Datenbanken IIA: Datenbank-Entwurf

(Wintersemester 2018/19)

Prof. Dr. Stefan Brass

Institut für Informatik

Übung: M.Sc. Mario Wenzel

Wichtigste Lernziele

Nach dieser Vorlesung sollten Sie ...

- ein Datenbank-Schema auch für (etwas) größere Anwendungen erstellen können.
- Korrektheit und Qualität von DB-Schemata bewerten und alternative Lösungen vergleichen können.
- mehrere ER-Notation benutzen können.
- mit mindestens einem DB-Entwurfswerkzeug praktisch umgehen können.
- Trigger und objektrelationale Features beherrschen.

Themen

- Datenbank-Projekte: Übersicht
- Fortgeschrittener Entity-Relationship-Entwurf

  Weitere ER-Konstrukte: Subklassen, mehrstellige Relationships, ...
- Oracle SQL Developer Data Modeler (ER-Entwurf)

  Praktische Erfahrung mit einer etwas größeren Aufgabe ("Projekt").
- Übersetzung von ER-Modell ins Relationenmodell
- Relationale Normalformen (vertieft)
- UML Klassendiagramme
- Trigger, Stored Procedures, Objektrelationales SQL

  Ggf. Reverse Engineering, Ontologien

Voraussetzungen

Dies ist eine fortgeschrittene DB-Vorlesung.
Vorausgesetzt werden:

- Relationales Modell
- SQL, insbesondere `CREATE TABLE`
- Nullwerte, Schlüssel, Fremdschlüssel
- Erster Eindruck vom ER-Modell inklusive Klassifizierung von Relationships (z.B. "one-to-many").
- Programmierkenntnisse
- (mehr oder weniger:) Englische Sprachkenntnisse

Zeit und Ort

Vorlesung (2 SWS):

- Mittwochs, 10^{15}–11^{45}, Raum 3.04 [StudIP]

Übung (2 SWS, Beginn 24.10.2018):

- Tafelübung und Rechnerübung:
  - ◇ Mittwochs, 12^{15}–13^{45}, Raum 3.04 [StudIP]

    Bzw. PC-Pool, Raum 3.34. Aufteilung Tafel/PC variabel.
  - ◇ Offenbar reicht eine Übungsgruppe nicht. Vielleicht Mittwochs 14–16?
- "Hands-on Experience" ist wichtig!

Ansprechpartner (1)

Dozent: Prof. Dr. Stefan Brass

- Email: `brass@informatik.uni-halle.de`

  Bitte in Betreff-Zeile Präfix `[dd18]` und aussagefähiger Text.
- Büro: Von-Seckendorff-Platz 1, Raum 313
- Telefon: 0345/55-24740
- Sprechstunde: Montags, 12^{00}–13^{00}
- Frühere Unis: Braunschweig, Dortmund, Hannover, Hildesheim, Pittsburgh, Gießen, Clausthal.
- Oracle8 Certified Database Administrator IBM Certified Advanced DBA (DB2 UDB 8.1).

Ansprechpartner (2)

Übungsleiter: M.Sc. Mario Wenzel

- Büro: Von-Seckendorff-Platz 1, Raum 315
- Telefon: 0345/55-24776
- Email: `mario.wenzel@informatik.uni-halle.de`

Sekretärin: Ramona Vahrenhold

- Büro: Von-Seckendorff-Platz 1, Raum 324
- Telefon: 0345/55-24750, Fax: 0345/55-27333
- Email: `vahrenho@informatik.uni-halle.de`

Webseite

`http://www.informatik.uni-halle.de/~brass/dd18/`

- Folien
- Hausaufgaben
- Alte Klausuren
- Verweise auf CASE-Tool Hersteller etc.
- Verweise auf Literatur im WWW

  Hinweise auf weitere interessante Quellen in Web (Links) sind willkommen!

Modulleistung/Prüfung

Klausur (voraussichtlich 27.03.2019, ggf. Änderung):

- Es können drei DIN A4-Blätter mit beliebigen Notizen verwendet werden.

  Sie dürfen Vorder- und Rückseite beschreiben oder bedrucken.

- Praktische Anwendung, kein Auswendiglernen.

  Z.B. ER-Entwurf, ER-Diagramme auf Äquivalenz prüfen, Übersetzung ins relationale Modell, Implikation von FAen, Normalformen.

- Sicher bestanden haben Sie mit 60% der Punkte.

  Nicht mit 50%. Nach Durchsicht der Klausur werden die genauen Grenzen festgelegt. Ab 95% ist eine 1.0 garantiert.

- Nachprüfung mündlich Mitte/Ende Mai 2019.

Studienleistung: Projekt (1)

- Größere Hausaufgabe (in mehreren Varianten), auch eigene Projekte nach Absprache möglich.

  Aber kein Recyclen von Aufgabenlösungen aus anderen Vorlesungen, auch keine bezahlte Aufgabe.

- Einzeln bearbeiten.
- Verlangt sind
  - ◇ Bericht (ca. 15 Seiten) und
  - ◇ Verteidigung (Seminarvortrag mit Diskussion, ca. 15–20 min Vortrag, 10 min Diskussion).
  - ◇ Kurzkommentierung eines anderen Berichts.

Studienleistung: Projekt (2)

- Beginnen Sie mit einer Projekteinführung und mindestens exemplarisch den Anforderungen (welche Fragen aus der Datenbank zu beantworten sind).
- Das ER-Schema sollte ca. 7–15 Entitytypen haben, möglichst auch fortgeschrittene Konstrukte (z.B. Subklassen) verwenden.

  Im Bericht soll es dazu eine gute Dokumentation geben, nicht nur das reine Diagramm. Mindestens für den Vortrag wird es nötig sein, das Diagramm in mehrere Teile zu zerlegen. Erfahrungsgemäß sind die Diagramme nicht mehr lesbar, wenn sie als Ganzes auf eine Folie kommen. Selbstverständlich wäre die Zerlegung in mehrere Teildiagramme auch für den Bericht sinnvoll.

Studienleistung: Projekt (3)

- Beim ER-Entwurf sollen auch Integritätsbedingungen dokumentiert werden.
- Die Übersetzung ins relationale Modell ist durchgeführen (bis zu den `CREATE TABLE`-Anweisungen), eventuelle Einschränkungen sind zu diskutieren.
- Es sind mindestens einige Beispieldaten zu laden.

  Dafür könnte auch etwas Programmierung nötig sein, wenn Sie eine größere Datensammlung im Internet finden (in irgendeinem Format), und diese als Tabellenzeilen in die Datenbank laden wollen.

Studienleistung: Projekt (4)

- Einige typische Anfragen sollen gezeigt werden.

  Um zu demonstrieren, dass das DB-Schema den Projektanforderungen gerecht wird.

- Es sind Trigger oder serverseitige Prozeduren und ggf. objektrelationale Konstrukte zu verwenden.

  Mindestens hier ist Programmierung unbedingt nötig. Stellen Sie die Einhaltung der Integritätsbedingungen sicher, entweder mit Triggern, oder indem Sie die Änderung der Daten nur über Stored Procedures ("Methoden") erlauben.

Studienleistung: Projekt (5)

- Ein Entwurfswerkzeug muss verwendet werden.

  Falls Sie nicht Oracle SQL Developer Data Modeler verwenden wollen, sollten wir vorher darüber reden. Das Werkzeug muss eine Trennung von ER-Entwurf und relationalem Schema erlauben (kein Entwurf direkt im relationalen Modell, oder mit einer ER-Notation, die schon Fremdschlüssel anzeigt).

- Sie müssen einen zufällig zugeordneten Bericht eines anderen Studenten lesen und kurz kommentieren (Feedback, "Code Review").

  Maximal eine Seite mit Fragen, Verbesserungsvorschlägen, Lob. Danach kann der Bericht nochmals geändert werden. Wir schauen uns die Kritik erst an, nachdem wir über den Bericht selbst entschieden haben. Sie können dem Mitstudenten also nicht schaden.

Studienleistung: Projekt (6)

- Termin für Verteidigung: 20.-22.03.2019 (?)

  Es ist wichtig, dass alle Teilnehmer dabei sind, da man auch aus den Projekten der anderen Teilnehmer und der Diskussion lernt. Erwartet wird ein Vortrag mit ca. 8–15 Folien. Es reichen zwei Tage Anwesenheit, falls aufgrund der Teilnehmerzahl drei Tage notwendig sind.

- Herr Wenzel und ich sind die Auftraggeber, Sie können uns fragen, wie bestimmte Anforderungen gemeint sind (normal: Kunde vergisst Details).

  Falls Sie sich für eine Aufgabe von uns entscheiden. Viele Studenten haben sich aber ein eigenes Projekt vorgenommen. Wenn Sie konkrete Fragen zu Teilproblemen haben, können Sie uns auch als Lehrende fragen. Wir werden aber nicht Fragen der Art beantworten, "Hier ist meine Lösung, ist die richtig?"

Studienleistung: Übung (1)

- In der Übung werden Aufgaben besprochen, die zum Teil vorher als Hausaufgaben ins Netz gestellt wurden, zum Teil reine Präsenzaufgaben sind.

  Sie müssen die Hausaufgaben nicht abgeben. Der Übungsleiter ist durch die Korrektur der Projekte schon stark belastet. Sie müssen sich allerdings mit den Hausaufgaben beschäftigen.

- In der Übung müssen Sie Ihre Lösung vorführen und erklären können (inkl. Fragen zum Umfeld).

  Sowohl vor der ganzen Gruppe, als auch direkt dem Übungsleiter. Ohne Unterstützung von ggf. anderen Gruppenmitgliedern.

Studienleistung: Übung (2)

- Verlangt wird regelmäßige und aktive Teilnahme an der Übung.

  Sie dürfen nicht zu häufig fehlen (drei Mal wären akzeptabel, es gibt einen Ermessensspielraum des Übungsleiters, ggf. Zusatzaufgabe).

- Wenn Sie bei jedem Übungstermin einen Beitrag geleistet haben (und nicht oft passen mussten), sind Sie auf der sicheren Seite.

  Die Studienleistung ist Bedingung für den erfolgreichen Abschluss des Moduls.

- Zum Teil in der Übung auch schon Arbeit am Projekt (am Rechner, mit Entwurfswerkzeug).

Zeitliche Belastung

- Diese Vorlesung hat 5 Leistungspunkte.

  Auch "credit points" genannt.

- Entspricht 150 Stunden studentischer Arbeitszeit:

Lernform	SWS	Stunden
Vorlesung	2	30
Selbststudium	0	30
Seminar, Tafelübung	1	15
Praktische Übung, Projekt	1	15
Projektverteidigung	0	15
Hausaufgaben	0	15
Projekt	0	30

Lehrbücher (1)

- Toby J. Teorey et al.:
  Database Modeling and Design. Logical Design.

  Weitere Autoren: Sam S. Lightstone, Tom Nadeau, H. V. Jagadish. Morgan Kaufmann, 2011, 5th Ed., ISBN 0123820200, 368 pages, 50.24 Euro.

- Sam Lightstone, Toby Teorey, Tom Nadeau:
  Physical Database Design.

  Morgan Kaufmann, 2007, ISBN 0123693896, 448 pages, 43.82.
  → mehr für Datenbanken II B.

- Database Design: Know it All (Know It All)

  Viele Autoren, u.a. Teorey (jeder 1–3 Kapitel).
  Morgan Kaufmann, 2008, ISBN 0123746302, 349 Seiten, 32.27 Euro.

Lehrbücher (2)

- Graeme C. Simsion, Graham C. Witt:
  Data Modeling Essentials, 3rd Edition.

  Morgan Kaufmann, 2004, ISBN 0126445516, 550 pages, 48.45 Euro.

- Graeme Simsion:
  Data Modeling: Theory and Practice.

  Technics Publications, 2007, ISBN 0977140016, 400 pages, 47.99.
  Aus Dissertation entstanden, nicht neuere Auflage obigen Buches.

- Bernhard Thalheim: Entity-Relationship Modeling.
  Foundations of Database Technology.

  Springer Verlag, 2000, ISBN 3540654704, 627 pages, 64.15 Euro.

Lehrbücher (3)

- Antoni Olivé:
  Conceptual Modeling of Information Systems.

  Springer, 2007, ISBN 3540393897, 455 pages, 58.80 Euro.

- Otto Rauh, Eberhard Stickel:
  Konzeptuelle Datenmodellierung.

  Teubner, 1997, ISBN 3-8154-2601-4.

- Carlo Batini, Stefano Ceri, Shamkant B. Navathe:
  Conceptual Database Design.

  Benjamin/Cummings, 1992, ISBN 0-8053-0244-1, 470 pages.

Lehrbücher (4)

- Richard Barker:
  CASE*Method: Entity Relationship Modelling.

  Addison-Wesley, 1990, ISBN 0-201-41696-4, nur noch gebraucht.

- Heli Helskyaho: Oracle SQL Developer Data Modeler for Database Design Mastery.

  McGraw Hill Education / Oracle Press, 2015, ISBN 0071850090, 336 Seiten.

- Peter Koletzke, Paul Dorsey:
  Oracle Designer Handbook, 2nd Edition.

  ORACLE Press, 1998, ISBN 0-07-882417-6, 1075 pages, nur noch gebraucht.

Lehrbücher (5)

- Grady Booch, James Rumbaugh, Ivar Jacobson:
  The Unified Modeling Language User Guide.

  Addison Wesley, 1999, ISBN 0-201-57168-4, 482 pages, 50 Euro.

- Martin Fowler, Kendall Scott:
  UML Distilled, Second Edition.

  Addison-Wesley, 2000, ISBN 0-201-65783-X, 185 pages, 35 Euro.

- Robert J. Muller: Database Design for Smarties — Using UML for Data Modeling.

  Morgan Kaufmann, 1999, ISBN 1-55860-515-0, 442 pages, 47 Euro.

Software: Oracle

- In Übung: Oracle SQL Developer Data Modeler.

  Installieriert im PC-Pool 3.32 und im Thin Client Pool 3.34. Er ist in Java geschrieben, kostenlos erhältlich und läuft auch unter Linux.
  [http://www.oracle.com/technetwork/developer-tools/datamodeler/]

- Der "Oracle Designer" wird nicht mehr weiterentwickelt, verwendet aber die gleiche ER-Notation und ist teils technisch noch interessant.

- Es gibt auch den Oracle JDeveloper als IDE.

  [http://www.oracle.com/technetwork/developer-tools/jdev/]
  Unterstützt u.a. Java, UML, HTML/JSP/JSF, Web Services, SQL, Oracle ADF (Application Development Framework, erlaubt visuelle, deklarative Programmierung von Web-Datenbank Anwendungen).

Software: Weitere (1)

- Sybase Powerdesigner

  [http://powerdesigner.de/]
  Im Pool installiert.

- ERwin

  [http://erwin.com/products/data-modeler]
  Wird im Pool installiert.

- Rational Rose

  [http://www.ibm.com/software/products/en/datamodeler]
  Im Pool installiert (von Softwaretechnik-Gruppe).

- Oracle JDeveloper

  [http://www.oracle.com/technology/products/jdev/index.html]

Software: Weitere (2)

- DB-MAIN

  [http://www.rever.eu/en/content/db-main-homepage]

- ER Studio

  [https://www.idera.com/er-studio-data-architect-software]

- Borland Together

  [https://www.borland.com/en-GB/Products/
  Requirements-Management/Together]

- Microsoft Visio

  [http://www.microsoft.com/office/visio/prodinfo/default.mspx]

Software: Weitere (3)

- Datanamic DeZign

  [http://www.datanamic.com/]

- CASE Studio: Toad Data Modeler

  [http://www.casestudio.com/]

- Dia: A Drawing Program

  [https://wiki.gnome.org/Apps/Dia] [http://dia-installer.de/]

- Schemaspy: Visualisierung relationaler Schemata

  [http://schemaspy.sourceforge.net/]

Software: Weitere (4)

- Visual Paradigm: Database Visual Architect

  [http://www.visual-paradigm.com/product/dbva/]

- IBM Rational Architect (Integration)

  [http://www-306.ibm.com/software/data/integration/rda/]

- Toad Data Modeler

  [http://www.quest.com/Toad-Data-Modeler/]

- Xcase DB Design Software / Data Modeling Tool

  [http://www.xcase.com/]

- Open Modelsphere (data, process, UML modeling)

  [http://www.modelsphere.org/] (open source, früher kommerziell)

Software: Weitere (5)

- DBDesigner 4

  [http://www.fabforce.net/dbdesigner4/] (open source)

- In der Wikipedia gibt es einen Vergleich verschiedener "Data Modeling Tools".

  [http://en.wikipedia.org/wiki/Comparison_of_data_modeling_tools] Kriterien sind u.a.: License (Commercial, Free, etc.), Supported Database Platforms, Supported Operating Systems (Windows, Linux, etc.), Supported Data Models (Conceptual, logical, physical), Supported Notations (IDEF1X, Crows feet, etc.), Forward Engineering, Reverse Engineering, Model/database synchronization, Support of Teamwork.

Software: Bemerkung

- Mit gewissen Einschränkungen kommt es für diese Vorlesung nicht sehr auf das Werkzeug an.

  Mir ist eine klare Trennung zwischen ER-Ebene und relationaler Ebene wichtig, manche Werkzeuge vermischen das. Auch sonst ist der Funktionsumfang der Werkzeuge natürlich unterschiedlich.

- Probieren Sie verschiedene Werkzeuge aus!

  Sie sollen einen Eindruck davon gewinnen, was solche Werkzeuge für Sie tun können.

- Es ist auch ein Lernziel der Vorlesung, dass Sie mit verschiedenen ER-Notationen arbeiten können.

  In Klausur gegebene ER-Schemata voraussichtlich in Oracle Designer Notation ("Barker Notation") oder als UML Klassendiagramm.

Verbesserung der Lehre

- Gute Lehre ist für mich wichtig.
  Ich möchte ein Datenbank-Lehrbuch schreiben.

- Vorschläge zur Verbesserung der Vorlesung sind sehr willkommen. Fragen sind sehr willkommen.

  Für Hinweise auf Fehlern in den Vorlesungs-Materialien, sowie nützliche Links für die WWW-Seite wäre ich besonders dankbar.

- Eine Vorlesung, gerade auch auf diesem Gebiet, lebt ganz wesentlich auch von der Diskussion.

  Es gibt öfters nicht die eine perfekte Lösung. Leider sind auch nicht alle Qualitätskriterien 100% objektiv.