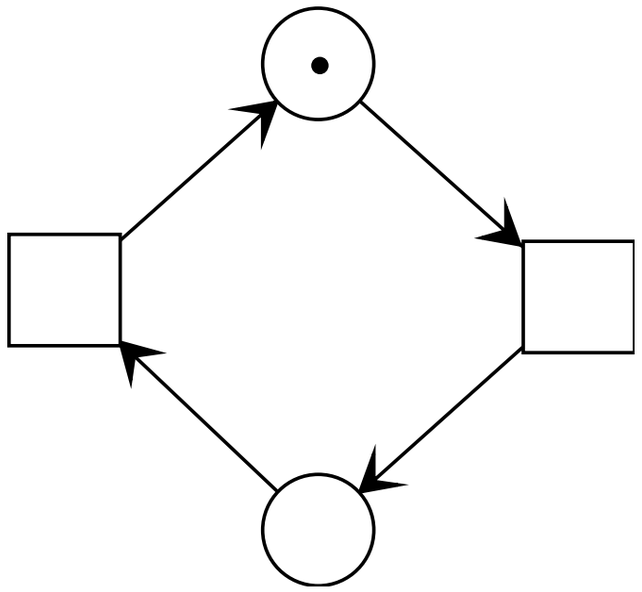


WEBBASIERTE ANALYSE UND SYNTHESE VON PETRINETZEN UND TRANSITIONSSYSTEMEN

ADRIAN JAGUSCH

VORGEHENSMODELLE

- Strukturierter Prozess zur (Software-)Entwicklung
- Nach **Erwin Grochla** (1921 – 1986)

- Voruntersuchung
- Ist-Aufnahme
- Ist-Kritik
- Sollkonzeption
- Umsetzung
- Evaluierung und Weiterentwicklung

INHALT

- Theoretische Grundlagen
- Ist-Aufnahme
- Ist-Kritik
- Sollkonzeption
- Umsetzung
- Evaluierung und Weiterentwicklung

THEORETISCHE GRUNDLAGEN

- Ein Petrinetz ist ein Quintupel PN = (P, T, F, B, s_0)

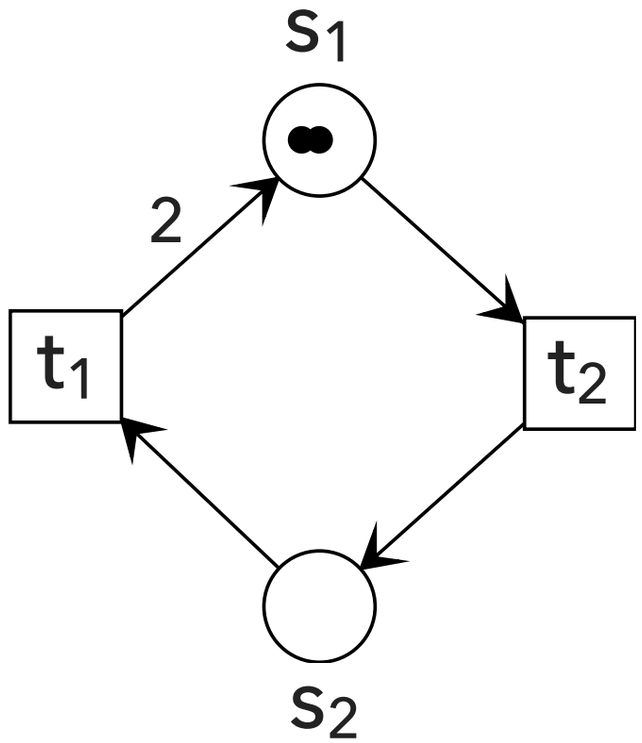
P: Menge der Stellen

T: Menge der Transitionen

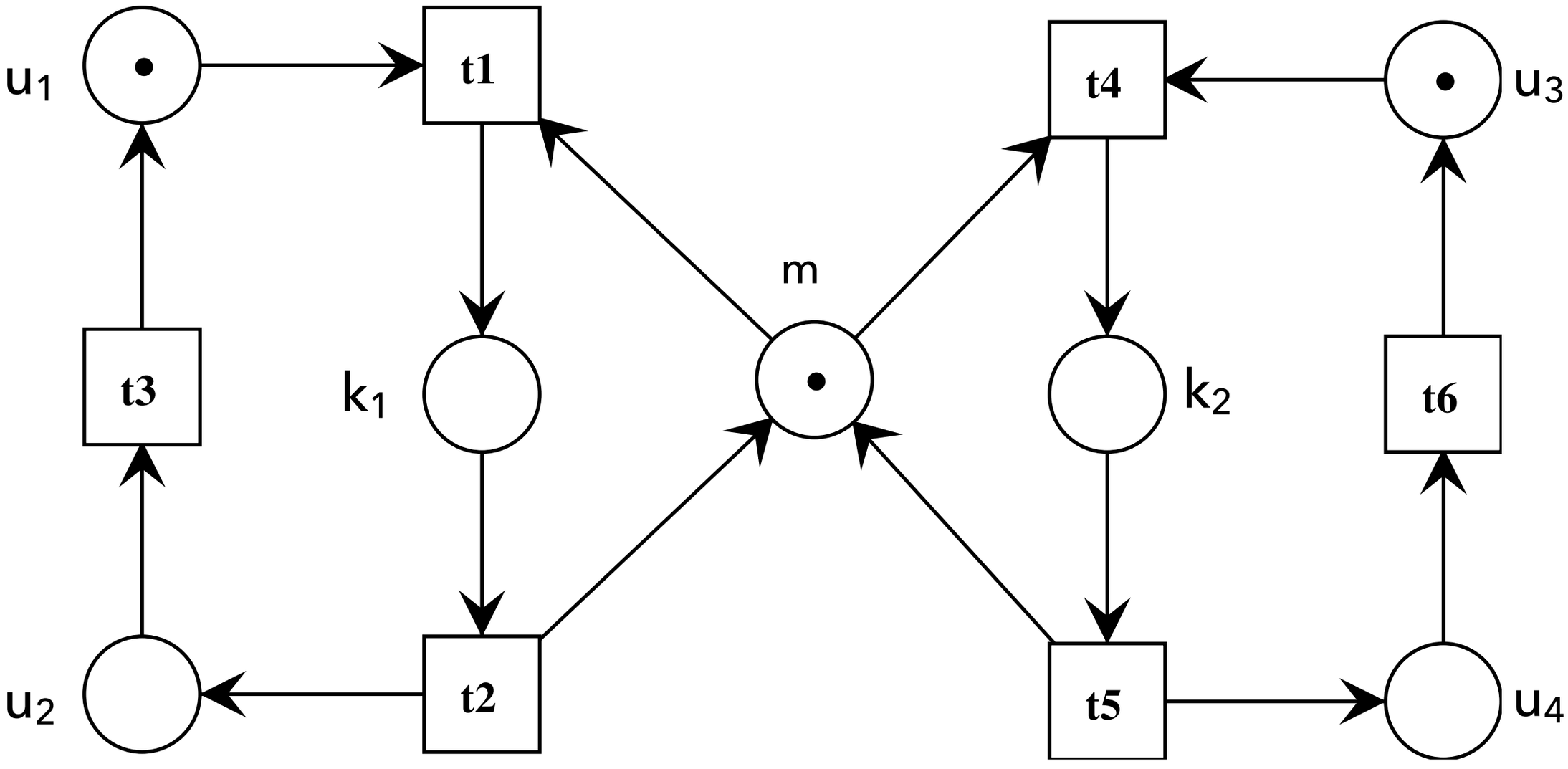
F: Forward-Matrix

B: Backward-Matrix

s_0: Initialzustand

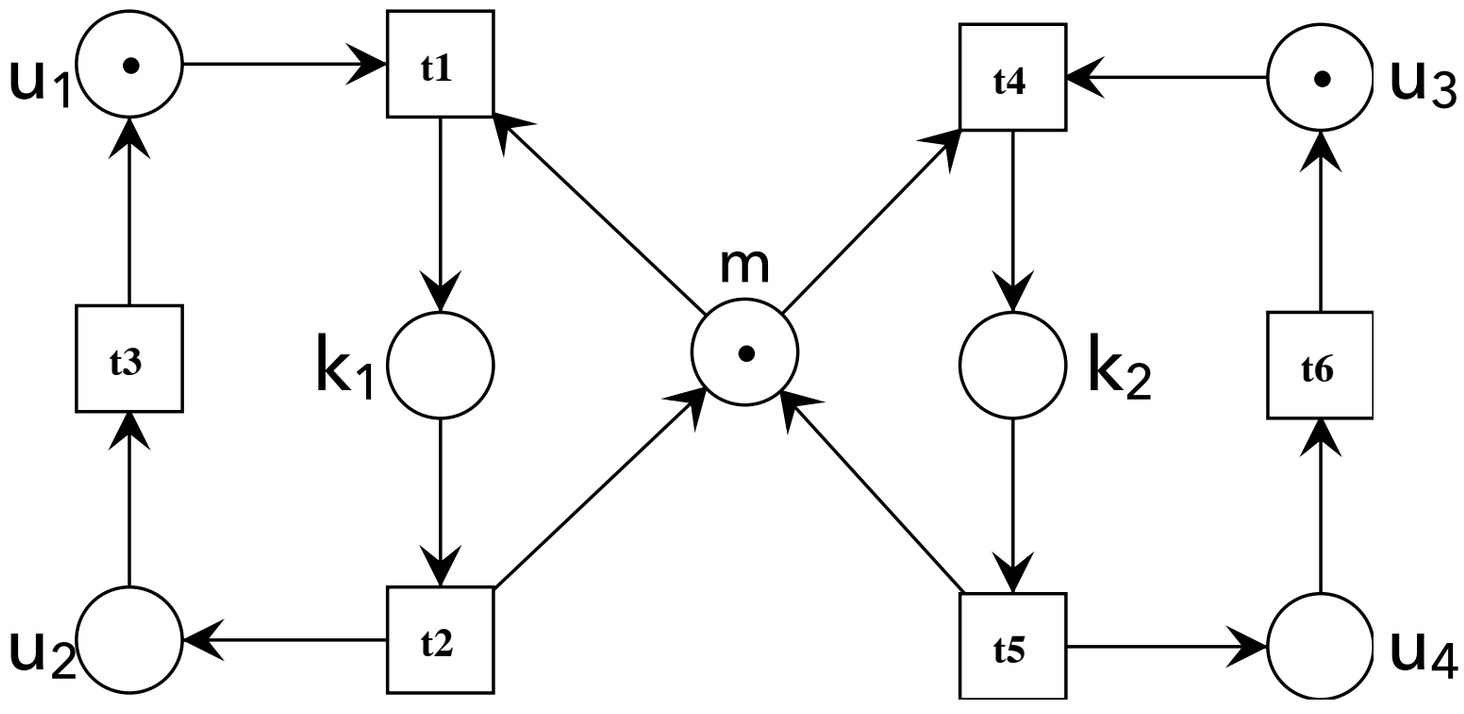


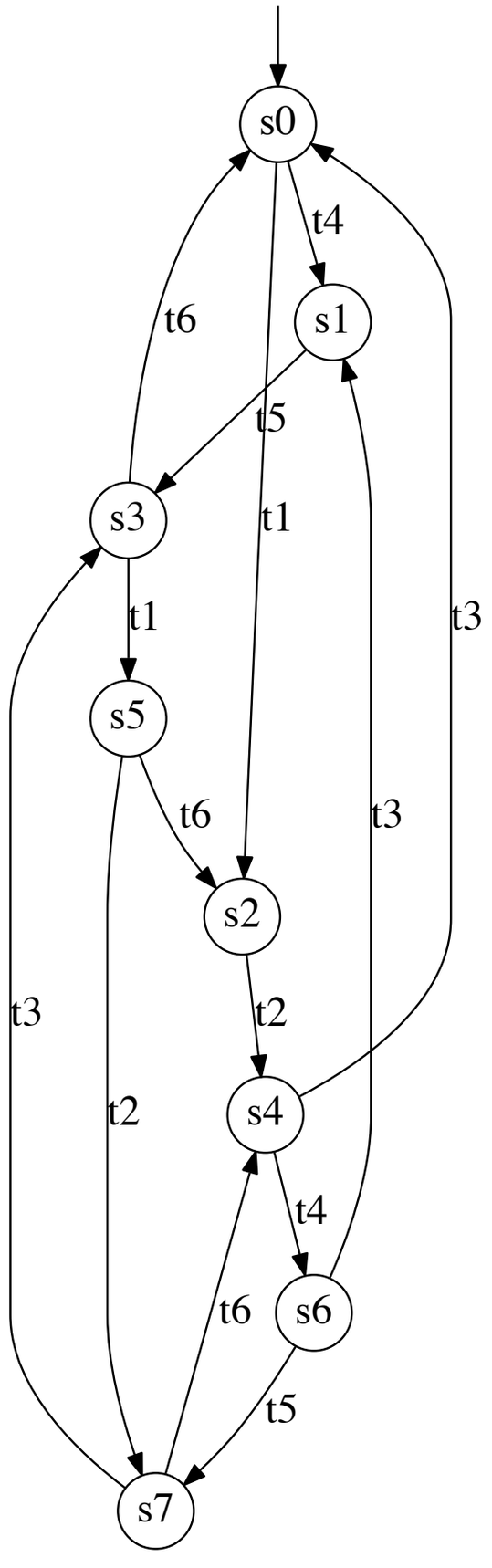
WECHSELSEITIGER AUSSCHLUSS



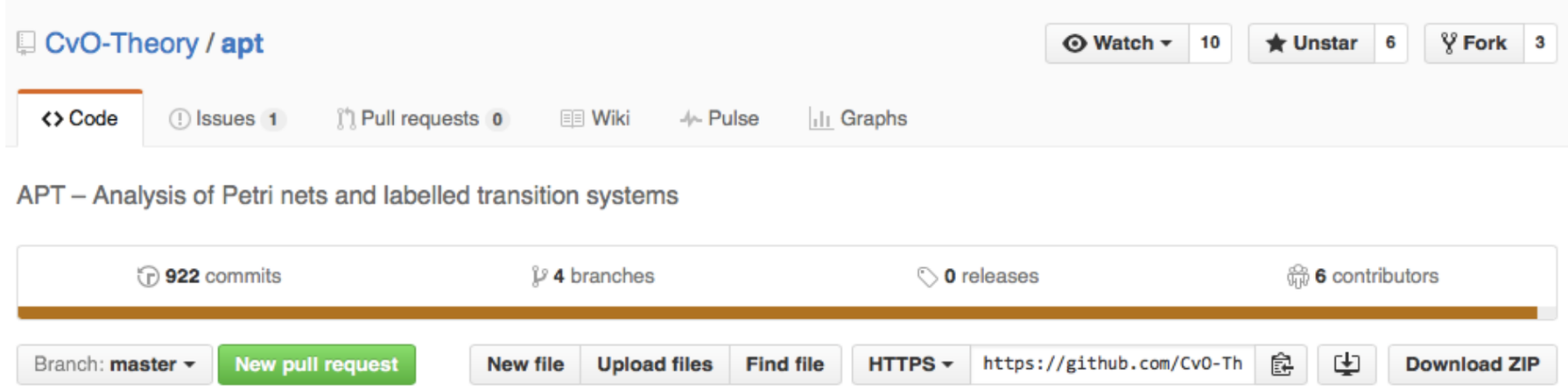
TRANSITIONSSYSTEME

- Erreichbarkeitsgraphen
- Überdeckungsgraphen





IST-AUFNAHME



- APT (Analyse von Petri-Netzen und Transitionssystemen)
- Entstanden 2012 im Rahmen einer PG
- In der Abteilung **Parallele Systeme** im Einsatz
  - Erweiterung: Synthese

APT

apt — -bash — 154×39

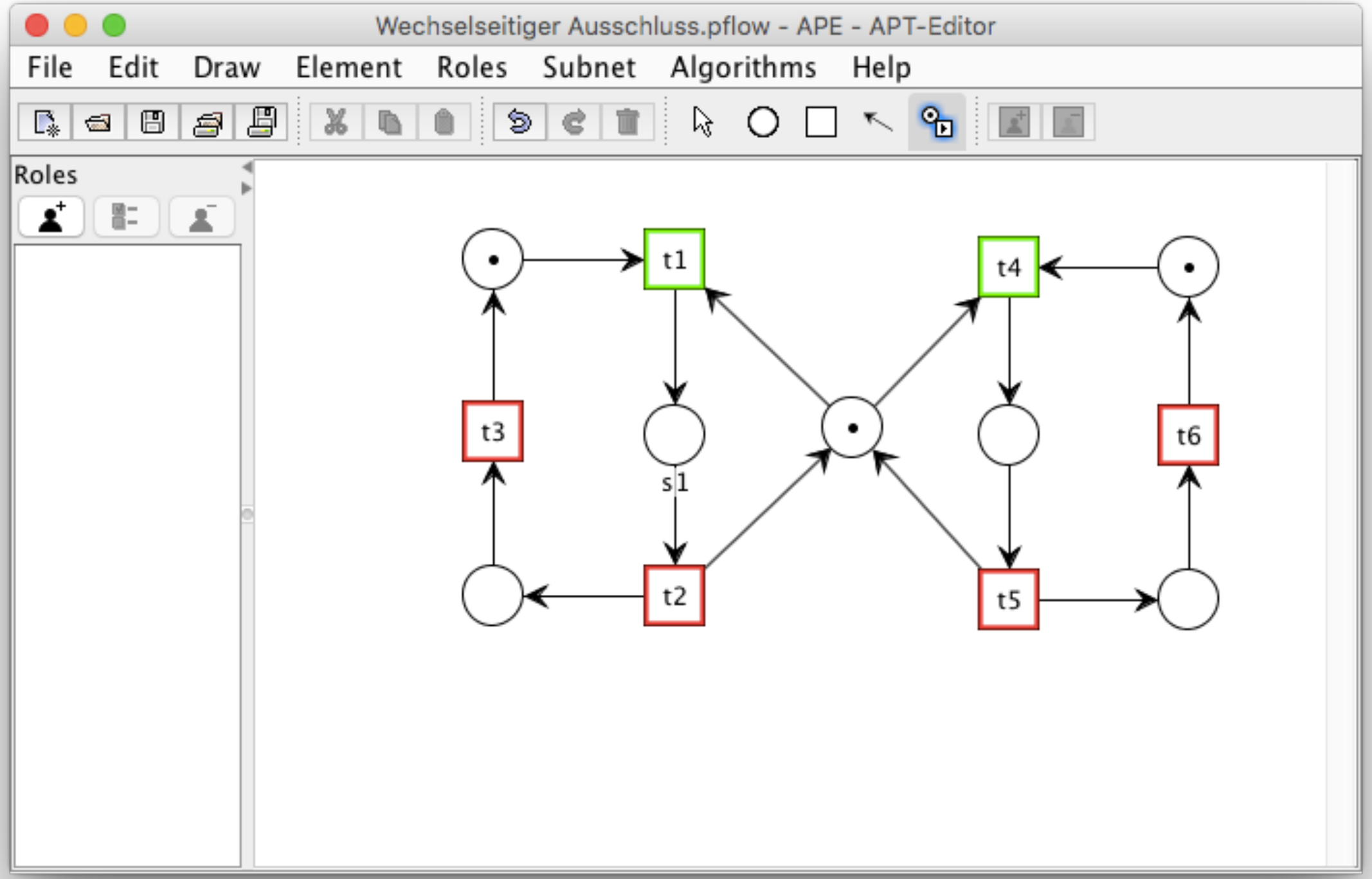
```
openwall:apt adrian-jagusch$ java -jar apt.jar
Usage: apt <module> <arguments>

Available modules:

Miscellaneous
=============
  draw                          Convert a Petri net or LTS to the Dot format used by Graphviz
  example_module                Lowercase a string
  help                          Get information about a module

Petri net
=========
  ac                            Check if a Petri net is asymmetric-choice
  backwards_persistent          Check if a Petri net or LTS is persistent
  bcf                           Check if a Petri net is behaviourally conflict free (BCF)
  bicf                          Check if a Petri net is binary conflict free (BiCF)
  bisimulation                  Check if the reachability graphs of two bounded labeled Petri nets or of two LTS or a combination of both are bisimilar
  bounded                       Check if a Petri net is bounded or k-bounded
  cf                            Check if a plain Petri net is conflict-free
  check                         Search for a Petri net which fulfills the given attributes
  check_all_cycle_prop          Check all cycle properties of a Petri net
  compute_pvs                   Compute parikh vectors of smallest cycles of a Petri net or LTS
  concurrent_coverability_graph Calculate the concurrent coverability graph of a Petri net in the step semantics
  conpres                       Check if a Petri net is concurrency-preserving
  coverability_graph            Compute a Petri net's coverability graph
  covered_by_invariant          Check if a Petri net is covered by an S-invariant or a T-invariant
  cycles_same_disjoint_pv       Check if the smallest cycles of a Petri net or LTS have the same or mutually disjoint parikh vectors
  cycles_same_pv                Check if the smallest cycles of Petri net or LTS have the same parikh vector
  examine_pn                    Perform various tests on a Petri net at once
  fc                            Check if a Petri net is free-choice
  fire_sequence                 Try to fire a given firing sequence on a Petri net.
  homogeneous                   Check if a Petri net is homogeneous
  info                          Report basic statistics about a Petri net.
  invariants                    Compute a generator set of S- or T-invariants
  isolated                      Check if a Petri net contains isolated elements
  isolated_elements             Find isolated elements in a graph
  isomorphism                   Check if two Petri nets have isomorphic reachability graphs
  k_bounded                     Find the smallest k for which a Petri net is k-bounded
```

APT

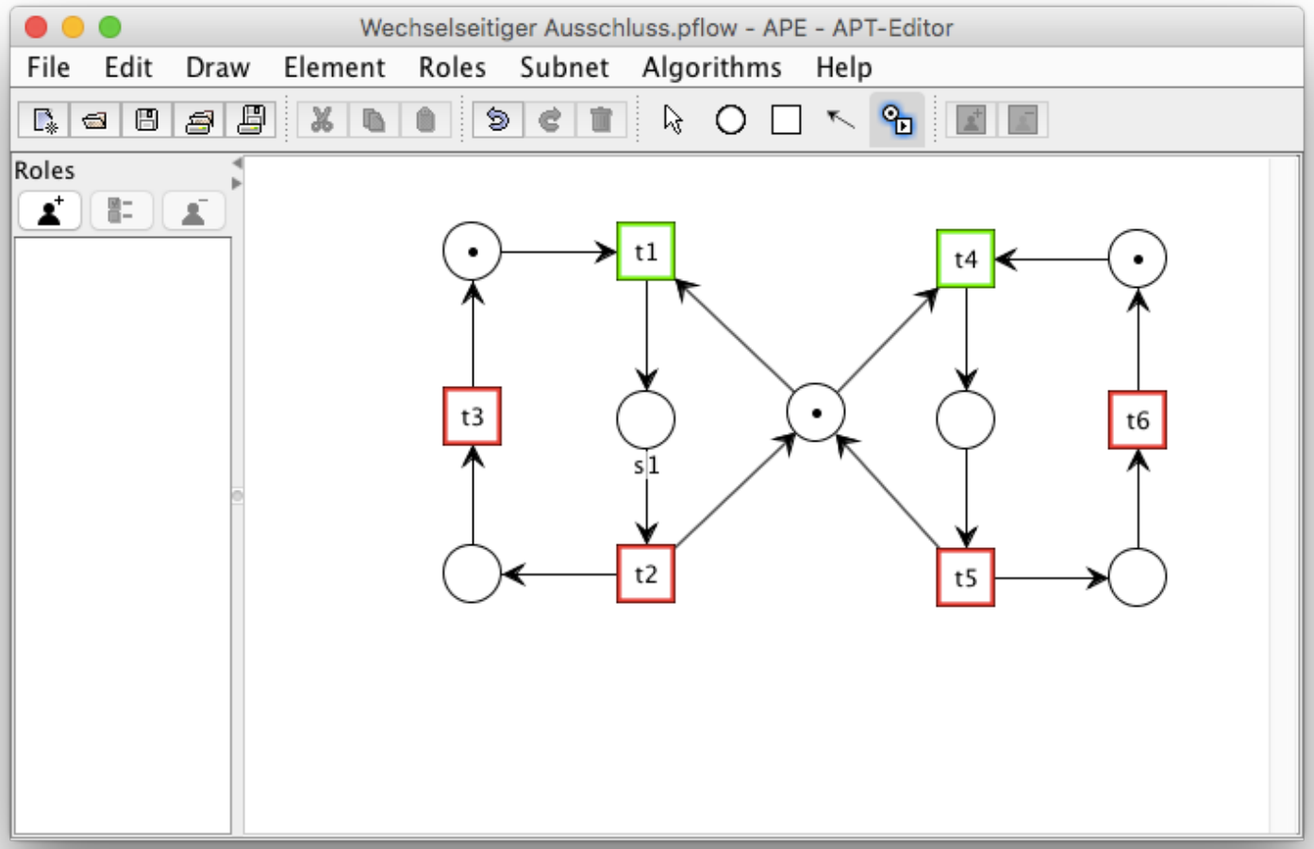
Wechselseitiger Ausschluss.apt

```
.name ""
.type LPN

.places
p0
p1
p2
p3
p4
p5
p6

.transitions
t0 [label="t5"]
t1 [label="t4"]
t2 [label="t3"]
t3 [label="t6"]
t4 [label="t2"]
t5 [label="t1"]

.flows
t0: {1*p0} -> {1*p6, 1*p2}
t1: {1*p1, 1*p6} -> {1*p0}
t2: {1*p3} -> {1*p5}
t3: {1*p2} -> {1*p1}
t4: {1*p4} -> {1*p3, 1*p6}
t5: {1*p5, 1*p6} -> {1*p4}

.initial_marking {1*p1, 1*p5, 1*p6}
```

APE (APT-EDITOR)



APE (APT-EDITOR)



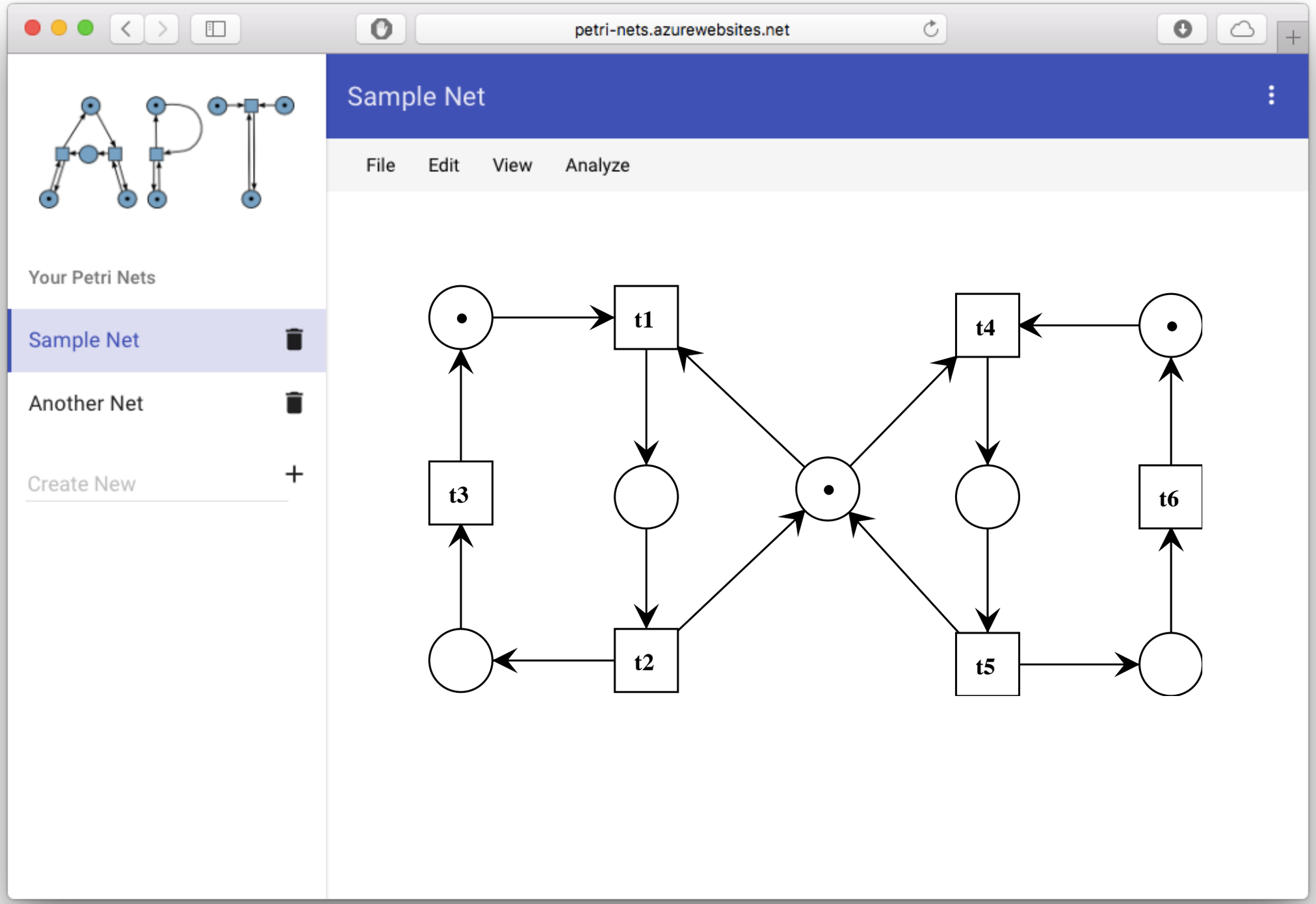
- Entstanden 2013 (Bachelorarbeit von Hillit Saathoff)
- Auf Basis des PNEditors

IST-KRITIK

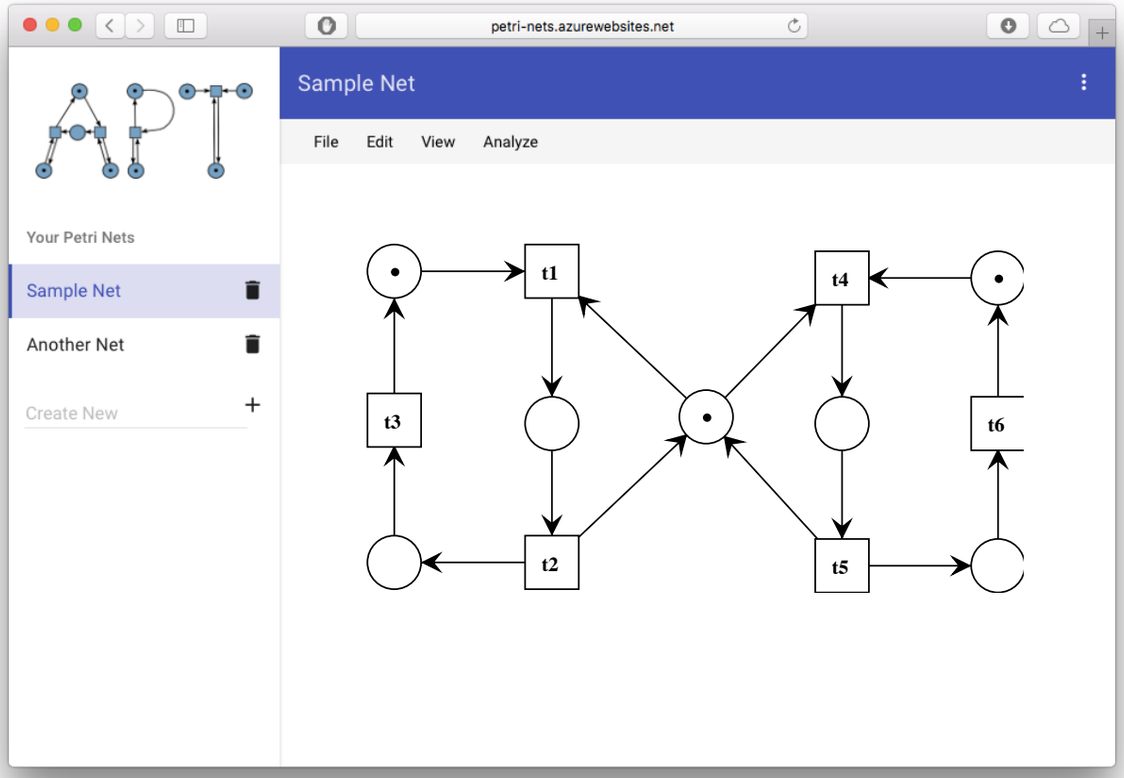
- Download des APT-GIT-Repositorys
- Kompilieren des Java-Projekts
- Download des APE-GIT-Repositorys
- Kompilieren des APE-GIT-Repositorys
- Erstellen des Petrinetzes in APE
- Exportieren des Petrinetzes als .apt-Datei
- Mit APT Petrinetz in Transitionssystem überführen
- Transitionssystem mit APT in .dot-Daten konvertieren
- Transitionssystem mit Graphviz visualisieren

WHAT ABOUT MOBILE?

SOLLKONZEPTION



SOLLKONZEPTION



- Responsive Webapp
- Auf Mobilgeräten nutzbar
- Code OpenSource auf GitHub, aber direkt nutzbar
- Analyse- und Synthese-Werkzeuge direkt aus dem Browser heraus nutzbar

TECHNOLOGIEN

GitHub

DEMO

(FRÜHER PROTOTYP)

ZEITPLAN

Dauer	Inhalte
1 Woche	Grundlagenrecherche, Entwicklungsumgebung
1 Woche	Einführung schreiben
1 Woche	Anforderungsanalyse
2 Woche	Framework-Evaluierung, Software-Architektur
4 Woche	Implementierung des Editors
3 Woche	Serverseitige Implementierung (APT-Anbindung)
1 Woche	Benutzertests und Verbesserungen
1 Woche	Schreiben von Einleitung und Fazit
2 Wochen	Korrekturlesen und Puffer
16 Wochen	

QUELLEN

- Best, Eike und Schlachter, Uli: *Analysis of Petri nets and transition systems*. In Proc. 8th Interaction and Concurrency Experience (ICE), Grenoble, 2015.
- Desel, Jorg und Javier Esparza: *Free choice Petri nets*. Vol. 40. Cambridge university press, 2005.
- Grochla, Erwin: *Grundlagen der organisatorischen Gestaltung*. Stuttgart: Poeschel, 1982.
- Saathoff, Hillit: *Ein grafisches Frontend für eine Algorithmensammlung zur Analyse von Petrinetzen und Transitionssystemen*. Carl von Ossietzky Universität Oldenburg, 2013.
- Schlachter, Uli u. a.: *Projektgruppe APT – Analyse von Petri-Netzen und Transitionssystemen*. Carl von Ossietzky Universität Oldenburg, 2013.

SOFTWARE

- AngularJS: http://angularjs.org
- D3: http://d3js.org
- Jetty: http://www.eclipse.org/jetty/
- GitHub: http://github.com
- Travis CI: http://travis-ci.org

ICONS

- Icon-Set unter CC-Lizenz: http://pelfusion.com/media-long-shadow-icons/

DANKE FÜR IHRE AUFMERKSAMKEIT!