Software Reuse Repositories

Andreas Kohlmaier

Inhalt

1. Motivation
   - Quality Improvement Paradigm
2. Software Reuse Systeme
   - Eigenschaften und Maße
   - Klassifizierung
3. Beispiele
   - CodeFinder
   - SFB 501 Experience Base
4. Ausblick

Motivation

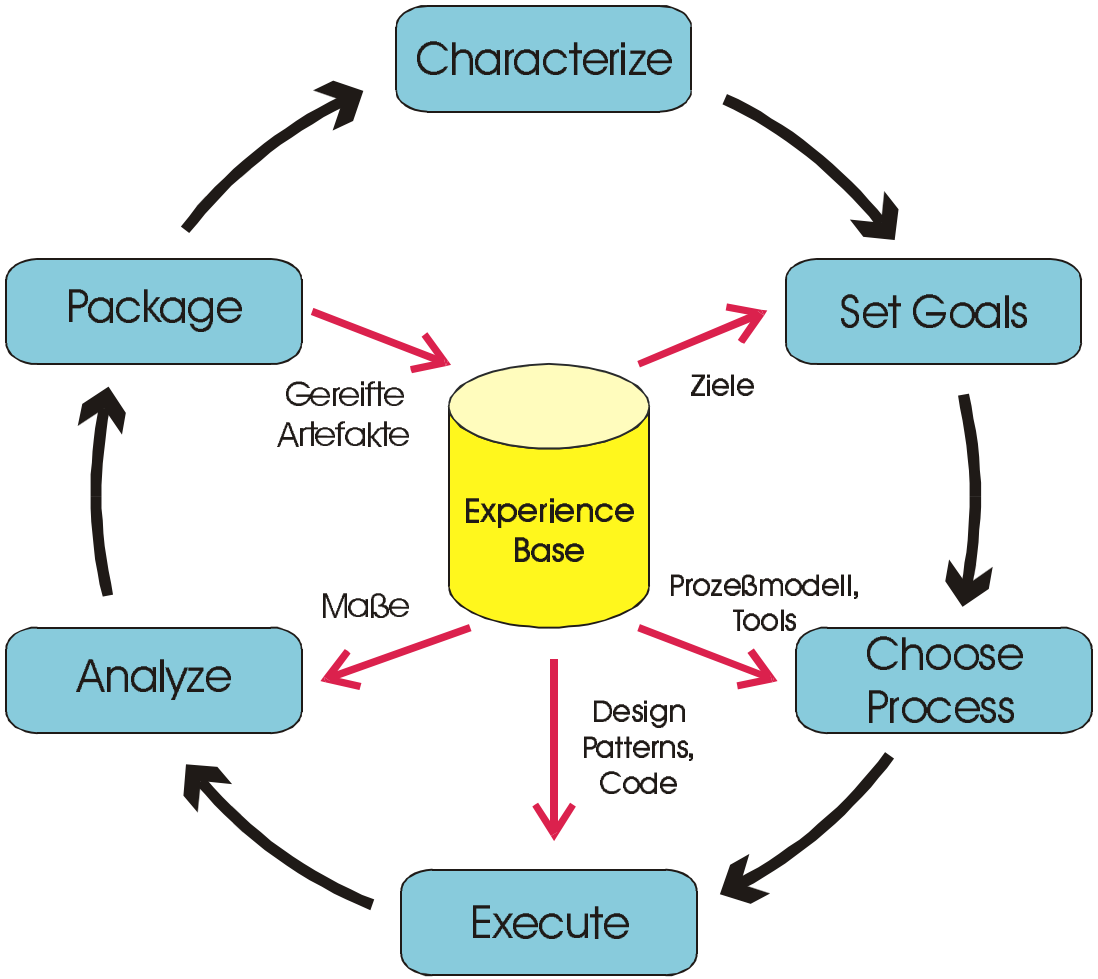
Vorteile durch Wiederverwendung:

- kürzere Entwicklungszeiten
- geringere Kosten
- höhere Qualität

Probleme bei der Durchführung:

- Artefakte vorhanden, jedoch nicht auffindbar.
- Artefakte wegen mangelnder Dokumentation nicht wiederverwendbar.
- Artefakte nur auf einem speziellen System verwendbar.

Quality Improvement Paradigm

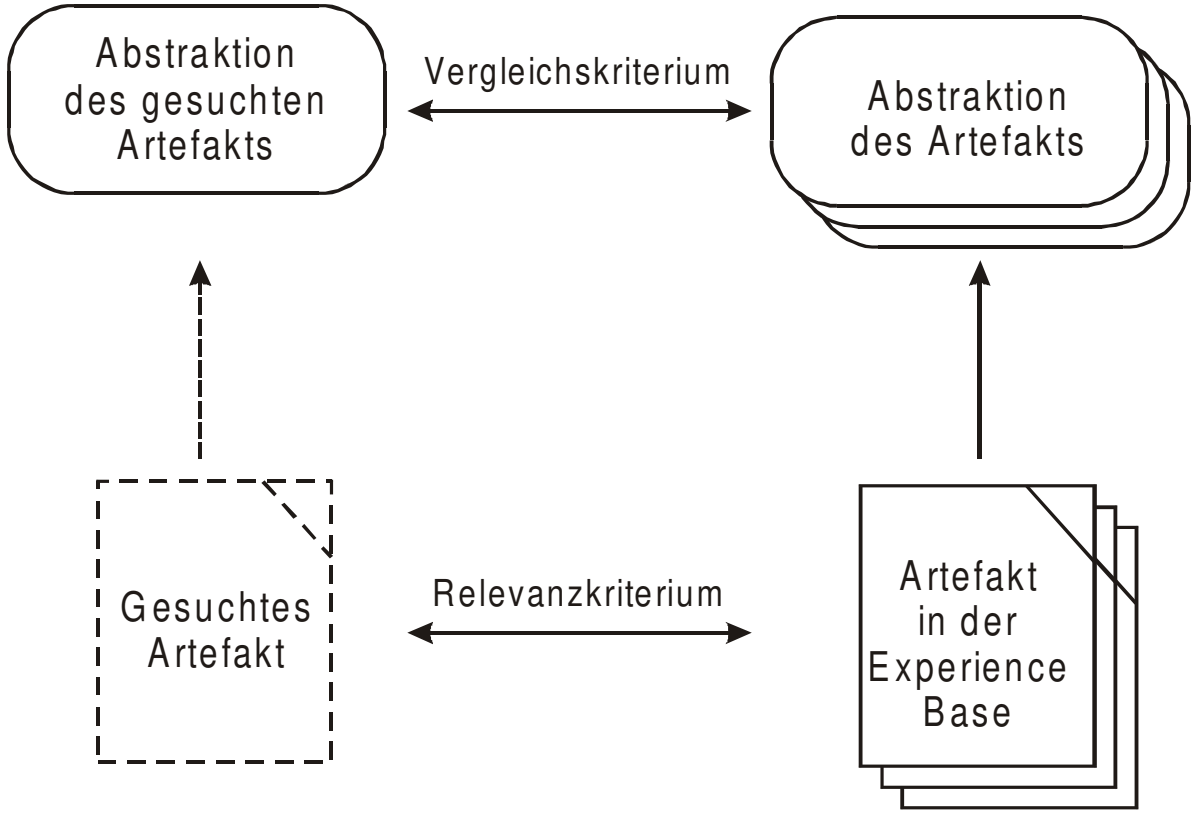


- Prozess zur Unterstützung der Wiederverwendung.
- Artefakte durchlaufen Reifeprozess.
- zentrale Einheit zur Speicherung (*Experience Base*).
- verschiedenartige Artefakte.
- Zusatzaufwand nötig.

Eigenschaften eines Software Reuse Systems

„Artefakte so speichern, dass sie in geeigneter Weise wiederverwendet werden können", Mili

- Relevanzkriterium
- Vergleichskriterium
- Art der Artefakte
- Abstraktion der Artefakte
- Struktur der Daten



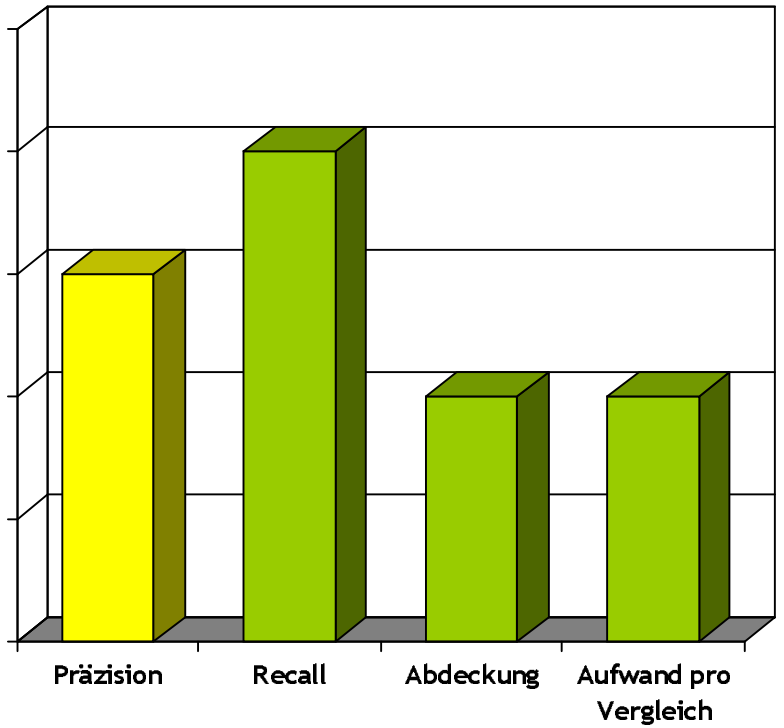
Maße

- Präzision
  Wie viele nicht relevante Artefakte werden zurückgeliefert?
- Recall
  Wie viele relevante Artefakte werden nicht zurückgeliefert?
- Abdeckung
  Welcher Bruchteil der Artefakte wird beim Retrievalprozess betrachtet?
- Aufwand pro Vergleich
  Wie aufwändig ist die Berechnung des Vergleichskriteriums für ein Artefakt?
- Initiale Kosten
- Laufende Kosten
- Benutzbarkeit

Information Retrieval

Artefakte werden als Textdokumente betrachtet und wie beim herkömmlichem Information Retrieval behandelt.

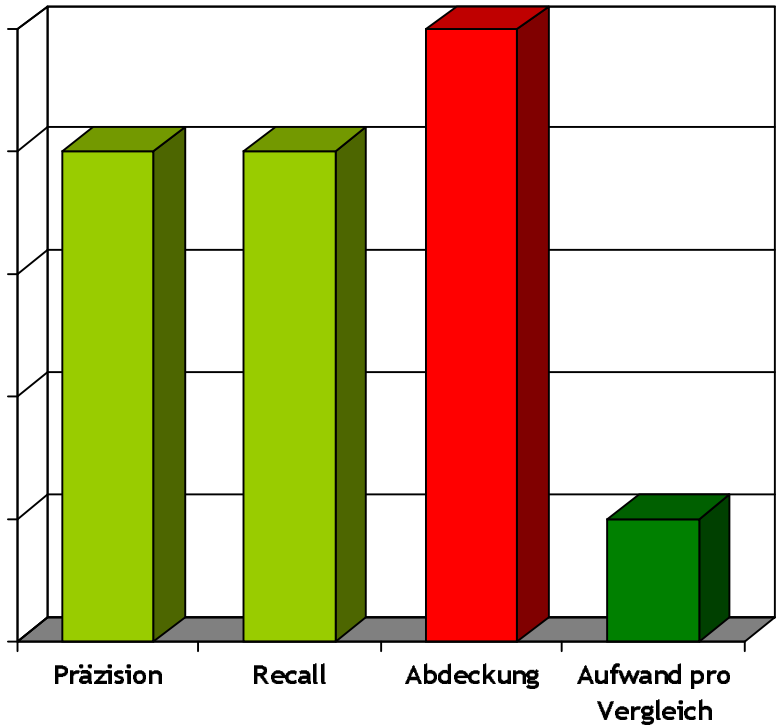
- Artefakte mit textuellem Inhalt.
- Keine explizite Abtraktion.
- Artefakt ist relevant, wenn es das Thema der Query behandelt.
- Artefakt wird zurückgeliefert, wenn es alle Worte der Query enthält.
- Strukturierung an Hand der Beziehungen zwischen den Artefakten.



Deskriptive Methoden

Zu jedem Artefakt wird meist manuell eine Beschreibung erstellt. Diese Beschreibung charakterisiert die wichtigsten Eigenschaften des Artefakts in einer einheitlichen Weise.

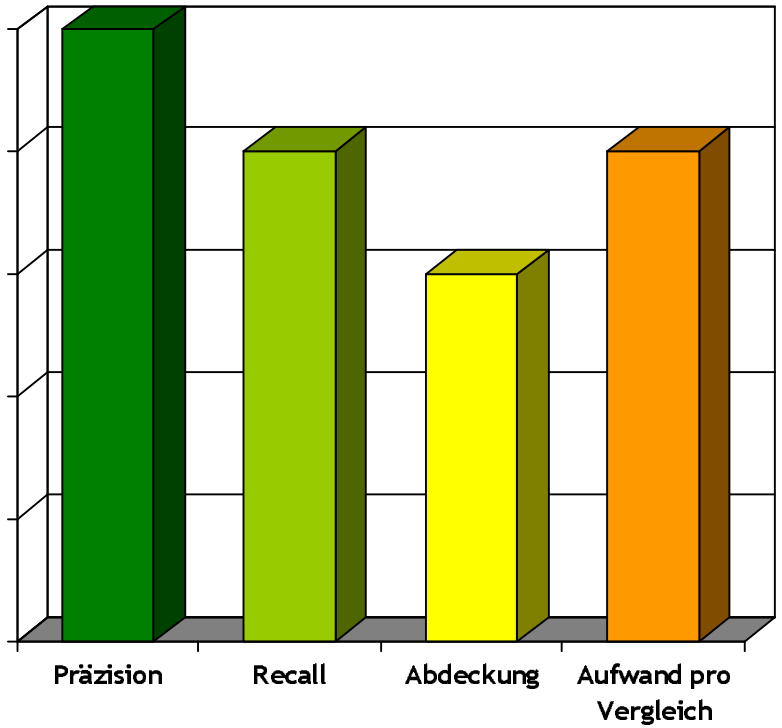
- Beliebige Artefakte.
- Abstraktion ist Vektor von beschreibenden Schlüsselworten aus kontrolliertem Vokabular.
- Artefakt ist relevant, wenn es das Thema der Query behandelt.
- Artefakt wird zurückgeliefert, wenn alle Schlüsselworte der Query in der Beschreibung des Artefakts enthalten sind.
- Meist keine Struktur modelliert.



Operationale Semantik

Ausführbarkeit wird ausgenutzt, um geeignete Artefakte auszuwählen. Dabei wird auf einer zufälligen Menge von Eingabedaten das Verhalten der Artefakte an Hand einer Orakelfunktion überprüft.

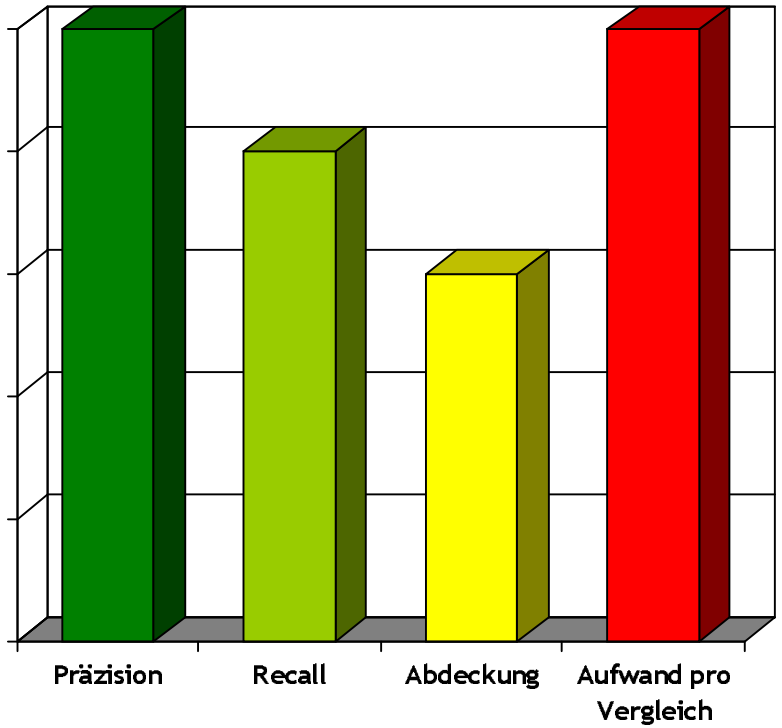
- Nur ausführbare Artefakte.
- Keine explizite Abstraktion, aber Signatur und Treiber zum Aufrufen der Artefakte.
- Artefakt ist relevant, wenn es das in der Orakelfunktion spezifizierte Ein-Ausgabeverhalten hat.
- Artefakt wird zurückgeliefert, wenn es für alle ausgewählten Beispielwerte das korrekte Verhalten zeigt.
- Strukturierung nach Signatur möglich.



Denationale Semantik

Formale Beschreibungen charakterisieren Artefakte. Expliziter Beweis garantiert gewünschtes Verhalten zurückgelieferter Artefakte.

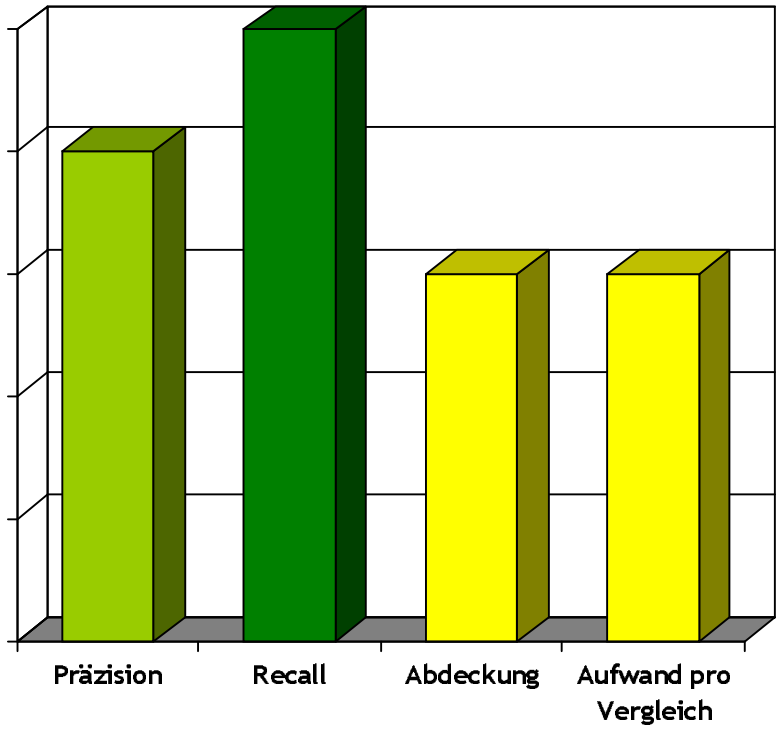
- Formal beschreibbare Artefakte.
- Abstraktion ist formale Beschreibung durch Signatur und Axiomenmenge.
- Artefakt ist relevant, wenn es mindestens die in der Query geforderten Ziele erfüllt.
- Artefakt wird zurückgeliefert, wenn die Ziele der Query aus den Axiomen des Artefakts herleitbar sind.
- Strukturierung nach Signatur möglich.



Topologische Methoden

Es existiert ein Distanzmaß zwischen der Anfrage und den gespeicherten Artefakten. Gesucht werden Artefakte mit minimaler Distanz zur Anfrage.

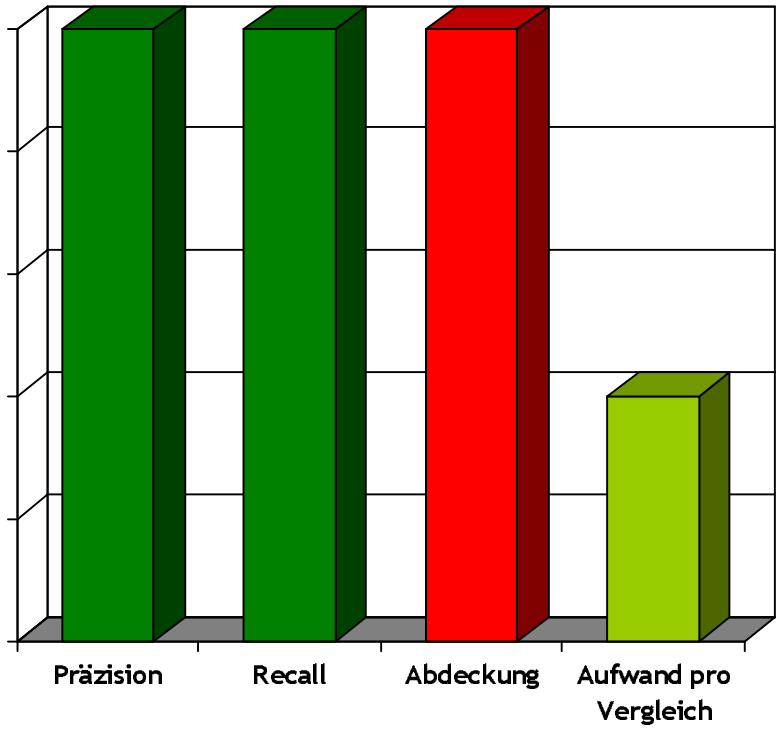
- Beliebige Artefakte.
- Abstraktionen sind Schlüsselworte oder objektorientierte Beschreibungen.
- Geeignetes Distanzmaß wichtig.
- Die der Anfrage am ähnlichsten Artefakte sind relevant.
- Artefakt wird zurückgeliefert, wenn das Distanzmaß einen Schwellwert unterschreitet.
- Strukturierung nach Distanz der Artefakte zueinander möglich.



Strukturelle Methoden

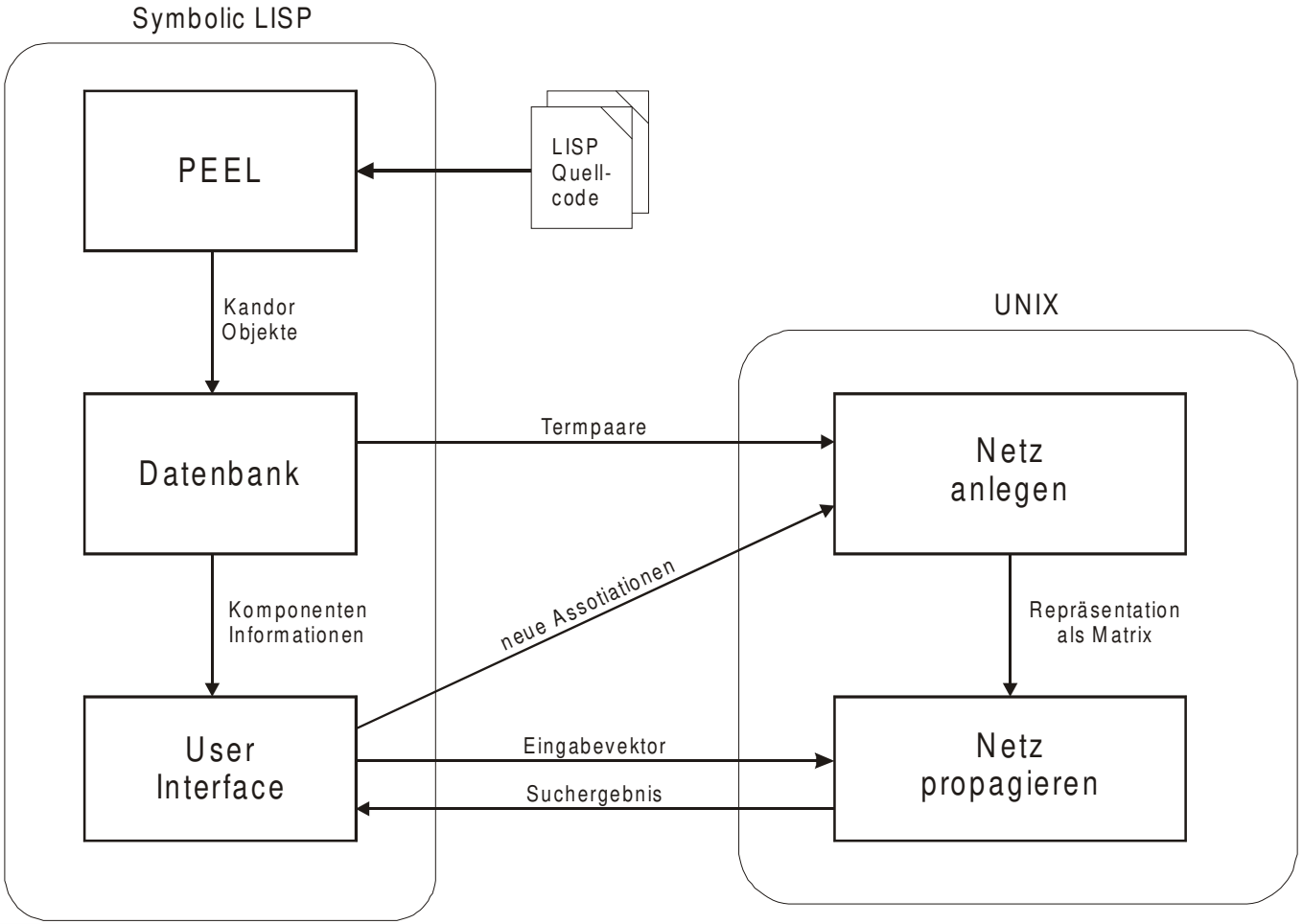
Auswahl der Artefakte allein auf Grund ihrer Struktur. Gesucht werden Artefakte die eine bestimmte Struktur haben, und somit leicht an die Anforderungen angepasst werden können.

- Meist universell einsetzbare Codefragmente oder Design Patterns.
- Abstraktion ist zur Laufzeit erzeugte Strukturbeschreibung oder eindeutiger Name.
- Artefakt ist relevant, wenn es eine Instantiierung der in der Query definierten Struktur ist.
- Vergleichskriterium ist identisch mit Relevanzkriterium.
- Struktur spielt keine Rolle.

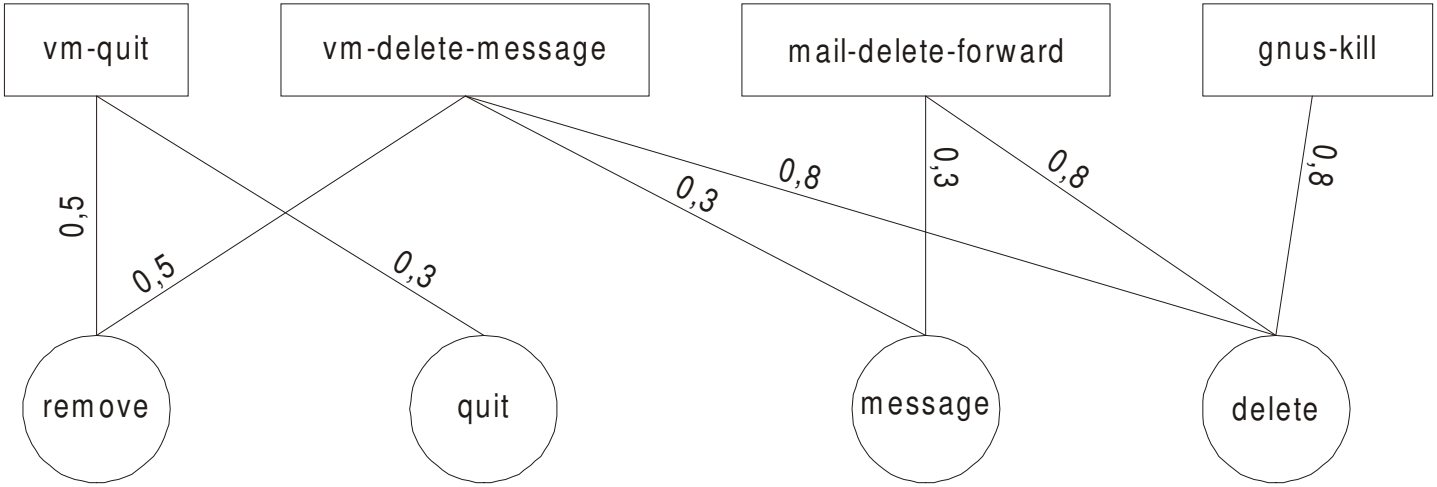


CodeFinder (1)

System zur Speicherung von LISP Funktionen, das die Nachteile der automatischen Indizierung durch einen lernenden Retrievalalgorithmus überwindet.



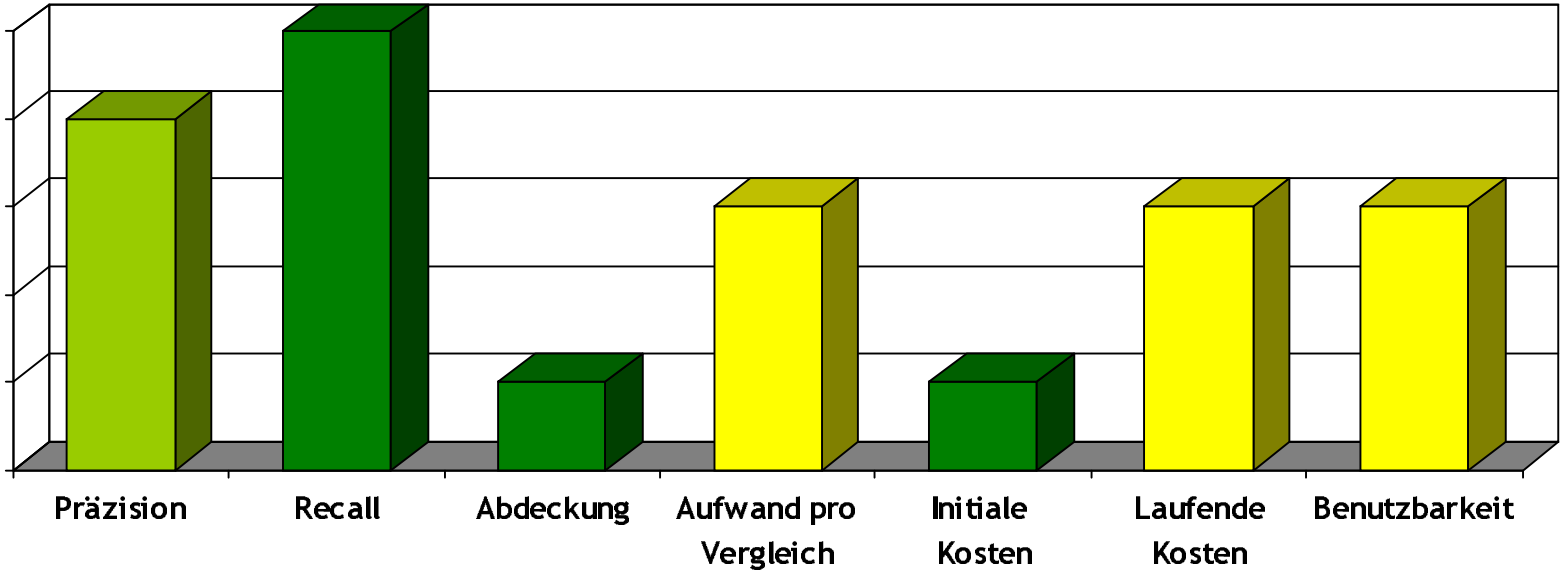
CodeFinder (2)



- Zunächst automatisch aus den Kommentaren im Quelltext generiert.
- Gewichte geben an, wie gut das Schlüsselwort ein Artefakt beschreibt.
- Gesuchte Schlüsselworte werden aktiviert.
- Propagierung des Netzes durch „spreading activation".
- Nach Stabilisierung werden Artefakte mit größter Aktivierung zurückgeliefert.
- Gewichte werden durch Feedback vom Benutzer verändert. Das System erlernt die Begriffe des Benutzers.

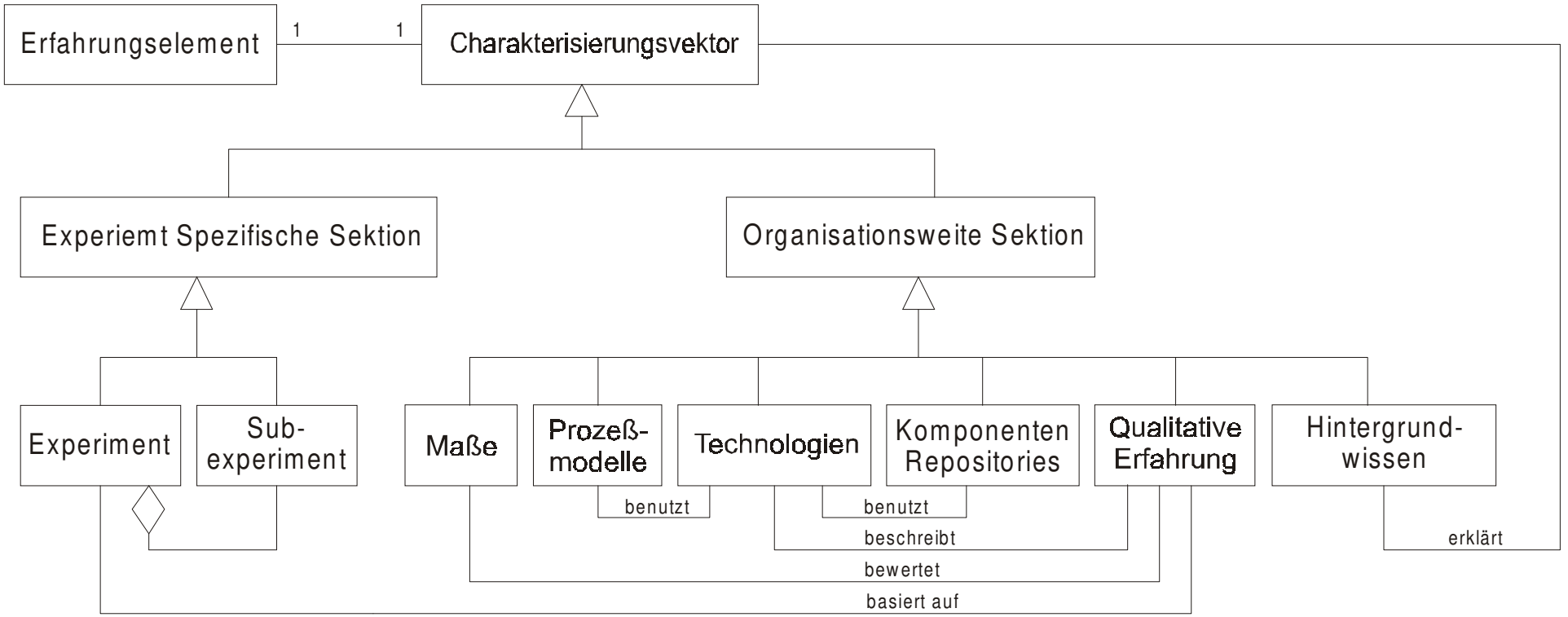
CodeFinder (3)

- System fällt in die Klasse der deskriptiven Methoden.
- Angestrebt wird Wiederverwendung durch Komposition.
- Beschreibungen werden zunächst automatisch erzeugt, und erst während der Benutzung weiter verfeinert.
- Neuronales Netz überwindet Nachteile der automatischen Indizierung.

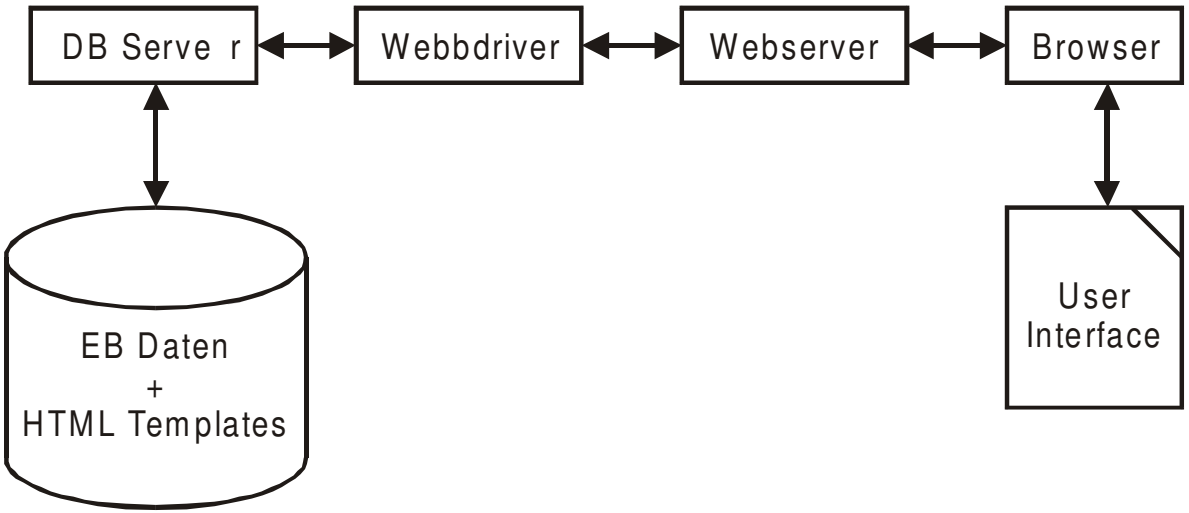


SFB 501 Experience Base

Stark strukturiertes Repository auf Grundlage des Quality Improvement Paradigms zur Speicherung verschiedenartiger Artefakte bei optimaler Verfolgbarkeit.



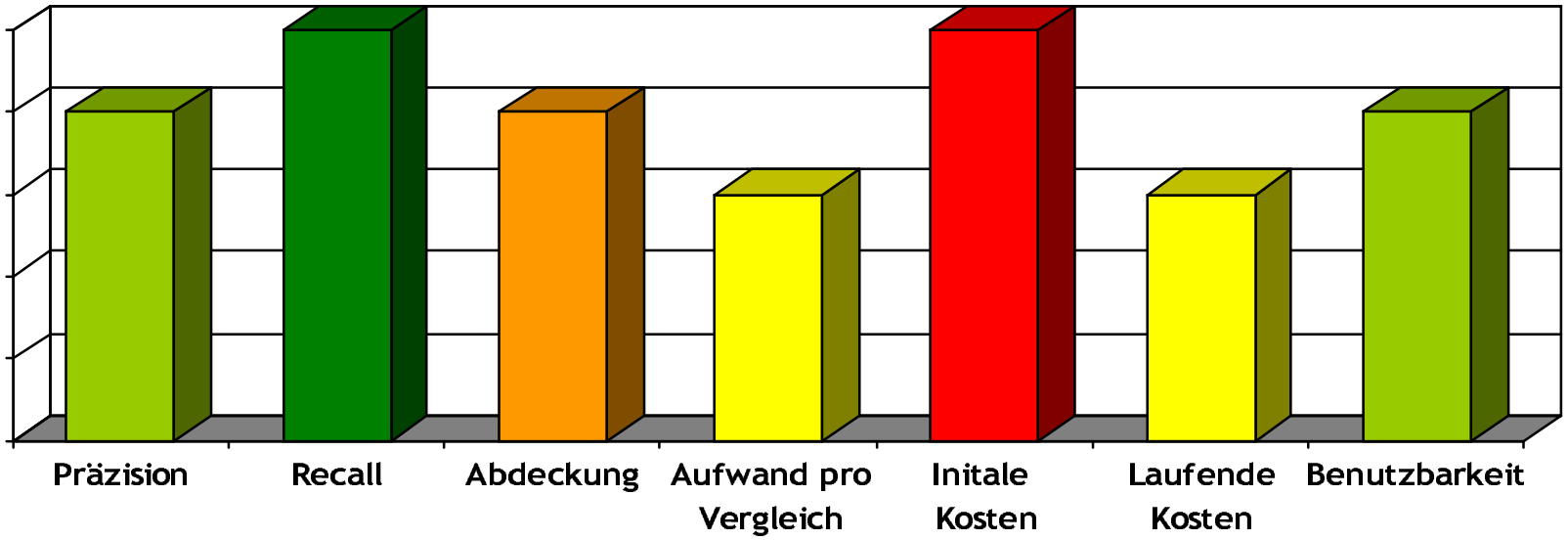
SFB 501 Experience Base (2)



- Erfahrungsdatenbank enthält Erfahrungselemente und HTML Templates der Benutzeroberfläche.
- Seiten werden dynamisch aus Metadatenkatalog der Datenbank erstellt.
- Ähnlichkeitsbasierte Suche über user-defined functions im Datenbankserver.
- Benutzung über ähnlichkeitsbasierte Suche oder direkte Navigation der Erfahrungselemente.
- Sicherheitsaspekte und Benutzerrollen.

SFB 501 Experience Base (3)

- System fällt in die Klassen der deskriptiven oder topologischen Methoden.
- Alle Artefakte des Softwareentwicklungsprozesses speicherbar.
- Ansätze eines prozessunterstützenden Systems durch Experiment Spezifische Sektion.
- Experience Base im Sinne des Quality Improvement Paradigms.



Ausblick

Integrierte Systeme:

- Prozessmodellierung
- Unterstützung während der Ausführung des Prozesses (Workflow Management System)
- Speicherung der produzierten Artefakte für spätere Wiederverwendung