

Visualisierung von UML-Diagrammen

Handout zum Vortrag

Christian M. Meyer

Motivation

- Layout von Hand ist aufwändig
- Automatisches Layout für MDA notwendig
- Viele Tools für automatisches Graphenlayout, Anbindung an führende Case-Tools
- Aber: Tools haben viele Nachteile
 - Hierarchische Vererbungsstrukturen nicht behandelt
 - Übersicht oft unzureichend
 - Ästhetische Mängel
- Spezielle Algorithmen für UML-Diagramme sinnvoll

Ziele

- Minimale Kreuzungen
- Minimale Knicke
- Einheitliche Richtung in Hierarchien
- Keine verschachtelten Hierarchien
- Beziehungen je nach Semantik betrachten
- Orthogonales Layout
- Kombinierte Vererbungspfeile
- Gut lesbare Kantenbeschriftungen

GoVisual-Algorithmus

```

procedure GoVisual( $G[V, A, E]$ )
 $G'[V', A', E'] := \text{preprocess}(G)$ ;
 $\mathcal{H} := \text{hierarchies}(G')$ ;
foreach  $H[V_H, A_H] \in \mathcal{H}$  do
     $P_H[V_H, A_H] := \text{upwardPlanarize}(H)$ ;
     $P_H^s := \text{makeSTGraph}(P_H)$ ;
     $P_H^i := \text{insertEdges}(P_H^s, A_H)$ ;
end;
 $\mathcal{P} := \{P_H \mid H \in \mathcal{H}\}$ ;
 $\Gamma := \text{makeClusterGraph}(G', \mathcal{P})$ ;
removeDummies( $\Gamma$ );
drawOrthogonal( $\Gamma$ );
return  $\Gamma$ ;
end.
    
```

Artikel

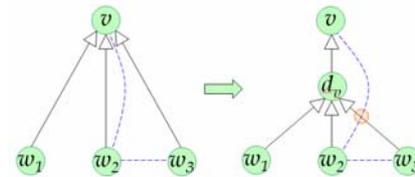
Gutwenger, Jünger, Klein, Kupke, Leipert, Mutzel:
A New Approach for Visualizing UML-Class Diagrams, SOFTVIS 2003: 179-188, 217-218.

Software

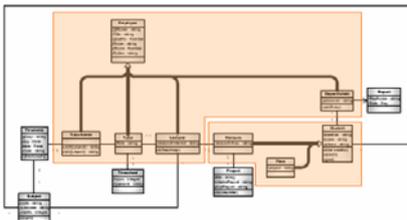
GoVisual-Algorithmus
 oreas GmbH
<http://www.oreas.de>

Schritte

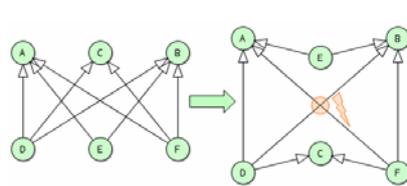
Vorbereitung:



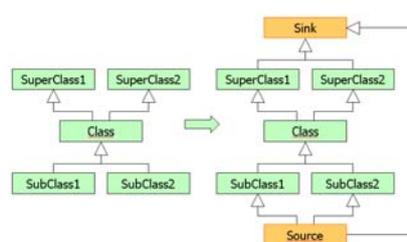
Hierarchien finden:



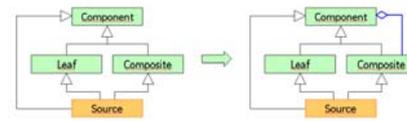
Gleichrichten und Planarisieren:



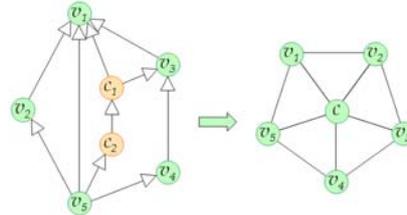
ST-Graph erstellen:



Assoziationen einfügen:

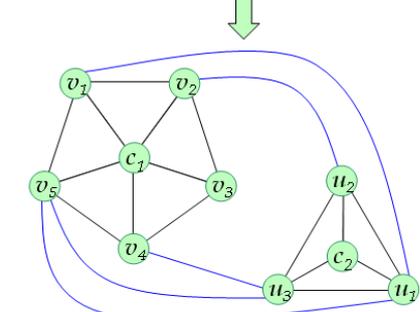
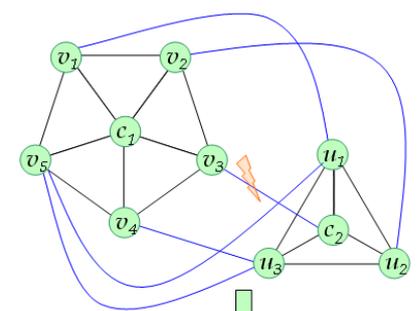


Wheel-Graph bilden:

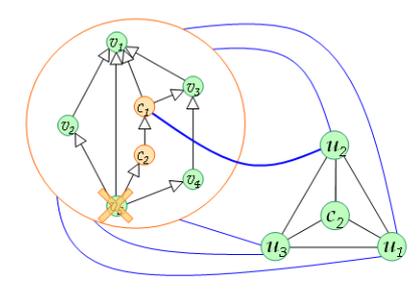


Schritte

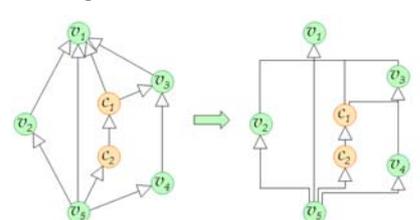
Planare Einbettung finden:



Wheel-Graph ersetzen:



Orthogonalisieren:



Kontakt und Folien

Christian M. Meyer
christian.meyer@stud.tu-darmstadt.de
<http://uni.christian-meyer.org>