count((*p:Point,(v.contains(p,false)==true)*));
// get the total number of points
long total = count((*Point*));
```

- In query/ rules

```
[   p:Point::>{
    p[x]*=1.1;
    p[y]*=1.2;
    p[z]*=1.1;
}   ]
```

Examples

- Validate growth rule based on point clouds coordinates



- Apply light model to fine grained organ