Bilgisayar Programcılığı   
2.Sınıf 1.Öğretim

No: 223010710026

Tuğçe Karakuş

ÖZET

Verilere hızlı ve kolay biçimde erişebilme isteği verilerin düzenli bir biçimde saklanması ihtiyacını doğurmuş dolayısıyla geçmişten günümüze bir biçimde hayatımızda kendisine yer bulmuştur. Geçmişle günümüzdeki fark kullanılan araç gereç teknik ve yaklaşımlar da ortaya çıkmaktadır örneğin geçmişte ev telefonuyla biriyle iletişime geçmek istendiğinde kişilerin adı soyadı telefon numaralarının tutulması bir veri tabanı örneğidir. Yine benzer bir şekilde devlet kurumları hastaneler ve ticari kuruluşlar da ihtiyaç duyulan verileri kâğıt dosyaları halinde belirli bir düzende arşivlemekteydiler. Ancak düzenlenmesi ve saklanması gereken veri miktarın artması ve bu verilerin fiziksel olarak uzun süre saklanabilmesi, korunabilmesi ve istendiğinde hızlıca erişilebilmesi kolay bir iş değildi. Söz konusu bu problem bilgisayar teknolojilerinin gelişmesinin gelişmelerinin temel nedenlerinden biri olmuştur.

**Veri Tabanına İlişkin Temel Kavramlar:**

**Veri** ham gözlemler, işlenmemiş gerçekler ya da izlenimlerdir. Harf, rakam ya da çeşitli sembol ve işaretler yardımıyla temsil edilir. Birbirleriyle ilişkilendirilip yorumlanmadıkları sürece tek başlarına bir anlam ifade etmezler.

**Bilgi** ise verinin işlenmiş ve karar verme sürecine destek olacak duruma

dönüştürülmüş biçimidir. Söz konusu işleme ve dönüştürme süreci; veri üzerinde

kaydetme, sınıflama, sıralama, hesaplama, özetleme, çoğaltma, analiz ve raporlama işlemlerinin uygulanması ile gerçekleştirilir. Bu işlemler sonucunda veri anlam kazanarak bilgiye dönüşmüş olur.

**Veri tabanı (database)** herhangi bir konuda birbiriyle ilişkili olan ve amaca uygun olarak

düzenlenmiş, mantıksal ve fiziksel olarak tanımlanmış veriler bütünüdür.

**Veri Tabanı Özellikleri:**

•Veri tabanı herhangi bir kurumda birden fazla uygulamada ortak olarak kullanılabilen verilerden oluşur.

• Veri tabanında sürekli niteliği olan veriler bulunur. Buna göre, girdi ya da çıktı verisi

olan ya da kurum için sürekli bir anlam ifade etmeyen geçici veriler veri tabanında

yer almaz.

• Veri tabanı, ortak kullanılan verilerin tekrarlanmasına izin vermeden çok amaçlı

kullanılmasına olanak verir.

• Veri tabanında saklanan veriler durağan nitelikte değişmez veriler değildir. Ekleme,

silme ya da güncelleme işlemleri ile veri tabanındaki veriler değiştirilebilir

***Veri tabanı Yönetim Sistemi (VTYS*-Database Management System);** Veri tabanı tanımlamak, veri tabanı oluşturmak, veri tabanında işlem yapmak, veri tabanının farklı kullanıcı yetkilerini belirlemek, veri tabanının bakımını ve yedeklemesini yapmak için geliştirilmiş programlar bütünüdür.

Veri tabanı ve veri tabanı yönetim sisteminin birlikte oluşturduğu bütün ise **veri tabanı**

**sistemi** olarak ifade edilir.

Geçmişten beri veriye erişim amacıyla farklı yaklaşımlar kullanılmıştır. Bu yaklaşımlardan

ikisi; sıralı erişim ve doğrudan erişim biçimindedir.

**Sıralı erişimde**, istenilen veriye ulaşılıncaya kadar ilgili dosyadaki tüm verilerin sırayla

okunması gerekir. Geçmişte kullanılan müzik kasetleri bu tür erişim kullanımına örnektir.

Bu kasetlerde şarkı olarak dinlediğimiz müzik verileri sırayla çalınmaktaydı. Kasette yer

alan 6 numaralı şarkıyı dinleyebilmek için ya ilk 5 şarkıyı dinlemek ya da kasetçaların

ileri düğmesine basarak 5 şarkılık verinin fiziksel olarak ileri sarılması gerekirdi.

Bu şekilde, dosyada yer alan tüm verilerin okunması zorunluluğu ve istenilen bilgiye

anında ulaşılamaması sıralı erişimin dezavantajı olarak ortaya çıkmaktadır.

**Doğrudan erişimde**, sıralı erişimin aksine istenilen veriye ulaşabilmek için o veriye kadar

olan diğer tüm verilerin okunması gerekmemektedir. Bu erişim biçiminde adından da

anlaşılacağı gibi istenilen veriye doğrudan erişim mümkündür. Bu erişim biçiminde, verilerin

yer aldığı fiziksel adresler birer indeks numarası ile tanımlanıp bu indeks numaraları

da ayrı bir dosya olarak saklanır.

|  |  |
| --- | --- |
| **Geleneksel dosya sistemlerinin sakıncaları** | **Veri tabanı yönetim sistemlerinin**  **üstünlükleri** |
| Veri tekrarı ve veri tutarsızlığına yol açar | Veri tekrarı ve veri tutarsızlığını önler |
| Veri paylaşımına olanak vermez. | Veri paylaşımına olanak verir |
| Uygulamalarda ihtiyaç duyulan  değişikliklerin gerçekleştirilebilmesi için  uzmanlık bilgisi gerektirir | Uzmanlık bilgisine ihtiyaç duyulmayacak  derecede kullanım kolaylığı sağlar |
| İstenilen veriye ulaşmada güçlükler  bulunur. | İhtiyaç duyulan veriye, tanımlanmış  kullanıcı yetkileri kapsamında kolaylıkla  erişilmesini sağlar. |
| Verilerin güvenliği ve gizliliği konusunda  sorun yaşanır | Veri güvenliği ve gizliliğini güçlü bir  biçimde yerine getirir. |
| Veriler ve uygulamalarla ilgili belirli bir  standart yoktur | Veriler ve uygulamalarla ilgili standart  yapı ve kuralların olması kullanım  kolaylığı sağlar |
| Verileri yedekleme ve kurtarma  konusunda güçlükler yaşanır | Verileri yedekleme ve kurtarma  konusunda kolaylık sağlayan programlar  barındırır. |

**VERİTABANI KULLANICILARI**

Veri tabanı ile herhangi bir şekilde etkileşimde olan kişi ya da kişiler veri tabanı kullanıcısı

olup aşağıdaki gibi sınıflandırılabilirler:

• Veri tabanı Sorumluları

• Veri tabanı Yöneticisi

• Veri tabanı Tasarımcısı

• Son Kullanıcılar

• Standart Kullanıcılar

• Sıradan ya da Parametrik Kullanıcılar

• Gelişmiş Kullanıcılar

• Bağımsız Kullanıcılar

• Sistem Analistleri ve Uygulama Programcıları

**Veri tabanı Sorumluları**

* **Veri tabanı Yöneticisi**

Veri tabanı yöneticisinin (database administrator) veri tabanına erişim yetkilerini belirleme,

veri tabanı kullanımının düzenlenmesi ve izlenmesini sağlama, ihtiyaç duyulan yazılım

ve donanım kaynaklarını edinme gibi sorumlulukları vardır. Ayrıca güvenlik ihlalleri ve kötü sistem yanıt süresi gibi sorunların çözümünden de sorumludur.

* **Veri tabanı Tasarımcısı**

Veri tabanı tasarımcısı (database designer) veri tabanında saklanacak olan verilerin tanımlanmasından ve bu verilerin depolanması ve gösterilmesi için gerekli olan uygun yapıların seçilmesinden sorumludur.

**Son Kullanıcılar**

Son kullanıcılar (end users), yaptıkları işler gereği veritabanına sorgulama ya da güncelleme

yapmak veya rapor türetmek için erişen kullanıcılardır

* **Standart son kullanıcılar:**Veritabanına nadiren erişim yapan fakat her seferinde farklı bilgi ihtiyacı olabilen kullanıcılardır
* **Sıradan ya da parametrik son kullanıcılar:** Bu kullanıcıların temel iş fonksiyonları, veri tabanı üzerinde sürekli bir sorgulama ve güncelleme yapmalarını gerektirir.
* **Gelişmiş son kullanıcılar*:*** VTYS’nin sağladığı özellikler yardımıyla ayrıntılı olarak

belirledikleri karmaşık gereksinimlerini karşılamak amacıyla veri tabanını kullanan

gruptur. Mühendisler, bilim adamları, işletme analistleri vb. bu gruptaki kullanıcılara

örnek olarak verilebilir.

* **Bağımsız son kullanıcılar:** Bu kullanıcılar menü kullanımı ya da araç çubukları gibi

grafiksel ögeler yardımıyla kullanım kolaylığı sağlayan hazır paket programlarını kullanarak kişisel veri tabanlarının sürekliliğini sağlar

**Sistem Analistleri ve Uygulama Programcıları**

**Sistem analisti** son kullanıcıların, özellikle de sıradan son kullanıcıların gereksinimlerini belirleyen ve standart işlemler yoluyla bu gereksinimleri karşılayabilecek ayrıntıları

belirleyen kişi ya da kişilerdir. **Uygulama programcıları** ise sistem analisti tarafından

belirlenen ayrıntıları program hâline getiren ve daha sonra test eden, hataları ayıklayan,

belgeleyen ve kaydedilmiş işlemler olarak sürekliliğini sağlayan kişilerdir

**Veri Modelleri**

Veri tabanı yaklaşımının temel karakteristiklerinden biri, veri tabanının bazı veri soyutlama

düzeyleri sağlamasıdır. Veri soyutlama (data abstraction), verilerin düzenlenmesi ve

depolanmasına ilişkin ayrıntıların gizlenmesi ve verinin daha iyi anlaşılmasını sağlamak

için veriye ilişkin temel özelliklerin vurgulanması anlamına gelir. Veri modeli ise söz konusu

bu soyutlamaları gerçekleştirebilmek için gerekli olan araçları sağlar.

**Veri modeli** (data model), bir veri tabanının mantıksal yapısını tanımlamada kullanılacak

kavramlar, işlemler ve kurallar bütünüdür. Veri tabanının mantıksal yapısı; veri

tipleri, veriler arasındaki ilişkiler, veri üzerinde uygulanacak kısıtlamalar vb.dir

**Veri Modellerinin Sınıflandırılması**

Yüksek düzeyli ya da kavramsal veri modelleri (conceptual data model), kullanıcıların

veri algılama biçimiyle ilişkili kavramları kapsar. Düşük düzeyli ya da **fiziksel veri modelleri** (physical data models), verinin bilgisayar ortamında nasıl depolanacağına ilişkin ayrıntıları tanımlayan kavramları kapsar. Fiziksel veri modelleri kavramları genellikle son kullanıcılar için değil, bilgisayar uzmanları için geliştirilir.

**Şemalar, Örnekler ve Veri tabanının Durumu**

Veri tabanının herhangi bir veri modeliyle tanımlanması **veri tabanı şeması** olarak adlandırılır. Söz konusu bu şema veri tabanının tasarlanması sürecinde oluşturulur ve sık sık değişiklik göstermez.

**Üç Şema Mimarisi**

Veri tabanı yaklaşımının önemli karakteristikleri aşağıdaki biçimde sıralanabilir:

1. Veri tabanı tanımlarını (şema) depolamak için kendi kendine tanım yapabilmeyi

sağlayan bir katalog kullanır.

2. Veri ve program izolasyonunu sağlar (program – veri bağımsızlığı; program – işlem

bağımsızlığı)

3. Çoklu kullanıcıyı destekler

Üç şema mimarisinin amacı kullanıcı uygulamalarını fiziksel veri tabanından ayırt etmektir.

Bu mimaride şemalar izleyen üç düzeyde tanımlanır:

**1. İçsel (fiziksel) düzey** (internal level), veri tabanının fiziksel depolama yapısını tanımlayan

içsel şemayı içerir. İçsel şema, veriyi depolama ayrıntılarının tamamını

ve veritabanına erişim yollarını tanımlayan fiziksel veri modelini kullanır.

**2. Kavramsal düzey** (conceptual level), kullanıcı topluluğu için tüm veri tabanının

yapısını tanımlayan kavramsal şemayı içerir. Kavramsal şema fiziksel depolama

yapısının ayrıntılarını gizler ve veri tabanında yer alan verilerin tipine, veriler arası

ilişkilere, kullanıcı işlemlerine ve kısıtlara ilişkin tanımlara yoğunlaşır

**3. Dışsal (görünüm) düzey** (external level), bir dizi dışsal şema ya da kullanıcı görünümü

içerir. Her dışsal şema bir grup kullanıcının ilgilendiği bazı veri tabanı

bölümlerini tanımlar. Böylece veri tabanının diğer kısmı bu kullanıcı grubundan

gizlenir. Her dışsal şema, bir yüksek düzey veri modelinde tasarlanan dışsal şema

tabanlı uygulama veri modeli kullanır.

**Veri Bağımsızlığı**

Veri bağımsızlığı iki başlıkta ele alınır:

**1. Mantıksal veri bağımsızlığı** (logical data independence), kavramsal şemanın dışsal

şemalarda ya da uygulama programlarında değişiklik yapılmaksızın değiştirilebilmesi

anlamına gelir. Kavramsal şemada değişiklik, veri tabanını genişletmek

(bir kayıt tipi ya da veri kalemi eklemek) kısıtları değiştirmek ya da veri tabanını

eksiltmek (bir kayıt ya da veriyi silmek) biçiminde olabilir.

**2. Fiziksel veri bağımsızlığı** (physical data independence), kavramsal şemada bir değişiklik

yapılmaksızın içsel şemada değişiklik yapma kapasitesidir. İçsel şemalarda değişiklik ihtiyacı fiziksel dosyalar üzerinde yeniden düzenleme ihtiyacı ile ortaya çıkabilir. Veritabanına ek erişim yapısının oluşturulması, veri tabanının bilgiyi çağırma performansının geliştirilmesi ya da veri tabanının güncellenmesi içsel şema değişikliklerine verilebilecek örneklerdir

**VERİTABANI TÜRLERİ**

Veri tabanında yer alacak veriler ve veriler arasında kurulacak ilişkiler mantıksal olarak

ilgili veri modeline göre yapılandırılır ve veri tabanları da buna göre sınıflandırılır.

Geçmişten günümüze kadar geliştirilmiş olan çok sayıda veri modeli, kullandıkları teknikler

açısından dört temel başlıkta incelenir. Bu dört veri modelinden hangisini kullandığına

bağlı olarak veri tabanları da aşağıda verilen dört başlık altında sınıflandırılabilir:

1. Hiyerarşik veri tabanı (Hierarchical database)

2. Ağ veri tabanı (Network database)

3. İlişkisel veri tabanı (Relational database)

4. Nesneye yönelik veri tabanı (Object oriented database)

**Hiyerarşik Veri tabanı**

Bu tür veri tabanlarında kullanılan veri modelinde kayıtlar, ilişkileri temsil eden ve ağaç yapısına benzeyen kök ve dallar biçiminde hiyerarşik bir yapıda oluşturulur. Bu yapı, başlangıç noktası ağacın kökü, bağlantılı kılınacak noktalar dallar ve ana dallara bağlı alt dallar olarak düşünülebilen bir yapı biçimindedir.

**Ağ Veri tabanı**

Hiyerarşik veri modelindeki ebeveyn-çocuk ilişkisinin yetersizliği ağ veri modeliyle giderilmeye çalışılmıştır. Buna göre her bağlantı noktası düğüm olarak ifade edilirse hiyerarşik yapıdan farklı olarak ağ veri modelinde, her düğümün birden fazla ebeveyn ve birden fazla çocuk düğümü ile bağlantısı olabilir.

**İlişkisel Veri tabanı**

Bu yapıda ilk iki veri modelinden farklı olarak birden çok ilişki biçimi

kullanılabilir. Günümüzde kullanılan veri tabanı yönetim sistemlerinin hemen hemen

hepsinde tercih edilen model ilişkisel veri modelidir

**Nesneye Yönelik Veri tabanı**

Verileri satırlar ve sütunlar biçiminde

düzenlemek için tasarlanmış olan veri tabanı yönetim sistemleri grafik unsurları ve multimedya unsurlarını kullanmaya pek uygun değildir. Bu nedenle bu eksikliği gidermek amacıyla nesneye yönelik veri modelleri geliştirilmiştir. Söz konusu bu modelleri kullanan veri tabanları da nesneye yönelik veri tabanı olarak adlandırılmaktadır

**VERİTABANI YÖNETİM SİSTEMİ YAZILIMLARI**

Günümüzde yaygın olarak bilinen veri tabanı yönetim sistemi yazılımları; MS SQL Server,

Oracle, MySQL, Sybase, MS Access, PostgreSQL, IBM DB2, Informix, Advantage

biçiminde sıralanabilir.

Bilgisayar Programcılığı   
2.Sınıf 1.Öğretim

No: 223010710026

Tuğçe Karakuş