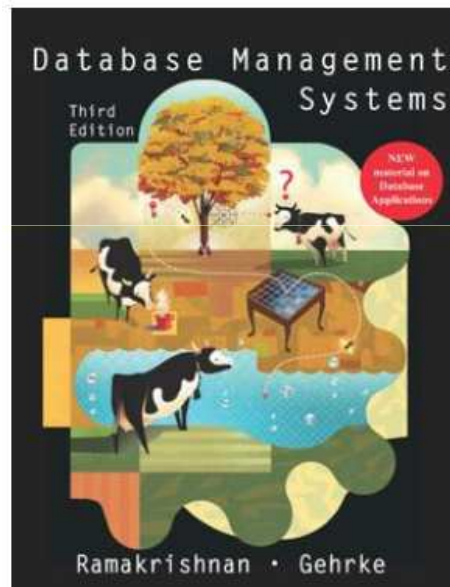


Veri Tabanı Yönetim Sistemleri

Öğr. Gör. Dr. İsmail İŞERİ
OMÜ Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

Kaynaklar



- <https://www.youtube.com/watch?v=EUzsy3W4I0g&list=PL812991E24CB4E28F>

Konular

- Veri tabanı nedir?
- VTYS nedir
- Veri tabanı bileşenleri ve aktörler Veri Modelleri
- İlişkisel Model
- Veri tabanı tasarımı
- Normalleştirme
- SQL (DML, DDL, DQL)
- MySql Üzerinde Örnek Uygulamalar ve MySql Workbench kullanımı – ORACLE Kurulum Kullanım
- Workshop

Veri Tabanı Yönetim Sistemi (VTYS)

- **Veri tabanı**; belirli bir veri modeli ile saklanan, sistematik erişim imkânı olan, yönetilebilir, güncellenebilir, taşınabilir, birbirleri arasında tanımlı ilişkiler bulunabilen bilgiler kümesidir.
- **VTYS**; bir veri tabanını yaratmak, üzerinde istenilen bilgiyi aramak, gerektiğinde bilgi eklemek-silmekdeğiştirmek ve veri tabanı ile ilgili her türlü işletimsel gereksinimleri karşılamak için kullanılan geniş kapsamlı yazılım sistemidir.
- **VTYS**, birbirleri ile ilişkili veri ve programlar topluluğundan oluşmaktadır.

Veri Modeli (Data Model)

- Veri modeli, verileri mantıksal düzeyde düzenlemek için kullanılan yapılar, kavramlar ve işlemler topluluğudur.
- Her VTYS belirli bir veri modeli kullanır.
- Veri tabanını tasarlayan kişi, veri modelinin yapılarını ve kavramlarını kullanarak mantıksal düzeydeki düzenlemeleri oluşturur. Daha sonra o veri modelini kullanan bir VTYS üzerinde bu düzenlemelere göre veri tabanı yaratılır.
- Günümüzde en çok kullanılan veri modeli “ilişkisel veri modeli” dir.

Veri Tabanı Mimarisi

- Veri tabanı mimarisi üç seviyeden oluşur:

- İç (Fiziksel) Seviye:

Verilerin nasıl saklanacağı ve bilgisayar tarafından nasıl işleneceği ile ilgilenen seviyedir. Veri tabanının performansı bu seviyenin iyi planlanmış olmasına bağlıdır. Örn: MyIsam, Innodb, ..

- Kavramsal Seviye:

İç seviye ile dış seviye arasında iletişimi sağlayan seviyedir.

- Dış (Alt) Seviye:

Kullanıcıların verileri gördüğü yada veri eklemek-silmek değiştirmek için kullandığı ekranlar dış seviyede yer alır.

Neden VTYS?

- VTYS yaklaşımında veri girişi ve depolanması veriye erişen uygulama programlarından bağımsızdır.
- Klasik dosya kullanımında ise, kayıt desenleri ve dosya yapılarında ortaya çıkabilecek en ufak bir değişiklik bile uygulama programlarının değiştirilmesine neden olmaktadır.

VTYS Kullanmanın Avantajları

- Gereksiz veri tekrarını önler.
- Tüm uygulamaların gereksinim duyduğu veriler birbirleri ile bütünleşik yapıdadır. Dolayısıyla veri kaynağı tek olarak tasarlanır, böylece veri tekrarı önlenmiş olur.
- Örneğin, Türkiye'deki il kodları ve isimleri hem personel alt sisteminde hem de pazarlama alt sisteminde ayrı ayrı tutulmaz

VTYS Kullanmanın Avantajları

- Veri bütünlüğünü (data integrity) sağlar.
- Veri bütünlüğü, verinin doğruluğunu ve tutarlılığını anlatmaktadır.
- Veri bütünlüğü için bazı kısıtlamalar tanımlanabilir.
- Örneğin, öğrenci bilgileri girilirken doğduğu il koduna 100 değeri girilirse, yanlış bilgi nedeniyle bu isteğin yerine getirilmemesi istenebilir.

VTYS Kullanmanın Avantajları

- Verilerin güvenliğini sağlar.
- Tüm verilere her kullanıcının kolayca erişebilmesi çoğu zaman istenmeyen bir durumdur.
- Her kullanıcıya çeşitli yetkiler atanarak, bu kullanıcının erişebileceği, değiştirebileceği ve silebileceği veriler ayrı ayrı tanımlanabilir.
- Örneğin, pazarlama bölümünde çalışan bir kullanıcının diğer personelin özlük bilgilerine ulaşması engellenebilir.

VTYS Kullanmanın Avantajları

- Aynı zamanda yapılan erişimlerde tutarsızlıkların ortaya çıkmasını önler.
- İki veya daha fazla kullanıcı aynı anda aynı veri üzerinde değişiklik yapmak istediğinde, yetkiye ve/veya kimin önce eriştiğine bakarak, birine önceliği verir, diğerlerini bekletir.

VTYS Kullanmanın Avantajları

- Veriler üzerinde merkezi denetim sağlar.
- Kullanıcılar işletim sistemi komutları ya da genel amaçlı programlama dilleri ile yazılmış uygulama programlarını kullanarak doğrudan veri tabanındaki verilere erişemezler ve bu verileri değiştiremezler.
- Veri tabanı kullanımı yalnız VTYS (DBMS - Data Base Management System) olarak adlandırılan yazılım sistemi aracılığıyla mümkündür.

VTYS Kullanmanın Avantajları

- Veri tabanının bilgisayar belleklerindeki fiziksel yapısı kullanıcılardan gizlenir.
- Kullanıcılara daha yalın mantıksal yapılar sunulur.
- VTYS, bir anlamda yüksek düzeyli programlama dili derleyicisi gibi davranarak kullanıcının, soyut terimler kullanarak veri tabanı ile ilişki kurmasını sağlar.
- Böylece kullanıcı, sistem tarafından kullanılan karmaşık veri gösterimleri ve algoritmaların ayrıntılarıyla uğraşmadan, neyin yapılmasını istediğini belirterek isteklerini ortaya koyabilir.

VTYS Kullanmanın Avantajları

- Veritabanı sisteminin kendi kendini tanımlayabilmesi
- Veritabanı sistemleri yalnızca veriyi değil verinin özelliklerini, kısıtları ve ilişkileri de tanımlar (meta)
- Meta dosya tabanlı sistemlerde uygulamanın içindedir, bu nedenle uygulamalar yalnızca tek bir veritabanı ile çalışabilir

VTYS Kullanmanın Avantajları

- Uygulamalar ile verinin izolasyonu
- Dosya tabanlı uygulamalarda dosya değiştiği anda bağlı tüm uygulamalar değişmelidir
- Veritabanı sistemine bağlı uygulamalarda veri ile uygulama birbirinden bağımsızdır

VTYS Kullanmanın Avantajları

- Verinin farklı görünümlerine ulaşım (views)
- Farklı kullanıcılar verinin farklı bölümlerine farklı sorgular ile ulaşabilir
- Örn: Bir kullanıcı yalnızca öğrenci not bildirimi bir diğeri ise öğrencinin alması gereken dersleri listeleyebilir

VTYS Kullanmanın Avantajları

- İhtiyaca göre veri tutma
- Dosya bazlı sistemlerde aynı veri birden çok kullanıcı tarafından tutuluyor olabilir
- DBMS sistemlerinde aynı veri tek bir merkezden idare edilir ve ortak veriler kullanıcılar arasında paylaşılır

VTYS Kullanmanın Avantajları

- Yedekleme ve geri yükleme özelliği
- Veriler düzenli olarak yedeklenebilir ve herhangi bir arıza durumunda geri yüklenebilir

VTYS Kullanmanın Avantajları

- Farklı arayüzler kullanabilme
- Kullanıcılara ve verinin çeşidine göre farklı ara yüzler aracılığıyla veriye erişilebilir

VTYS Kullanmanın Avantajları

- Verinin ilişkilendirilmesi
- Farklı veriler birbiri ile ilişkilendirilebilir

VTYS Kullanmanın Avantajları

- Tetikçiler (trigger), ve saklı yordamlar (stored procedures)
- Veri belli bir şekle geldiğinde otomatik olarak belli değişikliklerin gerçekleşmesi sağlanabilir

VTYS Kullanmanın Avantajları

- Standartların belirlenebilmesi
- Raporlama, iş akışları gibi standartlar ortak veritabanı kullanımı ile kurumlar içinde uygulanabilir

VTYS Kullanmanın Avantajları

- Hızlı uygulama geliştirme süreleri
- Dosya bazlı uygulamalar geliştirmek veritabanı uygulamaları geliştirmekten daha uzun sürmektedir

VTYS Kullanmanın Avantajları

- Esneklik
- Veritabanındaki verilerin değişmesine gerek kalmadan yeni yapılar eklenip çıkarılabilir

VTYS Kullanmanın Avantajları

- Doğru bilgiye anlık ulaşabilmek
- Güncel veriye kullanıcılar aynı anda ulaşabilir, yapılan değişiklikleri görebilir

1



Veri Tanımlama Dili (VTD)

- Veri tabanı tanımlarını VTYS'ye iletmek için kullanılan biçimsel dile Veri Tanımlama Dili (DDL: Data Definition Language) adı verilir.
- Veri Tanımlama Dili kullanılarak oluşturulan veri tabanı tanımları, VTYS'nin Veri Tanımlama Dili Derleyicisi olarak adlandırılacak bir bileşeni tarafından çözümlenir, varsa eksik ve yanlışları bulunarak kullanıcıya iletilir.
- Yanlışlardan arınmış veri tabanı tanımları VTYS tarafından uygun yapılara dönüştürülerek saklanır.

Veri Tanımlama Dili (VTD)

- Veri tabanı tanımlarının VTYS tarafından derlenerek saklanması veri tabanı yaklaşımının temel özelliklerinden biridir.
- Bu süreç sayesinde, veri tabanı tanımlarının yetkili kişiler tarafından bir kez yapılması, tanımların kalıcılığının sağlanması ve kullanıcıların bu tanımları kullanmaları ve bu tanımlara uygun işlem yapmaları sağlanmış olur.
- Programlama dili kullanılarak gerçekleştirilen dosya tabanlı uygulamalarda, bilgisayar belleklerinde yalnız veri saklanır. Uygulama programlarında, veri üzerinde yapılacak işlemler yanında verinin tanımı da yer alır.

Veri Tanımlama Dili (VTD)

- Veri tabanı yaklaşımında ise veri tanımlama ile veriler üzerindeki uygulama işlemleri birbirinden tamamen ayrılmıştır.
- Veri tanımlama ve daha önce yapılmış tanımları değiştirme yetkisi yalnız Veri Tabanı Sorumlusu (Database Administrator) olarak adlandırılan ve veri tabanının tümünden sorumlu olan kişi ya da kişilere aittir.
- Veri tabanı üzerinde işlem yapan, uygulamaları gerçekleştiren kullanıcıların ise veri tanımlama ya da mevcut tanımları değiştirme yetkisi yoktur.

Veri Tanımlama Dili (VTD)

- VTD ile yapılan tanımlarda veri tabanı tanımları içinde yer alan en önemli tanımlar şunlardır:
 1. Mantıksal düzeyde yapılan düzenlemeler oluşturulan yapılar ve her yapıda hangi verilerin yer aldığı.
 2. Her verinin türü, uzunluğu, varsa varsayılan değeri ve diğer özellikleri.
 3. Veriler arası ilişkiler ve her türlü kısıtlamalar.
 4. Fiziksel veri yapıları ile ilgili tercihler ve parametreler.
 5. Kullanıcı tanımları ve kullanıcıların hangi veriler üzerinde hangi işlemleri yapmaya yetkili olduklarına ilişkin tanımlar.

Veri Tanımlama Dili (VTD)

- Veri tabanı tanımları veri sözlüğü (data dictionary) olarak da adlandırılır.
- VTYS'nin fiziksel ortamında aşağıda belirtilen çeşitli veriler saklanır:
 1. Veri kütükleri (dosyalar)
 2. Veri tabanı tanımlarının yer aldığı veri sözlüğü
 3. veri kütüklerine ya da veri sözlüğüne erişim için kullanılan dizinler (indices)
 4. Veri değerlerine ve uygulamalara ilişkin istatistiksel veriler ve günlük kütükleri (logfiles)

1



Sorgu Dili ve Sorgu İşleyicisi

- Veri tabanı uygulamaları için kullanılan en yaygın araç sorgu dilidir.
- Kullanıcı, sorgu dili ile, gerçekleştirmek istediği işlemi yalın bir biçimde ifade eder.
- Kullanıcının oluşturduğu sorguda, neyin yapılmasının istendiği yer alır; bunun nasıl yapılacağı ise yer almaz.

Yapısal Sorgu Dili (SQL)

- IBM, deneysel bir ilişkisel veritabanı yönetim sistemi geliştirmek amacıyla System/R adlı bir proje başlatır.
- Bu sistem için SEQUEL (Structured English Query Language) adında bir sorgu dili geliştirilmeye başlanmıştır.
- 1979' da tamamlanan bu proje sırasında geliştirilen dilin adı SQL (Structured Query Language) olarak değiştirilmiştir.

ORACLE

- System/R projesini izleyen bir grup mühendis, ilişkisel veritabanındaki potansiyeli görmüş ve Relational Software, Inc. adında bir şirket kurmuşlardır (1979)
- Oracle adını verdikleri ilk ticari ilişkisel veri tabanı yönetim sistemini geliştirmişler ve SQL'i bu sistemde sorgu dili olarak kullanmışlardır.

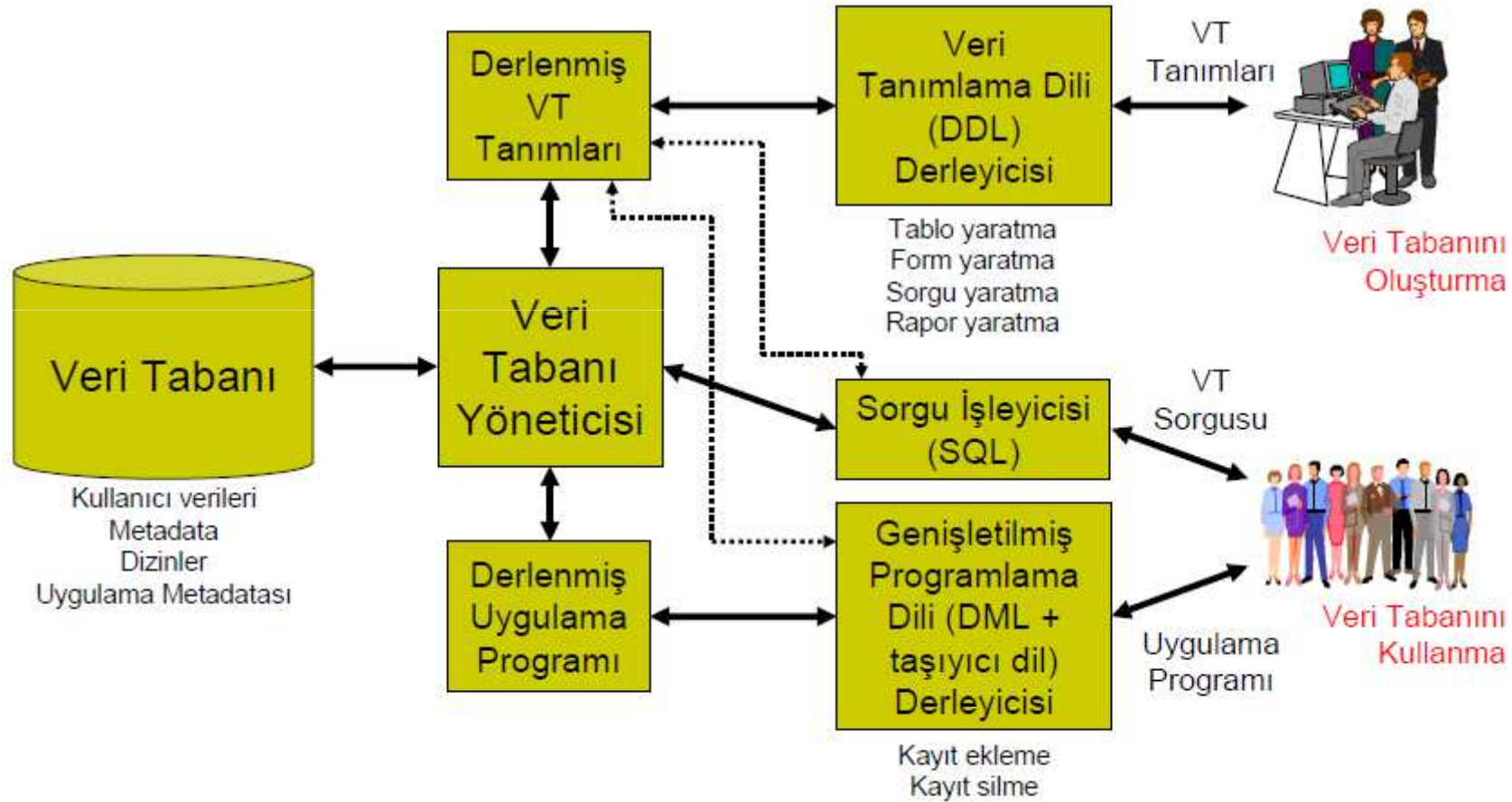
SQL Örneği

- İngilizce diline çok benzeyen SQL sorgu dilinde oluşturulmuş bir sorgu aşağıda yer almaktadır. SELECT ADI, SOYADI, ADRESİ FROM PERSONEL WHERE BÖLÜMNO = 17 AND GÖREVİ= 'Sekreter';
- Yukarıdaki sorgu ile “17 numaralı bölümde çalışan sekreterlerin ad, soyad ve adreslerinin” bulunmak istendiği kolayca anlaşılmaktadır.

Sorgu İşleyicisi

- VTYS'nin, sorguların işlenmesi ile ilgili görevleri gerçekleştiren bileşenine Sorgu İşleyici (Query Processor) adı verilir.
- Sorgu işleyicinin görevleri:
- Sorgunun sözdizimsel ve anlamsal özümlemesini yapmak.
- Kullanıcının verilen işlemi yapmaya yetkili olup olmadığını denetlemek.
- Sorguyu işletmek için kullanılabilecek algoritmaları (işletim senaryolarını) belirlemek ve “Query Optimizer” alt bileşeni yardımıyla en iyisini seçmek.
- Sorgunun işletimini gerçekleştirdikten sonra yanıtını oluşturup kullanıcıya iletme.

VTYS'nin Temel Bileşenleri (3)



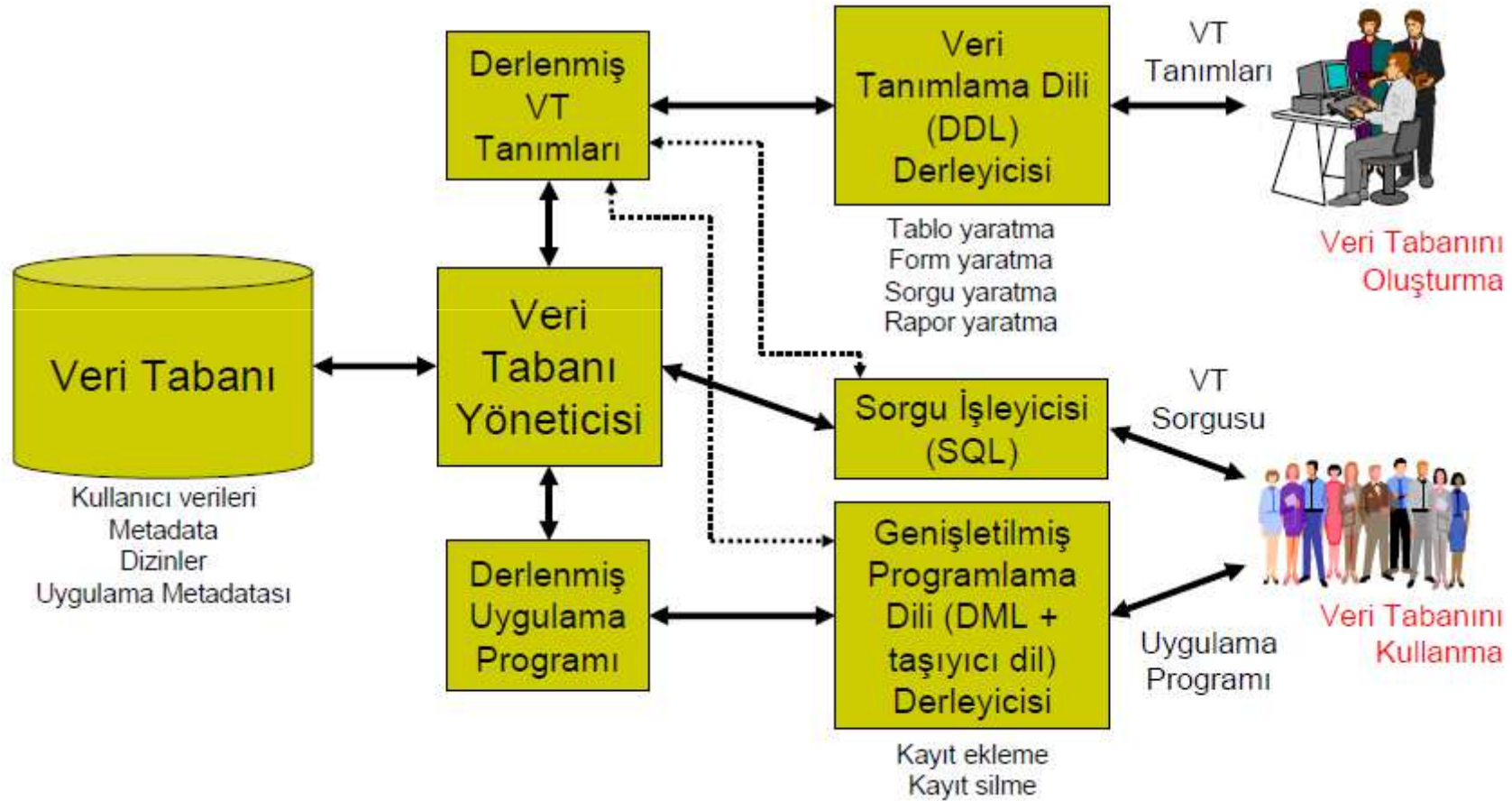
Veri İşleme Dili (DML)

- Veri tabanı üzerinde, veriyi değiştirme, silme ve güncelleme gibi sorgularla ifade edilemeyecek ya da sorgularla ifade edilmesi uygun olmayan işlemler de gerçekleştirilir.
- Bu işlemler için Veri İşleme Dili (DML: Data Manipulation Language) olarak adlandırılan bir dil kullanılır.

Genişletilmiş Programlama Dili

- Veri tabanı üzerindeki uygulamaları gerçekleştirmek için genişletilmiş bir dil oluşturulur.
- Bu dilde hem veri tabanı işlemlerini gerçekleştiren DML komutları, hem de diğer işlemleri gerçekleştiren C, Pascal, Cobol gibi genel amaçlı programlama dillerinden bir ya da birkaçı ile yazılmış taşıyıcı dil komutları yer alır.
- Oluşturulan genişletilmiş dil (DML + taşıyıcı dil) ile hazırlanan uygulama programları, VTYS'nin bileşeni olan genişletilmiş derleyicilerle derlenerek saklanır ve gerektiğinde çalıştırılarak kullanılır.

VTYS'nin Temel Bileşenleri (4)



Veri Tabanı Yöneticisi

Veri Tabanı Yöneticisi (Database Manager), kullanıcı isteklerini veri tabanı üzerinde gerçekleştiren ve çok kullanıcıli ortamdaki işletimsel sorunları çözümleyen yazılımlar bütünüdür.

VTYS'nin gerçekleştirdiği işlemlerin önemli bir bölümü bu katman tarafından gerçekleştirilmektedir

Veri Tabanı Yöneticisi

- Veri Tabanı Yöneticisinin birçok bileşeni vardır. Bunlardan en önemli iki tanesi şunlardır:
- Bellek Yöneticisi (Storage Manager)
- Kütük Yöneticisi (File Manager)
- Tampon Yöneticisi (Buffer Manager)
- Hareket Yöneticisi (Transaction Manager)

Bellek Yöneticisi

- Veri Tabanı Yöneticisinin, verilerin ikincil belleklerde saklanması ile ilgili işlevlerini yöneten kesimdir.
- Bir VTYS'nin Bellek Yöneticisi olarak, VTYS'nin altında çalıştığı işletim sisteminin kütük sistemi (file system) kullanılabilir.
- Ancak büyük boyutlu veri tabanlarını kurmak ve işletmek için kullanılan VTYS'ler için verimlilik çok önemlidir ve gerek ana bellek, gerekse ikincil bellek yönetiminin VTYS tarafından gerçekleştirilmesinde yarar vardır.

Bellek Yöneticisi

- Bellek Yöneticisi aşağıdaki iki bileşenden oluşur:
- Kütük Yöneticisi (File Manager)
- Verinin ikincil bellek kütüklerinde saklanmasını sağlayan ve ana bellek ile ikincil bellek arasında verinin bloklar halinde aktarılmasını sağlayan yazılımlardır.
- Tampon Yöneticisi (Buffer Manager)
- Kütük yöneticisi aracılığıyla ikincil bellekten getirilen veri bloklarını ana bellek sayfalarında saklayan, ana bellek sayfalarını yöneten ve gerektiğinde ana bellek sayfalarındaki veri bloklarının, kütük yöneticisi aracılığıyla ikincil belleklere yazılmasını sağlayan yazılımdır.

Hareket Yöneticisi

- Veri Tabanı Yöneticisinin, veri tabanı yaklaşımının üstünlüklerinden faydalanmasını sağlayan kesimdir.
- Veri tabanı yaklaşımının üstünlükleri arasında:
- Çok kullanıcı ortamda çok çeşitli kullanıcı isteklerinin eşzamanlı gerçekleştirilmesi ve bu arada veri tabanı ve uygulama tutarlılıklarının korunması,
- Veri tabanı üzerinde yapılan işlemlerin yazılım, donanım ya da güç kaynağı ile ilgili bozukluklar nedeniyle yitirilmemesi gibi özellikler bulunmaktadır.

Hareket Yöneticisi

- Hareket Yöneticisinde belirtilen hareket, bir bütün oluşturan ve tutarlılık açısından veri tabanı üzerinde birlikte gerçekleştirilmesi gereken işlemler bütünüdür.
- Tutarlılık açısından, bir hareketi oluşturan işlemlerin ya tümünün gerçekleştirilmesi, ya da hiçbirinin gerçekleştirilmemesi gerekir. Bu Özelliğe hareketin ACID (Atomicity, Consistency, Isolation, Durability) özelliği adı verilir.

Hareket Yöneticisi

- Diğer taraftan, veriler üzerinde değişikliğe neden olan (veri ekleyen, silen ya da verileri güncelleyen) hareketler birlikte işletildiğinde, henüz tamamlanmamış (ve belki de tamamlanmayarak geriye alınacak) bir hareket tarafından gerçekleştirilen değişiklik işlemleriyle oluşturulan veri değerlerinin diğer hareketler tarafından görülmemesi gerekir.

Hareket Yöneticisi

- Hareket yöneticisi, hem hareketlerin ACID özelliğinin bozulmamasını hem de birlikte (eşzamanlı) işletilmelerini sağlar.
- Bunu gerçekleştirmek için de kilitleme (locking), günlük tutma (logging) ve hareket tamamlama (transaction commitment) gibi teknikleri kullanır

Veritabanları Ön Plan Aktörleri

- Veritabanı Yöneticileri (DBA)
- Veritabanı tasarımcıları
- Son kullanıcılar
- Sistem analistleri ve uygulama programcıları

Veritabanları Arka plan Aktörleri

- DBMS sistem tasarımcıları ve uygulayıcıları
- Araç geliştiriciler
- Operatörler ve bakım verenler (network)

- Teşekkürler.
- Sorular?