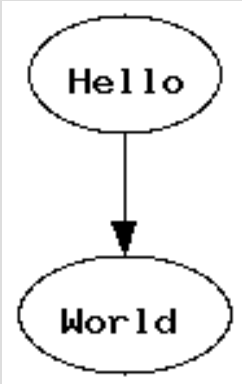
Gvglue.py

Generating cool graphviz dot files

What is Graphviz ?

- Represents structured informations as diagrams
- Easy to use and script
- “digraph G { Hello->World }” will produce :



Why Gvglue.py ?

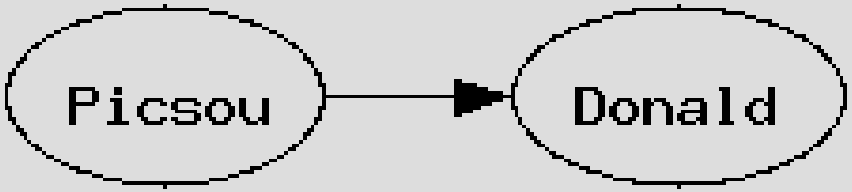
- Simple API to generate dot files
- Not a libgraphviz frontend
- No need to learn more than 3 functions to do anything you need from graphviz
- KISS spirit, because a graph does not need more than : “This is linked with that”
- Graphviz provides a set of programs that can read dot files and produces nice graphs

Why not using pydot or other frontend ?

- You need to learn both the frontend and graphviz
- Because.. well I couldn't put a node in a clusted belonging to an other cluster
- You face bugs that are not related to graphviz
- Not that easy to use

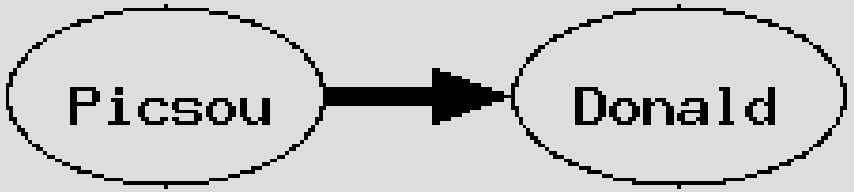
Simple example

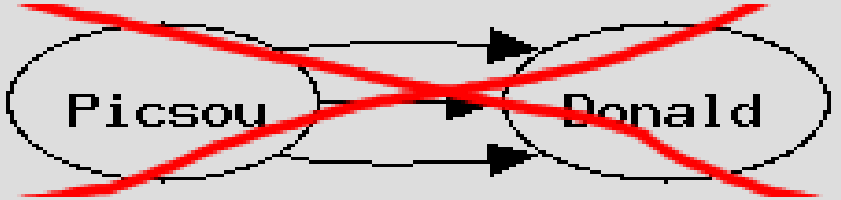
```
#!/usr/bin/env python


from gvglue import *

if __name__ == "__main__":
        gvg = GvGlue()

        radin = gvg.newItem("Picsou")
        maladroit = gvg.newItem("Donald")

        gvg.newLink(radin, maladroit)

        gvg.finish()
```



Graphs are for viewing

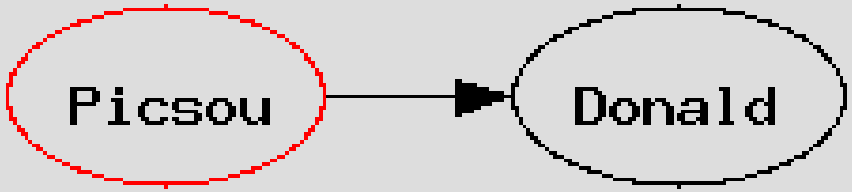
```
#!/usr/bin/env python

from gvglue import *

if __name__ == "__main__":
gvg = GvGlue()

radin = gvg.newItem("Picsou")
maladroit = gvg.newItem("Donald")

gvg.newLink(radin, maladroit)
gvg.newLink(radin, maladroit)
gvg.newLink(radin, maladroit)

gvg.finish()
```





Adding a property

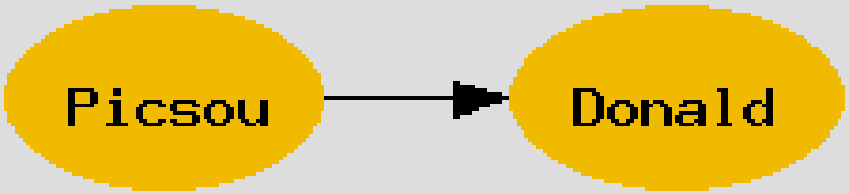
property_add(radin, "color", "red")



- All graphviz properties can be used here : http://www.graphviz.org/doc/info/attrs.html

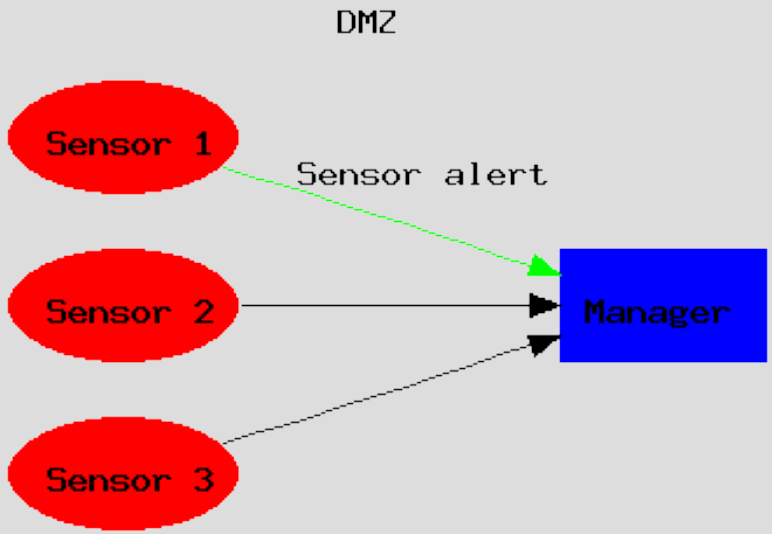
Defining styles

```
gvg = GvGlue()

radin = gvg.newItem("Picsou")
maladroit = gvg.newItem("Donald")

gvg.newLink(radin, maladroit)

gvg.properties_style_add("duck", "color", "#efba00")
gvg.properties_style_add("duck", "style", "filled")
gvg.properties_style_apply("duck", maladroit)
gvg.properties_style_apply("duck", radin)

gvg.finish()
```



Apply styles anywhere

```
gvg.properties_style_add("sensor", "shape", "ellipse")
gvg.properties_style_add("sensor", "style", "filled")
gvg.properties_style_add("sensor", "color", "red")

gvg.properties_style_add("manager", "shape",
"rectangle")
gvg.properties_style_add("manager", "style", "filled")
gvg.properties_style_add("manager", "color", "blue")

dmz = gvg.newItem("DMZ")
s1 = gvg.newItem("Sensor 1", dmz)
gvg.properties_style_apply("sensor", s1)
s2 = gvg.newItem("Sensor 2", dmz)
gvg.properties_style_apply("sensor", s2)
s3 = gvg.newItem("Sensor 3", dmz)
gvg.properties_style_apply("sensor", s3)

m1 = gvg.newItem("Manager", dmz)
gvg.properties_style_apply("manager", m1)

l1 = gvg.newLink(s1, m1, "Sensor alert")
gvg.property_add(l1, "color", "green")
gvg.newLink(s2, m1)
gvg.newLink(s3, m1)
```



Get it!

http://www.wallinfire.net/files/gvglue.py