



dgpf

Distributed Genetic Programming Framework

Alexander Podlich
podlich@student.uni-kassel.de

Sommersemester 2006

DGPF-Projekt:
Erstellung einer grafischen
Benutzeroberfläche

Überblick

1. Allgemeines zu DGPF
2. Die Aufgabe
3. Implementierung/Realisierung
4. Demo
5. Zusammenfassung

Übersicht – Kapitel 1

1. Allgemeines zu DGPF

Was ist DGPF?

2. Die Aufgabe

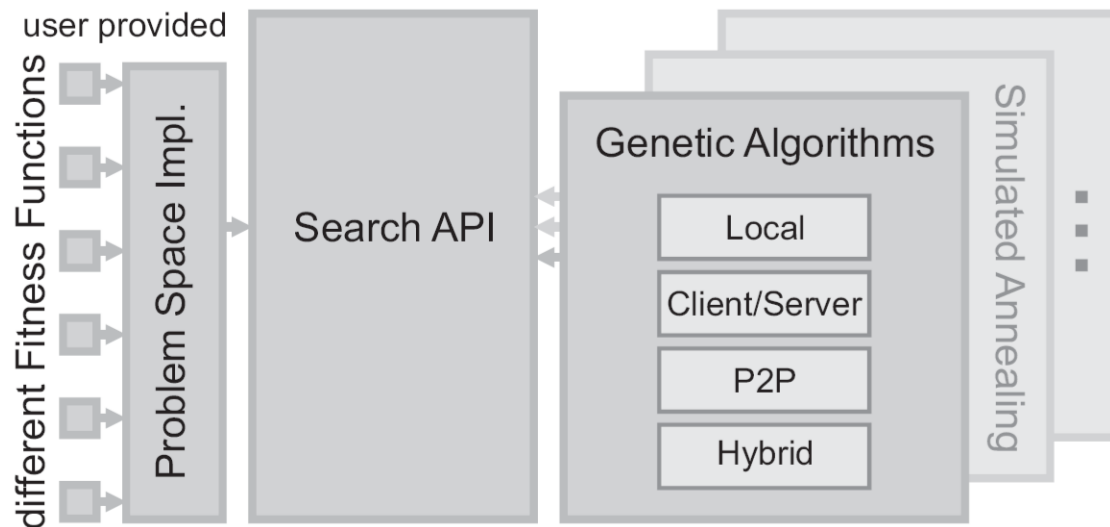
3. Implementierung/Realisierung

4. Demo

5. Zusammenfassung

1 Was ist DGPF?

- DGPF ist ein Java-Open-Source Sourceforge Projekt
- Benutzt z.B. Genetische Algorithmen, um automatisch Programme zu finden.
- **Hauptziel:** Finden von Algorithmen zur verteilten Problemlösung in (Sensor-) Netzwerken.



Übersicht – Kapitel 2

1. Allgemeines zu DGPF

2. Die Aufgabe

Anforderungen und Ziele

3. Implementierung/Realisierung

4. Demo

5. Zusammenfassung

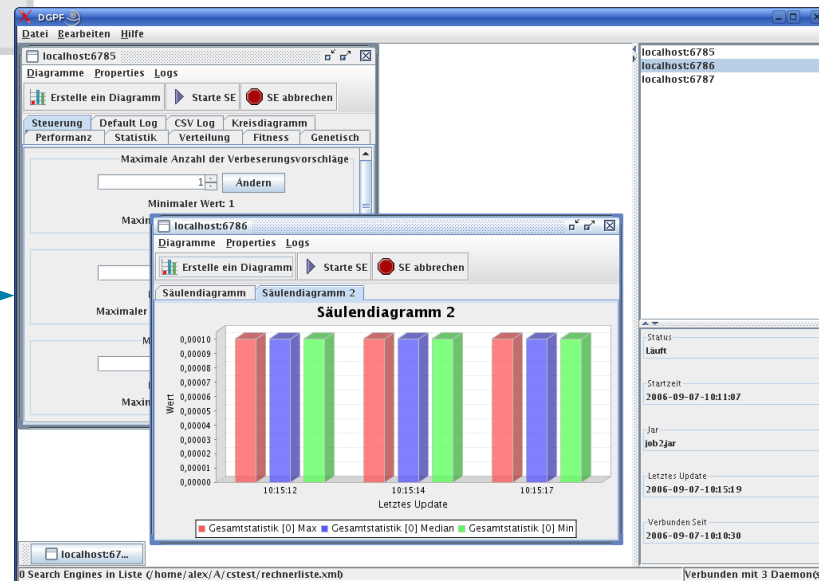
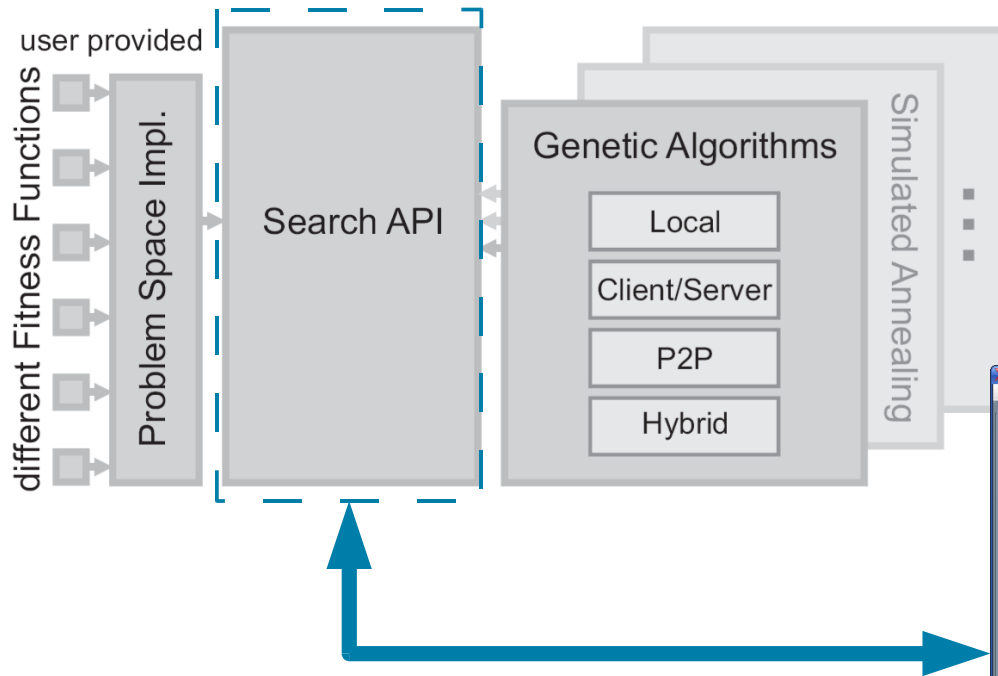
2. Anforderungen und Ziele

- **Erstellung einer graphischen Benutzeroberfläche**
 - Starten und Beenden einer Suche (im Netzwerk)
 - Darstellung, Auswertung und Aufzeichnung von statistischen Daten, die vom Suchalgorithmus erzeugt werden.
 - *Wie sieht es im Moment aus?*
 - *Wie ist die zeitliche Entwicklung?*
 - *Protokollierung und Abspeicherung einer Suche?*
 - Steuerung der Suchalgorithmen

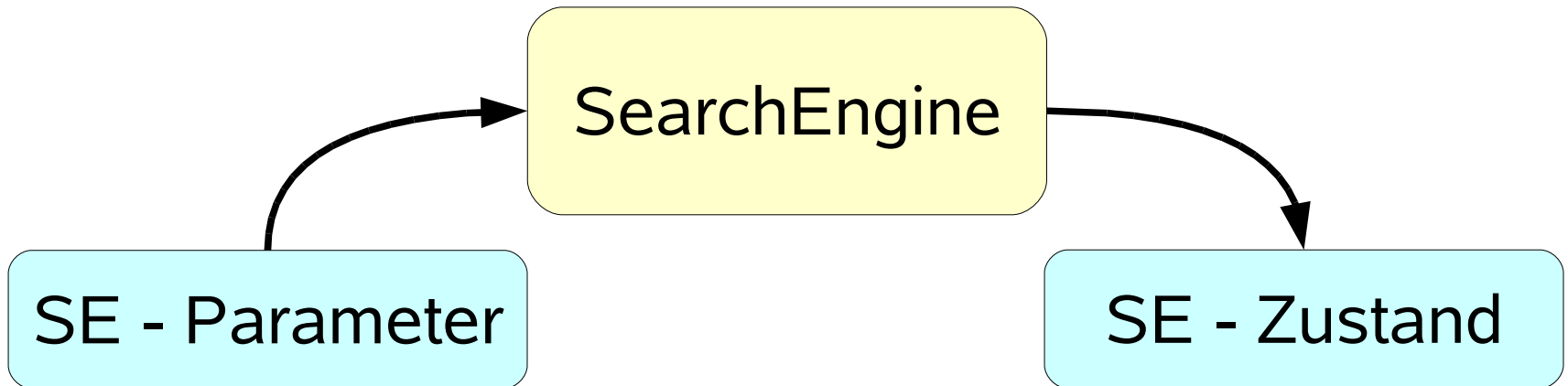
Übersicht – Kapitel 3

1. Allgemeines zu DGPF
2. Die Aufgabe
- 3. Implementierung/Realisierung**
 - 1. Mein Einstiegspunkt**
 - 2. Datengewinnung**
 - 3. Darstellung der statistischen Daten**
 - 4. Diagramme**
 - 5. Vorteile?**
 - 6. Weitere Funktionen**
4. Demo
5. Zusammenfassung

3.1 Mein Einstiegspunkt



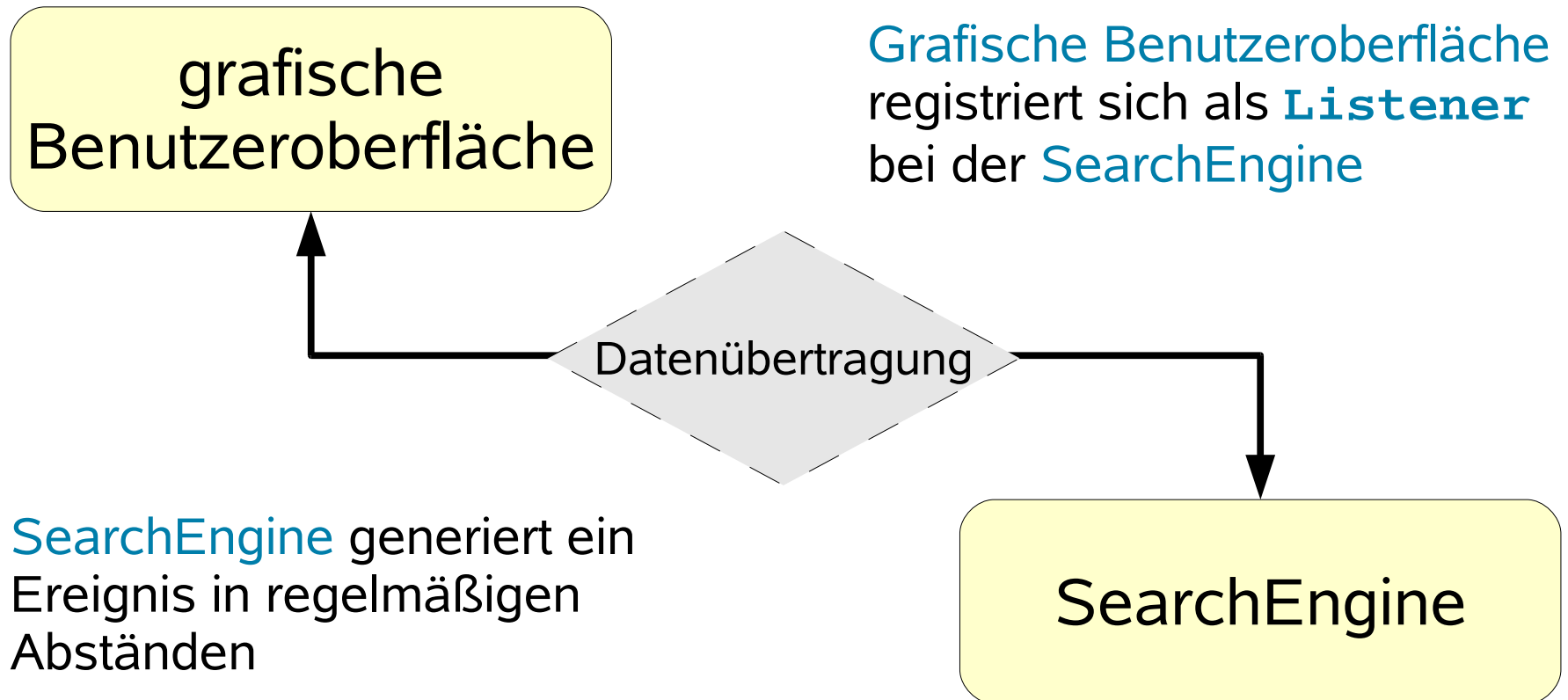
3.2 Datengewinnung - Überblick



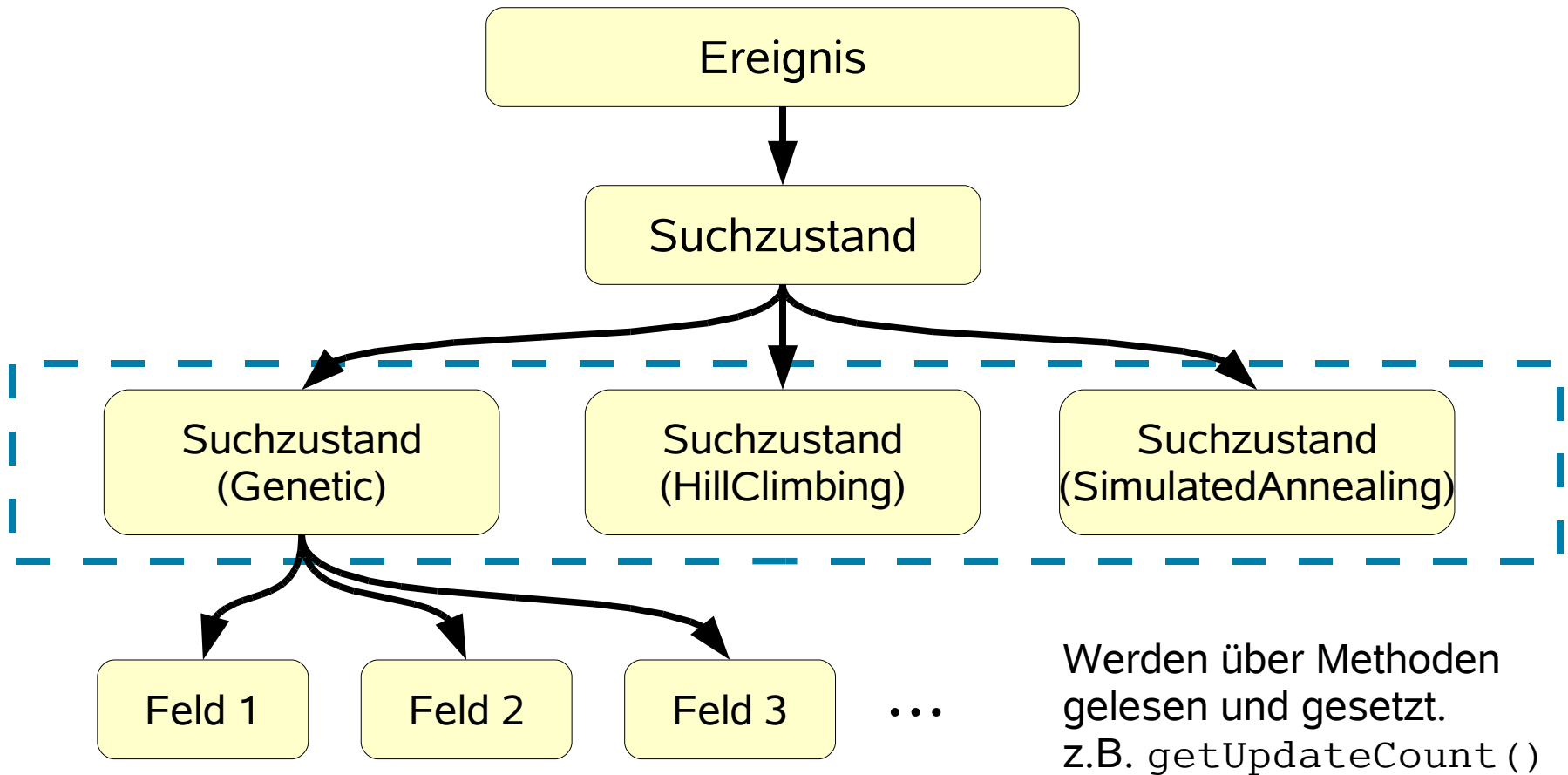
- Die Einstellungen für die nächste Generation.
- Der aktuelle Zustand der SearchEngine.

Parameter (z.B. Mutationsrate) können und dürfen nicht während einer Generation geändert werden.

3.2 Datengewinnung - GUI

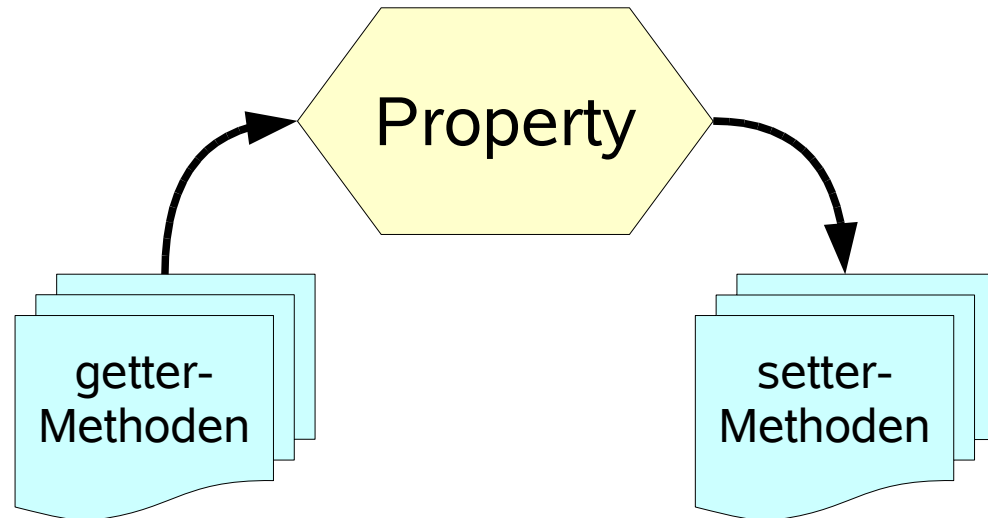


3.2 Datengewinnung - Interna

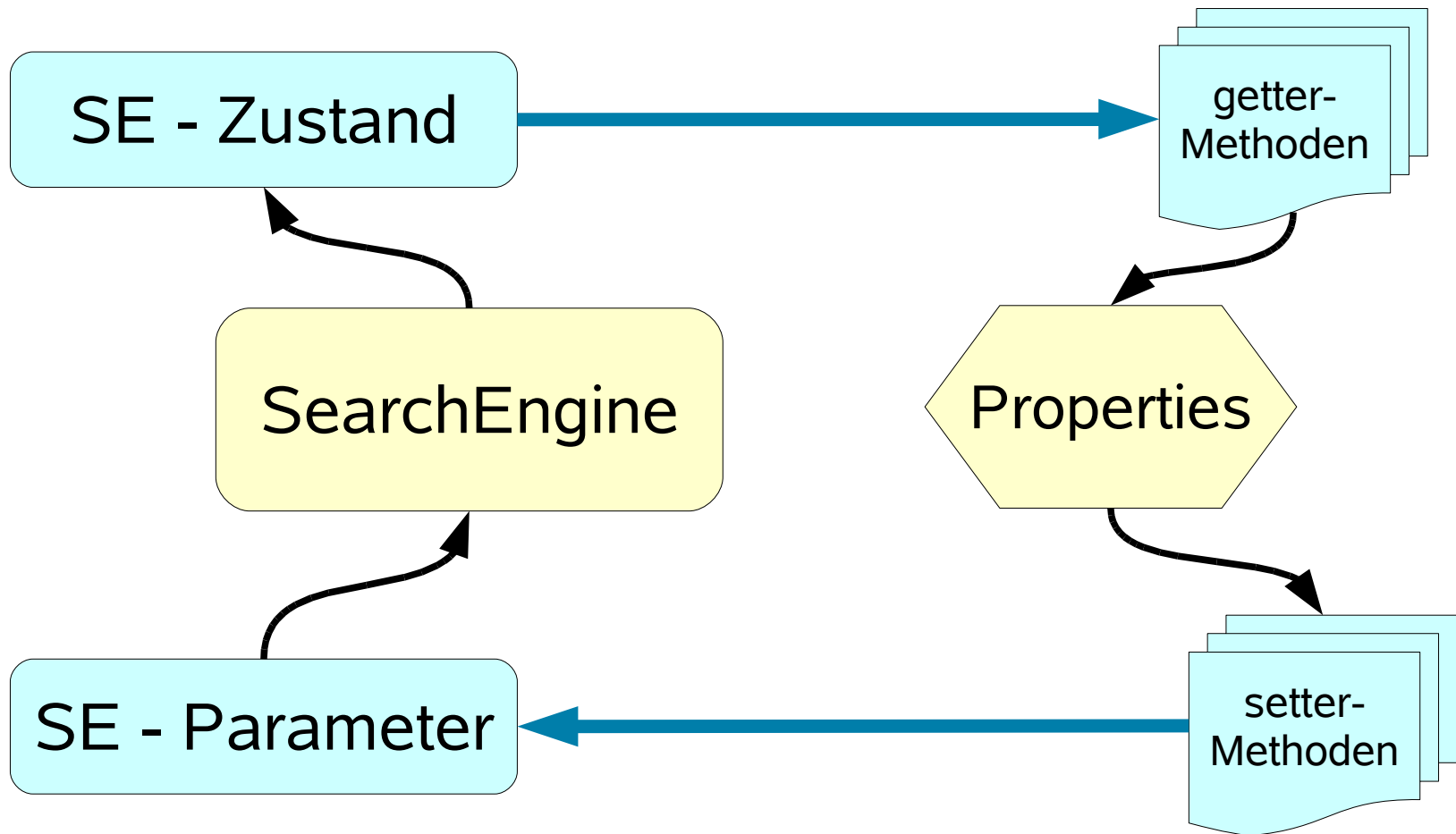


3.3 Darstellung der statistischen Daten

- 1. Möglichkeit
Implementierung einer Darstellung für **jeden** Suchalgorithmus.
- 2. Möglichkeit
Automatische Erstellung von Darstellungen mit Meta-Daten.

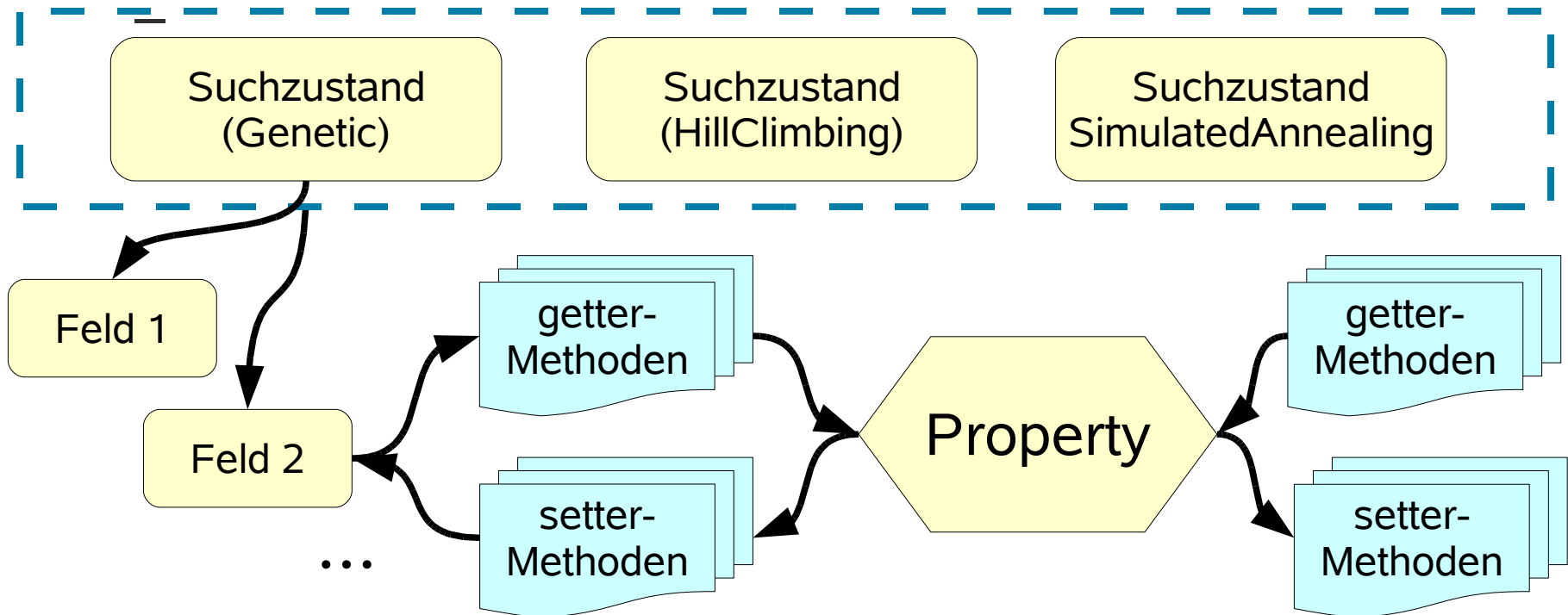


3.3 Darstellung der statistischen Daten



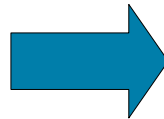
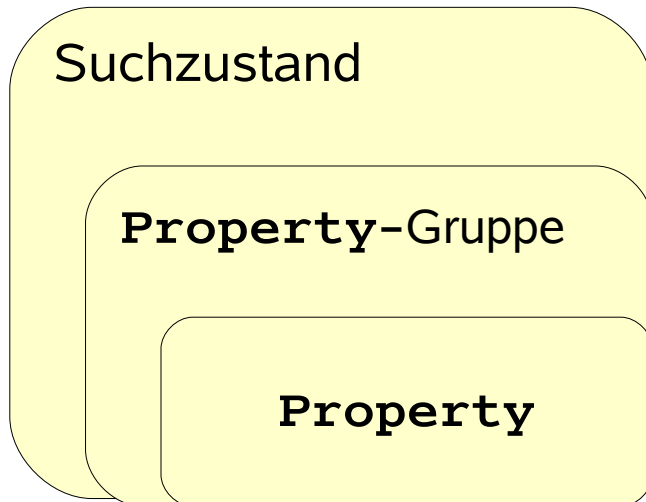
3.3 Darstellung der statistischen Daten

- Felder werden über Properties mittels *Java-Reflektion* ausgelesen und gesetzt.



3.3 Darstellung der statistischen Daten

- Eine Eigenschaft (**Property**) wird mit einer Schaltfläche (**Control**) dargestellt.
- Properties werden zur Übersicht in Gruppen zusammengefasst.



3.3 Darstellung der statistischen Daten

- Arten von Properties
 - Property (allgemein)
 - Integer-Property
 - Time-Property
 - ...

Anzahl der Updates (total)
193

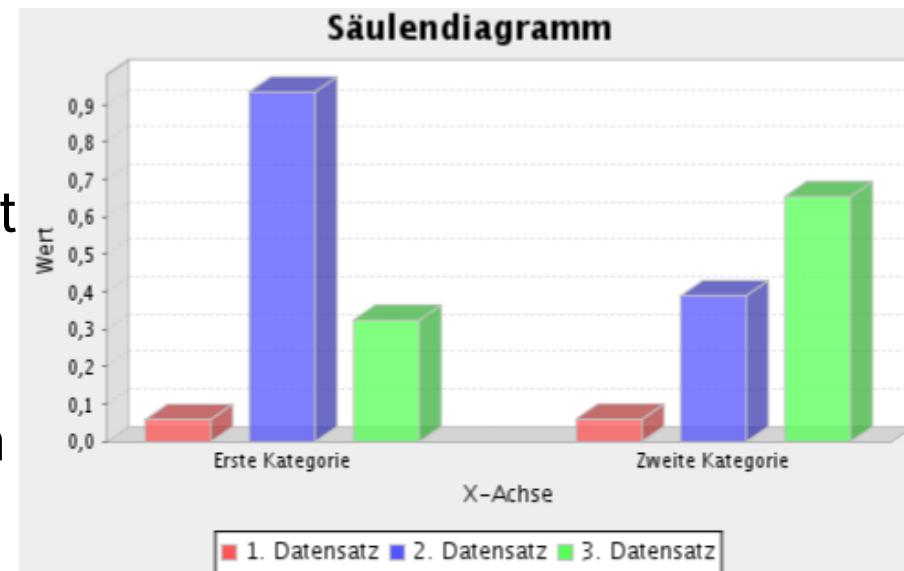
Maximale Anzahl der Erstellungsversuche
10
Minimaler Wert: 1
Maximaler Wert: 134217727

Letztes Update
2006-09-07-10:18:59

- Werte der Properties können geändert werden.

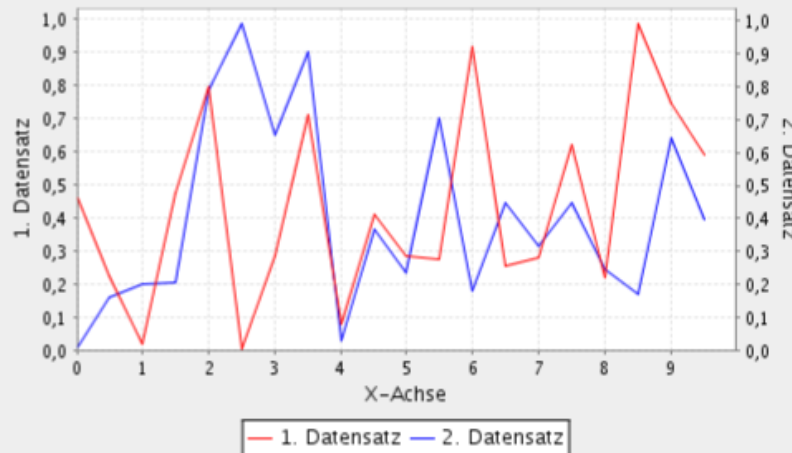
3.4 Diagramme (1)

- Grafische Darstellung von statistischen Daten wurde mit den *JFreeCharts* realisiert
 - Open-Source Sourceforge Projekt (LGPL-Lizenz)
 - Wird seit 6 Jahren entwickelt
 - Wird von ca. 40.000 bis 50.000 Entwicklern benutzt
 - Ermöglicht eine Vielzahl von Diagrammen.

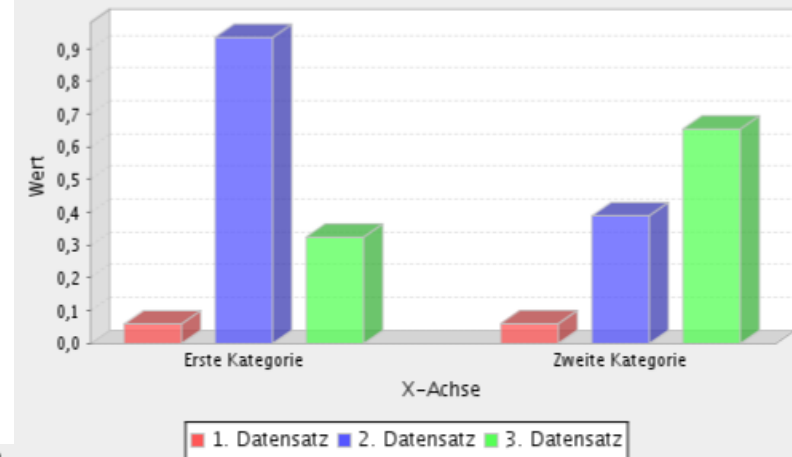


3.4 Diagramme (2)

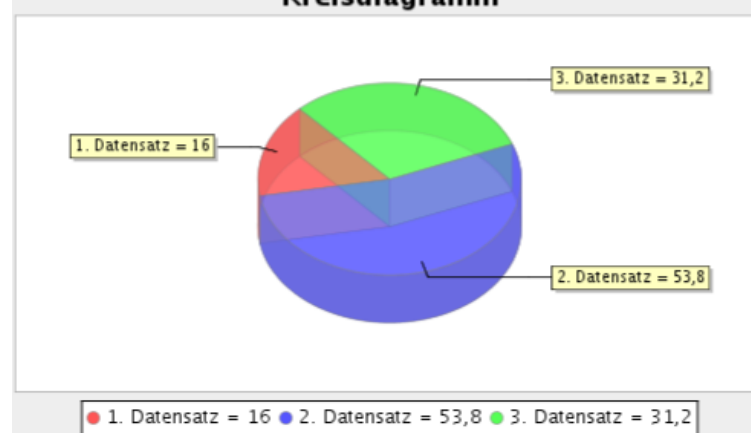
XY Liniendiagramm



Säulendiagramm



Kreisdiagramm



3.5 Vorteile?

- Welche Vorteile hat die vorgestellte Implementierung?
 - Properties sind unabhängig vom Suchzustand und Suchalgorithmus/Suchverfahren
 - Darstellungskomponenten und Diagramme sind unabhängig von Feldern des Suchzustandes
 - Bei neuen Feldern und Suchalgorithmen passt sich die grafische Benutzeroberfläche automatisch an



GUI muss nicht angepasst werden

3.5 Weitere GUI-Funktionen

- Anzeigen und speichern von Log-Dateien
 - Bereits vorhandene Klassen wurden integriert
 - Logger registriert sich *nicht* zusätzlich bei der SearchEngine. Die Daten werden in der GUI gemultiplext.
- Benutzerfreundlichkeit
 - Einstellungen werden gespeichert.
 - Erstellung von Diagrammen vor dem ersten Ereignis (**SearchStateEvent**).
- Ein- und Ausblenden von Log-, Diagramm- und Property-Tabs
- Sprachabhängige Darstellung

Übersicht – Kapitel 4

1. Allgemeines zu DGPF
2. Die Aufgabe
3. Implementierung/Realisierung

4. Demo

**Annähern von Optima mehrdimensionaler
mathematischer Funktionen**

5. Zusammenfassung

Übersicht – Kapitel 5

1. Allgemeines zu DGPF
2. Die Aufgabe
3. Implementierung/Realisierung
4. Demo
- 5. Zusammenfassung**

5. Zusammenfassung

- **Aufgabe:**
Erstellung einer graphischen Benutzeroberfläche
- **Umsetzung:**
Darstellung der Komponenten und der Diagramme ist nicht vom Suchalgorithmus und den Feldern des Suchzustands abhängig.
- **Ausblick**
 - Erstellung weiterer Diagramme
 - Editoren für die Erstellung einer Suchdefinition

Ende

Vielen Dank für eure Aufmerksamkeit!



Fragen?

