ALADIN: Generator für **A**ufgaben und **L**ösung(shilf)en **a**us **d**er **I**nformatik und angrenzenden Disziplin**en**

„(Didaktische) Herausforderung“ vor ALADIN

- 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
- …

Kurzfassung? Weglassen? Einkürzen!

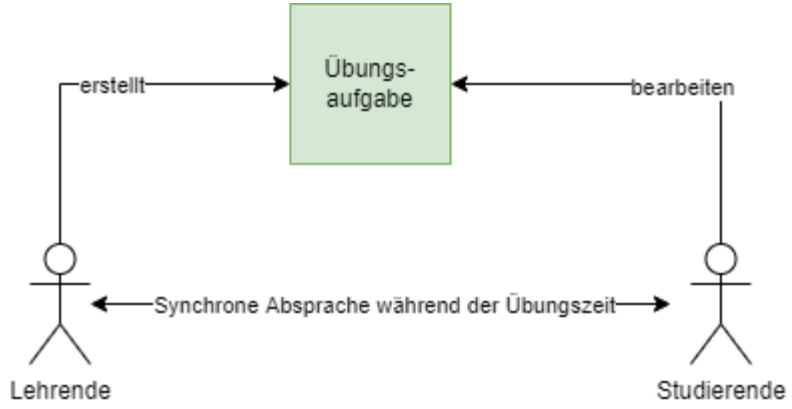
HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

- automatisches und zufallsbasiertes Erstellen und digitales Darbieten von Aufgaben
- Übung und Wiederholung
- parametrisieren, dass er in Abhängigkeit von Kompetenz und Wünschen der Studierenden leichte und schwere, umfangreiche und weniger umfangreiche Aufgaben generiert
- löst der Generator die Aufgaben auch schrittweise automatisch, gibt den Studierenden Hinweise zur Lösung der Aufgaben und hilft somit bei der Lösung der Aufgaben digital
- Die Studierenden müssen die Übungsaufgaben nicht während der Lehrveranstaltungen, sondern können sie in ihrer Selbststudienzeit (zu beliebiger Zeit), auch ohne Hilfe durch Lehrende, an einem beliebigen Ort und auch während ihrer Prüfungen lösen.
- Alle Interessierten können ALADIN, eine Open Educational Resource, einsetzen.

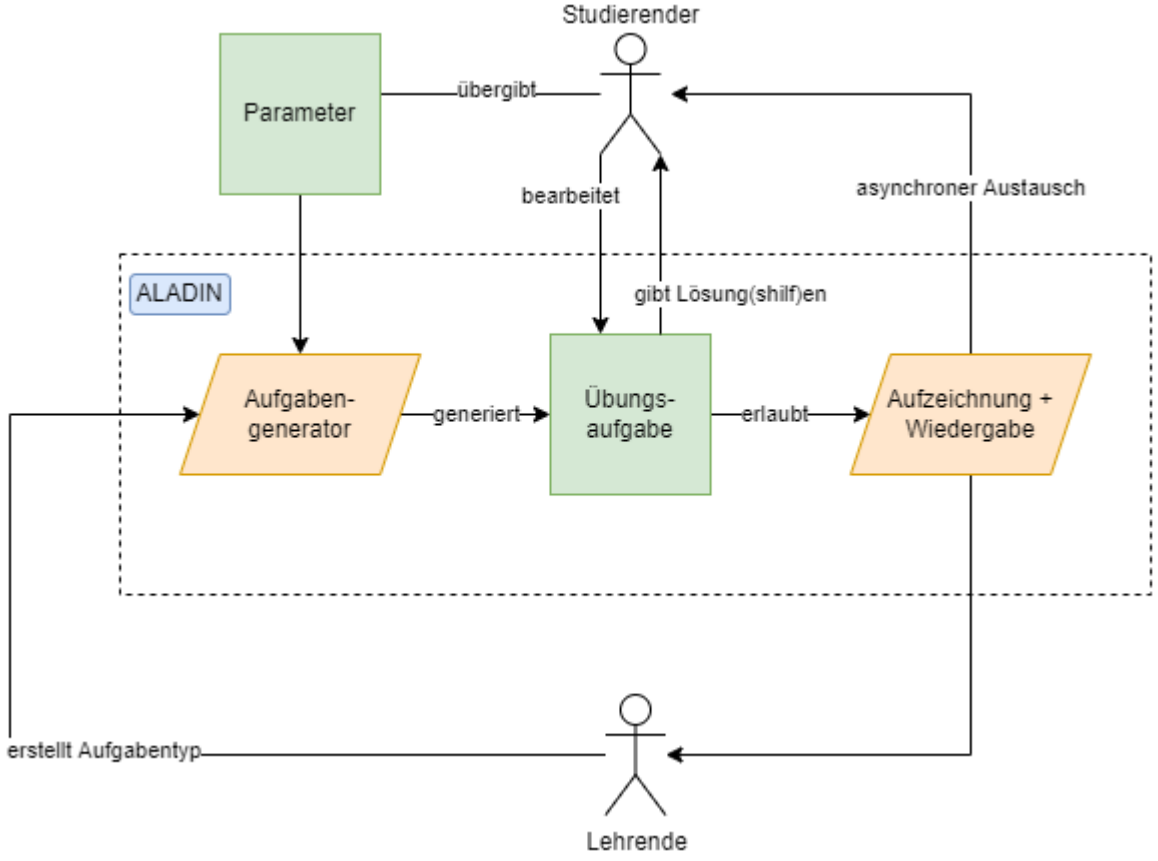
Graphische Alternative zu vorheriger Folie

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

Ablauf ohne ALADIN



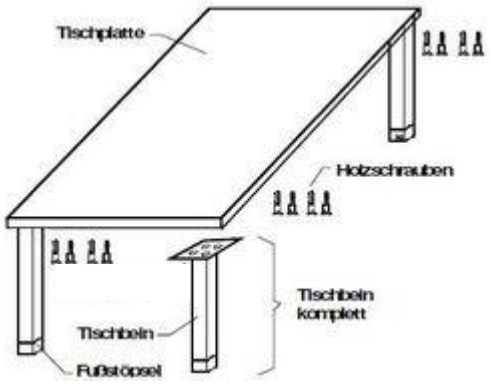
Ablauf mit ALADIN



Derzeitiger Leistungsumfang von ALADIN

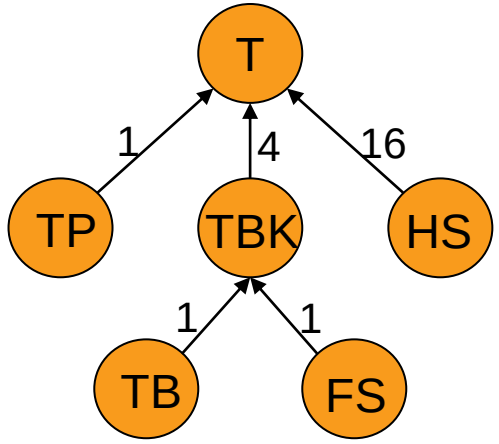
- unterstützte Aufgabentypen:
  - Stücklistenauflösung mittels dreier, unterschiedlicher Verfahren
  - SQL-Abfragen
  - Geostatistische Interpolationsverfahren (Inverse Distanzwichtung)
  - Shortest-Path-Algorithmen (Dijkstra)
  - Projektmanagement (Netzplan, Gantt)
- Aufzeichnung, Wiedergabe und Fortführung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

Stücklistenauflösung - Theorie



- Stückliste für 1 **T**isch
  - 1 **T**isch**p**latte
  - 4 **T**isch**b**eine **k**omplett
    - 1 **T**isch**b**ein
    - 1 **F**uß**s**töpsel
  - 16 **H**olz**s**chrauben

Gozintograph



Direktbedarfsmatrix D

	T	TP	TBK	HS	TB	FS
T	0	0	0	0	0	0
TP	**1**	0	0	0	0	0
TBK	**4**	0	0	0	0	0
HS	**16**	0	0	0	0	0
TB	0	0	**1**	0	0	0
FS	0	0	**1**	0	0	0

D

	T	TP	TBK	HS	TB	FS
T	0	0	0	0	0	0
TP	1	0	0	0	0	0
TBK	4	0	0	0	0	0
HS	16	0	0	0	0	0
TB	0	0	1	0	0	0
FS	0	0	1	0	0	0

*

	PB
T	5
TP	2
TBK	3
HS	12
TB	0
FS	7

Primärbedarf

Direktbedarfsmatrix D

	T	TP	TBK	HS	TB	FS
T	0	0	0	0	0	0
TP	1	0	0	0	0	0
TBK	4	0	0	0	0	0
HS	16	0	0	0	0	0
TB	0	0	1	0	0	0
FS	0	0	1	0	0	0

\+

= D^2

	T	TP	TBK	HS	TB	FS
T	0	0	0	0	0	0
TP	0	0	0	0	0	0
TBK	0	0	0	0	0	0
HS	0	0	0	0	0	0
TB	4	0	0	0	0	0
FS	4	0	0	0	0	0

\+

E

	T	TP	TBK	HS	TB	FS
T	1	0	0	0	0	0
TP	0	1	0	0	0	0
TBK	0	0	1	0	0	0
HS	0	0	0	1	0	0
TB	0	0	0	0	1	0
FS	0	0	0	0	0	1

\=

Gesamtbedarfsmatrix

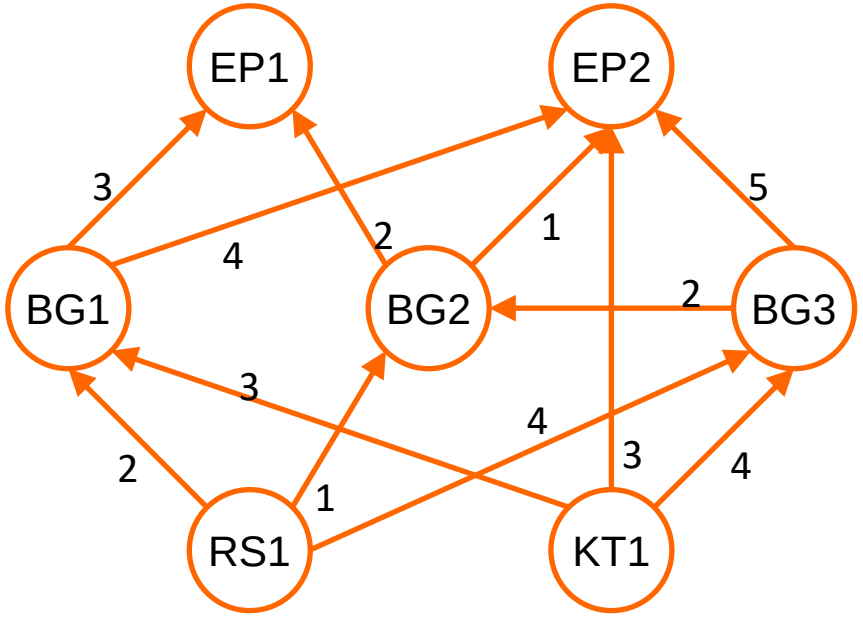
	T	TP	TBK	HS	TB	FS
T	1	0	0	0	0	0
TP	1	1	0	0	0	0
TBK	4	0	1	0	0	0
HS	16	0	0	1	0	0
TB	4	0	1	0	1	0
FS	4	0	1	0	0	1

\=

	SB
T	5
TP	7
TBK	23
HS	92
TB	23
FS	30

Sekundärbedarf

Stückliste: Gozintograph und Auflösung



Direktbedarfsmatrix D

	EP1	EP2	BG1	BG2	BG3	KT1	RS1
EP1	0	0	0	0	0	0	0
EP2	0	0	0	0	0	0	0
BG1	3	4	0	0	0	0	0
BG2	2	1	0	0	0	0	0
BG3	0	5	0	2	0	0	0
KT1	0	3	3	0	4	0	0
RS1	0	0	2	1	4	0	0

Primärbedarfsvektor P

EP1	30
EP2	70
BG1	0
BG2	10
BG3	0
KT1	0
RS1	0

gegeben: D und P
gesucht: S
Lösung:
$G = E + D^1 + D^2 + \ldots + D^k$ oder $G = (E - D)^{-1}$
E .. Einheitsmatrix
k .. Anzahl der Kanten im längsten Pfad im Graphen
S = G * P

Gesamtbedarfsmatrix G

	EP1	EP2	BG1	BG2	BG3	KT1	RS1
EP1	1	0	0	0	0	0	0
EP2	0	1	0	0	0	0	0
BG1	3	4	1	0	0	0	0
BG2	2	1	0	1	0	0	0
BG3	4	7	0	2	1	0	0
KT1	25	43	3	8	4	1	0
RS1	24	37	2	9	4	0	1

Sekundärbedarfsvektor S

EP1	30
EP2	70
BG1	370
BG2	140
BG3	630
KT1	3840
RS1	3400

Vorführung Stücklistenauflösung

- Version mit Matrizenmultiplikation

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

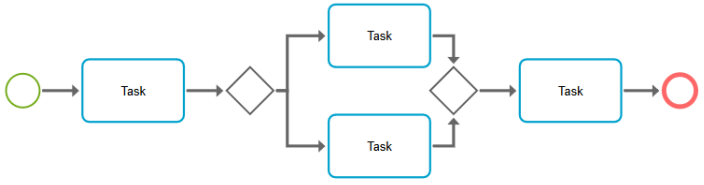
Name	Vorname
Decker	Dirk

Vorführung SQL-Abfragen

Replay – Motivation & Anwendungsfälle

- Möglichkeiten
  - Aufzeichnung aller Interaktionen
  - Wiedereinstieg an beliebiger Stelle
  - Vervollständigung des Lösungsversuchs als neue Aufzeichnung

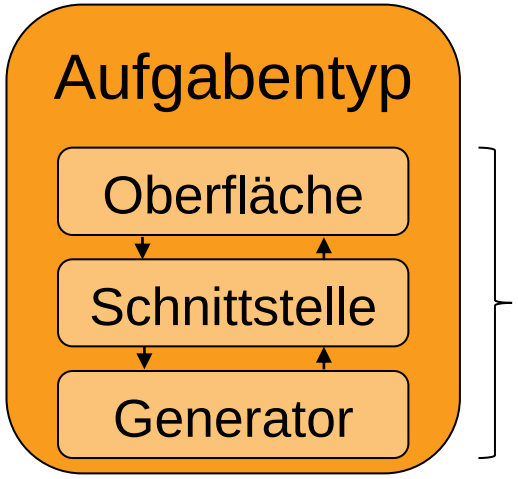
- Anwendungsfälle
  - „Zwischenspeichern“ des Bearbeitungszustandes
  - Asynchroner Austausch eines Lösungsversuchs mit Kommilitonen und Lehrpersonal
  - Aggregierte Auswertungen (anonymisiert)
    - Erkennung von „Verklemmungen/Engpässen“
    - Optimierung der Nutzeroberfläche und –pfade
  - Blended Learning



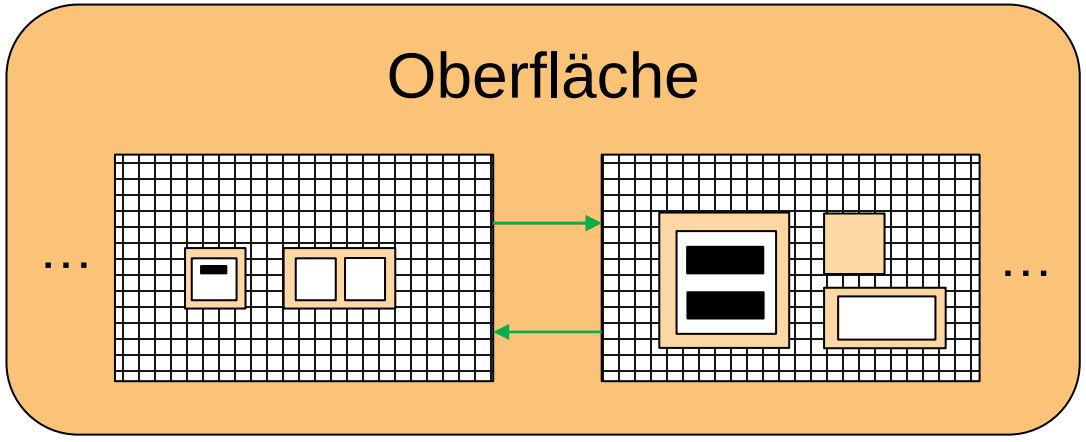
Aufzeichnung, Wiedergabe und Fortführung von Lösungsversuchen

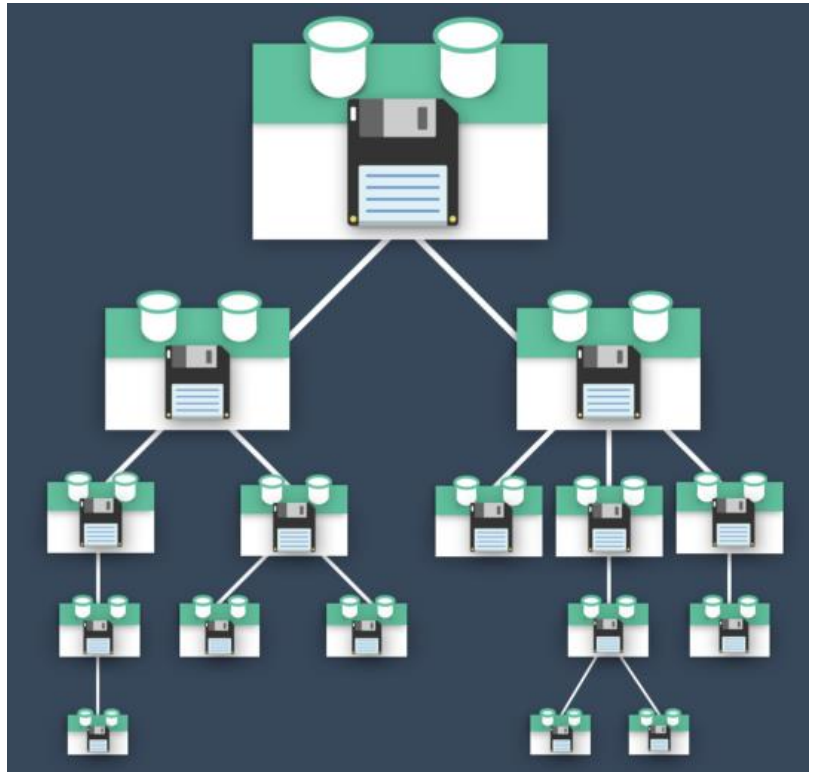
- Vorführung

Hinter den Kulissen



Deklarativ konfigurierbar
im JSON-Format





Ausblick I: neue Aufgabentypen

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

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

OPALADIN: ALADIN goes OPAL

- 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 ermöglichen
  - Harmonisierung des Lehrplans
  - Vereinfacht Anerkennung der Modulleistungen

Ausblick (kommt zum Schluss)

- Was wir noch für Aufgabentypen unterstützen wollen:
  - Spatial SQL
  - Datenfluss-, ERM- und UML-Modellierung
  - Kodierung (Faltungscodes, Huffman)
  - Prüfmuster / Paragraphennetzwerke für Rechtsfälle / Gesetze
  - Chemische Strukturformeln von Molekülverbindungen
  - Euler-Tonnetze/PLR-Regeln in der Musiktheorie
- Statistische Auswertungen zu Nutzerverhalten und Aufgabenbearbeitung
- OPALADIN:
  - Generierung „semantisch sinnvoller“ (andere, bessere Worte) Aufgaben
  - „Generalisierung“ der Aufgabentypen (eigentlich Aufgabengenerierung?)
  - (Visuell konfigurierbare) „programmierfreie“ Erstellung neuer Aufgabentypen
  - Integration in OPAL (und ONYX, falls möglich)

Ausblick II: ALADIN goes OPAL (OPALADIN)

HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN
UNIVERSITY OF APPLIED SCIENCES

- FACHLICH:
- „Generalisierung“ der Aufgabentypen
- „programmierfreie“ Erstellung neuer Aufgabentypen

- TECHNISCH:
- „von der Syntax 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!