



yFiles Class Library Overview

<https://www.yworks.com/>

Παναγιωτόπουλος Διονύσης

[Οκτώβριος 2020]



Η βιβλιοθήκη y-Files

- Η βιβλιοθήκη y-Files έχει σχεδιαστεί και αναπτυχθεί ολοκληρωτικά σε περιβάλλον Java και μπορεί να χρησιμοποιηθεί ανεξάρτητα σε οποιαδήποτε εφαρμογή στην πλατφόρμα της Java, αλλά και σε οποιοδήποτε λειτουργικό σύστημα:
 - Linux
 - Solaris
 - MacOS X
 - Microsoft Windows
- Η τελευταία έκδοση της βιβλιοθήκης είναι η y-Files 3.3 και απαιτεί Java 8 ή νεότερη έκδοση.
- Πρόκειται για ένα εμπορικό προϊόν το οποίο διανέμεται από την yWorks GmbH.



Χρήσιμα links

- Αρχικοποίηση IDE

(https://docs.yworks.com/yfilesjava/doc/api/#/dguide/getting_started-ide)

- IntelliJ IDEA 2016**

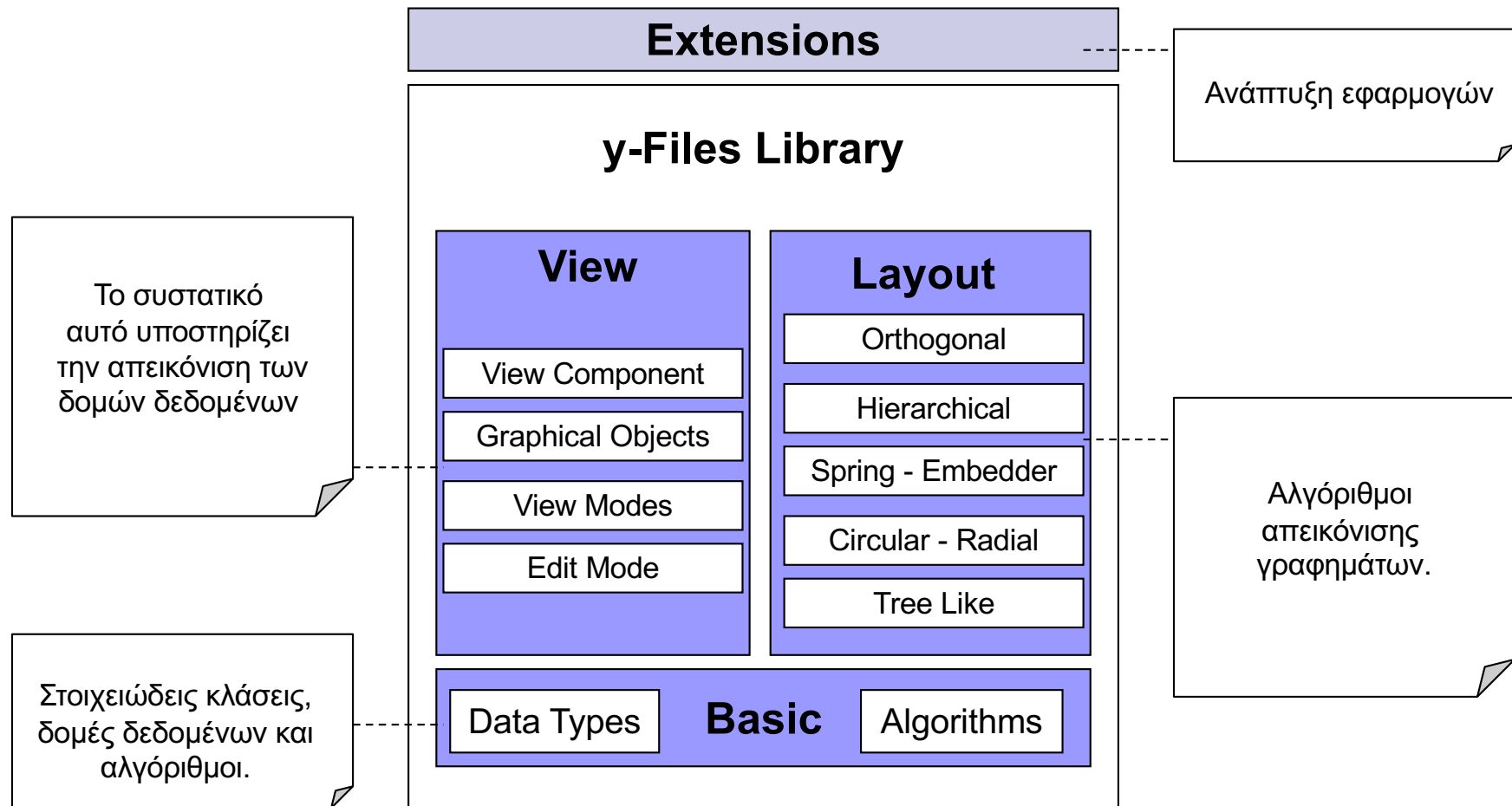
- Eclipse Neon**

- NetBeans 8.2**

- Documentation της βιβλιοθήκης

(<https://docs.yworks.com/yfilesjava/doc/api/#/home>)

Η δομή της βιβλιοθήκης y-Files





Το πακέτο `yfiles.algorithms`

- Περιέχει στοιχειώδεις κλάσεις:
 - Node – Edge
 - Graph
 - YList - NodeList - EdgeList



Η κλάση `yfiles.algorithms.Node`

- Κατασκευαστής:

- `protected Node(Graph g)`

- Μέθοδοι:

- `int degree(), outDegree(), inDegree()`

- `IEdgeCursor getEdgeCursor()`

- `IEdgeCursor getOutEdgeCursor(),
getInEdgeCursor()`

- `INodeCursor getNeighborCursor()`

- `Edge getEdgeFrom(Node source)`

- `int index()`



Η κλάση `yfiles.algorithms.Edge`

- Κατασκευαστής:

- `protected Edge(Graph g, Node v, Edge e1, Node w, Edge e2, GEI d1, GEI d2)`

- Μέθοδοι:

- `boolean isSelfLoop()`
- `Node source()`
- `Node target()`
- `Node opposite(Node v)`
- `int index()`



Η κλάση `yfiles.algorithms.Graph`

- Κατασκευαστής:

- `Graph()`
- `Graph(Graph graph)`

- Μέθοδοι:

- `boolean isEmpty()`
- `boolean contains(Edge e), contains(Node v)`
- `boolean containsEdge(Node v1, Node v2)`



Η κλάση `yfiles.algorithms.Graph`

■ Μέθοδοι:

- `Edge createEdge(Node v, Node w)`
- `Node createNode()`
- `IEdgeCursor getEdgeCursor()`
- `INodeCursor getNodeCursor()`
- `Edge[] getEdgeArray()`
- `Node[] getNodeArray()`
- `void removeEdge(Edge e), removeNode(Node v)`
- `int nodeCount(), edgeCount()`



Επίδειξη

```
Graph graph = new Graph();
Node tmpNodes[] = new Node[5];
for(int i = 0; i < 5; i++) {
    tmpNodes[i] = graph.createNode();
}
for(int i = 0; i < 5; i++) {
    for(int j = i+1; j < 5; j++) {
        graph.createEdge(tmpNodes[i],tmpNodes[j]);
    }
}
```



Οι κλάσεις INodeCursor & IEdgeCursor

```
for(INodeCursor nc = graph.getNodeCursor(); nc.ok(); nc.next())  
{  
    Node node = nc.node();  
    System.out.println(node);  
}
```

```
for(IEdgeCursor ec = graph. getEdgeCursor(); ec.ok(); ec.next())  
{  
    Edge edge = ec.edge();  
    System.out.println(edge);  
}
```



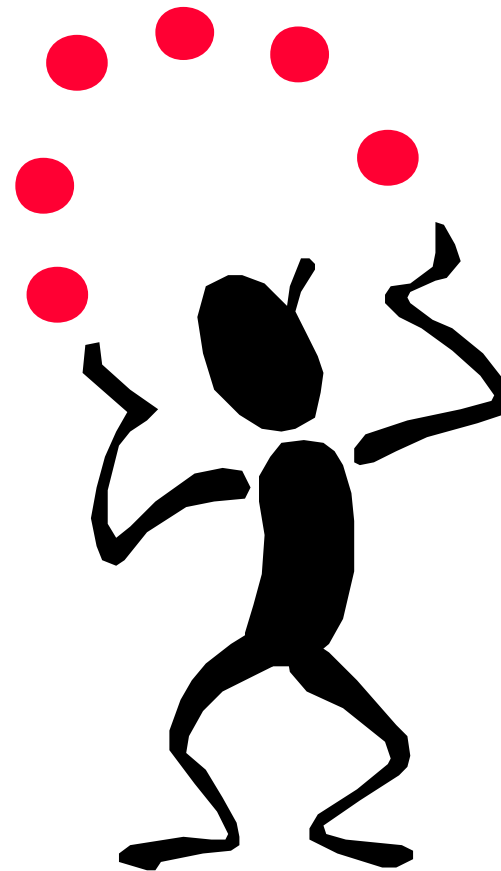
Διαπέραση κόμβων ή ακμών

```
for(IEdgeCursor ec = graph. getEdgeCursor(); ec.ok(); ec.next())  
{  
    graph.reverseEdge(ec.edge());  
}
```

```
for(INodeCursor nc = graph. getNodeCursor(); nc.ok(); nc.next())  
{  
    if(nc.node().degree() > 2)  
        graph.removeNode(nc.node());  
}
```

Επίδειξη

- GraphDemo
- RandomTreeGenerator





Οι κλάσεις INodeMap & IEdgeMap

- Παρέχει πρόσβαση σε δεδομένα που αφορούν έναν κόμβο/ακμή:
 - `INodeMap nodeMap = graph.createNodeMap();`
 - `IEdgeMap edgeMap = graph.createEdgeMap();`
- Μέθοδοι:
 - `void set(Object element, Object value)`
 - `void setBool(Object element, boolean value)`
 - `void setDouble(Object element, double value)`
 - `void setInt(Object element, boolean value)`
 - `Object get(Object element)`
 - `boolean getBool(Object element)`
 - `int getInt(Object element)`
 - `double getDouble(Object element)`

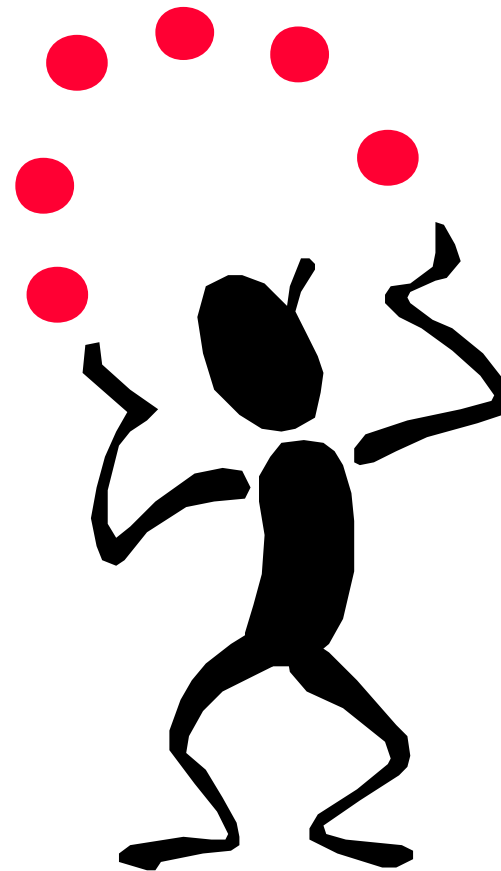


Παράδειγμα γραφήματος με βάρη

```
Graph graph = new Graph();
for(int i = 0; i < 10; i++)
{
    graph.createNode();
}
INodeMap map = graph.createNodeMap();
for(INodeCursor nc = graph. getNodeCursor(); nc.ok(); nc.next())
{
    map.setInt(nc.node(), nc.node().index());
}
for (INodeCursor nc = graph. getNodeCursor(); nc.ok(); nc.next())
{
    Node v = nc.node();
    System.out.println(v+" Weight: "+map.getInt(v));
}
```

Επίδειξη

- NodeMapTest
- ExtendedGraph





Το πακέτο `yfiles.algorithms`

- Περιέχει επίσης κλάσεις που υλοποιούν `standard` ρουτίνες σε γραφήματα. Π.χ.
 - Εύρεση κύκλων σε γράφημα.
 - Εύρεση συνεκτικών συνιστωσών.
 - Εύρεση `Spanning Tree`
 - κ.ο.κ.

- Στατικές μέθοδοι

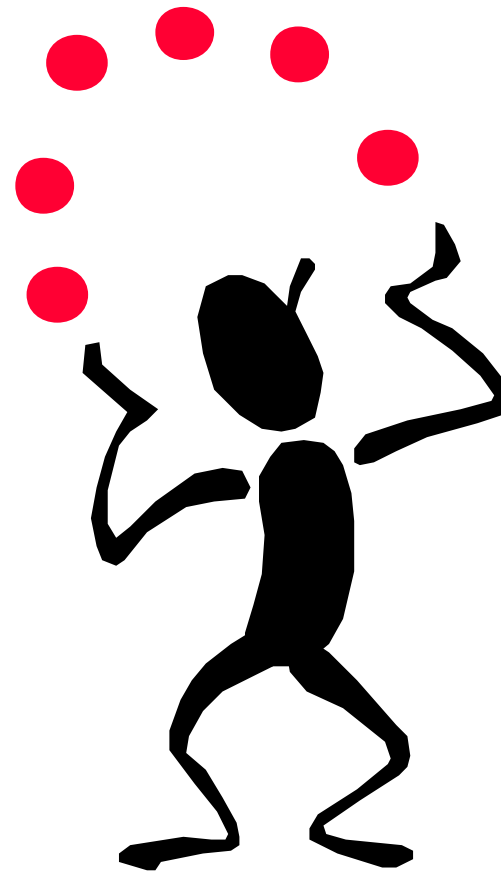


Βασικές κλάσεις του πακέτου yfiles.algorithms

- **Cycles**
findCycle(Graph g, boolean directed)
- **GraphChecker**
isPlanar(Graph g) or isTree(Graph graph)
- **GraphConnectivity**
isConnected(Graph g) or makeConnected(Graph g)
- **NetworkFlows**
calcMaxFlow(Graph g, Node t, Node s, IDataProvider d, IEdgeMap flow)
- **Paths**
findLongestPath(Graph g)
- **Trees**
isTree(Graph g) or isRootedTree(Graph g)

Επίδειξη

- AlgoDemo
- ConnectedDemo





Το πακέτο `yfiles.view`

- Σημαντικές κλάσεις:
 - `GraphComponent`
- Χρήση κλάσεων από το πακέτο `yfiles.graph`
 - `IGraph`
 - `IEdge`
 - `INode`



Το πακέτο `yfiles.graphml`

- Περιέχει κλάσεις:
 - για την αποθήκευση γραφημάτων σε αρχεία τύπου GraphML.
 - για την ανάκτηση γραφήματος από αρχείο τύπου GraphML.
- Βασική κλάση:
 - `GraphMLIOHandler`



Αποθήκευση Γραφήματος

```
public static void saveGraph(GraphComponent gc, String fileName)
{
    if (!fileName.endsWith(".graphml")) {
        fileName = fileName + ".graphml";
    }
    try {
        gc.exportToGraphML(fileName);
    }
    catch (java.io.IOException ioe) {
        //Do something...
    }
}
```



Ανάκτηση Γραφήματος

```
public static void loadGraph(GraphComponent gc, String fileName)
{
    if (name.endsWith(".graphml")) {
        try {
            gc.importFromGraphML(fileName);
        }
        catch (java.io.IOException ioe) {
            //Do something...
        }
    }
}
```



Μετατροπή IGraph σε Graph

- Κλάση yfiles.layout.YGraphAdapter
 - Κατασκευαστής: YGraphAdapter(IGraph)
 - Μέθοδοι:
 - Graph getYGraph()
 - IGraph getOriginalGraph()
 - INode getOriginalNode(Node v)
 - IEdge getOriginalEdge(Edge e)



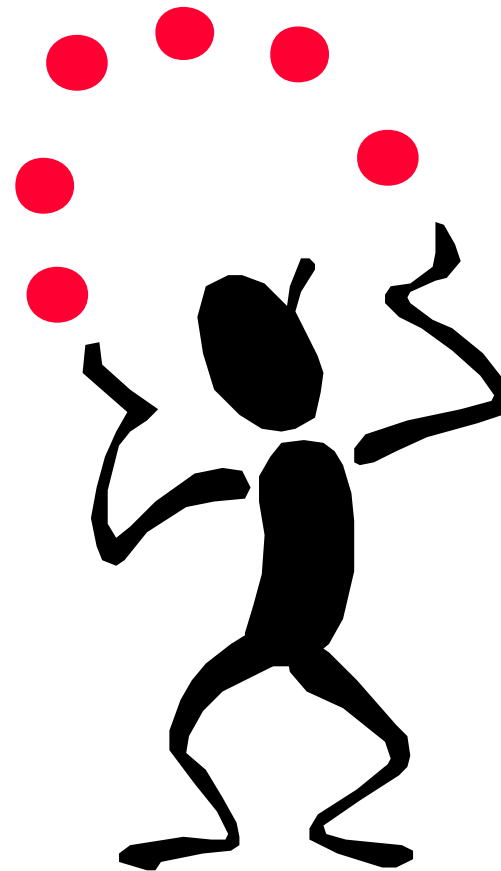
Η κλάση IGraph

■ Μέθοδοι:

- `setStyle(IEdge, IEdgeStyle)`
- `setStyle(INode, INodeStyle)`
- `setNodeCenter(INode v, PointD position)`
- `setNodeLayout(INode node, RectD layout)`

Επίδειξη


- VisualDemo





Η κλάση GraphComponent

- JComponent για την απεικόνιση γραφημάτων

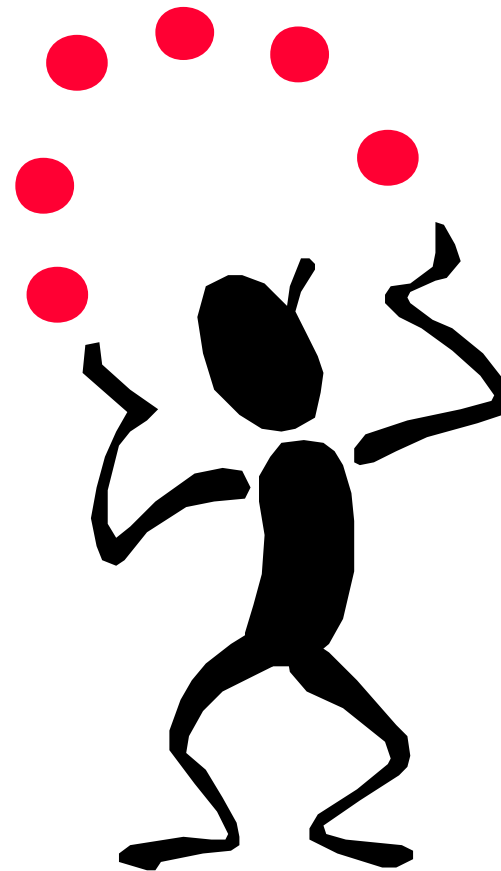


Παράδειγμα

```
public class SimpleFrame extends JFrame
{
    public SimpleFrame()
    {
        super("Simple Frame");
        setSize(600,600);
        setLayout(new BorderLayout());
        GraphComponent gc = new GraphComponent();
        //... read or create a graph
        add(gc, BorderLayout.CENTER);
        setVisible(true);
    }
}
```

Επίδειξη

- SimpleDemo





Zoom in / Zoom Out

- Η κλάση `GraphComponent` περιέχει μεθόδους που μπορούν να αλλάξουν το Zoom σε αυτό το Panel
- π.χ.
 - `gc.setZoom(new PointD(0.0,0.0), 1.2)`
 - `gc.setZoom(0.8*gc.getZoom())`
 - `gc.fitGraphBounds()`



Αλληλεπίδραση με τον χρήστη

- Η κλάση `GraphComponent` παρέχει την δυνατότητα επεξεργασίας του γραφήματος από τον χρήστη
- Π.χ.
 - `gc.setInputMode(new GraphEditorInputMode());`

Επίδειξη

- SimpleFrame

