

Projektierung von relationalen Datenbanksystemen

1. Anforderungsanalyse und -spezifikation

Prinzipiell identisch mit den allgemeinen Zielstellungen dieser Phase(n) des Software-Entwicklungsprozesses.

Ergebnis soll der bestätigte (z.B. im Pflichtenheft) Katalog aller Leistungsanforderungen an das Datenmodell bzw. die Datenbank-Applikation sein. Generell gliedert sich die Gesamtheit der **Anforderungen an Softwareprodukte** in:

- funktionale Anforderungen
- Qualitätsanforderungen
- Anforderungen an die Realisierung
- Anforderungen an die Einführung
- Anforderungen an die Projektentwicklung

Spezifische Fragestellungen für die Entwicklung von Datenmodellen bzw. Datenbank-Applikationen leiten sich daraus in folgenden Richtungen ab:

1. Wird das Datenmodell für eine konkrete Applikation, als Grundlage einer Datenbank-Applikation, für die Standard-Software-Einführung oder für das Re-Engineering entwickelt?
2. Welchen Komplexitätsgrad und welche Größenordnung werden die Anzahl der nach dem Datenmodell strukturierten Daten erreichen?
(Abschätzung der Datenelement-Typen, der Anzahl der Daten und der Anzahl der Domänen je Datenelement-Typ, Abschätzung der Anzahl paralleler Nutzer, Abschätzung der Anzahl externer Sichten)?
3. Wie wird die Stabilität der Datenbank-Applikation(en) eingeschätzt (stabile Dauerlösung, Befriedigung dynamischer Informationsbedürfnisse, vorwiegend stabiles Auswertungsprofil, vorwiegend benutzerindividuelle Abfragen, Historisierung von Daten)?
4. Welche Zeiteinflüsse sind für die zu speichernden Daten relevant
(Aktualisierungsfrequenz, Archivierungsrhythmus)?
5. Wird eine verteilte oder zentralisierte Datenverarbeitung notwendig / angestrebt?
6. Welche Klassifizierungen und Generalisierungen von Daten sind erforderlich um sinnvolle Aggregatinformationen zu gewinnen (z.B. Bildung von Summen und Durchschnittswerten für Management-Informationssysteme)?
7. Welche Systemschnittstellen für die Datenkommunikation sind zu bedienen?
8. Existieren gleich strukturierte Datenbestände für unterschiedliche Verantwortungsbereiche (Mandantenproblematik)?

2. Konzeptueller Entwurf

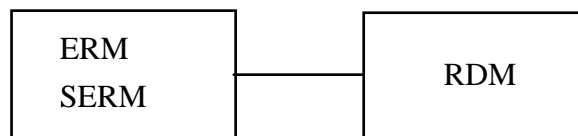
Phasen der ERM-Entwicklung

1. Bestimmung von Objekten und Beziehungen zwischen diesen Objekten
2. Definition von Entities und Relationships, Festlegen der Attribute und des Primärschlüssels
3. Strukturierung der Entities und Relationships zunächst unabhängig von späteren Zugriffspfad
4. Berücksichtigung der späteren Zugriffspfade

3. Logischer Entwurf

Zielstellung des logischen Entwurfs ist die Überführung des konzeptionellen Datenmodells auf ein datenbankunabhängiges formales Datenmodell.

Dem gegenwärtigen Anwendungsstand und Technologieniveau entsprechend ist die Beziehung



favorisiert.

1. Entwurf der Relationen

Regeln für die Ableitung von Relationen in 3NF aus dem ERM bzw. SERM

2. Transaktionsentwurf

Definition der Transaktion: Konsistenzerhaltende Operation auf eine Datenbank, die bei gegebener Konsistenz zu Beginn einer Operationsfolge die Konsistenz am Ende der Operationsfolge sichert.

Konsistenzerhaltende Maßnahmen sind bereits in hohem Maße in die DB-Technologie implementiert (Z.B. Satzsperrern).

Ziel des Transaktionsentwurfs in der Phase der logischen Datenmodellierung ist die Definition von Operationsfolgen, die die semantische Beziehung zwischen den Relationen einer Datenbank beinhaltet, da diese dem formalen DB-Modell nicht bekannt sind.

3. Externe Sichten

Externe Sichten (Views) werden in der Literatur häufig nur als **Nutzersicht** dargestellt. Sie erfüllen im DB-Anwendungs-Entwicklungsprozeß jedoch 3 Funktionen:

- Nichtnormalisierte Relation zur besseren Anpassung an Nutzerforderungen (Wiederherstellung der Redundanz),
- Unterstützung des Zugriffsschutzes,
- Projektbaustein.

4. Implementierungsentwurf

Zielstellung des Implementierungsentwurf ist die weitere Konkretisierung des logischen Datenmodells mit den Schwerpunkten

- der Bestimmung der Datentypen und zugehörigen Wertebereiche (Domänen).
- der Integritätsbedingungen und
- der Zugriff auf die Daten.

In dieser Phase der Datenmodellierung sind formulierte **Anforderungen** an die Datenbanktechnologie **an konkrete DBMS-Leistungsparameter anzugleichen**.

Ergebnis des Implementierungsentwurfes ist die konkrete Beschreibung der Datenstrukturen, der auf die Datenstrukturen definierten Integritätsregeln und Zugriffsrechte mit den geeignetsten Formulierungsmitteln des für die Implementierung gewählten DBMS.

Fernerhin sind geeignete Tools für die externen Sichten (hier im erweiterten Sinne nicht nur Views) auszuarbeiten.

5. Physischer Entwurf

Zielstellung des physischen Entwurfs ist die Optimierung der speicher- und/oder laufzeitspezifischen Eigenschaften des DB-Anwendungssystems.

Aufgabenprofil dieser Phase sehr stark produktabhängig. Wesentlichste Problemgebiete für relationale DBMS sind:

- Bestimmung der Indexdateien,
- Suchstrategiefestlegung in komplexen SQL-Anweisungsfolgen,
- Portionierung normalisierter Relationen,
- Abschätzung von Systemparametern (Z.B. Clustergrößen für Backend-Datenverwaltung).

6. Wartungsprobleme

Gegenstand ist Wartung des Datenmodells und der DB-Anwendung, nicht des Dateninhaltes.

Ziel der Wartung ist die dynamische Systemanpassung aus Gründen

- der dynamischen Veränderung des Datenmodells,
- des Erkenntniszuwachses über den Gebrauch des Datenmodells,
- der Fehlerbeseitigung.

Methodik vergleichbar mit DB-Anwendungsentwicklung generell, jedoch wesentlich stärker auf Endphasen konzentriert und mit eigenen Tools unterstützt.

Grundsatz: Wartung beim Entwurf planen!!!

Wartung von UDM:

Ziel Anpassung an Unternehmensdynamik.

Mittel: Data-Dictionary

Organisation: Permanente Organisationseinheit.

Wartung von DB-Anwendungen

- Schwerpunkte:
- Strukturwartung,
 - Zugriffsoptimierung,
 - Sichtenwartung,
 - Domänenwartung