

DB II B: DBMS-Implementierung

Übung 1: Organisatorisches

Prof. Dr. Stefan Brass

Martin-Luther-Universität Halle-Wittenberg

Wintersemester 2021/22

<http://www.informatik.uni-halle.de/~brass/dbi21/>

Inhalt

- 1 Inhalte
- 2 Voraussetzungen
- 3 Organisatorisches
- 4 Literatur
- 5 Software

Themen

Wie funktioniert ein DBMS? (→ Performance Tuning)

- Platten, RAID-Systeme, SANs, Pufferung/Caching
- Speicherverwaltung, Physische Parameter bei der Tabellendeklaration in Oracle
- Indexe: B-Bäume, ggf. Cluster, Hash, Bitmap
- Anfrage-Auswertungspläne, Algorithmen für Operationen der relationalen Algebra
- Anfrage-Optimierung
- Ggf. Backup und Recovery

Motivation

Warum diese Vorlesung hören?

- Neugierde: Wie funktionieren DBMS eigentlich?
- Damit auch besseres Verständnis für Datenbanksysteme, effektivere Anwendung.
- Teil der Vorbereitung für
 - Beruf “Datenbankadministrator” (DBA).
 - Zertifizierung als Oracle DBA.
 - Beruf “Berater für Datenbank-Tuning”.

Echte Experten sollen sehr gut bezahlt werden, weil sie den Firmen das Geld für eine größere Hardware sparen.

Sind Sie in der richtigen Vorlesung?

Allgemein:

- Dies ist eine fortgeschrittene Vorlesung über Datenbanken, insbesondere für den Master-Studiengang „Informatik“.

In diesem Studiengang kann sie als Basismodul der Vertiefungsrichtung „Datenbanken und Informationssysteme“ verwendet werden.

- Falls Sie keine Einführung in Datenbanken gehört haben, fehlen Ihnen wichtige Vorkenntnisse.

<https://users.informatik.uni-halle.de/~brass/db21/>

Für Wirtschaftsinformatiker:

- Diese Vorlesung ist nur für den Wahlbereich II anrechenbar.

Eine Anrechnung für den Wahlbereich I wäre theoretisch über einen Antrag an den Prüfungsausschuss denkbar, aber die Vorsitzende hat gerade deutlich gemacht, dass es zumindest keine schnelle Lösung geben wird (wenn überhaupt).

Inhalt

- 1 Inhalte
- 2 Voraussetzungen**
- 3 Organisatorisches
- 4 Literatur
- 5 Software

Voraussetzungen

Vorausgesetzt werden:

- Relationales Modell
- SQL (komplexere Anfragen, `CREATE TABLE`, `GRANT`)
- Operationen der relationalen Algebra
- Programmierkenntnisse, Datenstrukturen
Für die Studienleistung ist ernsthaft zu programmieren!
- ER-Diagramme (Grundkenntnisse)
- Englische Sprachkenntnisse

Nicht vorausgesetzt: Datenbanken IIA

SQL-Vorkenntnisse (1)

- Können Sie Anfragen in SQL schreiben?
 - A. Sehr gut (z.B. auch **GROUP BY** und **NOT EXISTS**).
 - B. Etwas (**SELECT ... FROM ... WHERE ...**).
 - C. Nein.
- Kennen Sie **CREATE TABLE**?
 - A. Ja.
 - B. Nein.
- Und **INSERT, UPDATE, DELETE**?
 - A. Ja.
 - B. Nein.

SQL-Vorkenntnisse (2)

- Können Sie Zugriffsrechte mit **GRANT** vergeben?
 - A. Ja.
 - B. Nein.
- Können Sie **CREATE INDEX** verwenden, um Anfragen zu beschleunigen?
 - A. Ja.
 - B. Nein.

Beispiel-Datenbank (1)

STUDENTEN

<u>SID</u>	VORNAME	NACHNAME	EMAIL
101	Lisa	Weiss	...
102	Michael	Grau	NULL
103	Daniel	Sommer	...
104	Iris	Winter	...

AUFGABEN

<u>ATYP</u>	<u>ANR</u>	THEMA	MAXPT
H	1	ER	10
H	2	SQL	10
Z	1	SQL	14

BEWERTUNGEN

<u>SID</u>	<u>ATYP</u>	<u>ANR</u>	PUNKTE
101	H	1	10
101	H	2	8
101	Z	1	12
102	H	1	9
102	H	2	9
102	Z	1	10
103	H	1	5
103	Z	1	7

Beispiel-Datenbank (2)

- **STUDENTEN**: enthält eine Zeile für jeden Studenten.
 - **SID**: "Studenten-ID" (eindeutige Nummer).
 - **VORNAME**, **NACHNAME**: Vor- und Nachname.
 - **EMAIL**: Email-Adresse (kann NULL sein).
- **AUFGABEN**: enthält eine Zeile für jede Aufgabe.
 - **ATYP**: Typ/Kategorie der Aufgabe.
 - Z.B. 'H': Hausaufgabe, 'Z': Zwischenklausur, 'E': Endklausur.
 - **ANR**: Aufgabennummer (innerhalb des Typs).
 - **THEMA**: Thema der Aufgabe.
 - **MAXPT**: Maximale/volle Punktzahl der Aufgabe.

Beispiel-Datenbank (3)

- **BEWERTUNGEN**: enthält eine Zeile für jede abgegebene Lösung zu einer Aufgabe.
 - **SID**: Student, der die Lösung abgegeben hat.
Dies referenziert eine Zeile in **STUDENTEN**.
 - **ATYP**, **ANR**: Identifikation der Aufgabe.
Zusammen identifiziert dies eine Zeile in **AUFGABEN**.
 - **PUNKTE**: Punkte, die der Student für die Lösung bekommen hat.
 - Falls es keinen Eintrag für einen Studenten und eine Aufgabe gibt: Aufgabe nicht abgegeben.

Adminer (1)

- Adminer ist eine Web-Schnittstelle zum Zugriff auf verschiedene Datenbanksysteme.
 - Aktuell (Version 4.7.7): MySQL, MariaDB, PostgreSQL, SQLite, MS SQL, Oracle, Firebird, SimpleDB, Elasticsearch and MongoDB.
- [<https://www.adminer.org/de/>]
[<https://en.wikipedia.org/wiki/Adminer>]
- Die wichtigste Funktion ist natürlich, dass man
 - SQL-Anfragen in ein Web-Formular eingeben kann,
 - auf den „Ausführen“-Knopf klicken, und
 - dann die Ergebnis-Tabelle angezeigt bekommt.
- Man kann aber z.B. auch nur durch Anklicken einer Tabelle das Schema und den Inhalt anzeigen lassen (GUI).

Adminer (2)

- Die Zugangsdaten unserer Installation stehen in StudIP, Reiter „Adminer“.
- [https://dbs.informatik.uni-halle.de/edb?pgsql=db&username=student_gast&db=postgres&ns=]
- Man kann Daten des Login-Bildschirms in der URL vorbelegen („ns“ steht für „Namespace“, d.h. Schema).
- Ein PostgreSQL Server kann mehrere Datenbanken verwalten.
Es gibt immer: postgres, template0, template1.
- Eine PostgreSQL-Datenbank enthält mehrere Schemata.
Es gibt immer das Schema public.

Adminer (3)

- Die Beispiel-DB steht im Schema „**studentenaufgaben_public**“.
Sie können das in der Auswahl-Box links relativ weit oben wählen.
Sie bekommen dann eine Liste/Tabelle mit den drei Tabellen „aufgaben“, „bewertungen“, „studenten“ angezeigt (unter der Überschrift „Tabellen und Views“). Es gibt noch weitere Überschriften für andere Arten von Datenbank-Objekten, davon gibt es aber keine in diesem Schema.
- Wenn Sie auf einen Tabellennamen klicken, bekommen Sie die Schema-Information zur Tabelle.
- Wenn Sie auf „zeigen“ links neben dem Tabellennamen klicken, werden die Tabellendaten (der Zustand) angezeigt.
- Wenn Sie auf „SQL-Kommando“ im Menu links klicken, bekommen Sie eine Dialogbox für SQL-Befehle.

SQL-Aufgabe: Testen Sie Ihr Wissen!

- Geben Sie Name und Vorname aller Studierenden aus, die die maximale Punktzahl in Hausaufgabe 1 haben.

„Hausaufgabe“ bedeutet: ATYP ist 'H'.

- Datenbank-Schema in Kurznotation:

- STUDENTEN(SID, VORNAME, NACHNAME, EMAIL^o)
- AUFGABEN(ATYP, ANR, THEMA, MAXPT)
- BEWERTUNGEN(SID→STUDENTEN,
(ATYP, ANR)→AUFGABEN, PUNKTE)

- Zur Kontrolle:

Es muss Lisa Weiss herauskommen.

Programmier-Fähigkeiten (1)

- Können Sie in einer üblichen Programmiersprache, wie etwa Java oder C++, programmieren?
 - A. Ja, gut (Erfahrung mit größeren Programmen)
 - B. Ja, etwas (soviel, wie man in einem Semester lernt)
 - C. Ein wenig
 - D. Nein.
- Können Sie in Java programmieren?
 - A. Ja.
 - B. Etwas.
 - C. Nein.

Programmier-Fähigkeiten (2)

- Können Sie in C++ programmieren?
 - A. Ja.
 - B. Etwas.
 - C. Nein.
- C++ ist etwas hardware-näher und könnte zu besserem Verständnis der Datenstrukturen (auf Byte-Ebene) und der Speicherverwaltung führen.
- Möchten Sie C++ lernen und in Übungen anwenden?
 - A. Ja.
 - B. Ein wenig.
 - C. Nein.

Simpler Test für Programmier-Fähigkeiten

- Schreiben Sie ein Programm (in Java oder einer anderen Sprache), das folgenden Text ausgibt:

```
INSERT INTO R VALUES(1, 'abc');  
INSERT INTO R VALUES(2, 'abc');  
INSERT INTO R VALUES(3, 'abc');
```

- Die Anzahl der Zeilen soll natürlich leicht zu modifizieren sein.

Für einen Performance-Test würde man eine große Datei benötigen.

- Dies ist eine ganz simple Übung, in dieser Vorlesung wird wesentlich mehr vorausgesetzt!

Die Aufgaben brauchen dann aber auch mehr Zeit.

Inhalt

- 1 Inhalte
- 2 Voraussetzungen
- 3 Organisatorisches**
- 4 Literatur
- 5 Software

Zeit und Ort (1)

Vorlesung:

- Donnerstags, 15³⁰–17⁰⁰, Raum 3.04
- Es gibt eine Videoaufzeichnung der Vorlesung.
 - Steht erst nach einigen Stunden in **StudIP**, Reiter „OpenCast“, zur Verfügung.
 - Wenn Sie zu spät kommen, sind Sie vermutlich auf dem Video zu sehen.
 - Wenn Sie Fragen stellen, sind Sie fast nicht zu hören. Stellen Sie Fragen!
- Möchten Sie an der Vorlesung in Präsenz teilnehmen?
 - A. Ja.
 - B. Gelegentlich.
 - C. Nein.

Der Hörsaal hat 93 Plätze (zu Corona-Zeiten 18). 3G-Regel (Sie brauchen einen Nachweis, Formular für Selbsttest). Löwen-App zur Registrierung.

Zeit und Ort (2)

Übung (beginnt 14.10.2021):

- Nur eine Gruppe: Donnerstags, 10³⁰–12⁰⁰.

Die Übung ist vor der Vorlesung, damit Präsenz-Teilnehmer und solche, die das Video schauen, auf dem gleichen Stand sind.

- Online als Videokonferenz (ohne Aufzeichnung).

Im [StudIP-Eintrag der Übung](#), Reiter „Meetings“.

- Ich werde mich bemühen, das Wichtigste auf Folien zu schreiben, die auf der Webseite veröffentlicht werden.

[\[https://users.informatik.uni-halle.de/~brass/dbi21/uebung.html\]](https://users.informatik.uni-halle.de/~brass/dbi21/uebung.html)

- Wenn Sie mehr als drei Mal fehlen müssen, reden Sie mit dem Dozenten.

Möglicherweise gibt es Extra-Aufgaben. Studierende, die ohnehin grenzwertige Vorkenntnisse haben, sollten möglichst immer dabei sein.

Wichtig!

- Webseite (Folien, Links, aktuelle Ankündigungen):
<https://users.informatik.uni-halle.de/~brass/dbi21/index.html>
- **Modul-Anmeldung nicht vergessen!**

Das gilt ganz besonders für Wirtschaftsinformatiker. Die Modulanmeldung ist nur in den ersten zwei Wochen der Vorlesungszeit möglich. Ohne Modulanmeldung kann man sich nicht zur Prüfung anmelden. Wenn man nicht zur Prüfung angemeldet ist, darf man nicht geprüft werden. Typischerweise fallen die Probleme erst am Ende des Semesters auf, wenn man schon viel Zeit in die Vorlesung investiert hat. Die Dopplung zwischen StudIP und Löwenportal ist unglücklich, aber für die Modulanmeldung muss man sich im Löwenportal eintragen.

Ansprechpartner (1)

Dozent: Prof. Dr. Stefan Brass

- Büro: Von-Seckendorff-Platz 1, Raum 313
- Telefon: 0345/55-24740
- Email: brass@informatik.uni-halle.de
- Frühere Unis: Braunschweig, Dortmund, Hannover, Hildesheim, Pittsburgh, Gießen, Clausthal.
- Oracle8 Certified Database Administrator
IBM Certified Advanced DBA (DB2 UDB 8.1).

Die Zertifizierungen sind total veraltet, aber sind doch ein für einen Professor relativ ungewöhnliches Zeichen detaillierter systemspezifischer Kenntnisse.

Aktuell sind Oracle 19C (und schon 21C), (es gab einen Sprung von 12 nach 18) und DB2 Version 11.5.5 (oder 12).

Ansprechpartner (2)

Erreichbarkeit bei Fragen:

- Hier in der Übung!
- Forum in **StudIP** (Eintrag der Vorlesung)
- Sprechstunde: Montags, 12¹⁵–13⁰⁰
Raum 313, Tel. 0345/55-24740
- Online-Sprechstunde: Montags, 16³⁰–16⁴⁵ (und nach Bedarf)
[\[https://mluconf.uni-halle.de/b/ste-p6j-fmd-i21\]](https://mluconf.uni-halle.de/b/ste-p6j-fmd-i21)
- EMail: brass@informatik.uni-halle.de

Bitte nur, wenn es nur Sie persönlich betrifft. Bei allgemeinen Fragen zu Vorlesung und Hausaufgaben wäre das Forum besser. Verwenden Sie eine aussagekräftige Betreffzeile, möglichst mit „[dbi21]“. Wenn Sie nach drei Tagen keine Antwort haben, schicken Sie die EMail nochmal.

Ansprechpartner (3)

Sekretärin: Ramona Vahrenhold

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

Weitere Mitglieder der DB-Gruppe:

- PD Dr. Alexander Hinneburg,
Raum 314, Tel. 55-24732, hinnebur@...
- MSc Mario Wenzel,
Raum 315, Tel. 55-24776, mario.wenzel@...

Studienleistung (1)

- Studienleistung: 67% der Hausaufgabenpunkte
 - Die Hausaufgaben sind einzeln zu bearbeiten.
 - Die Hausaufgaben werden nur stichprobenartig korrigiert, aber in der Übung besprochen.

Wenn Ihre Hausaufgabe nicht korrigiert wurde, bekommen Sie vorläufig die volle Punktzahl. Es ist möglich, dass Hausaufgaben noch nachträglich korrigiert werden, und die Punktzahl dann gesenkt wird. Damit müssen Sie insbesondere dann rechnen, wenn Sie wegen eines Plagiats auffallen, oder wegen der Abgabe einer Lösung, die offensichtlich kein sinnvoller Lösungsversuch ist.
 - Wenn Sie sich in der Übung freiwillig melden, vorzurechnen, werde ich nur in extremen Fällen Punkte abziehen.

Sollte ich Ihre Hausaufgabe zufällig später für die Stichprobe auswählen und dann doch Punkte abziehen, können Sie mich daran erinnern, dass Sie freiwillig vorgerechnet haben.

Studienleistung (2)

- Falls Plagiate bei Hausaufgaben oder sehr still bei Übungen: Änderung (Verschärfung) der Regeln möglich.

Modulbeschreibung: „Aktive Mitarbeit in den Übungen inkl. Kurzvorträge über Hausaufgaben und Beantwortung von Fragen zum Umfeld der Aufgaben“.

- Helfen Sie Ihren Mitstudierenden, aber geben Sie nicht Ihre Lösung zum Abschreiben.

Minimal 0 Punkte für alle Beteiligten bei „zu ähnlichen“ Lösungen.
Im Wiederholungsfall keine Studienleistung in diesem Jahr!

- Es ist möglich, dass es in den Übungen Präsenzaufgaben gibt, die auch Hausaufgabenpunkte liefern.

Diese Aufgaben können nur bei Übungsteilnahme abgegeben werden. Sie machen aber nur einen kleinen Teil der Punkte aus (maximal 15%).

Klausur (1)

- Voraussichtlich Papier-Klausur in Präsenz am 31.03.2022.
- Voraussichtlich mündliche Prüfungen um den 29.09.2022.
- **Die Termine der Klausuren sind noch nicht endgültig!
Melden Sie Überschneidungen frühzeitig!**
- Bei der Klausur sind drei DIN A4 Blätter mit Notizen erlaubt.
Sie können die Vorder- und Rückseite bedrucken oder beschreiben oder 6 Blätter einseitig mitbringen. Es ist eine gute Klausurvorbereitung, sich selbst das Wichtigste auf begrenztem Platz in eigener Strukturierung aufzuschreiben (inkl. „SQL Quick Reference“). Sie können sich natürlich auch etwas aus dem Netz drucken (wenn Sie sich damit nicht vorher beschäftigen, wird es Ihnen in der Klausur nichts nutzen, nur Zeit kosten). Irgendwie exotischere Tabellen aus dem Oracle Data Dictionary werden in der Klausur genannt (möglicherweise aber zu viele, auch solche, die Sie nicht brauchen).

Klausur (2)

- Es ist nicht garantiert, dass man mit 50% der Punkte in der Klausur bestanden hat.

Garantiert ist das Bestehen mit 60% (die Grenze wird häufig bis in die Nähe der üblichen 50% gesenkt, aber durch die elektronische Klausur brauchen wir auch etwas Flexibilität).

Ab 95% der Punkte ist die 1.0 garantiert, aber auch diese Grenze wird häufig noch etwas gesenkt bis max. ca. 90%.

- Man kann die Klausur nur mitschreiben, wenn man angemeldet ist. Dazu ist die Modulanmeldung nötig.
- Man bekommt die Leistungspunkte für dieses Modul nur, wenn man auch die Studienleistung hat (Hausaufgaben, s.u.).

Man kann die Klausur ohne Studienleistung mitschreiben, aber das macht wenig Sinn: Die Hausaufgaben sollen ja auf die Klausur vorbereiten.

Lehrbücher (1)

DB-Lehrbücher mit Teil über DBMS-Implementierung:

- Alfons Kemper / André Eickler:
Datenbanksysteme. Eine Einführung. 10. Auflage.
De Gruyter, 2015, ISBN 3110443759, 880 Seiten, 49.95 €.
- Ramez Elmasri / Shamkant B. Navathe:
Fundamentals of Database Systems, 7th Ed.
Pearson India, 2017, ISBN 933258270X, ca. 29.00 €
Global Ed.: Prentice Hall, 2016, ISBN 1-292-09761-2, 1272 S., 39.16 €
Für diese Vorlesung interessanten Teile fehlen in deutscher Ausgabe.
- Raghu Ramakrishnan, Johannes Gehrke:
Database Management Systems. 3rd Edition.
McGraw-Hill, 2002, ISBN: 0071151109, 1065 pages.

Lehrbücher (2)

Lehrbücher speziell über DBMS-Implementierung:

- Gunter Saake, Andreas Heuer, Kai-Uwe Sattler:
Datenbanken: Implementierungstechniken, 4. Aufl.

MITP-Verlag, 2019, ISBN 3958457797, 736 Seiten.

- Hector Garcia-Molina, J. D. Ullman, J. Widom:
Database Systems: The Complete Book. 2nd Ed.

Prentice Hall, 2008, ISBN: 0131354280, 1248 pages.

Nur der zweite Teil ist über DBMS-Implementierung. Früher waren es zwei Bände. Der zweite heißt "Database System Implementation". Eventuell noch erhältlich, aber man vergleiche Preis.

Lehrbücher (3)

Lehrbücher über DBMS-Implementierung, Forts.:

- Theo Härder, Erhard Rahm: Datenbanksysteme: Konzepte und Techniken der Implementierung.
2. Aufl., Springer, 2001, ISBN 3-540-42133-5, 582 pages, 40 Euro.
- Jim Gray, Andreas Reuter: Transaction Processing: Concepts and Techniques.
Morgan Kaufmann, 1993, ISBN 1-55860-190-2, 1070 pages, 96 Euro.
- Markus Winand: SQL Performance Explained (Deutsche Ausgabe).
Markus Winand, Wien, 2012, ISBN 978-3-9503078-1-8, 204 Seiten.

Lehrbücher (4)

Oracle:

- Mark Gurry, Peter Corrigan:
Oracle Performance Tuning, 2nd Edition.
- John Watson, Roopesh Ramklass, Bob Bryla:
OCA/OCP Oracle Database 11g All-in-One Exam Guide:
Exams 1Z0-051, 1Z0-052, 1Z0-053.

O'Reilly, 1996, ISBN 1-56592-237-9, 942 pages, 50 Euro.

McGraw-Hill/Oracle, 2009, ISBN: 0071629181, 1050 pages, CD.

- Oracle Documentation ist online verfügbar

[<https://docs.oracle.com/en/database/oracle/oracle-database/19/books.html>]

Empfehlenswert z.B. Oracle Concepts Manual.

Lehrbücher (5)

IBM DB2:

- Raul Chong, Xiaomei Wang, Michael Dang, Dwaine R. Snow: Understanding DB2: Learning Visually with Examples, 2nd Ed.

IBM Press, 2007, ISBN: 0131580183, 1056 Seiten, 50.99 Euro.

- DB2 Handbücher sind online:

[<https://www.ibm.com/support/pages/db2-version-111-linux-unix-and-windows-english-manuals>]

[https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.welcome.doc/doc/welcome.html]

- Tutorials zur DB2 Zertifizierung sind online:

[<https://www.ibm.com/developerworks/data/tutorials/db2-cert6101/index.html>]

Software in dieser Vorlesung

- In der Übung: Oracle (mit DBA-Rechten).

Der Dozent ist Oracle Certified Professional (Oracle 8 DBA). Diese Vorlesung ist daher etwas Oracle-lastig. Ich begrüße es aber ausdrücklich, wenn Sie andere Systeme ausprobieren und mir über Gemeinsamkeiten und Unterschiede berichten. Letztendlich ist diese Vorlesung kein Oracle-Kurs, sondern wir wollen die allgemeinen Prinzipien und Probleme studieren. Da DBMS-Internas aber notwendigerweise sehr systemspezifisch sind, und wir das zumindest exemplarisch konkret machen wollen, tun wir das anhand von Oracle. Es ist das System, mit dem ich mich am besten auskenne.

- Geplant ist, auch PostgreSQL auszuprobieren, und (gemeinsam) etwas über die Implementierung zu lernen.

Es ist ja interessant, die Lösungen für bestimmte Probleme in beiden Systemen zu vergleichen.

Software: Oracle

- Verbreitetes kommerzielles System. Vermutlich Marktführer.

Ich persönlich habe mit Oracle gearbeitet seit meinem Studium.

Es ist das System, was ich am besten kenne. Recht gut standard-kompatibel.

Oracle war die erste kommerzielle SQL-Datenbank überhaupt (1979).

- Die “Express Edition” kostet nichts:

[<https://www.oracle.com/database/technologies/appdev/xe.html>]

Gibt es für Windows und Linux. Bei Linux funktioniert jedenfalls die Installation unter CentOS problemlos (ähnlich zu Red Hat Linux).

Maximal 4 GB/11 GB Datenbank, 1 Kern, 1 GB RAM, u.a. Einschränkungen.

- Oracle ist auch Anbieter von betriebswirtschaftlicher Standard-Software (Konkurrent zu SAP).

Durch viele Zukäufe, u.a. Siebel Systems (CRM), PeopleSoft, JD Edwards.

Die Datenbank war aber die Basis der Firma. Jetzt auch Cloud Anbieter.

Software: IBM DB2

- Weiteres wichtiges kommerzielles DBMS.

Edgar F. Codd, der Erfinder des relationalen Datenbankmodells, arbeitete im IBM Forschungslabor in San Jose, als er diese Ideen entwickelt hat. Sein Artikel "A Relational Model of Data for Large Shared Data Banks" erschien 1970 (in den Communications der ACM). 1981 bekam er den Turing Preis. SEQUEL, eine frühere Version von SQL, wurde von Chamberlin, Boyce et al. 1974 im gleichen Forschungslabor entwickelt. System/R, einer der beiden ersten Forschungs-Prototypen einer relationalen Datenbank, wurde dort 1976/77 entwickelt. Der andere war Ingres von Michael Stonebraker (Berkeley).

- Kostenlose Express-C Version (läuft aus?)

[<https://www-01.ibm.com/marketing/iwm/iwm/web/pickUrxNew.do?source=swg-db2expressc>]

Für Entwickler gibt es diese neueren Versionen, die zum Test kostenlos sind (?):

[<https://www.ibm.com/us-en/marketplace/ibm-db2-direct-and-developer-editions/purchase>]

PostgreSQL (1)

- POSTGRES wurde von Michael Stonebraker (+ Team) an der University of California at Berkeley ab 1985 entwickelt.
Michael Stonebraker hatte zuvor das INGRES-System entwickelt (ebenfalls in Berkeley), eines der beiden ersten relationalen Datenbanksysteme (das andere war System R, Forschungsprototyp von IBM). Michael Stonebraker bekam 2014 den Turing Award, eine Art Nobelpreis für Informatik [<https://amturing.acm.org/byyear.cfm>].
- POSTGRES war Vorreiter der objektrelationalen Datenbanken.
Michael Stonebraker gründete zusammen mit anderen die Firma Illustra Information Technologies, die eine kommerzielle Version von POSTGRES herausbrachte. Sie wurde 1997 an Informix verkauft (Informix 2001 von IBM).
- 1994 entwickelten Andrew Yu und Jolly Chen (Studenten in Berkeley) eine Version mit SQL-Unterstützung (statt der Datenbank-Sprache QUEL/POSTQUEL).

PostgreSQL (2)

- Im Ranking von DB-Engines.com belegt PostgreSQL den vierten Platz (hinter Oracle, MySQL, Microsoft SQL Server).
[\[https://db-engines.com/de/ranking\]](https://db-engines.com/de/ranking)
Siehe auch [\[https://www.datanyze.com/market-share/databases\]](https://www.datanyze.com/market-share/databases).
- Folgende Plattformen werden mit fertigen Installations-Paketen unterstützt [\[https://www.postgresql.org/download/\]](https://www.postgresql.org/download/):
 - Linux (Red Hat/CentOS/Fedora, Debian, Ubuntu, SuSE)
 - BSD (FreeBSD, OpenBSD)
 - Windows
 - macOS
 - Solaris
- Der Quellcode ist auch frei verfügbar.

Open Source DBMS (1)

- Es gibt inzwischen recht viele Open Source DBMS, bei denen man also den Quellcode einsehen kann:

- PostgreSQL

[<https://www.postgresql.org/ftp/source/>]

[https://wiki.postgresql.org/wiki/Developer_FAQ]

[<http://www.postgresql.org/developer/coding/>]

[<http://doxygen.postgresql.org/index.html>]

- MySQL (gehört seit 2010 Oracle)

[<https://dev.mysql.com/doc/internals/en/>]

- MariaDB (Abspaltung von MySQL)

[<https://mariadb.com/kb/en/mariadb/>

[source-getting-the-mariadb-source-code/](https://mariadb.com/kb/en/mariadb-internal-source-getting-the-mariadb-source-code/)]

[<https://mariadb.com/kb/en/library/mariadb-internals-documentation/>]

Open Source DBMS (2)

- Weitere Open Source DBMS:

- SQLite

- [<https://www.sqlite.org/src/doc/trunk/README.md>]

- HSQLDB

- [<http://hsqldb.org/web/hsqldbDevelopment.html>]

- H2 Database

- [<https://www.h2database.com/html/main.html>]

- Firebird

- [<http://www.firebirdsql.org/en/code-repository/>]

- Apache Derby

- [<https://db.apache.org/derby/>]

- Vormals Cloudscape, dann IBM Cloudscape, IBM DB2 SQL Syntax.

- War auch mal als “Java DB” im JDK enthalten.