

# Graphen zeichnen mit dot

Dominik Wagenführ

Ubucon 2011

15. Oktober 2011

# Übersicht

**Einleitung**

**Knoten**

**Kanten**

**Graphen**

**Beispiele**

# Einleitung

- ▶ Teil von Graphviz<sup>1</sup>
- ▶ dot ist Name der Sprache ...
- ▶ ... aber auch hierarchischer Layout-Algorithmus
- ▶ andere Layout-Algorithmen: neato, circo
- ▶ Dokumentation im Web<sup>2</sup>

---

<sup>1</sup><http://graphviz.org/>

<sup>2</sup><http://graphviz.org/Documentation.php>

# Einleitung – Anwendungen

- ▶ Abhängigkeitsdiagramm
- ▶ Netzwerkdarstellung
- ▶ Strukturanalyse von Matrizen<sup>3</sup>
- ▶ Maschinenkonfiguration

---

<sup>3</sup><http://www2.research.att.com/~yifanhu/GALLERY/GRAPHS/index.html>

# Einleitung – Erstes Beispiel

- ▶ ungerichteter Graph:

```
graph G {
    Hallo -- Welt;
}
```

- ▶ Übersetzen:

```
$ dot example-graph1.dot
```

- ▶ Ergebnis zeigt dot-Ausgabe, samt Position der Elemente:

```
graph G {
    node [label="N"];
    graph [bb="0,0,70,108"];
    Hallo [pos="35,90", width="0.97", height="0.50"];
    Welt [pos="35,18", width="0.92", height="0.50"];
    Hallo -- Welt [pos="35,72 35,61 35,47 35,36"];
}
```

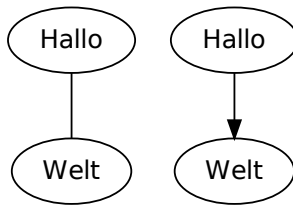
# Einleitung – Erstes Beispiel

- ▶ Übersetzen als PNG-Bild

```
$ dot -Tpng -O example-graph1.dot
```

- ▶ weitere Formate sind svg, pdf, eps und viele mehr<sup>4</sup>
- ▶ gerichteter Graph:

```
digraph G {  
    Hallo -> Welt;  
}
```



ungerichtet

gerichtet

<sup>4</sup><http://graphviz.org/content/output-formats>

# Knoten – Allgemeines

```
digraph G {  
    NODE1;  
    /* Kommentar */  
    // noch ein Kommentar  
}
```

- ▶ Abschluss durch Semikolon
- ▶ Kommentare zwischen `/* ... */` oder hinter `//`



**NODE1**

# Knoten – Beschriftung

```
digraph G {  
  NODE1 [label="Kopf "];  
  NODE2 [label="Körper",fontcolor="red"];  
  NODE3 [label="Arme\nBeine",fontcolor="0.8,1.0,1.0"];  
  NODE4 [label="Füße",fontcolor="#00ff0055"];  
}
```

- ▶ `label` – Beschriftung im Knoten
- ▶ `fontcolor` – Farbe (als Wort, RGB<sup>5</sup> oder HSV<sup>6</sup>)
- ▶ Trennung per `\n`



<sup>5</sup><https://secure.wikimedia.org/wikipedia/de/wiki/RGB-Farbraum>

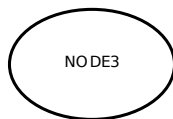
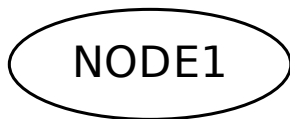
<sup>6</sup><https://secure.wikimedia.org/wikipedia/de/wiki/HSV-Farbraum>



# Knoten – Beschriftung

```
digraph G {  
    NODE1;  
    NODE2 [fontname="Times"];  
    NODE3 [fontsize=5];  
}
```

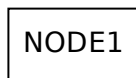
- ▶ `fontname` – Schriftart
- ▶ `fontsize` – Schriftgröße



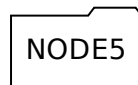
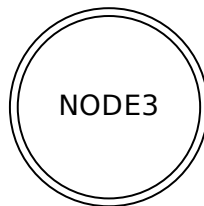
# Knoten – Form

```
digraph G {  
  NODE1 [shape=box];  
  NODE2 [shape=plaintext, label="Bezeichnung"];  
  NODE3 [shape=doublecircle];  
  NODE4 [shape=point];  
  NODE5 [shape=folder];  
}
```

- ▶ shape – Form des Knotens<sup>7</sup>



Bezeichnung

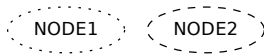


<sup>7</sup><http://graphviz.org/content/node-shapes>

# Knoten – Aussehen

```
digraph G {  
    NODE1 [style=dotted];  
    NODE2 [style=dashed];  
    NODE3 [style=invis];  
    NODE4 [style=rounded, shape=pentagon];  
    NODE5 [style=filled, fillcolor="purple"];  
    NODE6 [penwidth=2.0];  
}
```

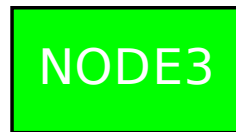
- ▶ `style` – Aussehen
- ▶ `penwidth` – Linienstärke



# Knoten – Globale Einstellungen

```
digraph G {  
    node [shape=box,fillcolor=purple,fontcolor=white,  
        style=filled];  
    NODE1;  
    NODE2 [style=dotted,fontcolor=blue];  
    NODE3 [fillcolor=green];  
}
```

- ▶ **node** – legt Aussehen für alle folgenden Knoten fest



# Knoten – Größe

```
digraph G {  
    NODE1 [shape=box];  
    NODE2 [shape=box,height=0.4,width=2];  
    NODE3 [shape=box,regular=true,height=0.8];  
    NODE4 [shape=box,height=0.8,labelloc=t];  
    NODE5 [shape=box,height=0.1];  
    NODE6 [fixedsize=true,shape=box,height=0.1];  
}
```

- ▶ `height` – Höhe, `width` – Breite
- ▶ `regular` – Höhe und Breite gleich
- ▶ `fixedsize` – feste Größe, unabhängig vom Inhalt
- ▶ `labelloc` – Position der Bezeichnung (`t`, `c`, `b`)



NODE1

NODE2

NODE3

NODE4

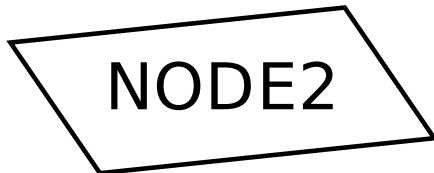
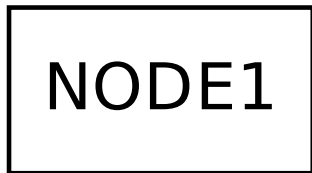
NODE5

~~NODE6~~

# Knoten – Drehung

```
digraph G {  
  NODE1 [shape=box];  
  NODE2 [shape=box,orientation=15.0];  
}
```

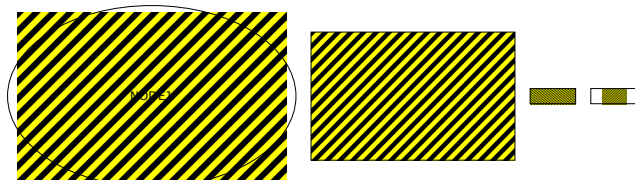
- ▶ `orientation` – Drehung für polygon-basierte Formen



# Knoten – Bilder

```
digraph G {  
  NODE1 [image="warning.png"];  
  node [image="warning.png",label="",shape=box];  
  NODE2 [height=0.25];  
  NODE3 [height=0.25,fixedsized=true];  
  NODE4 [height=0.25,fixedsized=true,imagescale=true];  
}
```

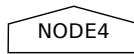
- ▶ **image** – Pfad zu einer Datei, Text liegt aber im Vordergrund
- ▶ Größe ändern mit **height**, **width** und **fixedsized**
- ▶ **imagescale** – Skalierungsmethode



# Knoten – Polygone

```
digraph G {  
    NODE1 [shape=polygon];  
    NODE2 [shape=polygon,sides=5];  
    NODE3 [shape=polygon,sides=5,skew=0.85];  
    NODE4 [shape=polygon,sides=5,distortion=-0.65];  
    NODE5 [shape=polygon,sides=5,peripheries=3];  
}
```

- ▶ **polygon** – Vieleck
- ▶ **sides** – Anzahl der Seiten
- ▶ **skew** – Neigung
- ▶ **distortion** – Verzerrung
- ▶ **peripheries** – Anzahl Rahmenlinien

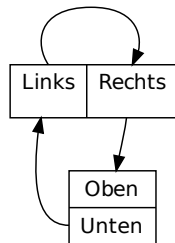




# Knoten – Records

```
digraph G {  
  node [shape=record];  
  NODE1 [label="<links> Links|<rechts> Rechts"];  
  NODE2 [label="{<oben> Oben|<unten> Unten}"];  
  NODE1:links:n -> NODE1:rechts:n;  
  NODE1:rechts -> NODE2:oben;  
  NODE2:unten:w -> NODE1:links;  
}
```

- ▶ **record** – tabellenartige Struktur
- ▶ Definition von Ports zum Andocken der Pfeile
- ▶ vordefinierte Ports bei allen Knoten (Himmelsrichtungen)



# Knoten – HTML

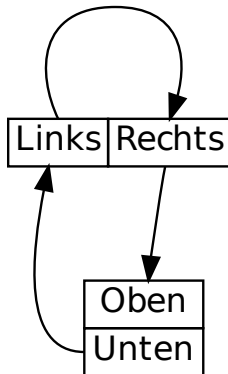
```
digraph G {
  node [shape=plaintext];

  NODE1 [label=<
    <TABLE BORDER="0" CELLBORDER="1" CELLSPACING="0"><TR>
      <TD PORT="links">Links</TD>
      <TD PORT="rechts">Rechts</TD>
    </TR></TABLE>
  >];
  NODE2 [label=<
    <TABLE BORDER="0" CELLBORDER="1" CELLSPACING="0"><TR>
      <TD PORT="oben">Oben</TD>
    </TR><TR>
      <TD PORT="unten">Unten</TD>
    </TR></TABLE>
  >];

  NODE1:links:n -> NODE1:rechts:n;
  NODE1:rechts -> NODE2:oben;
  NODE2:unten:w -> NODE1:links;
}
```

# Knoten – HTML

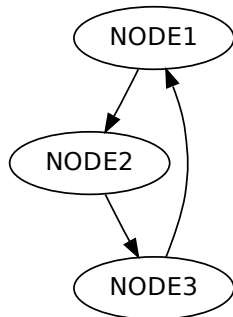
- ▶ es werden nur gewisse HTML-Befehle verstanden
- ▶ Benutzung auch bei Kanten möglich



# Kanten – Allgemeines

```
digraph G {  
    NODE1;  
    NODE2;  
    NODE1 -> NODE2;  
    NODE2 -> NODE3 -> NODE1;  
}
```

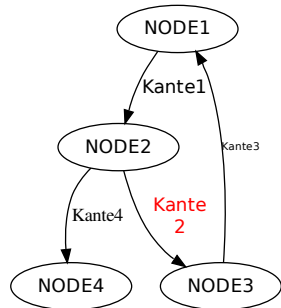
- ▶ Kanten durch Pfeile -> oder --
- ▶ auch Verkettung möglich
- ▶ neue Knoten müssen nicht extra deklariert werden



# Kanten – Beschriftung

```
digraph G {  
    NODE1 -> NODE2 [label="Kante1 "];  
    NODE2 -> NODE3 [label="Kante\n2",fontcolor=red];  
    NODE3 -> NODE1 [label="Kante3",fontsize=8];  
    NODE2 -> NODE4 [label="Kante4",fontname="Times"];  
}
```

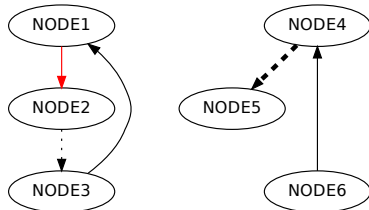
- ▶ **label** – Beschriftung der Kante (rechts davon)
- ▶ Trennung per **\n**
- ▶ **fontcolor** – Schriftfarbe (als Wort, RGB oder HSV)
- ▶ **fontsize** – Schriftgröße
- ▶ **fontname** – Schriftart



# Kanten – Aussehen

```
digraph G {  
  NODE1 -> NODE2 [color=red];  
  NODE2 -> NODE3 [style=dotted];  
  NODE3 -> NODE1 [weight=0];  
  NODE4 -> NODE5 [style=dashed,penwidth=4];  
  NODE5 -> NODE6 [style=invis];  
  NODE6 -> NODE4 [weight=10];  
}
```

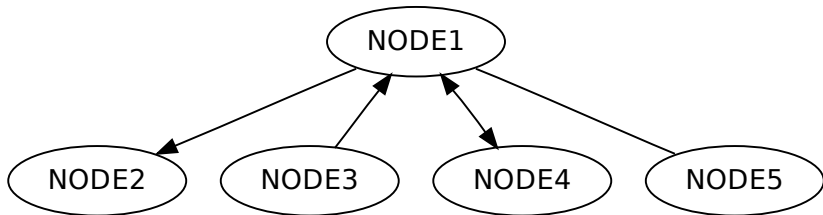
- ▶ **color** – Linienfarbe
- ▶ **style** – Aussehen
- ▶ **penwidth** – Linienstärke
- ▶ **weight** – „Krümmung“ der Linie



# Kanten – Pfeilrichtung

```
digraph G {  
  NODE1 -> NODE2 [dir=forward];  
  NODE1 -> NODE3 [dir=back];  
  NODE1 -> NODE4 [dir=both];  
  NODE1 -> NODE5 [dir=none];  
}
```

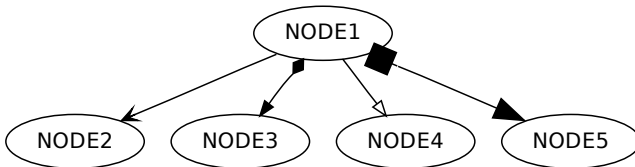
- ▶ `dir` – Pfeilrichtung



# Kanten – Pfeile

```
digraph G {  
  NODE1 -> NODE2 [arrowhead=vee];  
  NODE1 -> NODE3 [arrowtail=diamond];  
  NODE1 -> NODE4 [arrowhead=onormal];  
  NODE1 -> NODE5 [arrowtail=box,arrowsize=2];  
}
```

- ▶ **arrowhead** – Aussehen am Kopf
- ▶ **arrowtail** – Aussehen am Ende
- ▶ **arrowsize** – Pfeilspitzengröße

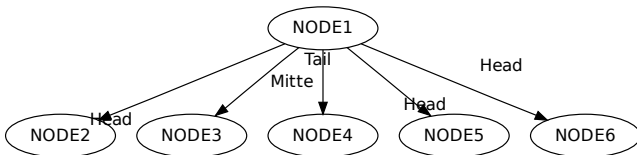




# Kanten – Noch mehr Beschriftung

```
digraph G {  
  NODE1 -> NODE2 [headlabel="Head"];  
  NODE1 -> NODE3 [label="Mitte"];  
  NODE1 -> NODE4 [taillabel="Tail"];  
  NODE1 -> NODE5 [headlabel="Head",labeldistance=1];  
  NODE1 -> NODE6 [headlabel="Head",labeldistance=6];  
}
```

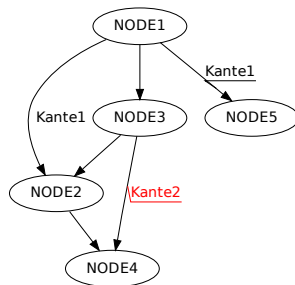
- ▶ **headlabel** – Beschriftung am Kopf
- ▶ **taillabel** – Beschriftung am Ende
- ▶ **labeldistance** – Abstand
- ▶ Mehr: **labelangle**, **labelfontcolor**, **labelfontname**, **labelfontsize**



# Kanten – Sonstiges

```
digraph G {  
  NODE1 -> NODE2 [label="Kante1"];  
  NODE1 -> NODE3 -> NODE2 -> NODE4;  
  NODE1 -> NODE5 [label="Kante1",decorate=true];  
  NODE3 -> NODE4 [label="Kante2",decorate=true,  
    fontcolor=red];  
}
```

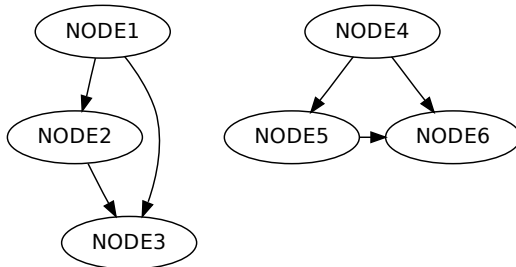
- ▶ **decorate** – Unterstreichung der Beschriftung bis zur Kante



# Kanten – Sonstiges

```
digraph G {  
    NODE1 -> NODE2; NODE1 -> NODE3;  
    NODE2 -> NODE3;  
    NODE4 -> NODE5; NODE4 -> NODE6;  
    NODE5 -> NODE6 [constraint=false];  
}
```

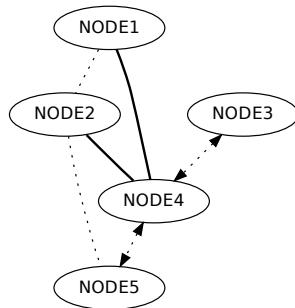
- ▶ **constraint** – keinen neuen Rank für Kante einfügen



# Kanten – Globale Einstellungen

```
digraph G {  
  edge [style=dotted,dir=none];  
  NODE1 -> NODE2 -> NODE5;  
  NODE3 -> NODE4 -> NODE5 [dir=both];  
  NODE2 -> NODE4 -> NODE1 [style=bold];  
}
```

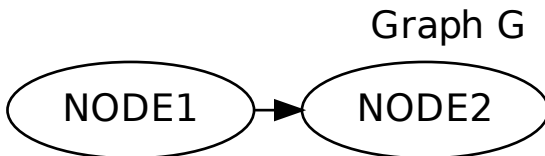
- **edge** = legt Aussehen für alle folgenden Kanten fest



# Graphen – Beschriftung

```
digraph G {  
    label="Graph G";  
    labelloc=t;  
    labeljust=right;  
    NODE1 -> NODE2 [constraint=false];  
}
```

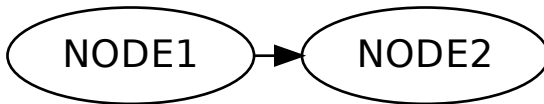
- ▶ `label` – Beschriftung
- ▶ `labelloc` – Platzierung der Beschriftung (oben oder unten)
- ▶ `labeljust` – Ausrichtung der Beschriftung



# Graphen – Beschriftung

```
digraph G {  
  label="Graph G";  
  fontsize=20;  
  fontcolor=red;  
  NODE1 -> NODE2 [constraint=false];  
}
```

- ▶ `fontsize` – Schriftgröße
- ▶ `fontcolor` – Schriftfarbe

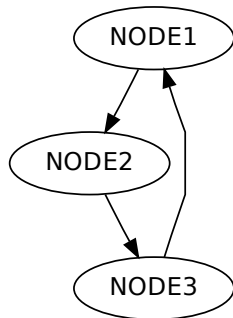


Graph G

# Graphen – Pfeilaussehen

```
digraph G {  
    splines=ortho;  
    NODE1 -> NODE2 -> NODE3 -> NODE1;  
}
```

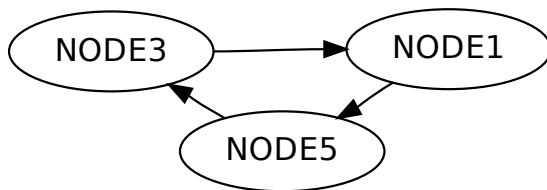
- ▶ `splines` – Pfeilaussehen im gesamten Diagramm
- ▶ Möglichkeiten: `spline|line|ortho|none`



# Graphen – Knotenabstände

```
digraph G {  
    overlap=false;  
    sep=0.5;  
    NODE3 -> NODE1 -> NODE5 -> NODE3;  
}
```

- ▶ **overlap** – Algorithmus, um Knotenüberlappung zu entfernen
- ▶ **sep** – Mindestabstand zwischen Knoten (nicht dot)
- ▶ **nodesep** – Mindestabstand zwischen Knoten (nur dot)

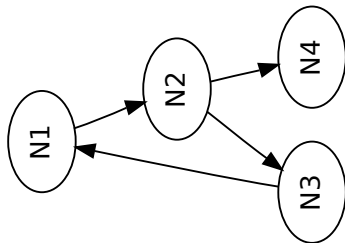




# Graphen – Layout

```
digraph G {  
    landscape=true;  
    N1 [root=true];  
    N3 -> N1 -> N2; N2 -> N4;  
}
```

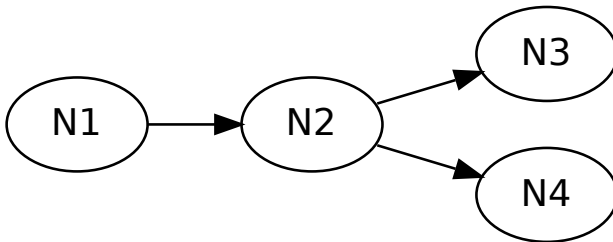
- ▶ **root** – Festlegung, was der Wurzelknoten ist
- ▶ **landscape** – Hoch- oder Querformat



# Graphen – Layout

```
digraph G {  
    rankdir=LR;  
    N1 -> N2 -> N3; N2 -> N4;  
}
```

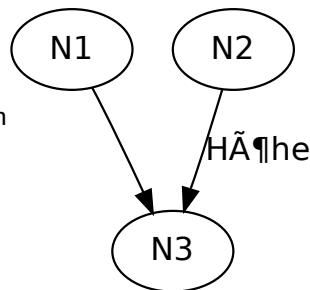
- ▶ **rankdir** – Richtung des Graphen
- ▶ Möglichkeiten: TB|BT|LR|RL



# Graphen – Sonstiges

```
digraph G {  
  charset="latin1";  
  concentrate=true;  
  N1 -> N3;  
  N1 -> N3;  
  N2 -> N3 [label="Höhe"];  
}
```

- ▶ **charset** – Schriftkodierung (normal: utf8)
- ▶ **concentrate** – fasst gleiche Pfeile (auch Teilwege) zusammen



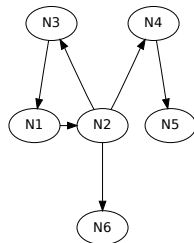
# Graphen – Subgraphen

```

digraph G {
  ranksep=1;
  subgraph G1 {
    rank=same;
    N1; N2; N5;
  }
  subgraph G2 {
    rank=min;
    N3; N4;
  }
  subgraph G3 {
    rank=sink;
    N6;
  }
  N1 -> N2 -> N4 -> N5;
  N2 -> N3 -> N1; N2 -> N6;
}

```

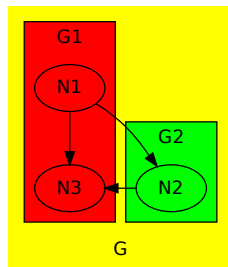
- ▶ **subgraph** – Graph innerhalb eines Graphen, Name und Schlagwort optional
- ▶ **rank** – Rank des Subgraphen und der Knoten im Gesamtgraph
- ▶ **same|min|max|sink|source**
- ▶ **ranksep** – Abstand zwischen zwei Subgraphen



# Graphen – Cluster

```
digraph G {
  label="G";
  bgcolor=yellow;
  subgraph clusterG1 {
    label="G1";
    bgcolor=red;
    N1->N3;
  }
  subgraph clusterG2 {
    label="G2";
    fillcolor=green;
    style=filled;
    N2;
  }
  N1 -> N2 -> N3;
}
```

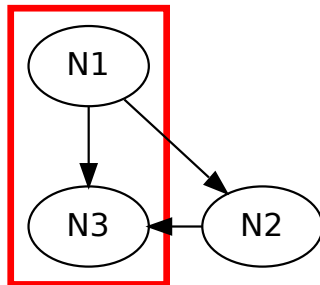
- ▶ Spezielle Subgraphen, Name muss mit **cluster** anfangen
- ▶ **bgcolor** – Hintergrundfarbe
- ▶ alternativ **fillcolor** und **style=filled**
- ▶ **label** – Beschriftung des Clusters



# Graphen – Cluster

```
digraph G {  
  subgraph clusterG1 {  
    pencolor=red;  
    penwidth=3.0;  
    N1->N3;  
  }  
  subgraph clusterG2 {  
    peripheries=0;  
    N2;  
  }  
  N1 -> N2 -> N3;  
}
```

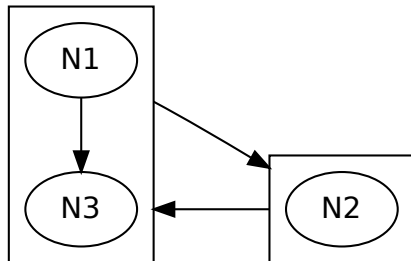
- ▶ **pencolor** = Rahmenfarbe (alternativ **color**)
- ▶ **penwidth** = Rahmenstärke
- ▶ **peripheries** = Rahmen sichtbar (0 oder 1)



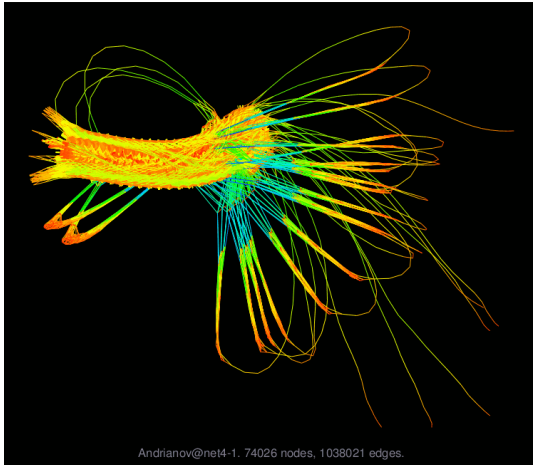
# Graphen – Cluster

```
digraph G {  
    compound=true;  
    nodesep=1.0;  
    subgraph clusterG1 {  
        N1->N3;  
    }  
    subgraph clusterG2 {  
        N2;  
    }  
    N1 -> N2  
        [ltail=clusterG1,  
         lhead=clusterG2];  
    N2 -> N3  
        [ltail=clusterG2,  
         lhead=clusterG1];  
}
```

- ▶ **compound** – Kanten an Clustergrenzen abschneiden
- ▶ **lhead** – echtes Ende der Kante
- ▶ **ltail** – echtes Beginn der Kante



# Beispiele – Matrix-Strukturanalyse



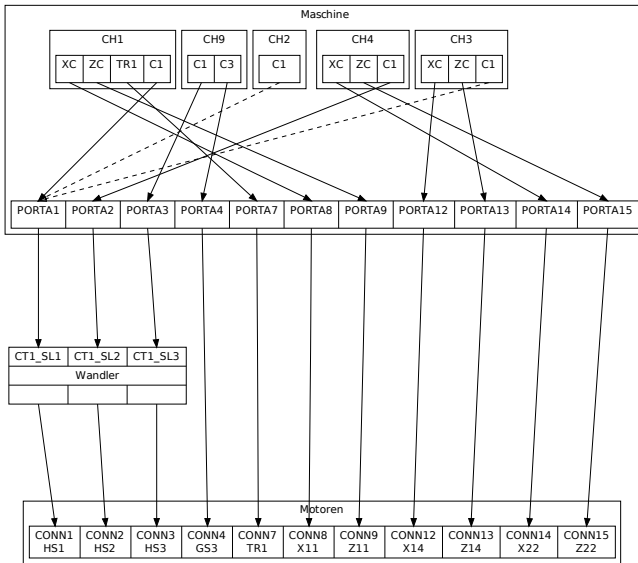
Copyright Yifan Hu, AT&T

<http://www2.research.att.com/~yifanhu/GALLERY/GRAPHS/index.html>

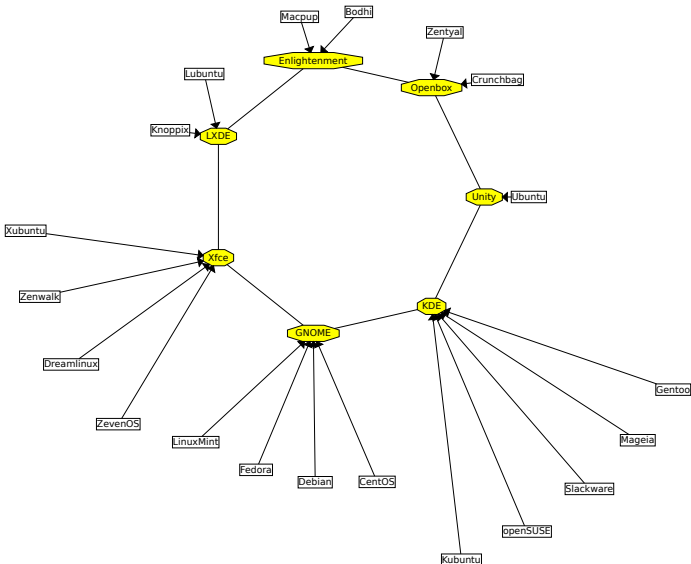
Bild unterliegt nicht der CreativeCommons-Lizenz!



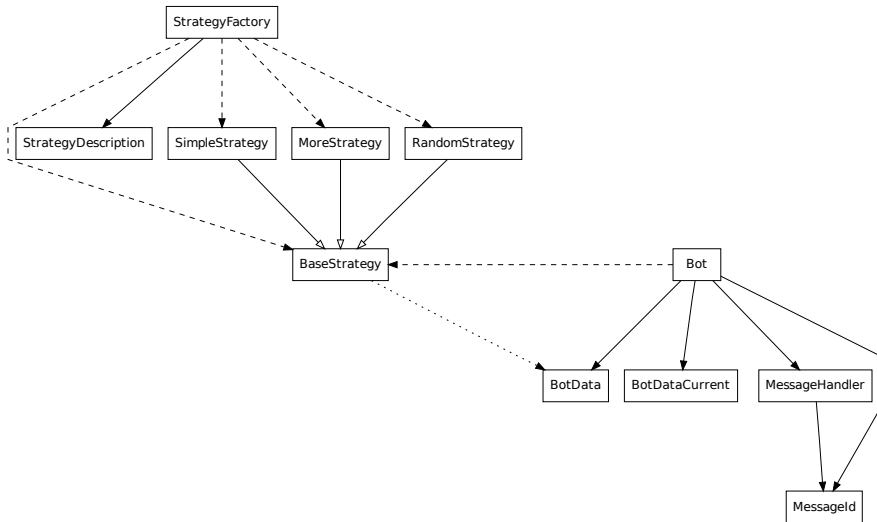
# Beispiele – Maschinenkonfiguration



# Beispiele – Linux-Desktop-Umgebungen



# Beispiele – Klassenabhängigkeiten



freiesMagazin-Programmierwettbewerb (Dummy-Bot)

# Ende

Vielen Dank für die Aufmerksamkeit  
und viel Spaß beim Nachmachen!

Folien und Beispiele unterliegen der Creative Commons  
„Namensnennung-Weitergabe unter gleichen Bedingungen 3.0 Deutschland“.



Copyright 2011 Dominik Wagenführ