Aufgabe GBIS
(TPCW Benchmark)

17.04.2002

(Markus Bon)

- Kurze Vorstellung der Aufgabe
- Allgemeines
- Bemerkungen zur Strategie
- Einteilen der Gruppen, Ausgabe der Dokumente und Accountanträge

Domäne: Web-basierte Anwendungssysteme

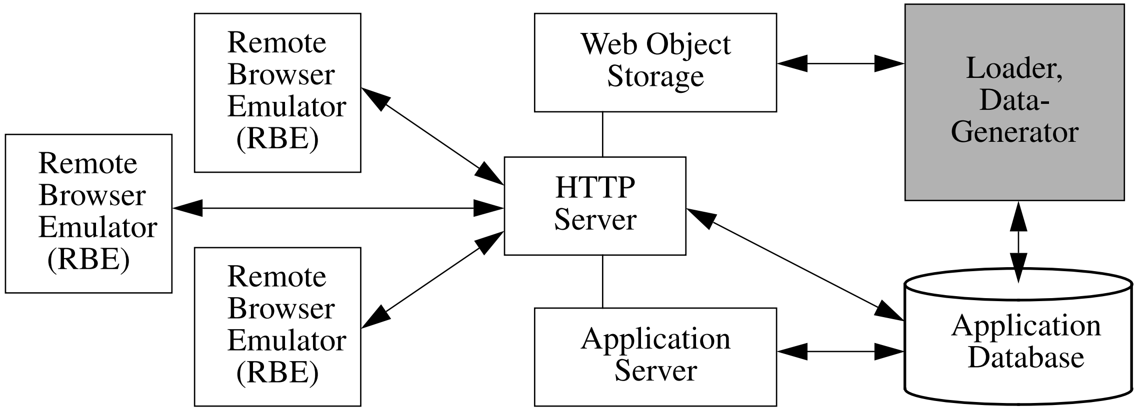
Szenario: Leistungsbewertung von E-Commerce-Systemen
anhand der TPC-W Spezifikation

TPC-W: Simulation eines web-basierten Buchhandel

Notwendigkeit: Aufbau einer geeigneten Datenbasis

Ziel: Entwicklung eines Softwaresystems (in Java) zur Erzeugung der Datenbasis

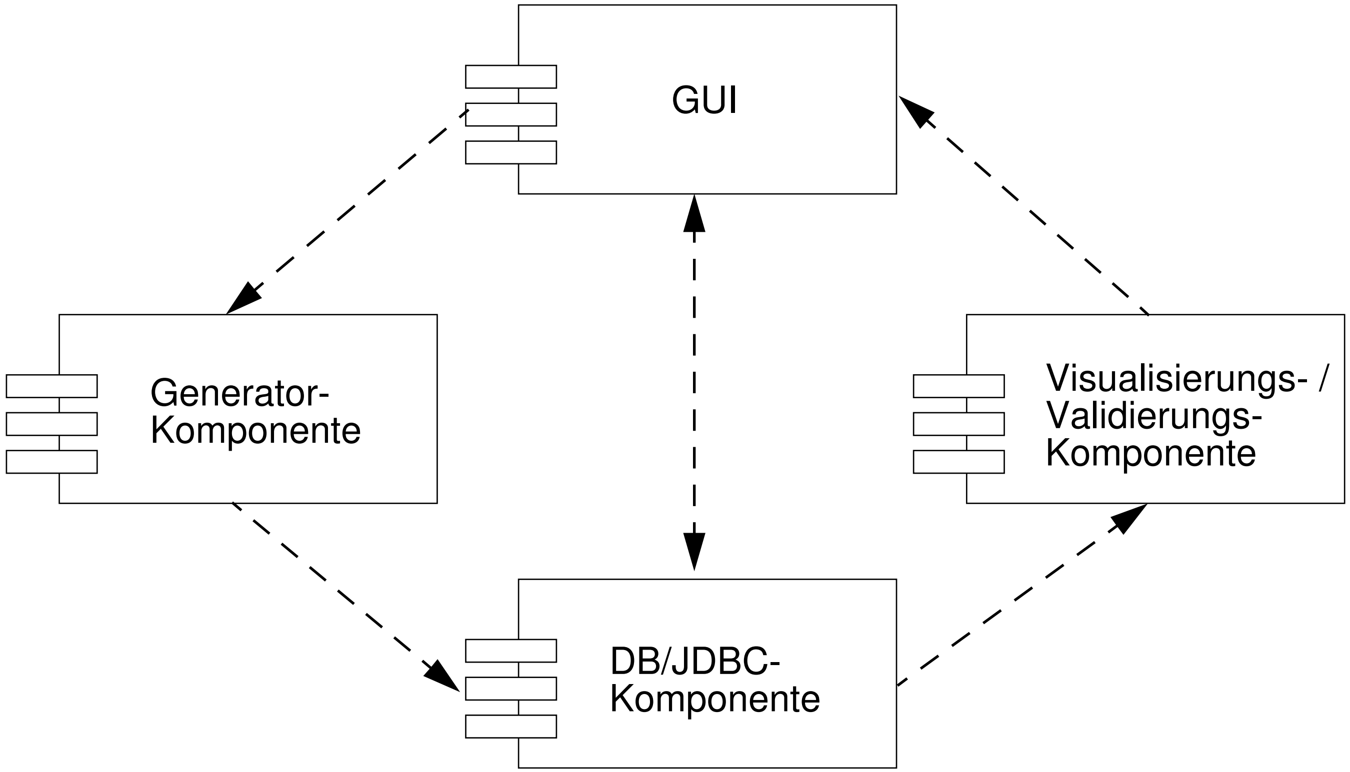
Übersicht:



Datenbasis:

- differenzierte Anforderungen an
  - Verteilung,
  - Umfang und
  - Struktur der Daten
- Verwaltung durch Datenbanksystem (Zugriff über JDBC)

Das Teilsystem zur Population der Datenbank ist im Praktikum zu entwickeln.



Komponenten definieren Packages!

* DB / JDBC:

- Eine Datenbank mit entsprechendem Schema ist vorgegeben
  und muß *nicht* modelliert werden.
- Zugriff auf der roulett mittels „dbaccess“.
- Zugriff vom SCI-Cluster durch eigene Java-Programme mittels JDBC
- DB-Komponente kapselt alle DB- / JDBC-Aspekte
  und versteckt alle Informix Eigenheiten und Erweiterungen (bspw. SQL-Dialekt)
- Auch: Connection-Management (An- / Ab-Meldung) evtl. Transaktionskontrolle

* Generator:

Paragraph 4 der TPC-W Spezifikation definiert Anforderungen an

- Generatorkomponente und
- Zufallsgenerator

- grafische Oberfläche zur komfortablen Bedienung des SW-Pakets
  - alle Funktionen sollen dem Benutzer auf einfache Art und Weise zugänglich sein
- geforderte Funktionen:
  - Herstellen/Beenden einer Verbindung zur Datenbank
    - Eingabe von JDBC-URL, Benutzer und Passwort
  - Generieren einer neuen Datenbank
    - Einstellen der Parameter (#EBs, #Items)
    - Starten/Beenden/Unterbrechen der Generierung
    - Statusanzeige

* geforderte Funktionen:
  - Anzeige verschiedener Statistiken zu gewähltem Attribut
    - Anzeige von Minimum, Maximum, Durchschnitt, #Werte, #verschiedener Werte
      (bei String-Attributen sind diese Angaben auf die Länge zu beziehen)
    - grafische Darstellung der Werteverteilung
      (z. B. als Kuchen, Linien oder Blockdiagramm, Auswahl sei dem Benutzer möglich)
  - Auswahl von Sätzen einer Tabelle mittels ID als Suchschlüssel
    - Eingabe einer ID durch Benutzer oder zufällige Erzeugung
      Anzeige des gefundenen Satzes

* Durchlaufen aller Phasen eines Softwareentwicklungsprozesses (Wasserfallmodell)
    - Anforderungsanalyse
    - Systementwurf und Modellierung
    - Kodierung
    - Integration und Validation
* Abgaben in der Regel Mittwochs, 8.00h
* Besprechung Freitags nach Vereinbarung mit dem Betreuer

Betreuer:	Markus Bon Ulrich Marder Elke Moritz
Hiwis:	Martin Husemann Andreas Kuempel Thorsten Lang Marco Müller
WWW:	http://wwwdbis.informatik.uni-kl.de/courses/praktika/swp/2002/index.html
Newsgruppen:	kl.inf.swp kl.inf.swp.discussion
Gruppe:	http://roulett:8888/swp/gbis/gruppen/swpg**.html lokal: swpg**u1/public_html/index.html *Zur Dokumentation nutzen - zählt mit zur Abgabe!*
Abgaben:	E-Mail Subject: „Abgabe[1-4], Gruppe[19-24]“

Maximal 100 Punkte möglich:

- 50 Punkte ergeben sich aus der Bewertung der Gruppenleistung
- 50 Punkte ergeben sich aus der Leistung des Einzelnen (Testat/Kolloquium/allg. Eindruck)

Zum Bestehen:

- Eine Aufgabe gilt als bestanden, wenn min. 60 Punkte erreicht wurden.

* Abhängigkeiten zwischen Tabellen:
    - Bestehen Fremdschlüsselbeziehungen zwischen zwei Tabellen, so ist darauf zu achten, daß alle Tupel an der Relation teilnehmen.
    - Beispiel 1: Jede Bestellung hat auch min. 1 Bestellposition
    - Beispiel 2: Jeder Autor hat min. 1 Buch geschrieben
    - Zahlungsinformationen, I_Related, ...

* Schema-Definition:
    - Gibt's auf der Materialien-Seite

- **Grundsatz**: *Tabellen nacheinander vollständig füllen!*
- Länderdaten sind vorgegeben
- Es bietet sich an, zuerst vollständig in der vorgegebenen Kardinalität die Adressen zu generieren, da sie von keiner anderen Tabelle abhängig sind.
- Danach können die Kundendaten vollständig erzeugt werden.
- Analog kann bei Autoren und Produkten verfahren werden.
- **Bestellungen**: Bestellungen, Bestellpositionen und Zahlungsinformationen „gemeinsam“ generieren.

- **Also**:
  - Eine zufällige Anzahl Bestellpositionen erzeugen und merken!
  - Die Bestellung erzeugen und merken!
  - Die Zahlungsinformation erzeugen und merken!
  - Die gemerkten Objekte zusammen in die DB schreiben.
  - Durch das „Merken“ erspart man sich viele DB-Anfragen und hält die DB konsistent