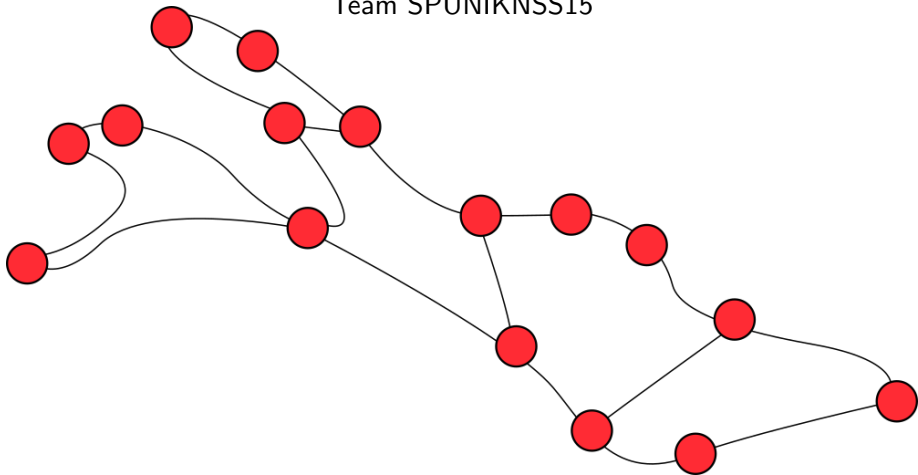


# ISGCI und Subgraph-Isomorphismen

Team SPUNIKNSS15



# *Information System on Graph Classes and their Inclusions*

## Aufbau und Funktionalität

- ▶ Java Projekt zur Entdeckung und Analyse von Beziehungen zwischen Graph-Klassen
- ▶ Relevant für unser Projekt:
  - ▶ `teo.isgci.smallgraph`: Implementierung von Smallgraphs, Configurations, Families
  - ▶ `smallgraphs.xml`: Große Sammlung von Smallgraphs, Configurations, Families
  - ▶ `teo.isgci.appl.FindISG`: Applikation zum Finden induzierter Subgraphen von Konfigurationen und Familien.  
Resultat: `output.xml`

# Der VF2 Algorithmus

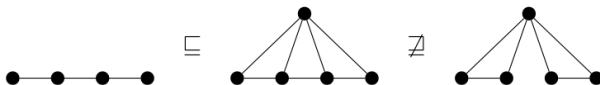
- ▶ VF2:

$$(G_1, G_2) \mapsto \begin{cases} \text{True,} & \text{falls } G_2 \text{ isomorph zu induz. Subgraph von } G_1 \\ \text{False,} & \text{sonst} \end{cases}$$

- ▶ VF2 ist

- ▶ allgemein
- ▶ vergleichsweise effizient
- ▶ und erlaubt semantische Matchings

(Cordella, Foggia, Sansone, Vento).



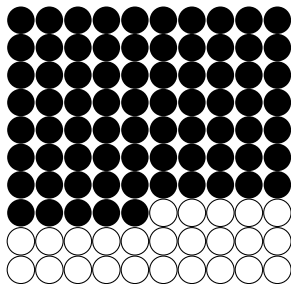
- ▶ Zwei getrennte Aufgabenfelder: ISGCI und VF2
- ▶ Deshalb: Aufteilung des Teams und der Milestones
  - ▶ **Milestone 1:** Implementierung des VF2-Algorithmus
  - ▶ **Milestone 2:** Ersetzung der Adjazenzmatrizen in ISGCI durch `jgraphT`-Objekte
  - ▶ **Milestone 3:** Integration des VF2 in ISGCI

# Herausforderungen

1. Analysephase
2. Teamkoordination
3. Tools

# Laufzeit von FindISG - (1)

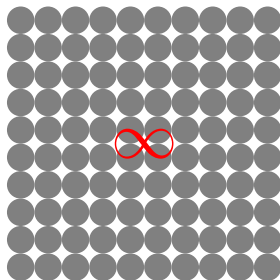
Vor unseren Änderungen.



**150 Minuten**

## Laufzeit von FindISG - (2)

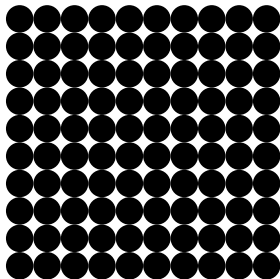
Verwendung der ersten Version des Java-VF2 anstelle der C++ Version.



über 12 Stunden

## Laufzeit von FindISG - (3)

Weglassen der Erzeugung sämtlicher Subgraphen, stattdessen direkter Test auf Subisomorphie.

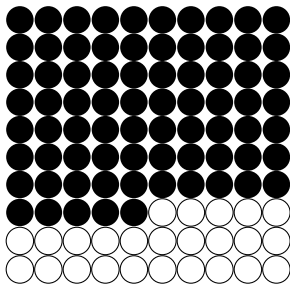


200 Minuten



## Laufzeit von FindISG - (4)

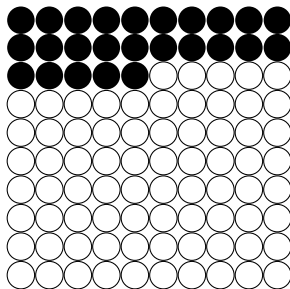
Konsequente Verwendung von VF2 in allen Isomorphietests, VF2 sortiert zudem nach Knotengrad vor.



**wieder 150 Minuten**

## Laufzeit von FindISG - (5)

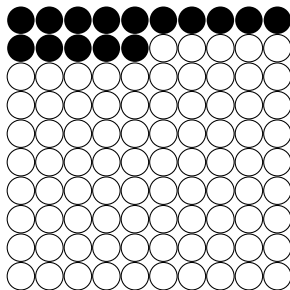
Parallelisierung der Subisomorphieberechnung für große Graphen.



**50 Minuten**

# Laufzeit von FindISG - (6)

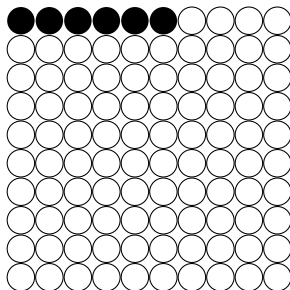
Caching von Kanten im VF2.



**30 Minuten**

# Laufzeit von FindISG - (7)

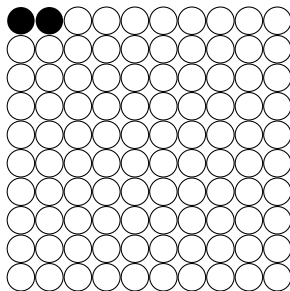
Isomorphietest auf dem Graphen bzw. Komplement mit weniger Kanten.



**12 Minuten**

# Laufzeit von FindISG - (8)

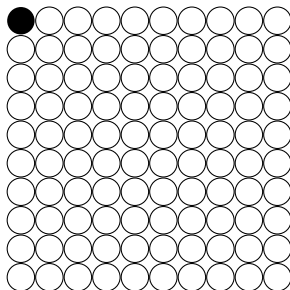
Weiterentwicklung des VF2...



**4 Minuten**

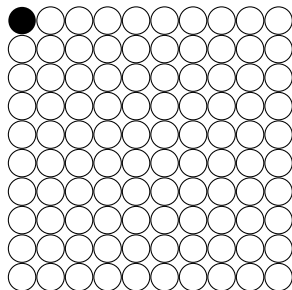
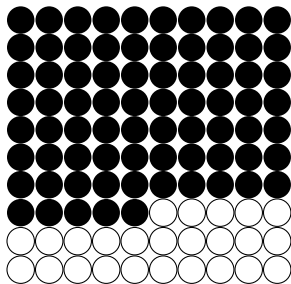
# Laufzeit von FindISG - (9)

...VF2 nochmals verbessert.



**2 Minuten**

# FindISG - Resultat



**Von 150 Minuten auf 2 Minuten**