

DATA-WAREHOUSE-PRAKTIKUM

Universität Leipzig, Institut für Informatik Abteilung Datenbanken Prof. Dr. E. Rahm V. Christen, L. Lange, F. Rohde, B. Uhrich

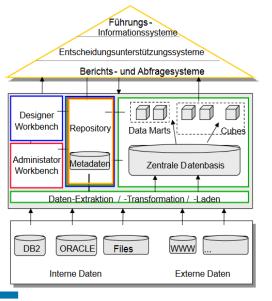
{christen, lange, rohde, uhrich}@informatik.uni-leipzig.de http://dbs.uni-leipzig.de

ORGANISATORISCHES

- Ziel: Realisierung eines typischen DWH-Projekts
 - Kennenlernen der echten, praktischen DWH-Probleme
- Zielgruppe:
 - Master-Studierende Data Science
 - Kenntnisse:
 - Notwendig: Data Warehousing Verständnis
 - Hilfreich: VL Data Mining, DB-Praktikum
- Ablauf:
 - Gruppenarbeit mit 2 Studierenden pro Gruppe
 - Bearbeitung von 3 Aufgaben → jeweils Testat
- Aufgabenstellung und Informationen:
 - Moodle

DATA-WAREHOUSE

- Ausgangsproblem
 - Viele Unternehmen haben Unmengen an Daten, ohne daraus ausreichend Informationen und Wissen für kritische Entscheidungsaufgaben ableiten zu können.



ETL-Prozess:

- Extraktion
 Laden der Quelldaten in temporären Arbeitsbereich
- Transformation
 Anpassung an das Zielschema
 Datenbereinigung und
 Integration
- Laden
 Data Cube Erstellung

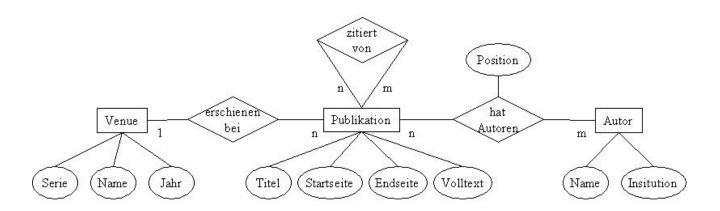
SZENARIO: ZITATIONSANALYSE

- In wissenschaftlichen Arbeiten werden andere Arbeiten zitiert
- Anzahl der Zitierungen als Indikator für den wissenschaftlichen Einfluss (Impact) und Qualität
 - Wie häufig wird eine Publikation zitiert?
 - Wie häufig werden Publikationen des Venues (Konferenz oder Journal) im Durchschnitt zitiert?
 - Wie ist die durchschnittliche Zitierungszahl von Autoren?
- Beziehungen zwischen Personen, Institutionen, Publikationen und Fachbereichen
 - Welche Autoren zitieren welche anderen Autoren?
- Verlagerung von Forschungsschwerpunkten

DATENQUELLEN

- DBLP Bibliography:
 - Manuelle gepflegte Website, die komplette Listen verschiedener Venues aus dem Informatik-Bereich enthält.
- ACM Digital Library:
 - Portal der Association for Computing Machinery
 - enthält ebenfalls komplette Listen verschiedener Venues
- Google Scholar:
 - Suchmaschine für wissenschaftliche Publikationen
- Relevante Teilmenge der Daten steht als CSV- und XML-Dateien zur Verfügung

ZIELSCHEMA



AUFGABEN

1. Datenimport

- Import der XML- und CSV-Dateien
- Datenextraktion mittels TSQL
- Relationale Speicherung der Daten dem Zielschema entsprechend wobei Zusatzinformationen beibehalten werden sollen

2. Data Cleaning

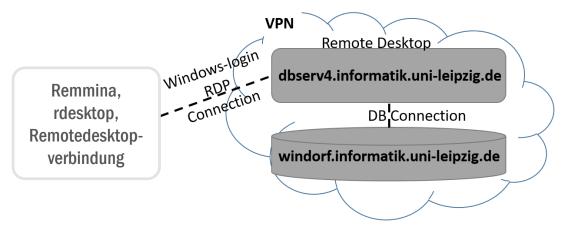
- Objekt-Matching: Erkennen gleicher Publikationen in verschiedenen (oder gleichen) Datenquellen
- Daten-Normalisierung: Normalisierung der Institutionsnamen
- Ableitung neuer Daten: Identifikation von Selbstzitierungen

3. Cube-Erstellung, OLAP und Data Mining

- Star-Schema-Erstellung und Datenimport
- OLAP-Analyse, MDX-Anfragen
- Data Mining: Assoziationsregeln zur Bestimmung ähnlicher Venues

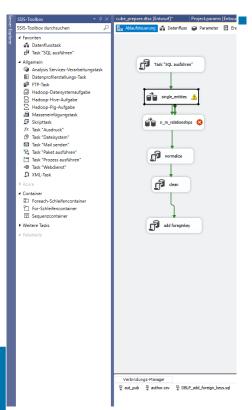
ENTWICKLUNGSUMGEBUNG

- Bereitstellung vollständiger Entwicklungsumgebung
 - Windows-Accounts nach Themeneinschreibung



- Vorteil: kein Installationsaufwand, bessere Hilfestellung mgl.
- Nachteil: steile Lernkurve, Tool "gewöhnungsbedürftig"

REALISIERUNG



SQL Server Data Tools innerhalb von Visual Studio

- Drag&Drop-Workflow-Erstellung (keine direkte Programmierung)
- Per Remote-Desktop-Verbindung: dbserv4.informatik.uni-leipzig.de
- Client-Anwendung für zentralen
 Datenbankserver: SQL Server 2019 auf windorf.informatik.uni-leipzig.de

REALISIERUNG

- SQL Management Studio
 - Erstellung von SQL-Skripten (Tabellenerstellung, Datentransformation,

etc.)

```
Datei Bearbeiten Ansicht Abfrage Projekt Debuggen Extras Fenster Hilfe
 📆 + 🖂 - 🔯 🖼 💋 🔽 Neue Abfrage 😘 😘 😘 🚜 🚜 🖽 📇 🛷 - 🙉 - 😂 + 📖 🙈 👂
                                                                                                                              289
                              · ! Ausführen » Debuggen = V D 中日 T 中日 O T O 至 2 速度 %;
57 July dwhorak00
                                                                 * P X ACM_create.sgl - wi...ATIK\christen (76)) X SQLQuery1.sgl - win...ATIK\christen (75))*
 Verbinden * 🐉 🐩 = 🔻 🕝 🕉
                                                                             USE [duhprak_30]

☐ Windorf (SQL Server 15.0.4390.2 - INFORMATIK\christen)

                                                                               if OBJECT_ID('dbo.acm_author_tmp') IS NOT NULL drop table [dbo].[acm_author_tmp]
   Datenbanken
                                                                              if OBSECT_IN('dbo.acm_citing_pub_tmp') 15 NOT NULL drop table [dbo].[acm_citing_pub_tmp] if OBSECT_IN('dbo.acm_publication_tmp) if OBSECT_IN('dbo.acm_publication_tmp) if OBSECT_IN('dbo.acm_publication_tmp) if OBSECT_IN('dbo.acm_pub_tmp') 15 NOT NULL drop table [dbo].[acm_publication_tmp] if OBSECT_IN('dbo.acm_pub_tmp') is NOT NULL drop table [dbo].[acm_pub_tmp']
      🗵 🗀 Systemdatenbanken

    Datenbank-Momentaufnahmen

      iii dwhprak_30
                                                                              if DBJECT_ID('dbo.acm_venue_series_tmp') IS NOT NULL drop table [dbo].[acm_venue_series_tmp]
      iii dwhprak christen
      ill dwhprak rost
                                                                              The ACM Data is imported in temporary tables first. After that, the data will be normalized into the final schema
      ☐ [] dwhprak00

    Datenbankdiagramme

         Tabellen
                                                                             CREATE TABLE [dbo].[acm_author_tmp] (
            Systemtabellen
                                                                                   -- defined by XSD
            [pos] tinyint,

    dbo.acm_author_institution

                                                                                   [institution] nvarchar(255),
            @ dbo.acm_author_publication
                                                                                   [publication id] numeric(20.0)

    dbo.acm_cited_by

            and dbo.acm insitution normalized
                                                                                 manning table between publication
            III dbo.acm institution
                                                                             CREATE TABLE [dbo] [acm_citing_pub_tmp]
            a dbo.acm institution fuzzy grouped
                                                                                   [pk_id] bigint NOT NULL PRIMARY KEY IDENTITY,

 dbo.acm_publication

            i dbo.acm_venue

    ■ dbo.acm_venue_publication

                                                                                     line number in source 30th
            dbo.acm_venue_series
                                                                                  [publication_id] numeric(20,0) NOT NULL.
            ... has cited publication B
            a dbo.cube AuthorPublication
                                                                                   [id] bigint NOT NULL,

    dho cube Publication

                                                                                   [text] nvarchar(4000)
            III dbo.cube Time
            and the cube Title
                                                                              CREATE TABLE [dbo] [acm_publication_tmp] (

 dbo.cube VenueSeries

            dbo.dblp_acm
                                                                                   [title] nvarchar(1000)
            [fulltext] nvarchar(255)
            iii dbo.dblp_author_publication
                                                                                   [start_page] int,
            @ dbo.dblp.gs
            dbo.dblp_publication
            [publication id] numeric(20.0)
            i dbo.dblp_venue_publication
            dbo.dblp venue series
                                                                             CREATE TABLE [dbo].[acm_venue_tmp] (
            @ 3 dbo.gs_author
                                                                                   [id] bigint,
            dbo.gs_author_publication
            [text] nvarchar(255),
            i dbo.gs_publication

    □ dba.gs_venue

                                                                                   [publication_id] numeric(20,0)
            i dbo.gs_venue_publication
            CREATE TABLE [dbo].[acm_venue_series_tmp] (
            i dbo.mine_AssoRegeln
                                                                                   [id] nvarchar(255)
         Sichten

    Programmierbarkeit

                                                                                   [publication_id] numeric(20,0)
         ■ Service Broker
```

ORGANISATORISCH

- Jeder Aufgabe ist ein Tutorial zugeordnet
 - Beschreibung der Aufgabe
 - Grundlegende Vorgehensweise (inkl. Screenshots) & Hinweise
- Software-Ergebnis sind ausführbare Projekte, welche im Testat ausgeführt/begutachtet werden
 - Terminabsprache rechtzeitig individuell mit Betreuer per E-Mail
 - <u>Deadlines</u> siehe Moodle(Testat 1 bis zum 13. Dezember)