# Veritabanı Yönetim Sistemleri

*Veritabanı Yönetim Sistemleri, kullanıcılar adına düşük seviye veri işlemlerini yerine getirip, kullanıcıya standart bir komut arayüzü veya grafik arayüzü ile veri yönetimi fonksiyonları sağlayan uygulamalarıdır.*

Veri tabanı yönetim sistemi(VTYS), yeni bir veritabanı oluşturmak, veri tabanını düzenlemek, geliştirmek ve bakımını yapmak gibi çeşitli karmaşık işlemlerin gerçekleştirildiği birden fazla programdan oluşmuş bir yazılım sistemidir. Veri tabanı yönetim sistemi, kullanıcı ile veri tabanı arasında bir arabirim oluşturmaktadır ve veri tabanına her türlü erişimi sağlar.

Veri tabanı yönetim sistemi, veri tabanını tanımlama, veri tabanını oluşturma ve üzerinde işlem yapma yeteneği olan bir yazılım sistemidir. Veri tabanının tanımlanması: veri tabanını oluşturan verilerin tip ve uzunluklarının belirlenmesidir. Veri tabanını oluşturulması ise veri için yer belirlemesi ve saklama ortamına verilerin yüklenmesini ifade eder. Veri tabanı üzerinde işlem yapmak; belirli bir veri üzerinde sorgulama yapmak, meydana gelen değişiklikleri yansıtmak için veri tabanın güncellenmesi ve rapor üretilmedi gibi işleri temsil eder. Ayrıca veri tabanı yönetim sistemi, verinin geri çağrılmasını sağlar. Veri tabanına yeni kayıt eklemek, eskileri çağırmak ve gerekli düzeltmeleri yapmak yoluyla, verinin bakımını ve sürekliliğini gerçekleştirir, kayıtlara yeni veri eklemek ve yeni kayıtlar oluşturmakla, veri tabanını genişletir.

Bir veri tabanı yönetim sistemi yazılım, karışık ve kapsamlı bir yazılımdır. Kısaca şu işleri gerçekleştirir:

1. Veri tabanını oluşturmak ve yönetmek
2. Veri tabanyna erişim yetkisi olan kullanıcılara erişim izni vermek
3. Kullanıcı isteklerine uygun olarak veriye erişimi sağlamak
4. Veri tabanında güncellemeler yapmak.

Veri tabanı yönetim sistemi bu işlerini bazı özel alt diller (Structured Query Language,SQL v.b) kullanarak gerçekleştirir. Veri tabanı yönetim sistemi SQL komutunu ve diğer işlevlerini, aşağıdaki adımları sırayla gerçekleştirerek yapar:

1. Kullanıcı, veri tabanına erişimi olan bir yazılım içinde bu SQL komutunu yazar.
2. Veri tabanı yönetim sistemi bu isteği alarak der ve analiz eder.
3. Görünüm seviyeden kavramsal ve fiziksel seviyelere doğru denetleme yapar.
4. Gerekli kayıtlan içeren disk dosyalarının yerini öğrenir.
5. Gerekli olan kayıtları okumak için kontrolü işletim sistemine bırakır.
6. İşletim sistemi gerekli veriyi disk dosyalarından okuyup veri tabanı yönetim sisteminin bufferlarına aktarır, kontrolü geri verir.
7. Veritabanı yönetim sistemi bufferlarından verileri alarak oradan istenilen terminale gönderir.

**Veri Tabanı Sistemi Kullanımının Avantajları**
Veri tabanına duyulan ihtiyacın asıl nedeni merkezi kontroldür. Veri tabanı, veri tabanı yöneticisi tarafından yönetilir ve sistem merkezi olarak veriler güncellenir. Veri tabanlarının avantajlarını kısaca şöyle sıralayabiliriz:

1. Verilerin tekrarı azalır. Böylece verilerde meydana gelen karışıklıklar önlenmiş olup bellekten tasarruf edilir
2. Yanlışlıkların giderilmesi. Tutarsız (hatalı) bilgilerin girişi engellenir. Dosya sisteminde bir değişildik olduğu zaman dişer bilgilerinde değiştirilmesi gerekiyordu
3. Verinin paylaşımı sağlanır. Aynı veri tüm kolaylığıyla dişer kullanıcılar tarafindan da kullanılır
4. Bilgilerin standartlaşması sağlanır. Belli standartlar varsa bu standartlara uyulması gerekir, ayrıca yeni standartlaşmalara doğru adımların atılması

**Veri Tabanı Sistemlerinin Dezavantajları**
Dosya sisteminden veritabanı sistemine geçmenin bir çok yararı olmakla beraber bazı dezavantajlarda beraber gelmektedir

1. Veri tabanı sistemi iyi tasarlanmadığı taktirde, bazı bileşenler iyi bir şekilde oluşturulmamışsa sistem dosya sistemine göre çok daha kötü çalışabilir.
2. Veri tabanı sisteminin kurulumu ve bakımı klasik dosya sistemine göre pahalı olabilir.

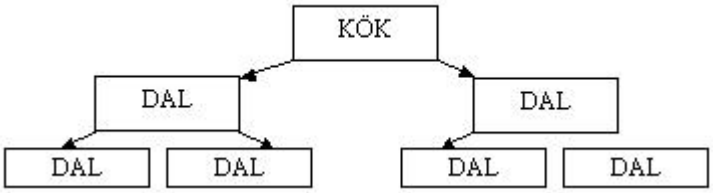
Verilerin tekrarı azalır. Böylece verilerde meydana gelen karışıklıklar önlenmiş olup bellekten tasarruf edilir,

Veri modeline göre veritabanı yönetim sistemleri

Yapısal olarak bütün veri tabanları bir değildir. Veri tabanları verileri saklama ve onlara erişme bakımından farklı tiplere ayrılır.

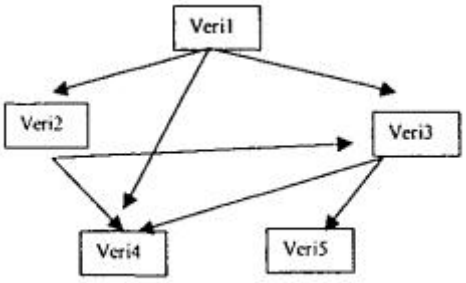
**Hiyerarşik Veri Tabanları**

Bu veritabanı tipi, ana bilgisayar ortamlarında çalışan yazılımlar tarafından kullanılmaktadır. Bu türde en çok kullanılan yazılım, IBM tarafından çıkarılan IMS' dir. Uzun bir geçmişe sahip olmasına rağmen, PC ortamına uyarlanan hiyerarşik veri tabanları

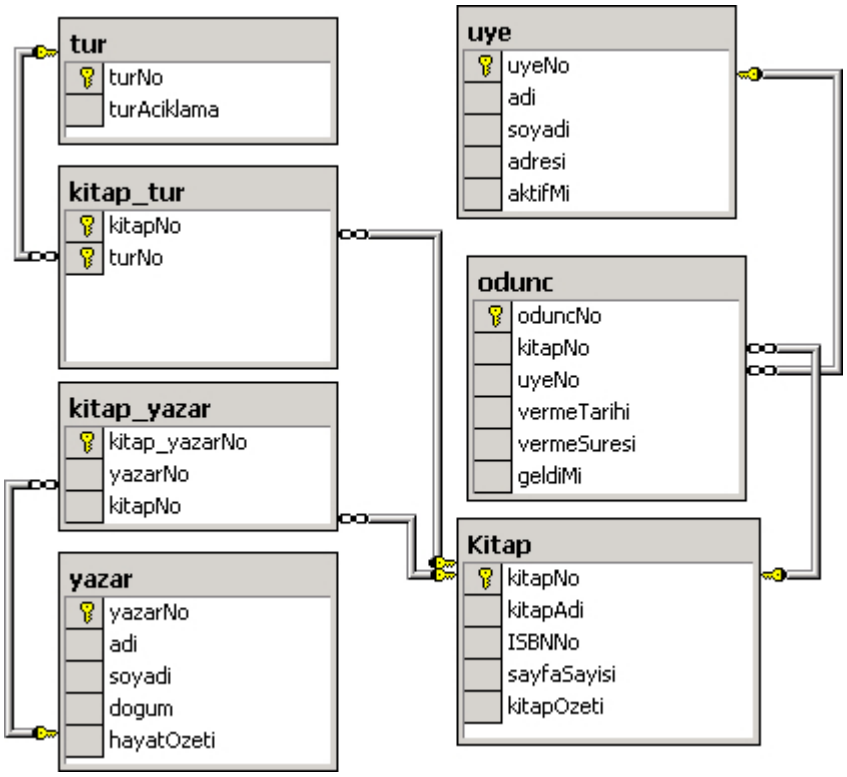



yoktur.
Hiyerarşik veri tabanları, bilgileri bir ağaç (tree) yapısında saklar. Kök (Root) olarak bir kayıt ve bu köke başlı dal (Branch) kayıtlar bu tip veritabanının yapısını oluşturur. Yukarıda böyle bir veri tabanının yapısı gösterilmektedir.

**A) Veri Tabanları**
Hiyerarşik veri tabanlarının yetersiz kalmasından dolayı bilim adamlarının ortak çalışması sonucu ortay konulmuş bir veri tabanı türüdür
A? veri tabanları verileri ağaçların daha da geliimiş hali olan graflar (ağacın kendiside özel bir graftır.) şeklinde saklarlar. Bu yapı en karışık yapılardan biridir.




**İlişkisel Veri Tabanları**
E.F. Codd Tarafından Geliştirilmiştir. Bu sistemde veriler tablo seklinde saklanır. Bu veri 'tabanı yönetim sisteminde veri alış verişi için özel işlemler kullanılır. Bu işlemlerde tablolar operandlar olarak kullanılır. Tablolar arasında ilişkiler belirtilir. Bu ilişkiler matematiksel bağıntılarla (ilişkilerle) temsil edilir. Günümüzde en çok tercih edilen ilişkisel veri tabanı ayrıca detaylı olarak anlatılacaktır.




**Nesneye Yönelik Veri Tabanları**
Günümüzde nesne kavramı her yerde kullanılmaya başladı. Pek çok kelime işlemci ve hesap tablosu programlarının alıştığımız görünümlerine artık bir de nesneler eklendi. Ancak bu gerçek anlamda bir nesneye yönelik yazılım demek değildir.
Yüzde yüz nesneye yönelik bir yazılımın tamamen nesneye temelli çalışması gerekir. Yazılımın mutlaka nesneye yönelik bir dilde yazılmış olması beklenir. Fakat Windows gibi işletim sistemi üzerinde çalışan yazılımlar bu özelliklere tümüyle sahip değiller. Sadece nesne kavramını kullanarak bazı ek özellikler sunar.
Nesneye yönelik veri tabanı da , C++ gibi nesneye dayalı bir dille (OOPL) yazılmı? olan ve yine C++ gibi nesneye dayalı (OOPL) bir dille kullanılan veri tabanı anlamına geliyor. Günümüz teknolojisinde yüzde yüz nesneye yönelik bir veri tabanı yaygın olarak kullanıma sunulmuş değildir. Ancak nesneye yönelik veri tabanlarının bazı üstünlükleri olacağından söz ediliyor.

İlişkisel veri tabanları ile karşılaştırıldığında nesneye yönelik veri tabanlarının sahip olması gereken üstünlükler şunlardır:

1. Nesneler, bir tabloda yer alan bir kayıttan çok daha karmaşık yapıya sahiplerdir ve daha esnek bir yapıda çok daha kullanışlı düzenlenebiliyorlar.
2. Nesneye dayalı bir veri tabanında, yapısı gereği arama işlemleri çok hızlı yapylabilir. Özellikle büyük tablolarla uğraşırken ilişkisel veri tabanlarından çok daha hızlı sonuca ulaşırlar. Ancak çalışma mantığı tümüyle değişir

Tüm bu özellikler tamamen nesneye yönelik olan veri tabanları için geçerlidir. Bazı ilişkisel veri tabanları ile çalışan yazılımlarda da nesnelerin bazı özellikleri kullanır, ama nesneye yönelik veri tabanı bunu kendini ilişkisel veri tabanı kurallarına uydurarak gerçekleştirebilir.