

Veri Tabanı Sistemleri

Giriş

H. Turgut Uyar Şule Öğüdücü

2002-2012

1 / 31

License



©2002-2012 T. Uyar, Ş. Öğüdücü

You are free:

- ▶ to Share – to copy, distribute and transmit the work
- ▶ to Remix – to adapt the work

Under the following conditions:

- ▶ Attribution – You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- ▶ Noncommercial – You may not use this work for commercial purposes.
- ▶ Share Alike – If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

Legal code (the full license):

<http://creativecommons.org/licenses/by-nc-sa/3.0/>

2 / 31

Konular

Veri İşleme

Giriş
Kayıt Dosyaları

Veri Tabanı Yönetim Sistemleri

Giriş
İstemci - Sunucu
SQL

3 / 31

Veri İşleme

- ▶ büyük miktarda verinin etkin biçimde tutulması ve işlenmesi
- ▶ temel işlemler:
 - ▶ yeni veri ekleme
 - ▶ olan verilerde değişiklik
 - ▶ veri silme
 - ▶ sorgulama: planlı - plansız
- ▶ **CRUD**: create - read - update - delete

4 / 31

Veri Türleri

- ▶ *kalıcı veriler*: tutulacak bilginin doğası gereği bulunması zorunlu olan veriler
- ▶ *geçici veriler*
 - ▶ *çıkış verileri*: kalıcı verilerden türetilen veriler (sorgu sonuçları, raporlar v.b.)
 - ▶ *giriş verileri*: sisteme yeni giren, henüz işlenmemiş veriler
 - ▶ kalıcı verilere eklenebilir
 - ▶ kalıcı verilerde değişikliklere yol açabilir
 - ▶ hiç kullanılmayabilir

5 / 31

Roller

- ▶ *son kullanıcılar*: veri üzerinde işlem yapanlar
 - ▶ teknik konularda bilgileri olmadığı varsayılır
- ▶ *uygulama programcıları*: son kullanıcıların kullandıkları programları yazarlar

6 / 31

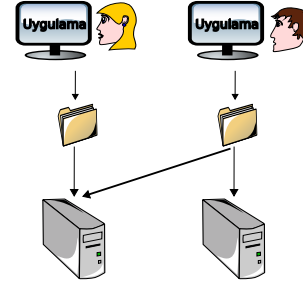
Uygulama Örneği

Örnek (öğrenci verileri)

- Öğrenci İşleri:
öğrencinin adı, numarası,
bölümü, aldığı dersler,
stajları v.b.
- Kitaplık:
öğrencinin adı, numarası,
bölümü, aldığı kitaplar v.b.
- ortak veriler:
öğrencinin adı, numarası,
bölümü v.b.
- uygulamaya özel veriler:
öğrencinin aldığı dersler,
stajlar, kitaplar v.b.

7 / 31

Kayıt Dosyaları



- her uygulamanın kendi verileri var
- her uygulama verilerini
kendi yönettiği dosyalarda tutuyor

8 / 31

Tekrarlılık

- aynı veri birden fazla yerde tutuluyor
 - disk alanı israfı

Örnek

- öğrenci adı, numarası ve bölümü Öğrenci İşleri'nde ayrı,
Kitaplık'ta ayrı tutuluyor

9 / 31

Tutarsızlık

- birden fazla yerde tutulan veriler farklılık gösterebilir

Örnek

- aynı öğrencinin adı Öğrenci İşleri'nde "Victoria Adams",
Kitaplık'ta "Victoria Beckham" görünebilir

10 / 31

Bütünlük Bozulması

- bilginin doğruluğunu sağlamak zordur

Örnek

- "Kontrol ve Bilgisayar Mühendisliği" bölümü kapatılır
ama öğrencilerinin bölüm verisi eskisi gibi kalır

11 / 31

Yeni Uygulamalarda Zorluklar

- her yeni uygulama için benzer işlerin yeniden yapılması gerekir

Örnek

- Burs İşleri için de uygulama yazılacak

12 / 31

Politika Boşlukları

- ▶ kurum uygulamalarında standart eksikliği
 - ▶ yaklaşım, yöntem, programlama dili farklılıkları
 - ▶ uygulamalar arasında veri alışverişi
- ▶ her birim yalnızca kendi gereksinimlerine göre karar verir

13 / 31

Güvenlik

- ▶ ayrıntılı güvenlik izinleri tanımlamak zor
- ▶ güvenlik yalnızca işletim sistemine bağlı

14 / 31

Veriye Bağımlılık

Tanım

veriye bağımlılık:

uygulama kodunun veri düzeni ve erişim yöntemine bağımlı olması

- ▶ uygulamada değişiklik yapmak çok zor

15 / 31

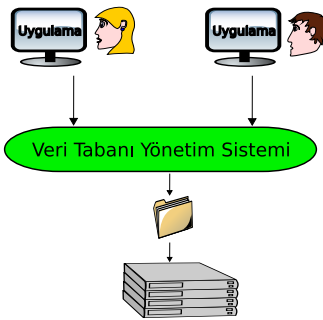
Veriye Bağımlılık

Örnek

- ▶ öğrenci numarası Öğrenci İşleri'nde katar, Kitaplık'ta sayı
- ▶ Öğrenci İşleri kayıtlarında öğrenci numarası için B-ağacı dizin tutuluyor
 - ▶ arama yapılırken B-ağacı algoritmaları kullanılıyor

16 / 31

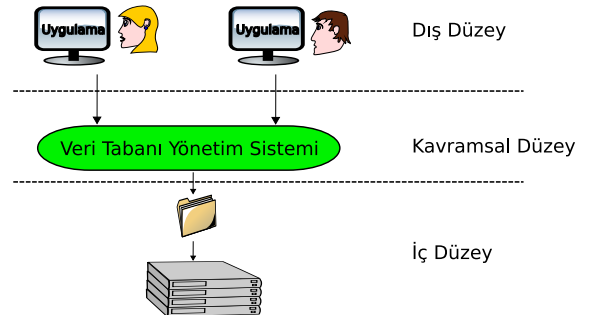
Veri Tabanı Yönetim Sistemleri



- ▶ veriler ortak bir sistemde tutuluyor
- ▶ uygulamalar verilere ortak bir arayüz üzerinden erişiyor

17 / 31

ANSI/SPARC Mimarisi



18 / 31

Dış Düzey

- ▶ son kullanıcı açısından dış düzey:
 - ▶ verinin kendine gereken altkütmesi
 - ▶ kullandığı uygulama programının arayüzü
- ▶ uygulama programcısı açısından dış düzey:
 - ▶ kullandığı programlama dili
 - ▶ bu dile veri tabanı işlemleri için yapılan ekler:
veri altdili

19 / 31

Kavramsal Düzey

- ▶ kavramsal düzey: verinin bütünü
- ▶ veriden bağımsızlığın sağlandığı düzey
- ▶ **katalog:**
verinin içeriğini betimleyen tanımlar
 - ▶ veri tabanları
 - ▶ veri tipleri, bütünlük kısıtlamaları
 - ▶ kullanıcılar, yetkiler, güvenlik kısıtlamaları

20 / 31

İç Düzey

- ▶ iç düzey: gerçekleştirme ayrıntıları
- ▶ verinin nasıl temsil edildiği:
 - ▶ dosyalar, kayıtlar
- ▶ veriye nasıl erişileceği
 - ▶ işaretçiler, dizinler, B-ağaçları

21 / 31

Dönüşümler

- ▶ veri bağımsızlığı için düzeyler arasında dönüşümler

Örnek (kavramsal - dış)

- ▶ öğrenci numarasını
Öğrenci İşleri uygulamasına katar,
Kitaplık uygulamasına sayı olarak sun

Örnek (kavramsal - iç)

- ▶ öğrenci numarası için dizin oluştur

22 / 31

Yönetici Roller

- ▶ **veri yöneticisi:** kararları verir
 - ▶ hangi veriler tutulacak?
 - ▶ hangi veriye kim erişebilir?
- ▶ **veri tabanı yöneticisi:** kararları uygular
 - ▶ kavramsal - dış/iç düzey dönüşümlerini tanımlar
 - ▶ sistem başarımını ayarlar
 - ▶ sistemin sürekliliğini sağlar

23 / 31

VTYS İşlevleri

- ▶ veri tanımlama dili
- ▶ veri işleme dili
- ▶ veri işleme isteklerinin bütünlük ve güvenlik açısından değerlendirilmesi
- ▶ eşzamanlı isteklerin uygun biçimde yürütülmeleri
- ▶ başarımlar

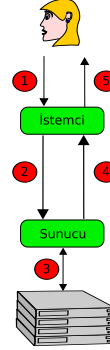
24 / 31

İstemci - Sunucu Yapısı

- **sunucu:**
VTYS işlevlerini yerine getirir
- **istemci:**
kullanıcı ile sunucu arasında etkileşimi sağlar
 - hazır paketler (sorgu dili işleyiciler, rapor üreticileri vs.)
 - uygulama programcılarının yazdıkları

25 / 31

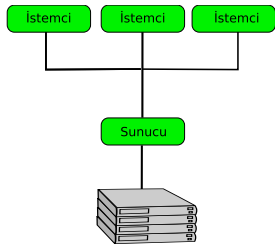
Yapı



- istemci ile sunucu aynı makinada ya da farklı makinada olabilirler

26 / 31

Çok İstemci / Tek Sunucu



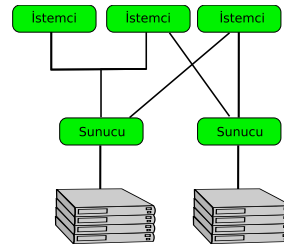
- birden çok istemci bir sunucuya bağlanarak çalışabilir

Örnek (Banka)

- merkezde bir sunucu
- şubelerde istemciler

27 / 31

Çok İstemci / Çok Sunucu



- sunucular da dağıtık olabilir

Örnek (Banka)

- her şube kendi hesaplarının sunucusu (ve istemcisi)
- her şube diğer şube hesaplarının istemcisi

28 / 31

SQL

- **Structured Query Language**
 - veri tanımlama dili
 - veri işleme dili
 - genel amaçlı programlama dilleriyle etkileşim
- **tarihçe**
 - 1970'lerde IBM başlatıyor
 - standartlar: 1992, 1999, 2003

29 / 31

SQL Ürünleri

- Oracle
- IBM DB2, Progress, MS-SQL, Sybase
- açık: PostgreSQL, MySQL, Firebird
- gömülü: SQLite, MS Access

30 / 31

Kaynaklar

Okunacak: [Date](#)

- ▶ Chapter 1: An Overview of Database Management
 - ▶ 1.4. [Why Database?](#)
 - ▶ 1.5. [Data Independence](#)
- ▶ Chapter 2: [Database System Architecture](#)