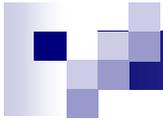


Multimediale Algorithmen und Datenstrukturen Assessments

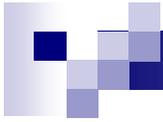
Entwicklung eines Assessment-Systems zur
Generierung, Durchführung und Auswertung von
Aufgaben im Bereich Algorithmen und
Datenstrukturen

Markus Krebs



Assessment-Systeme: Allgemein

- Erworbenes Wissen vertiefen
- Unterstützung des Lernprozesses
- Selbstbeurteilung (Stärken, Schwächen)



Assessment: 3 Phasen

- Vorbereitung
- Durchführung
- Evaluation



Vorbereitungsphase

- Generierung:
 - Automatisch
 - Durch Person
- Repository
 - Meta-Daten: wie z.B.:
 - Aufgabentyp (Übung, Klausur, etc.)
 - Schwierigkeitsgrad
 - Herkunft (Generator, Person)
 - Erstellungsdatum



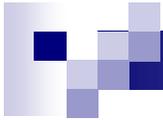
Durchführungsphase

- Aufgaben anfordern und bearbeiten
- Visualisierung: JEDAS
- Aufgabentypen:
 - Restrictive-Mode
 - Free-Mode
 - Fault-Mode
- Feedback-System
- Besonderes:
 - Undo-Funktion
 - Speichern/Laden von Aufgaben



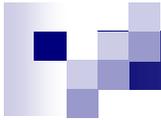
Evaluationsphase

- Einsenden von Lösungen
- Auswertung:
 - Automatisch
 - Durch Person (z.B. Tutor)
- Statistiken
 - Student → Selbsteinschätzung zum Klassendurchschnitt
 - Professor → Problembereiche der Studenten (Anpassung)



Umsetzung: Allgemein

- Vorerst für Bäume
 - AVL-Baum
 - Binomial-Queue
 - Exemplarisch für Fibonacci-Heaps
- Unterteilung der Aufgabenstellung in kleinere durchzuführende Schritte
 - Bsp.: delete-min unterteilen in Sub-Operationen



Umsetzung: Auto. Generierung (1)

- Aufgabenstellung (Beispiel, Fibonacci-Heap):
 - 3 Inserts
 - 2 Deletes
 - 2 decrease-key
 - 1 delete-min
- „Suche im Zustandsraum mit Bewertung“ um Aufgaben verschiedenen Schwierigkeitsgrades zu erhalten
- Abfrage prozeduralen Wissens (wenig deklaratives)



Umsetzung: Auto. Generierung (2)

- Fibonacci-Heap: Bewertungsparameter für..
 - #cuts
 - #Markierungs-Änderungen
 - #link
 - #meld
 - Neuer FHeap (bei insert)



Umsetzung: Auto. Generierung (3)

■ Fibonacci-Heap

- n Schlüssel

- n+1 Intervalle

 - $O(n)$ inserts

 - $O(n)$ deletes

 - $O(n^2)$ decrease-keys

 - $O(1)$ delete-mins

- Permutationen der Aufgabenstellung durchrechnen

 - Verzweigungsgrad sehr hoch → max. Tiefe 3-4

■ Erzeugen einer Aufgabenqueue der Sub-Operationen

Programm – Screenshot (FHeap)

FibHeap Analyzer V0.01 by Markus Krebs

Current operation: deletemin()
Consolidating root list
Entering node with rank 1

```
DECREASE-KEY 96 4 -> 9  
DELETE 96 -> 8  
DECREASE-KEY 96 6 -> 8  
DECREASE-KEY 96 21 -> 8  
DECREASE-KEY 24 4 -> 5  
DECREASE-KEY 64 4 -> 5  
DECREASE-KEY 94 4 -> 5  
DELETE-MIN (5) -> 4  
DELETE 5 -> 4  
DELETE 24 -> 4  
DELETE 64 -> 4  
DELETE 94 -> 4  
DECREASE-KEY 24 6 -> 4  
DECREASE-KEY 64 6 -> 4  
INSERT 4 -> 3  
DECREASE-KEY 7 4 -> 3  
INSERT 6 -> 2  
INSERT 21 -> 2
```

0.0 fps
0:00:00.000
100%



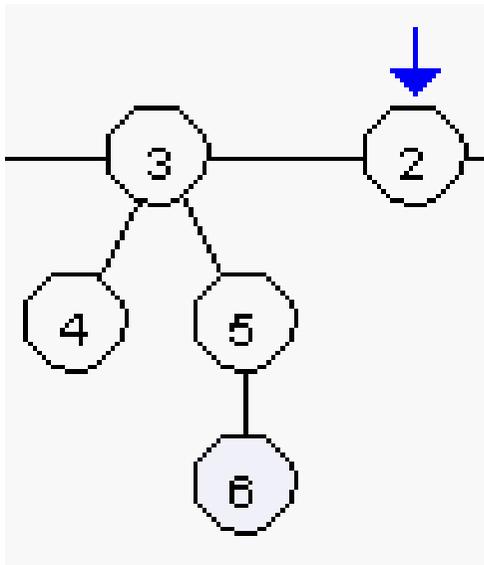
Umsetzung: Bearbeitung

- Grad der Detailierung (2 Ansätze):
 - Vertex-Format-based
 - Function-based
- Problem:
 - Nicht-Determinismus bei Fibonacci-Heap
 - Veränderung des Schwierigkeitsgrades
 - Komplizierte Evaluation
 - Bearbeitungsphase so gestalten, dass die Einzelaktionen deterministisch sind
 - Beispiel: nächste Folie

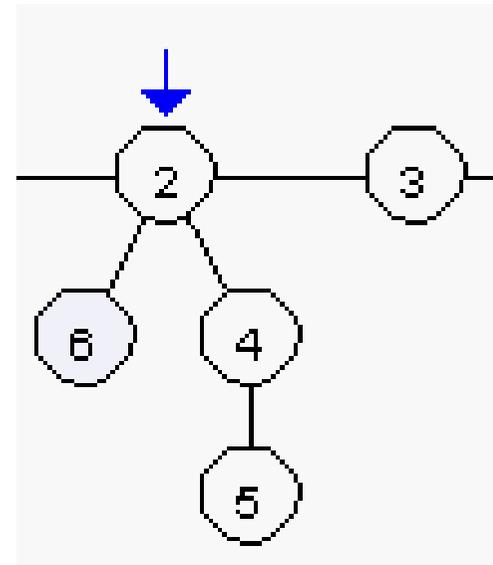
Problem: Beispiel

Aufgabenstellung:

Insert 1, Insert 2, ... , Insert 6, delete-min, delete-min



1 → 2 → 3 → 4 → 5 → 6 → delete-min



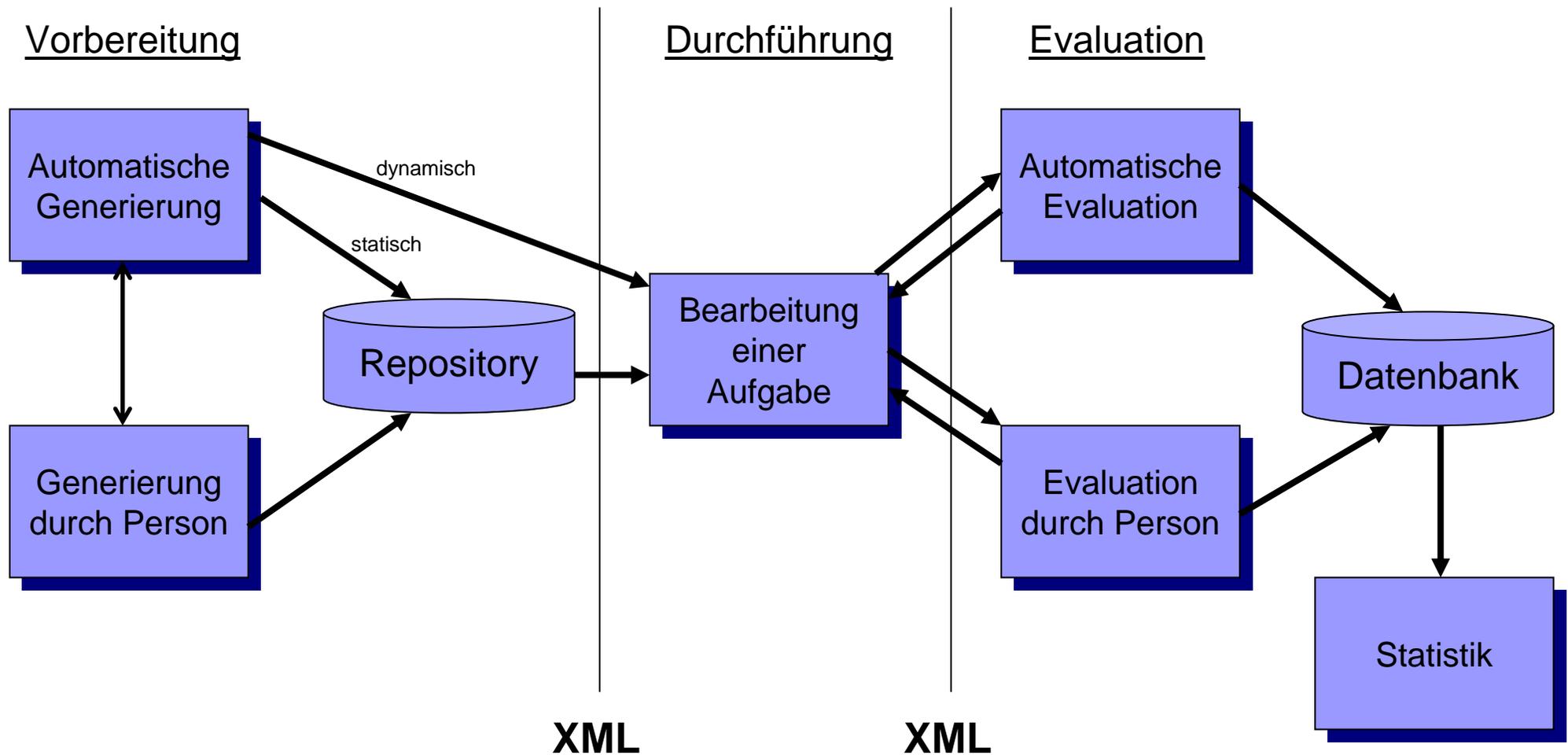
1 → 3 → 4 → 5 → 6 → 2 → delete-min



Umsetzung: Evaluation

- Advanced Computer Tutoring (ACT):
 - Immediate Feedback and Error Correction
 - Error-Flagging
 - Demand-Feedback
 - No-Tutor
- Vergleichsprogramm notwendig
- Zurückbringen auf richtigen Pfad

Ziel: Java-Programme





Erweiterung: Jedas-Klassen

■ Gründe:

- Ratings

- FaultMode

- Generierung einer falschen Animation notwendig

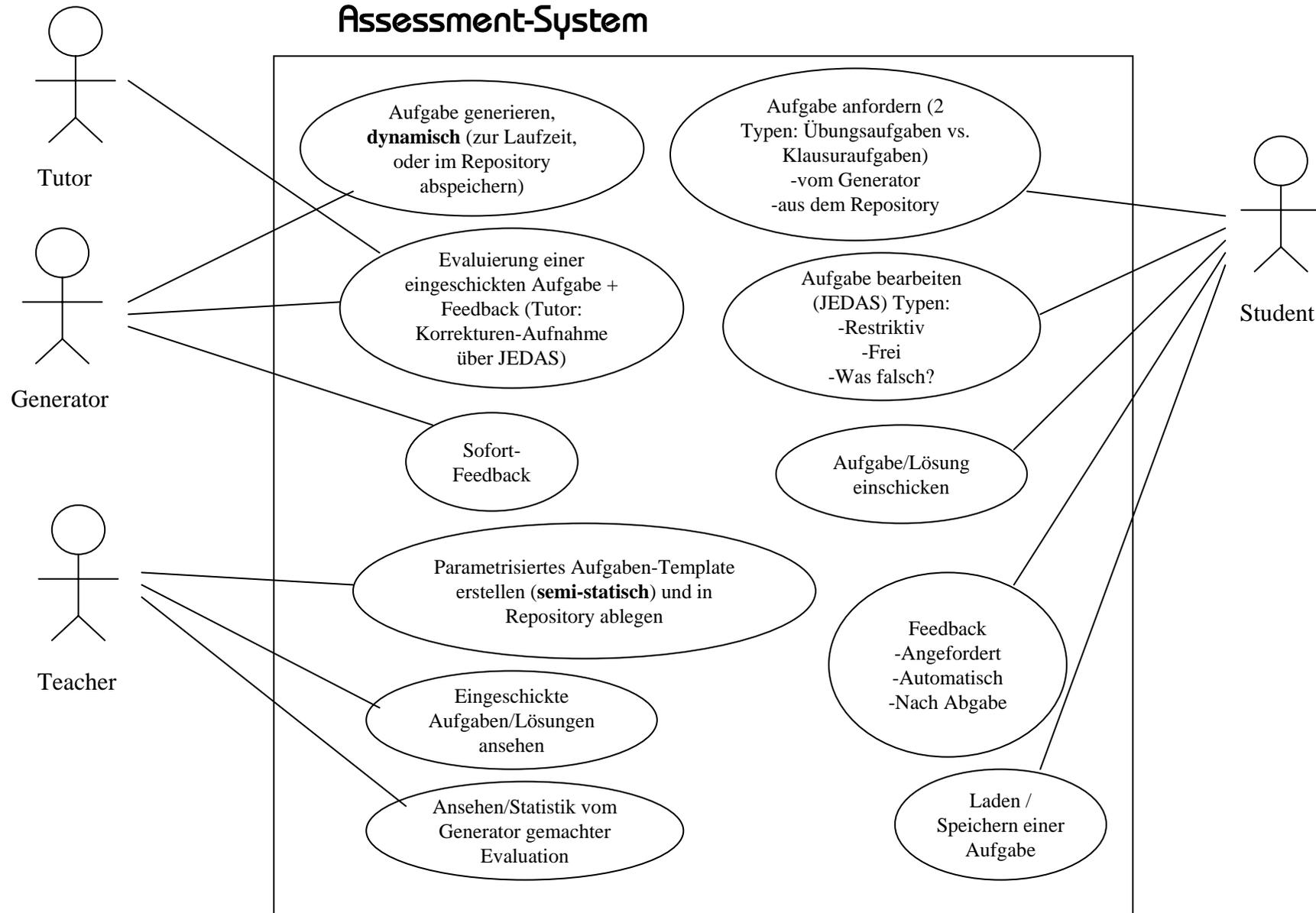
- Undo

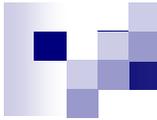
- Kopieren der Animationsstruktur

- Einzeloperationen müssen direkt ansprechbar sein

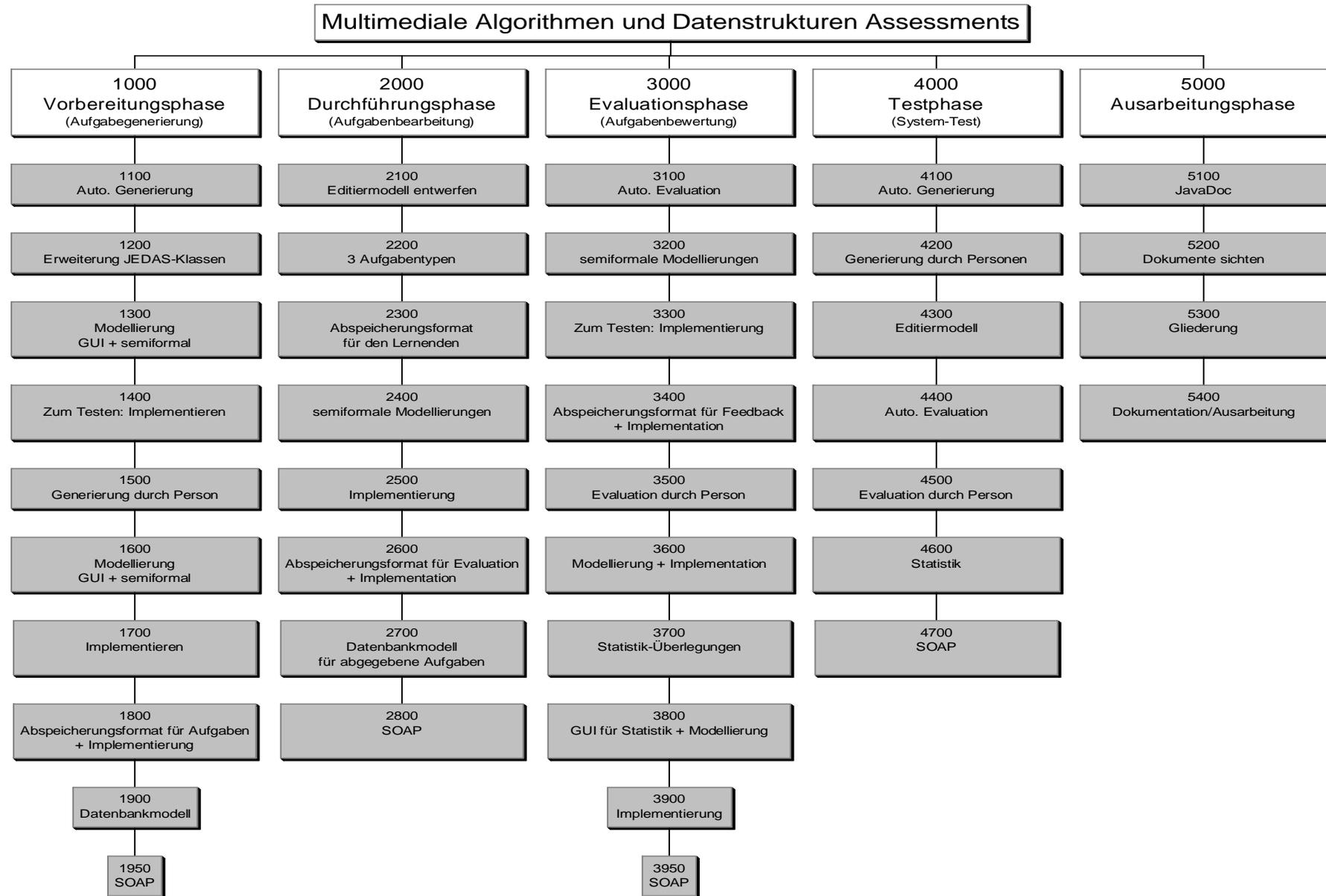
- Fibonacci-Heap: cut, meld, link, etc.

Use-Case Diagramm





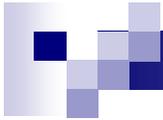
Projektstrukturplan





Literatur

- Brusilovsky, Peter and Miller, Phillip, „Web-based Testing for Distance Education“
 - Pathak, S. and Brusilovsky, P., „Assessing Student Programming Knowledge with Web-based Dynamic Parametrized Quizzes“
 - Tscherter, Vincent, Lamprecht, Reto and Nievergelt, Jürg, „Exorciser: Automatic Generation and Interactive Grading of Exercises in the Theory of Computation“
 - Albert T. Corbett, John R. Anderson „Locus of Feedback Control in Computer-Based Tutoring: Impact on Learning Rate, Achievement and Attitudes“
 - John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger and Ray Pelletier „Cognitive Tutors: Lessons Learned“
 - JEDAS – Java EDucational Animation System (<http://ad.informatik.uni-freiburg.de/jedas>)
- ➔ Spezielle Aufgaben, Betrugelimination, Tutortypen, Grundsätze, Empirische Studien



Ausblick

- Vorbereitung
 - Automatisch (Bewertung, Zustandsraum)
- Durchführung
 - Grad der Detailierung
- Evaluation
 - verschiedene Tutortypen

→ Fragen ?