

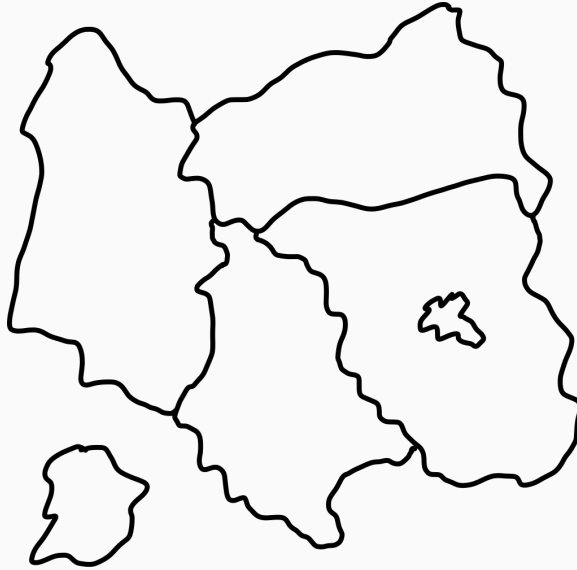
# Einführung Constraints

---

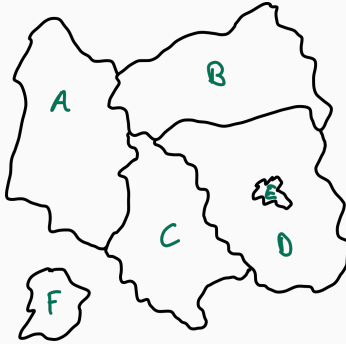
Carsten Gips (HSBI)

Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

## Motivation: Einfärben von Landkarten



# Einfärben von Landkarten: Formalisierung

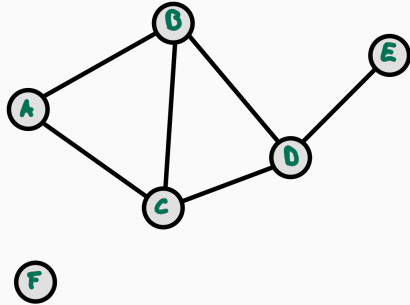
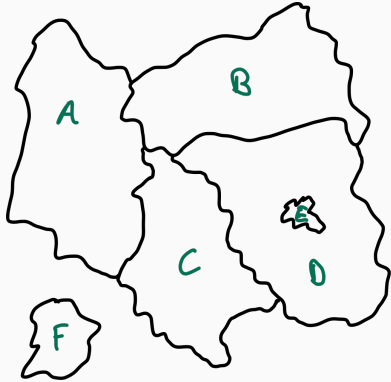


- **Variablen:** A, B, C, D, E, F
- **Werte:** {*red*, *green*, *blue*}
- **Constraints:** Benachbarte Regionen müssen unterschiedliche Farben haben
- **Mögliche Lösung:** Zuweisung an Variablen ("Belegung")  
 $\{A = \textit{red}, B = \textit{blue}, C = \textit{green}, D = \textit{red}, E = \textit{blue}, F = \textit{blue}\}$

# Definition: Constraint Satisfaction Problem (CSP)

- Ein CSP  $\langle V, D, C \rangle$  besteht aus:
  - Menge von **Variablen**  $V = \{V_1, V_2, \dots, V_n\}$
  - Je  $V_i$  nicht leere **Domäne**  $D_i = \{d_{i,1}, d_{i,2}, \dots, d_{i,m_i}\}$
  - Menge von **Constraints**  $C = \{C_1, C_2, \dots, C_p\}$   
(Randbedingungen, Abhängigkeiten zwischen Variablen)
- Zuweisung/Belegung (*Assignment*)  $\alpha$ :
  - Zuweisung von Werten an (einige/alle) Variablen:  $\alpha = \{X = a, Y = b, \dots\}$   
(aus den jeweiligen Wertebereichen)
  - **Konsistente Belegung**: Randbedingungen sind nicht verletzt
  - **Vollständige Belegung**: Alle Variablen sind belegt
- **Lösung** eines CSP: Vollständige und konsistente Belegung

# Constraint-Graph



- **unär:** betrifft einzelne Variablen

Beispiel:  $A \neq red$

- **binär:** betrifft Paare von Variablen

Beispiel:  $A \neq B$

- **höhere Ordnung:** betrifft 3 oder mehr Variablen

- **Präferenzen:** “soft constraints”

Beispiel: “rot ist besser als grün”

Abbildung über Gewichtung  $\Rightarrow$  Constraint-Optimierungsproblem (COP)

# Constraints – Wertebereiche

- **Endliche Domänen:**  $d$  Werte  $\Rightarrow O(d^n)$  mögliche Zuweisungen  
(exponentiell in der Zahl der Variablen)
- **Unendliche Domänen:** reelle Zahlen, natürliche Zahlen  
 $\Rightarrow$  Keine Auflistung der erlaubten Wertekombinationen mehr möglich  
 $\Rightarrow$  Übergang zu Gleichungen/Ungleichungen:  $job_1 + 5 < job_2$ 
  - lineare Constraints
  - nichtlineare Constraints

## Historische Unterscheidung:

- **Constraint Satisfaction:** endliche Domänen, kombinatorische Methoden
- **Constraint Solving:** unendliche Domänen

# CSP sind überall ...

- Stundenpläne (Klassen, Räume, Zeiten)
- Konfiguration (Computer, Autos, ...)
- Fahrpläne (Zug, Flug, ...)
- Planung von komplexen Projekten
- Sudoku :-)
- ...



- Definitionen und Begriffe:
  - Variable, (un-) endliche Domänen, Wertemenge
  - Constraint, Arität, CSP
  - Zuweisung, Lösung, ...



Unless otherwise noted, this work is licensed under CC BY-SA 4.0.