ALADIN: Generator für SQL-Aufgaben und Lösungshilfen – von der Syntaktik zur Semantik

„(Didaktische) Herausforderung“ vor ALADIN

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

- nur wenige Übungsaufgaben
- kaum unbekannte Aufgaben zum selbständigen Üben
- keine Skalierung der Aufgaben hinsichtlich Schwierigkeitsgrades und Umfangs
- keine Musterklausuren zu Prüfungsvorbereitung
- Lösungshilfen nur durch Lehrenden möglich ➔ erheblicher Aufwand
- keine motivierenden Impulse für Lernprozesse
- keine orts- und zeitflexible Lehre
- keine Selbstkontrolle beim Lernen durch Abgleich mit Musterlösungen
- kein selbstorganisiertes und selbsttätiges Lernen

„(Didaktische) Ziele“ von ALADIN

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

- bekannte Lösungsansätze wiederholt selbsttätig auf zufällig generierte Probleme anwendbar
- Orientierung des Schwierigkeitsgrads an individueller Leistungsfähigkeit
- leistungsgerechte Aufgaben für heterogene Zielgruppen
- hohe Problemlösungskompetenz der Studierenden ➔ höherer Studienerfolg
- Generierung von Online-Selbsttests und elektronischen Test- oder Probeklausuren und sofortiges automatisches und leistungsabhängiges Feedback ➔ weniger Aufwand
- fachlich und zeitlich unbegrenzt wiederverwendbar
- Generierung der Aufgaben parametrisier- und somit den Lehrinhalt aktiv mitgestaltbar
- Lernen mit eigener Geschwindigkeit
- zeitlich, räumlich und institutionell flexibel nutzbar
- Erweiterbarkeit um neue Aufgabentypen
- Vernetzung der Studierenden
- Feedback an/von Lehrende/n
- …

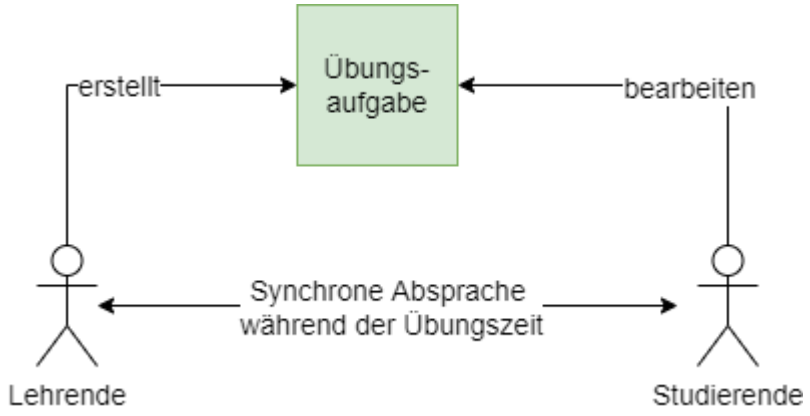
Derzeitiger Leistungsumfang von ALADIN

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

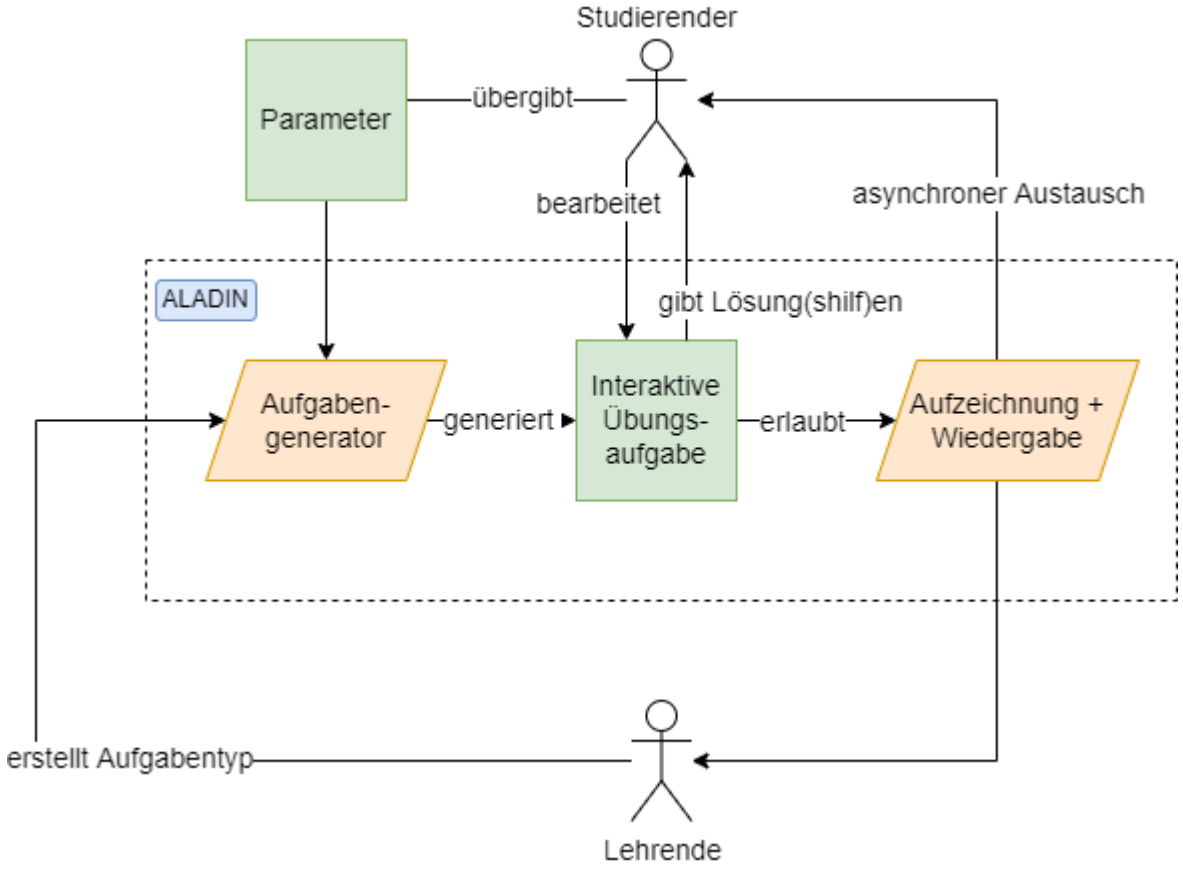
- unterstützte Aufgabentypen:
  - Stücklistenauflösung mittels dreier, unterschiedlicher Verfahren
  - **SQL-Abfragen**
  - Geostatistische Interpolationsverfahren (Inverse Distanzwichtung)
  - Shortest-Path-Algorithmen (Dijkstra)
- Aufzeichnung, Wiedergabe und Fortführung von Lösungsversuchen
- zum großen Teil deklarative Erstellung neuer Aufgabentypen

Lernen und Lehren mit und ohne ALADIN

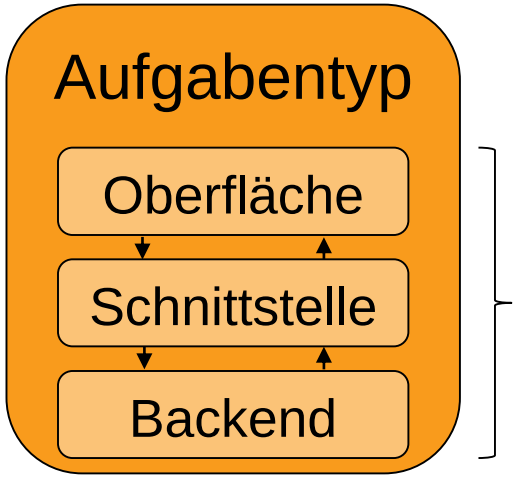
Ablauf ohne ALADIN



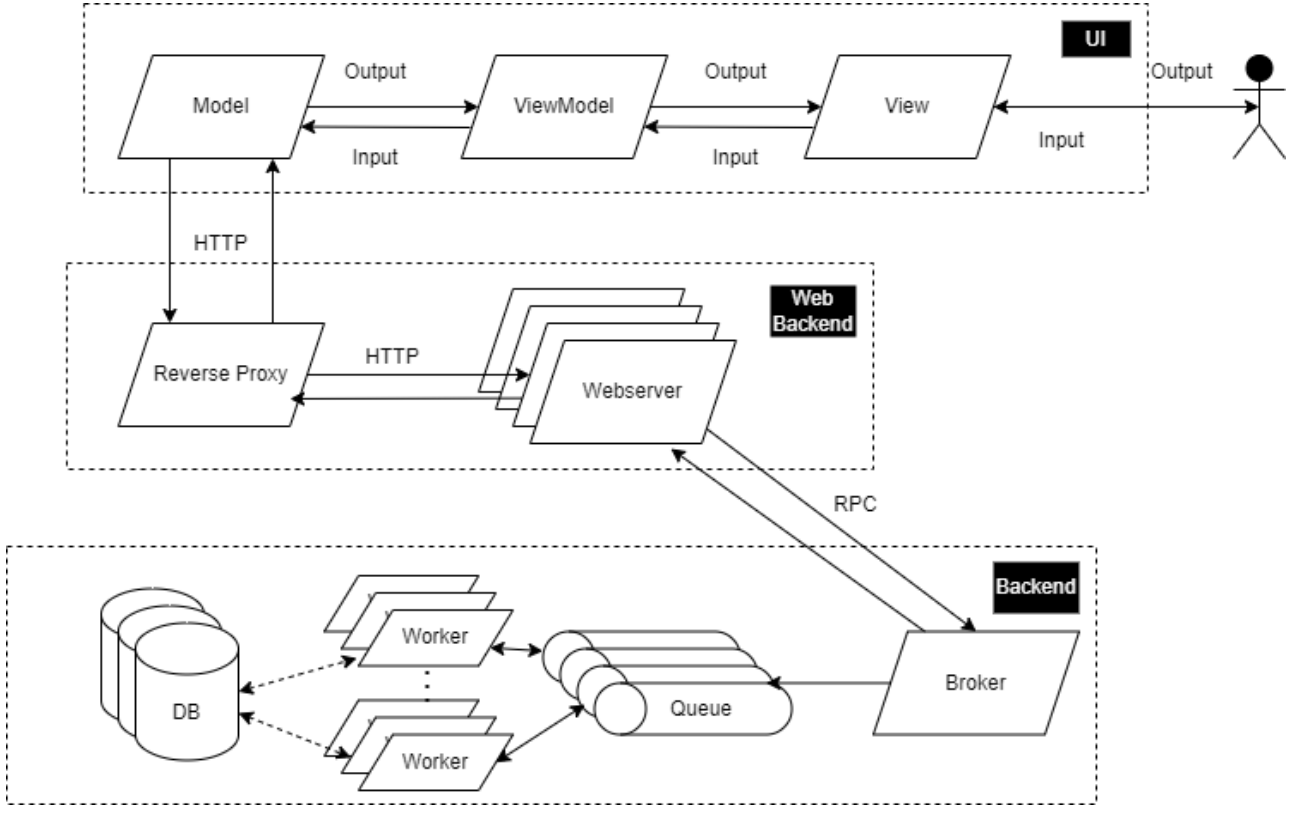
Ablauf mit ALADIN



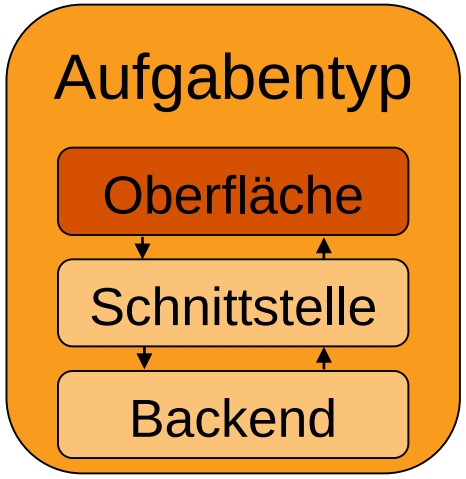
Hinter den Kulissen von ALADIN

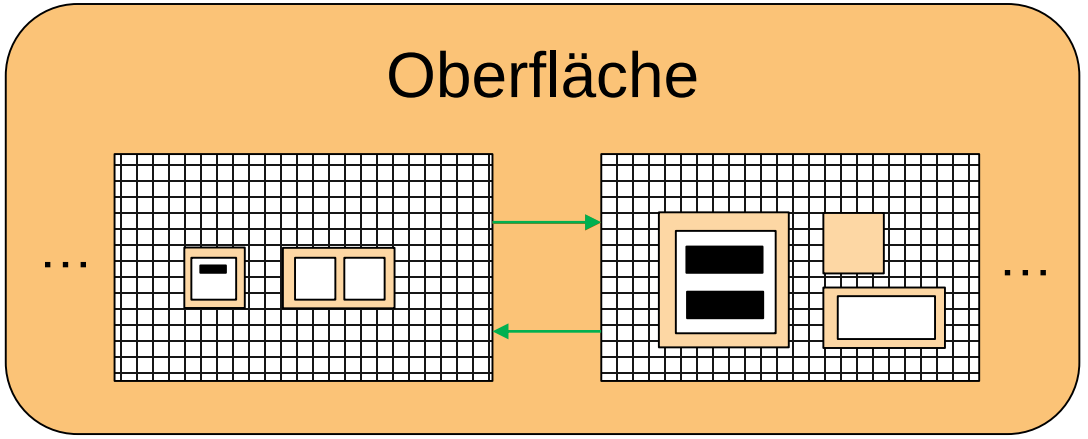


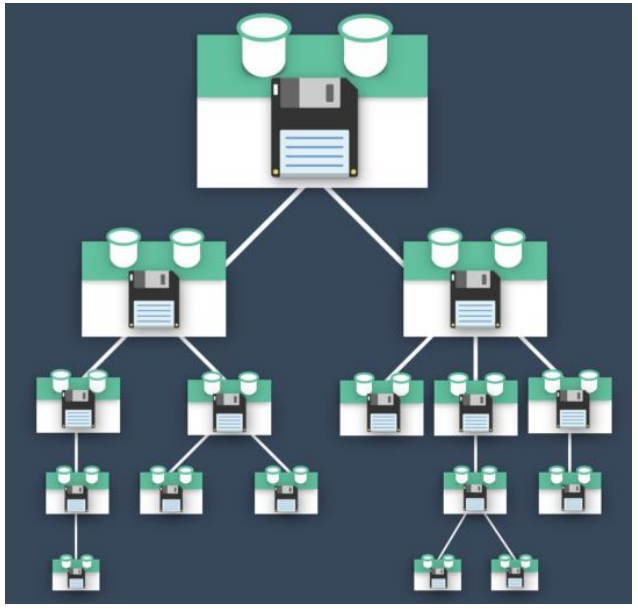
Deklarativ konfigurierbar
im JSON-Format

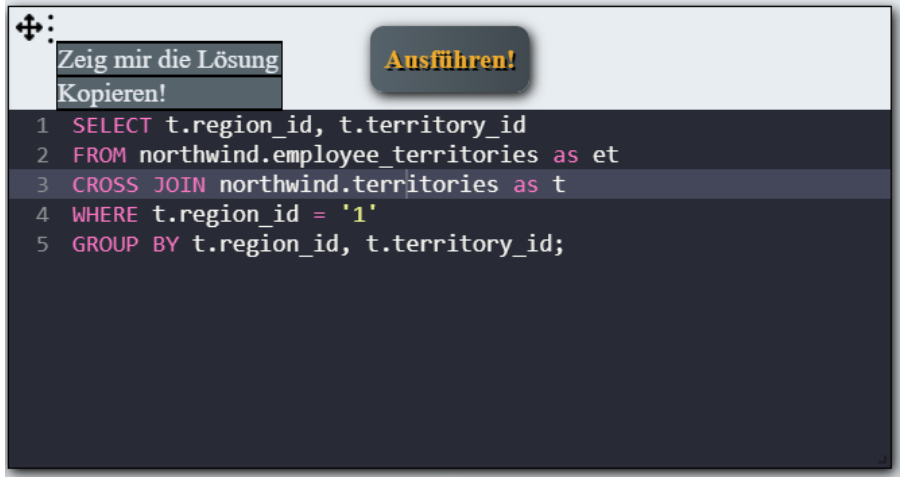


Hinter den Kulissen von ALADIN



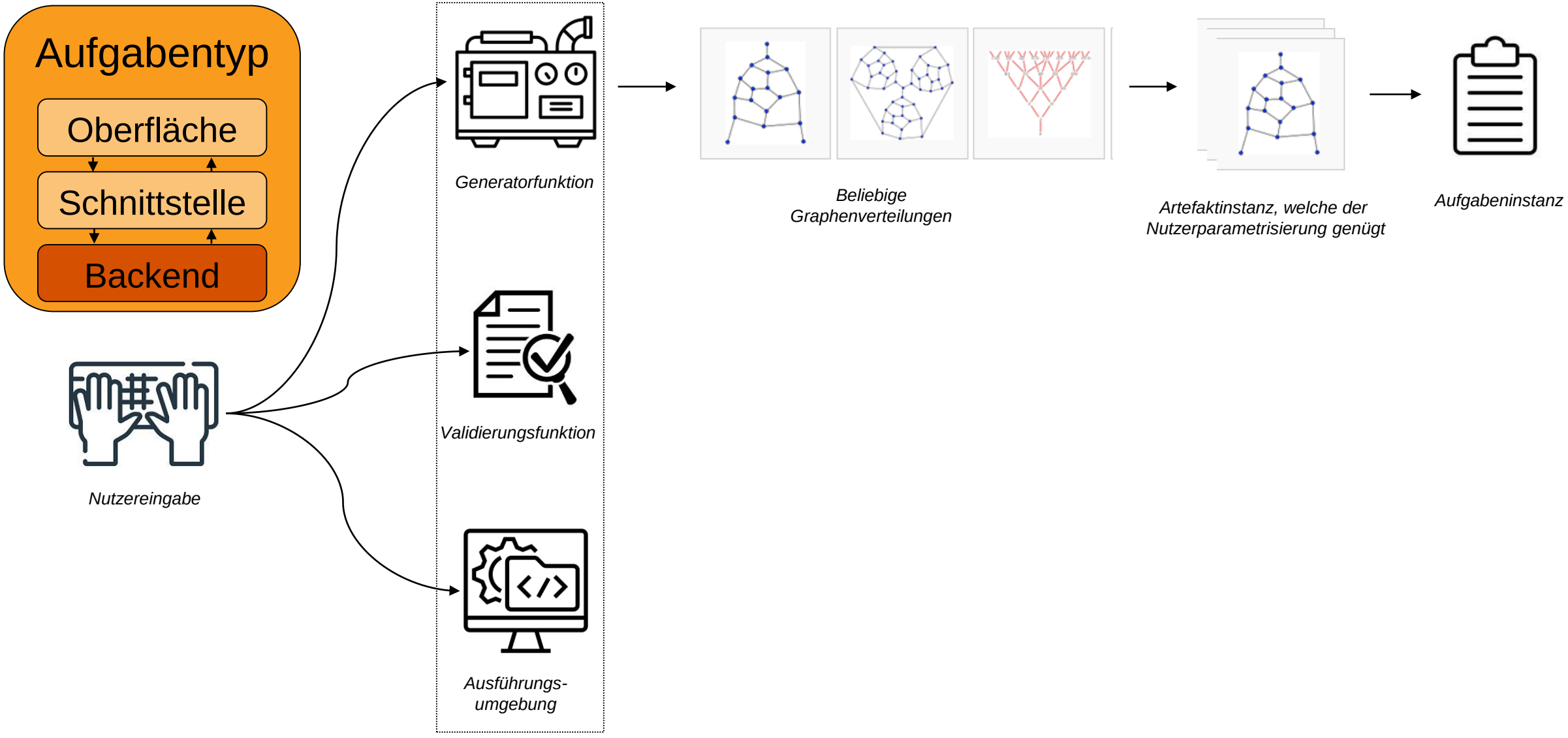




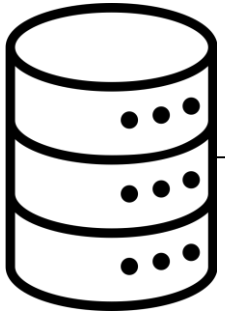


- Möglichkeiten:
  - „Atomare“ Fehlerbehandlung und Lösungshilfe
  - Aufzeichnung aller Interaktionen
  - Abspielen aller Interaktionen
  - Wiedereinstieg an beliebiger Stelle
  - Vervollständigung des Lösungsversuchs als neue Aufzeichnung

Hinter den Kulissen von ALADIN

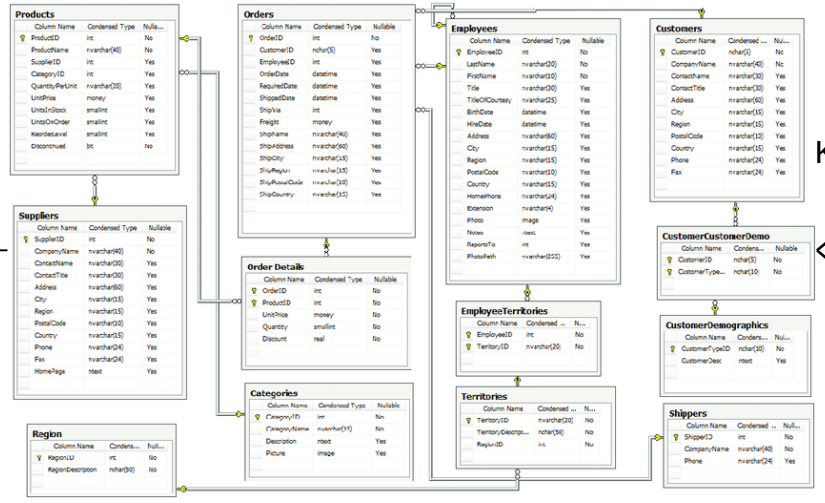


Algorithmus zur Generierung von SQL-Statements



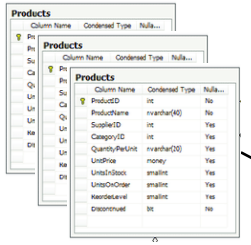
Importierte DB

Introspection

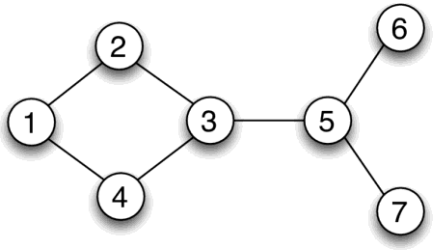


Knoten

Kanten



Transformation



Traversierung

(Abbildung der Joins)

(1, 2)

(1, 4)

(2, 3)

:

(1,2,3,4,5,7)

Zufallswahl

(unter Berücksichtigung der maximalen Join-Länge)

(1, 2, 3)

Zufallswahl

(unter Berücksichtigung der Nutzer-parametrisierung)

SELECT
WHERE
GROUP BY
HAVING
⋮

```
SELECT Book.title AS Title,
       count(*) AS Authors
FROM   Book
JOIN   Book_author
  ON   Book.isbn = Book_author.isbn
GROUP BY Book.title;
```

Transformation

Wie hoch ist die Anzahl an Autoren je Buch?

Limitation auf semantischer Ebene

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

Patient				
ID	Name	Vorname	Geburtsdatum	Geschlecht
0	Mustermann	Max	01.01.2000	m
1	Decker	Dirk	31.12.1999	m
2	Räubertochter	Ronja	03.02.1952	w
3	Lustig	Lea	04.05.1965	w

Patientenzustand			
ID	PatientenID	Status	Erfassungsdatum
0	0	Genesen	14.04.2020
1	1	Geimpft	01.06.2021
2	2	Geimpft	21.08.2021
3	3	Infiziert	05.12.2020
4	1	Infiziert	01.01.2022

- Welche Patienten wurden trotz Impfung infiziert?
- SQL-Abfrage:

```
SELECT p.Name, p.Vorname FROM Patient AS p
JOIN Patientenzustand AS pz ON p.ID = pz.PatientenID
WHERE pz.Status = ‘Infiziert‘
AND pz.PatientenID IN
        (SELECT PatientenID FROM Patientenzustand
        WHERE Status = ‘Geimpft‘ AND Erfassungsdatum < pz.Erfassungsdatum);
```

Ergebnis	
Name	Vorname
Decker	Dirk

Limitation auf semantischer Ebene

Bilde die Schnittmenge, welche die korrespondierenden Einträge der beiden Tabellen „Patient“ und „Patientenzustand“ enthält. Gib die Spalten „Name“ und „Vorname“ aus. Es sollen nur Daten ausgegeben werden für die „Zustand“ ‘Infiziert‘ ist und die Ausprägungen von „ID“ in einer Untermenge liegen, für die „Zustand“ ‘Geimpft‘ ist und „Erfassungsdatum“ kleiner ist als „Erfassungsdatum“ der Obermenge.

```
SELECT p.Name, p.Vorname FROM Patient AS p
JOIN Patientenzustand AS pz ON p.ID = pz.PatientenID
WHERE pz.Status = ‘Infiziert‘
AND pz.PatientenID IN
        (SELECT PatientenID FROM Patientenzustand
        WHERE Status = ‘Geimpft‘
        AND Erfassungsdatum < pz.Erfassungsdatum);
```

Finde Name und Vorname der Patienten, welche den Status ‘Geimpft‘ und nachfolgend den Status ‘Infiziert‘ aufweisen.

- Nicht hinreichend inferierbare Informationen
  - Kardinalität
  - Semantische Beziehung

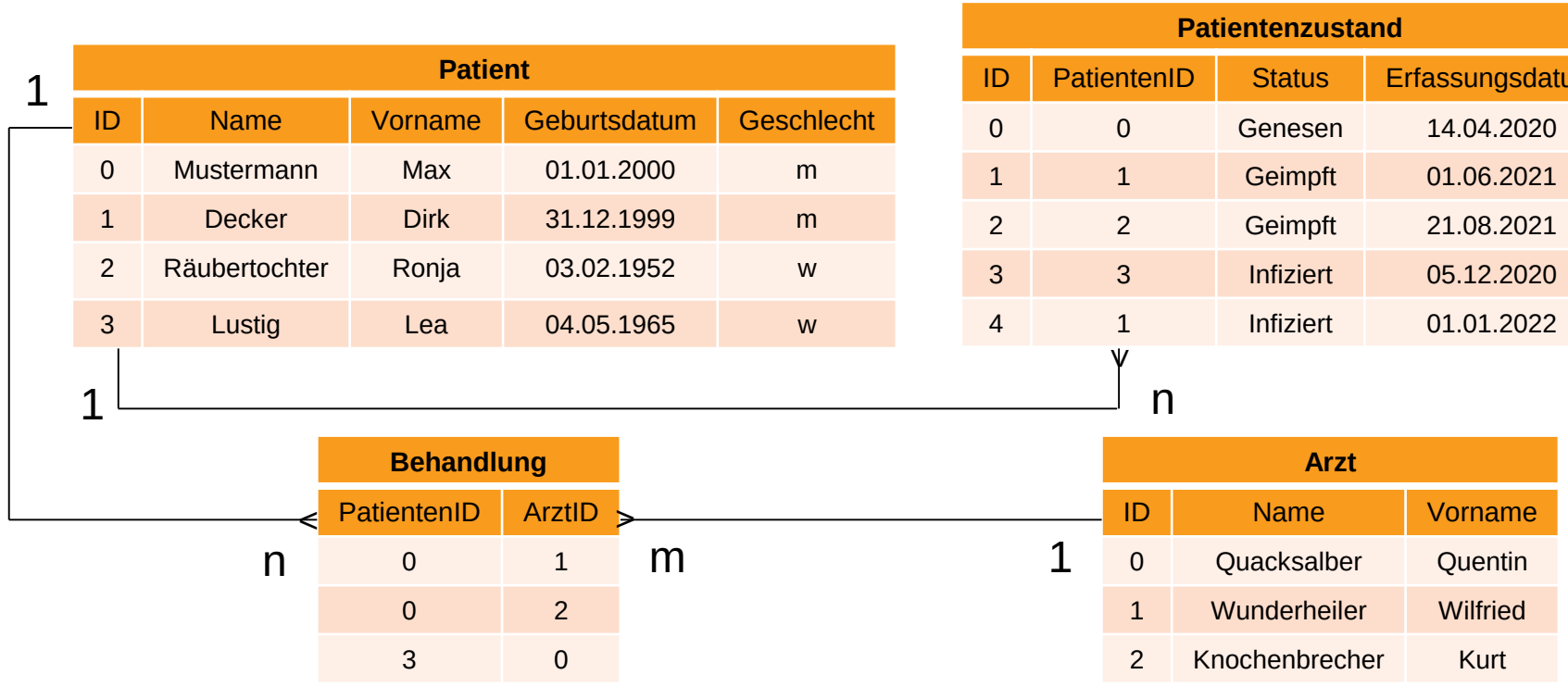
Welche Patienten wurden trotz Impfung infiziert?

Limitation auf semantischer Ebene

Finde Name und Vorname der Patienten, welche den Status ‘Geimpft‘ und nachfolgend den Status ‘Infiziert‘ aufweisen und von ‘Quentin Quacksalber‘ behandelt werden.

Patient ---Behandlung --- Arzt

Rezipient <-------Aktion-------- Akteur



Patient

ID	Name	Vorname	Geburtsdatum	Geschlecht
0	Mustermann	Max	01.01.2000	m
1	Decker	Dirk	31.12.1999	m
2	Räubertochter	Ronja	03.02.1952	w
3	Lustig	Lea	04.05.1965	w

Patientenzustand

ID	PatientenID	Status	Erfassungsdatum
0	0	Genesen	14.04.2020
1	1	Geimpft	01.06.2021
2	2	Geimpft	21.08.2021
3	3	Infiziert	05.12.2020
4	1	Infiziert	01.01.2022

Behandlung

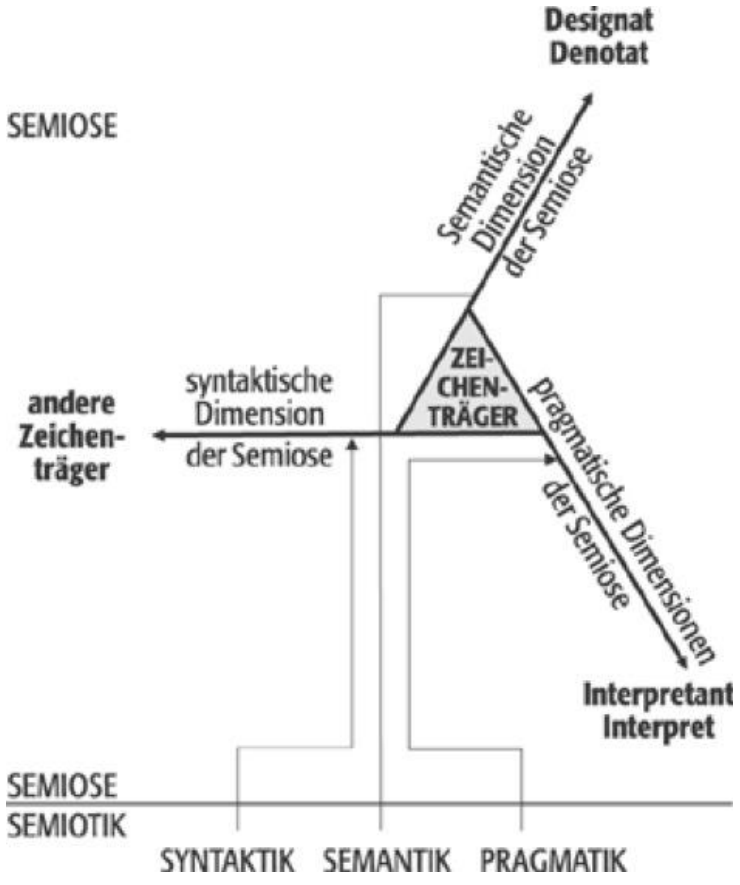
PatientenID	ArztID
0	1
0	2
3	0

Arzt

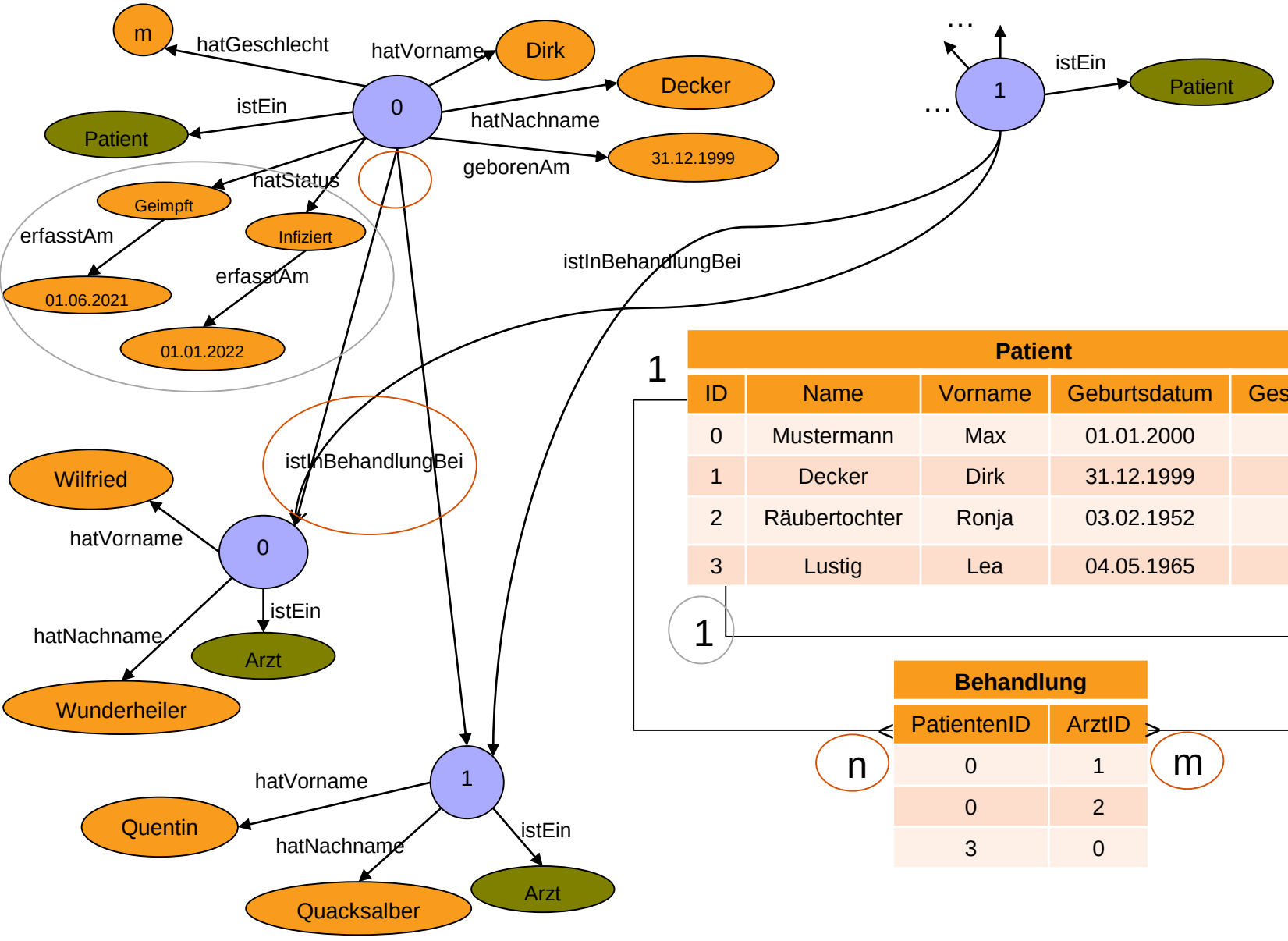
ID	Name	Vorname
0	Quacksalber	Quentin
1	Wunderheiler	Wilfried
2	Knochenbrecher	Kurt

Von der Syntaktik zur Semantik

- Wir kennen die Symbole
  - Tabelle, Attribute, Metainformationen
- Wir kennen die Syntaktik
  - Fremdschlüsselbeziehungen, Tabelle <-> Tabelle
  - Tabellenattribute, Tabelle <-> Attribut
  - Metainformationen, Attribut <-> Metainformation
    (Datentyp, Schlüsselausprägung etc.)
- Problem:
  - Fehlendes semantisches Wissen
    - *(Rollen und Generalisierung/Spezialisierung)*
    - Semantische Bedeutung von Fremdschlüsselbeziehungen
    - Insb. Interaktionsrichtungen
- Lösungsmöglichkeiten:
  -> Händische Annotation bestehender Datenbanken
  -> Generieren von relationalen Datenbanken aus Datenbestand mit vorliegenden semantischen Informationen



Projektion mittels Semantischer Netze / Wissensdatenbanken



1

Patient

ID	Name	Vorname	Geburtsdatum	Geschlecht
0	Mustermann	Max	01.01.2000	m
1	Decker	Dirk	31.12.1999	m
2	Räubertochter	Ronja	03.02.1952	w
3	Lustig	Lea	04.05.1965	w

Patientenzustand

ID	PatientenID	Status	Erfassungsdatum
0	0	Genesen	14.04.2020
1	1	Geimpft	01.06.2021
2	2	Geimpft	21.08.2021
3	3	Infiziert	05.12.2020
4	1	Infiziert	01.01.2022

1 — n

Behandlung

PatientenID	ArztID
0	1
0	2
3	0

n — m

Arzt

ID	Name	Vorname
0	Quacksalber	Quentin
1	Wunderheiler	Wilfried
2	Knochenbrecher	Kurt

1

Angepasster Algorithmus zur Generierung von SQL-Statements

Extrahierter Teilgraph

Semantische Information

Introspection

Generierte DB

Knoten

Kanten

Transformation

Traversierung

(1, 2)
(1, 4)
(2, 3)
:

Zufallswahl

(unter Berücksichtigung der maximalen Join-Länge)

(1, 2, 3)

Zufallswahl

(unter Berücksichtigung der Nutzer-parametrisierung)

SELECT
WHERE
GROUP BY
HAVING

```
SELECT Book.title AS Title,
       count(*) AS Authors
FROM   Book
JOIN   Book_author
  ON   Book.isbn = Book_author.isbn
GROUP BY Book.title;
```

Wie hoch ist die Anzahl an Autoren je Buch?

Transformation

Frei zugängliche Wissensdatenbanken

- Häufige Einschränkungen
  - Größtenteils rein taxonomischer Natur
    - Bspw. isA-, hasA-Beziehungen
  - Keine einheitlichen Schnittstellen und Repräsentationen
  - Meist einsprachig (Englisch)
  - Meist keine lexikalische Informationen

ConceptNet
An open, multilingual knowledge graph

DBpedia

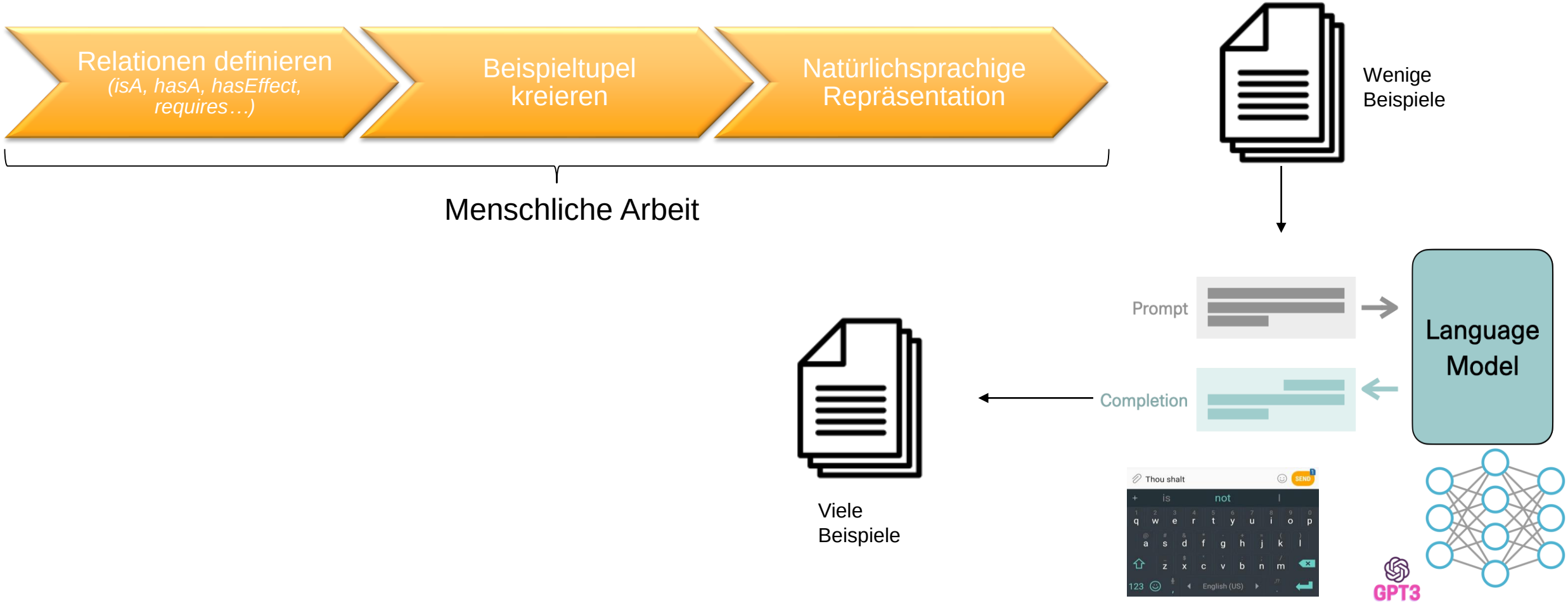
CYCORP
CYC: The Common Sense Knowledge Base

Frame Net

WORDNET

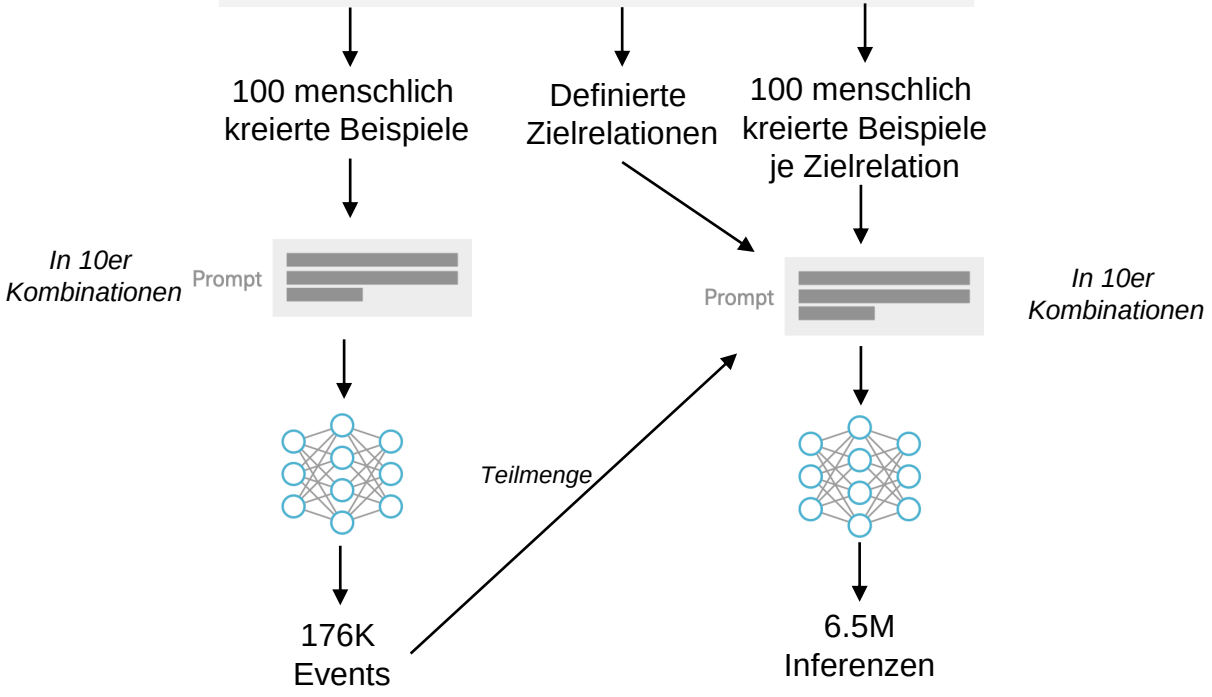
Dynamische Erzeugung von Wissensdatenbanken

- Generative Erzeugung der Wissensdatenbank nach Anforderungsprofil des jeweiligen Aufgabentyps, mit minimalen Aufwänden



Beispiel: Symbolic Knowledge Distillation [2]

Event	Relation	Inferenz
X starts running	**xEffect** *so, X*	gets in shape
X and Y engage in an argument	**xWant** *so, X wants*	to avoid Y
X learns to type fast	**xNeed** *X needed*	to have taken typing lessons
X steals his grandfather's sword	**xEffect** *so, X*	is punished by his grandfather
X takes up new employment	**xIntent** *because X wants*	to be self sufficient



Relation	ATOMIC$^{20}_{20}$	ATOMIC^{10x}
HinderedBy	77,616	**1,028,092**
xNeed	100,995	**760,232**
xWant	109,098	**730,223**
xIntent	54,839	**965,921**
xReact	62,424	**1,033,123**
xAttr	113,096	**884,318**
xEffect	90,868	**1,054,391**
Total Count	608,936	**6,456,300**
Est Total Cost	~$40,000	**~$6,000**
Est Cost Per Triple	~$0.06	**~$0.001**

Corpus	Accept	Reject	N/A	Size	Size (div)
ATOMIC$^{20}_{20}$	86.8	11.3	1.9	0.6M	0.56
ATOMIC^{10x}	78.5	18.7	2.8	**6.5M**	**4.38**
	88.4	9.5	2.1	5.1M	3.68
	91.5	6.8	1.7	4.4M	3.25
	94.3	4.6	1.1	3.6M	2.74
	95.3	3.8	1.0	3.0M	2.33
	96.4	2.7	0.8	2.5M	2.00

Strenge des „Critics“

Einführung des „Critic“

Klassifikationsmodell, das auf 10K, durch Crowdsourcing bewerteter zufällig gewählter Tupel trainiert wurde

[2] Symbolic Knowledge Distillation: from General Language Models to Commonsense Models

[3] LAMA [4] COMET

SQL - Relationsschema

Mögliche Relationen:

- isA-Beziehung
  - X ist ein Patient; X ist ein Arzt; X ist ein Covid-Status…
- hasA-Beziehung (X = Instanz einer „Klasse“)
  - X hat einen Nachnamen; X hat einen Covid-Status
- isAbleTo-Beziehung (X, Y = Instanzen einer „Klasse“)
  - X kann eine Behandlung durchführen; Y kann sich behandeln lassen; X kann Y behandeln
- hasDate-Beziehung (X = Datumsinstanz, Y = Instanz einer „Klasse“)
  - am X wurde Y geboren; am X wurde Y erfasst
- occuredBefore/occuredAfter
  - Status X vor Status Y („Geimpft“ vor „Infiziert“)

Ermöglicht:

- Datenbankgenerierung
- Aufgabenstellung, welche näher an einer Formulierung durch den Menschen liegt

Integration in ALADIN

- Nutzung von Open-Source Alternativen (GPT-Neo/-J/-X)
- Definition von Relationen und Erstellen von Beispieltupeln durch Lehrende/Entwickler bei Bedarf an neuen Aufgabentypen
- „Flagging“ von möglicherweise inakzeptablen Tupeln während der Bearbeitung durch Studenten (integriertes Crowdsourcing)
- Eröffnet domänenspezifische und von Semantik abhängige Aufgabentypen
  - Biologie, Jura, Medizin, etc.
  - Modellierungsaufgaben (ERM, UML, BPMN, etc.)
  - …

Ausblick I: neue Aufgabentypen

- aus ALADIN-II-Antrag:
  - Terminierung
  - Spatial SQL
  - Netzplantechnik
  - PERT
  - Datenfluss-, ERM- und UML-Modellierung.
- aus OPALADIN-Antrag:
  - Kodierung (Faltungscodes, Huffman)
  - Prüfmuster / Paragraphennetzwerke für Rechtsfälle / Gesetze
  - Chemische Strukturformeln von Molekülverbindungen
  - Euler-Tonnetze/PLR-Regeln in der Musiktheorie

Ausblick II: ALADIN goes OPAL (OPALADIN)

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

- fachlich/inhaltlich
  - „Generalisierung“ der Aufgabentypen
  - „programmierfreie“ Erstellung neuer Aufgabentypen
  - statistische Auswertungen zu Nutzerverhalten und Aufgabenbearbeitung
- technisch:
  - „von der Syntaktik zur Semantik“ …
  - Integration in OPAL (und ONYX)
  - Technische Umsetzung mittels LTI-Schnittstelle und Shibboleth-Nutzer
  - Einbettung in OPAL-Kurse als Abschluss der jeweiligen Lektionen
  - Eigenständige Nutzung ermöglichen (bspw. analog zu LAVA-Kursen)
  - Hochschulübergreifende Nutzung

Fragen & Diskussion

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

Vielen Dank für die Aufmerksamkeit!