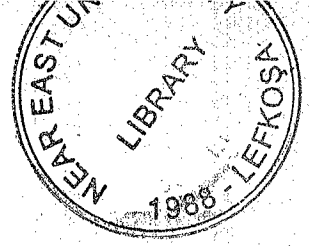


PERRAN ÖZBEN
KÜTÜPHANECİLİK ÖDÜLÜ
YARIŞMASI



VERİ TABANI YÖNETİM SİSTEMLERİ
VE KÜTÜPHANELERDEKİ UYGULAMALARI
İÇİN BİR ÇÖZÜM ÖNERİSİ

AYKUT ARIKAN
İSTANBUL ÜNİVERSİTESİ
EDEBİYAT FAKÜLTESİ
KÜTÜPHANECİLİK BÖLÜMÜ
DOKÜMANTASYON VE ENFORMASYON ANABİLİMDALİ
4. SINIF ÖĞRENCİSİ

İSTANBUL, 1994



İÇİNDEKİLER

YASAL UYARILAR

ÖNSÖZ

0 GİRİŞ

1 VERİ TABANI YÖNETİM SİSTEMLERİ

1.1 VERİ TABANI KAVRAMI

1.2 VERİ TABANI YAKLAŞIMININ AMAÇ VE KAPSAMI

1.3 VERİ TABANI GELİŞTİRME SÜRECİ

1.4 VERİ TABANLARININ TÜRLERİ

1.5 VERİ TABANI YÖNETİM SİSTEMLERİNİN İŞLEVLERİ

1.6 VERİ TABANI UYGULAMA YAZILIMLARI

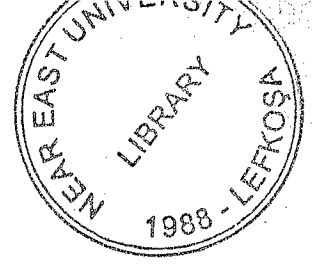
2 KÜTÜPHANELERDE VERİ TABANI YÖNETİM SİSTEMİ UYGULAMALARI VE BİR ÇÖZÜM ÖNERİSİ

2.1 VERİ TABANI YÖNETİM SİSTEMİ UYGULAMALARININ KÜTÜPHANELERE GETİRDİKLERİ YENİLİKLER

2.2 KÜTÜPHANELERDE VERİTABANI YÖNETİM SİSTEMİNE DAYALI UYGULAMALAR İÇİN PARADOX FOR WINDOWS VERİ TABANI YÖNETİM SİSTEMİ ALTINDA GELİŞTİRİLEBİLİCEK BİR ÇÖZÜM ÖNERİSİ

3 SONUÇ

KAYNAKLAR



* YASAL UYARILAR

Bu araştırmanın bütün hakları Aykut Arıkan'a aittir. Araştırma metni, Aykut Arıkan'ın yazılı izni olmaksızın, hiç bir biçimde hiç bir ortama kopyalanamaz, kısmen ya da tamamen çoğaltılamaz; kaynak gösterilmeden metinden alıntı ya da aktarma yapılamaz.

Araştırma da çözüm olarak önerilen "Bölünmüş Bibliyografik Veri Tabanı Yapısı'nın (B.B.V.T.Y.'nin) bütün hakları Aykut Arıkan'a aittir. B.B.V.T.Y.'yi kullanmak isteyen kamuya ait Arşiv, Kütüphane, Belge-Bilgi Merkezleri ve benzeri kuruluşların kar amacı gütmeyen kurum içi bilgi işlem birimleri, söz konusu kuruluşlarda, bunu ancak Aykut Arıkan'ın yazılı izni ile ücretsiz olarak kullanabilirler.

B.B.V.T.Y.'nin bütün ticari hakları Aykut Arıkan'a aittir. Aykut Arıkan'dan yazılı izin alınmaksızın, B.B.V.T.Y.'yi oluşturan dosyalar, veri tabanı yapıları, alanlar, veri yapıları, ve özellikle alanlar arası ilişkiler ya da benzeri B.B.V.T.Y.'ye has unsurlar, alan türleri, alan adları, dosya adları, nesne ya da ilişki adları değiştirilerek veya bunlara her hangi bir ekleme yapılarak dahi ticari yazılım ve ürünlerde kullanılamaz, taklit edilemezler.

B.B.V.T.Y. özellikle Borland International şirketinin PARADOX* FOR WINDOWS** adlı ürününün 4.5 sürümü için tasarlanmıştır. Bu ürün dışındaki veri tabanı yönetim sistemlerinde geliştirilecek olan uygulamalarda meydana gelebilecek olan aksaklık ve sorunların sorumluluğu, ya da uygulama, kullanıcı ve veri girişi hataları Aykut Arıkan'a, ya da Borland International'a ait değildir.

Yukardaki metnin aksine davrananlar hakkında İstanbul Mahkemeleri yetkilidir.

19.02.1994
Aykut Arıkan

* PARADOX ve BORLAND, BORLAND INTERNATIONAL, INC.'E ait tescilli ticari markalardır.

** WINDOWS, WINDOWS CORPORATION'a ait tescilli ticari markadır.

Gifted By

Mesut Yalvaç

Türkiye gün geçtikçe ilerlemekte, kimi kesimleri en son teknolojiye ayak uydurmaya çalışmakta kimileri ise hala ortaçağın karanlığını özlemektedir.

İkinci gruba söyleyecek sözümüz yok, ancak günümüzde değişim, gelişme ve ilerleme kavramlarını benimseyenlerin bu kavramların gereğini yapıp değişimin sadece fiziksel değil düşünsel de olması gerektiğini, gelişmenin sadece yapısal değil, zihinsel de olması gerektiğini ve ilerlemenin doğrusal değil çok boyutlu olması gerektiğini görmeleri gerekiyor.

Amacım burada yeni değerleri savunmak değil; onu yapanlar zaten televizyonda yorumculuk yapıyorlar. Amacım profesyonelliği ve mesleğe saygıyı savunmak.

Bir kütüphaneci düşünün ki son teknolojiyi Türkiye'ye getirmek ve kütüphane alanında uygulamak için 1990'larda 1970'lerin yaklaşımı olan MARC'ı savunuyor. Bir başkasını düşünün ki bir iletişim standardı olan MARC'ı veri tabanı yapısı olmadığı halde, üstelik de veri tabanı yaklaşımı MARC ile kapsam bakımından taban tabana zıt olduğu halde, veri tabanının yapısını MARC biçiminden alıyor.

Mesleğimize saygı duymamız gerekiyor. Profesyonellik ne ayıptır, ne şımarıklıktır ne de burnu büyüktür. Profesyonellik mesleğinde en iyisini yapmaktır.

Eğer mesleğimiz veri tabanı yaklaşımını bilmemizi gerektiriyorsa vakit kaybetmeden öğrenmeliyiz. Bu bakımdan bu araştırmanın getirdiği yeni bakış açısının Türkiye için bir başlangıç olmasını diliyorum.

"Ortaya ortak bir bibliyografik veri tabanı yapısı konursa ulusal düzeyli bir MARC biçimine gerek kalmaz" dediğimde bunu ortaya koymam konusunda bana gerekli motivasyonu veren ve bu araştırma metnini 1994 Perran Özben Kütüphanecilik Ödülü yarışmasına göndermeme neden olan hocam Prof. Dr. Meral ALPAY'a teşekkürü bir borç bilirirm.

Ayrıca araştırma evresinde beni yazılım, belge ve bilgi ile destekleyen Bimeks Bilgi İşlem ve Dış Tic. Ltd. Şti. genel müdürü Doç. Dr. Vedat AKGIRAY'a da bu içten desteği için teşekkür ederim.

19.2.1994
Aykut Arıkan

Bu araştırmanın amacı öncelikle Türkiye'de kuramsal temelsizlik ve bilgi eksikliği nedeniyle uygulamalarda yanlışlara ve aksaklıklara gidilen veri tabanı yönetim sistemleri konusuna kuramsal açıdan genel bir temel sağlamak, uygulamalara yöntem, yaklaşım ve kapsam konusunda bilgi sağlayarak onları desteklemektir.

Bu nedenle araştırmanın ilk bölümünde veri tabanı yönetim sistemleri yaklaşımı tarihsel gelişimi içinde, yöntem ve kuram açısından bir bütün olarak irdelenmiştir.

Kütüphane ve Bilgi bilimleri ile bu kadar yakın olan bir konunun kütüphane alanındaki uygulamaları, getirdiği yenilikler, yanlışlar, eksikler ve yapılması gerekenler ikinci bölümde somut bir örnek çerçevesinde ele alınmıştır. Bir çözüm olarak önerilen Paradox for Windows veri tabanı yönetim sistemi, konuya yatkınlığı açısından tercih edilmiş ve her hangi bir aksaklık ile karşılaşılmamıştır. Ancak bunun ötesine geçip ürün için "en iyisi" demek ne olasıdır ne de tutarlı bir yaklaşımdır.

Araştırma getirdiği çözüm önerisi ile Türkiye için yeni bir yaklaşımın başlangıcı olabilirse amacına ulaşmış sayılacaktır.

1 Veri Tabanı Yönetim Sistemleri

1.1 Veri Tabanı Kavramı

Günümüzde gerek kütüphanecilik gerekse bilgisayar konulu yayınlarda veri tabanı kavramı çoğunlukla kendisinden başka bir kavram olan veri tabanı yönetim sistemi kavramının yerine, sanki onun bir kısaltmasıymış gibi, yanlış olarak kullanılmaktadır. Oysa veri tabanı kavramı, aşağıda da ele alınacağı gibi, özellikle bilgisayar ortamındaki uygulamalarda, verilerin yüklü bulunduğu örgütlü teknik düzenegi (kütüphanecilik bakış açısıyla bir belgeyi) belirtirken, veri tabanı yönetim sistemi kavramı bu düzenek ile işlem yapmaya, onu söz konusu örgütlenme içinde yönetmeye yarayan bir tür uygulama yazılımını belirtmektedir.

İlk bakışta hiç te önemsenmeyen başka bir hata ise, özellikle bilgisayar piyasasında, veri tabanı yönetim sistemi olarak hiç bir ayırım yapılmadan, veri tabanı yönetim sistemleri pazarında "baskın bir eğilimi temsil ettiğinden"¹, değişik türlerdeki veri tabanı yönetim sistemlerinden sadece biri olan ilişkisel veri tabanı yönetim sisteminin kast edilmesidir.

Bu hata halkaları birbirlerine eklenip de ortaya çıkan zincirleme hata yukarıki gibi çözümlendiğinde konunun gerçek boyutları ortaya çıkmaktadır: günümüzde veri tabanı kavramıyla aslında kast edilen, veri tabanı yönetim sistemlerinin bir türü olan ilişkisel veri tabanı yönetim sistemleridir. Bu bakımdan veri tabanı kavramının ortaya çıkmasını sağlayan ve ilişkili diğer kavramların tek tek incelenerek konunun çözümlemeli olarak ele alınması gereklidir:

Veri:

Kavramın İngilizce karşılığı olan "date" [çoğ. data] Latince'deki "dare" [vermek] fiilinin "miş" yapıtı ortacı olan "datum" sözcüğüne dayanmaktadır²; sonuçta "verilmiş olan" gibi bir anlam çıkmaktadır. Dolayısıyla da "veri" sözcüğünün kavramı tam olarak karşıladığı söylenebilir.

- 1 ilişkisel sistemler ve diğerleri. — s. 38-44 . — Bilgisayar . — (Temmuz 1993). — s. 38'de
- 2 American heritage dictionary. — 2nd college ed. — Boston : Houghton Mifflin Company, c. 1982. — s. 366-367'de

Türkçe Sözlük, kavramı "bir araştırmanın, bir tartışmanın, bir usamlamanın [zihinsel bir yargılamanın] temeli olan ana öge" olarak ele almaktadır.

Öte yandan American Heritage Dictionary kavramı "bilgi, özellikle çözümleme için örgütlenmiş ya da bir kararın temellendirilmesi için kullanılan bilgi olarak"⁴ tanımlamaktadır. Bu noktada kavramın en önemli yanlarından biri ortaya çıkıyor: kavram bilgi kavramıyla yakından ilişkilidir.

Veri, zihinsel bir yargılamaya temel olan başlıca ögedir ve sonuçta bilgiden farklı bir şey olmayan (ya da bir tür çıkarsanmış bilgi olan) bir yargı veya kararı ortaya çıkartmaktadır. Yani bilgi, veri olarak adlandırılan başka bilgilerden çıkarsamalar yoluyla türetilir.

Öyleyse veri, işlenerek yeni bilgiler türetilen bilgidir, yani, bir tür işlenmemiş, ham bilgidir. Ancak burada kullanılan "ham bilgi" ifadesi aslında yanıltıcı bir ifade. Konuyu bu noktada örneklerle açmak gerekir:

Örneğin 3 sayısı bir veridir. Aynı zamanda kendi anlamı dışında herhangi bir bilgi vermediğinden ham bir bilgidir. Bu sayı her hangi bir çocuğun yaşı, bir öğrencinin aldığı not ya da bir nesneye ait bir ölçü (en, boy, yükseklik v.b. gibi büyüklükler) olabilir. Sayı diğer her hangi bir nesne ile tanımlanmış bir ilişkiye girdiği zaman ortaya bir bilgi çıkmaktadır. Örneğin "3" sayısı "araba" adlı nesne ile "metre cinsinden uzunluğu olma" olarak tanımlanan ilişkiye girdiğinde ortaya "arabanın metre cinsinden uzunluğu 3'tür" gibi önerme biçiminde bir bilgi çıkmıştır.

Ancak sözgelimi yukardaki önermede anılan "araba" adlı nesne "garaj" adlı kümenin bir ögesi olsun. Aynı kümenin diğer iki ögesi de "kamyonet" ve "otobüs" adlı nesneler olsun ve "kamyonet" adlı nesne "4" sayısı, "otobüs" adlı nesne de "5" sayısı ile "metre cinsinden uzunluğu olma" olarak tanımlanan ilişkiye girdiğinde şu önermelere ulaşılır:

- (1) "arabanın metre cinsinden uzunluğu 3'tür"
- (2) "kamyonetin metre cinsinden uzunluğu 4'tür"
- (3) "otobüsün metre cinsinden uzunluğu 5'tir"

Aslında başlı başına birer bilgi olan bu üç önermeden çıkarasama yoluyla bir dördüncüsüne de ulaşılabilir:

- (4) "garajdaki araçların metre cinsinden uzunluklarının ortalaması 4'tür"

3 Türkçe sözlük. — genişl. 7. bs. — [Ankara] : Türk Dil Kurumu, 1983. — s. 1253'de

4 American heritage dictionary...

İşte ham bilgi kavramının yanıltıcılığı da burada yatar: 1, 2 ve 3 numaralı önermeler aslında 4 numaralı önermeye ulaşmak için çıkarsamaya tutulacak birer veridirler, ancak bunlara ham bilgi demek doğru değildir, çünkü üçüde başlı başına birer bilgidir.

Bu nedenle veri kavramını, işlenmeye hazır bilgi ögesi olarak açıklamak yukarda sözü edilen yanılgıyı ortadan kaldıracığından daha uygundur.

Bilgi:

Burada da bilgi kavramının, Türkçe'nin yapısından kayanaktanan, ve çelişkililere yol açabilecek bir eşseslilik nedeniyle, kendi içindeki iki anlamını ayırmsamak gereklidir:

Bilgi terimi birbirinden farklı iki kavram olan "enformasyon" ve "birikim" (ing. knowledge) kavramlarını karşılayabilmektedir. Ancak bu araştırmada genel kullanım gözönünde bulundurularak "bilgi" terimi doğrudan doğruya "enformasyon" kavramı yerine kullanılmıştır.

Kavramın "veri" kavramı ile olan ilişkisi göz önünde bulundurulacak olursa şu sonuca da ulaşılabilir: "bilgi işlenmeye hazır bilgi ögeleri ya da veriden çıkarsanacak sonuç, varılacak yargıdır".

Veri tabanı:

Yukarda da belirtildiği gibi veri tabanı kavramı özellikle bilgisayar ortamındaki uygulamalarda, verilerin yüklü bulunduğu örgütlü teknik düzenegi (kütüphanecilik bakış açısıyla bir belgeyi) belirtir. Ancak gene de kavramı çeşitli bakış açılarından irdelemekte yarar vardır:

Yazılım pazarındaki en gelişmiş ticari ilişkisel veri tabanı yönetim sistemi yazılımlarından biri olan Paradox, veri tabanı kavramını özellikle kütüphane ve bilgi bilimlerinin yaklaşımlarına koşutluk gösterecek tarzda, "örgütlü bir bilgi ya da veri dermesi" olarak tanımlamaktadır. Bilgi dermesinin örgütlenmesinin bu iki bilim dalının temel alanlarından biri olduğu düşünülecek olursa kavramın özellikle bilgi bilim açısından önemi açıkça

5 Yontar, Aysel. Bilgi bilime giriş [ders notları]. — [İstanbul] : [y.y.], [1992]. — Not : Metin teksirdir. — s. 6'da

6 Paradox for Windows. Getting started. — Scotts Valley, California : Borland International, 1992. — s. 2'de

ortaya çıkmaktadır: kütüphaneler, arşivler ve belge merkezleri, örgütlü bir belge dermesinin yönetildiği sistemlerdir; bilgi merkezleri ise bunun diğer bir boyutu olan örgütlü bilgi dermelerinin yönetildiği sistemlerdir.

Buna göre denebilir ki veri tabanları bilgi merkezlerinin temel unsurudur. Veri tabanı yönetim sistemleri ise bilgi merkezleri ile işlev bakımından koşut hatta aynıdırlar.

Dermenin bilgi mi yoksa veri mi dermesi olduğu sorusunun bu tanımda yanıtlanmamış olması yukarıda da söz edilen veri ve bilgi kavramlarının anlam yakınlığından kaynaklandığı bir gerçektir. Ancak bu iki kavramın ayrımı da göz önünde bulundurularak veri kavramını benimsemek gerekmektedir.

Tanımın çözümlenmesi sonucunda ulaşılan nokta ise veri tabanının bir veri dermesi olduğu, ve ayırd edici özelliğinin ise örgütlenmesi olduğudur.

Ancak önemli olan bir diğer nokta ise yukarıki tanımda veri tabanının fiziksel biçimine değinilmediğidir. Özellikle bilgisayar çevrelerine yapılan önemli bir hata da burada yatmaktadır: kullanılagelenin aksine veri tabanı kavramı özellikle bilgisayara ait bir kavram değildir, yani veri tabanının fiziksel biçimi ille de bilgisayar ortamı değildir. Örgütlü olan her biçimdeki, her türden ve her ortamdaki veri dermesi bir veri tabanıdır: örneğin bir telefon rehberi, bir sözlük, her hangi bir çizelge özellikle de "kütüphanelerin kart katalogları" hep birer veri tabanıdırlar.

Kütüphane otomasyonu konusundaki araştırmaları ve çalışmalarıyla uluslararası düzeyde bir üne sahip olan Robert M. Hayes veri tabanı kavramını "kullanıma uygun, makine dilindeki bir veri dosyası" olarak tanımlamaktadır. Kullanıla gelen bilgisayar ortamındaki veri tabanı kavramının biçimsel temelidir bu: veri tabanı makine diline uygun bir dosyadır, yani bir bilgisayar dosyasıdır.

Yetkin danışma kaynaklarından American Heritage Dictionary ise veri tabanı kavramını şöyle tanımlamaktadır: "örneğin bilgisayarda olduğu gibi kolay ve hızlı bilgi erişimini sağlayacak şekilde düzenlenmiş veri topluluğu".

7 Paradox for Windows...

8 Hayes, Robert Mayo. Handbook of data processing for libraries / Robert M. Hayes. — 2nd ed. — Los Angeles : Melville Publishing Company, 1974. — s. 656'da

9 American heritage dictionary... — s. 366'da

Tanımda ilk göze çarpan nokta Paradox'un tanımına koşut olarak, bilgisayar dosyasının veri tabanının fiziksel biçimlerinden sadece biri olduğudur. Ancak bilgisayar ortamının hızlı ve kolay bilgi erişimini sağlaması bu tanımın asıl önemli yanıdır. Bu hız ve kolaylığı ise söz konusu veri dermesinin düzeni, yani veri tabanı örgütlenmesi sağlamaktadır.

Veri tabanının örgütlenme düzenini ise yazılım piyasasındaki ticari veri tabanı yönetim sistemi yazılımlarının özellikle 80'li yıllarda en yaygın olarak kullanılanlarından biri olan, Dbase 3 Plus adlı yazılım, veri tabanı kavramına önerdiği tanımda şöyle açıklamaktadır: "bir çok varlık hakkındaki aynı türden bilgi"¹⁰. Yani bir veri tabanı dosyası yapısında birden çok varlık hakkında bilgi vardır, ve bu bilgi hepsi için aynı düzendedir: örneğin değişik elementlere ait bir veri tabanında her elementin atom numarası, özgül ağırlığı, elektron sayısı v. b. gibi elementlerin doğasını açıklayan bilgiler bulunabilir. Bu bilgilerin düzeni ise tüm elementler için aynıdır, yani her elementin atom numarası, özgül ağırlığı v.b. bu veri tabanında yer almaktadır.

Bu tanım ilerde de ele alınacağı gibi söz konusu yazılımın doğası gereği özellikle ilişkisel veri tabanlarının yaklaşımına koşutluk göstermektedir. Ancak gene de veri tabanlarının örgütlenme düzeni hakkında bir fikir verdiğinden kayda değerdir.

Veri tabanı kavramıyla sıkça karıştırılan bir kavram ise veri bankası kavramıdır. İki kavram arasındaki ayrım özellikle son zamanlarda ortadan kalktığından her iki kavram veri tabanı terimi altında birleşmiştir. Bunun nedenini veri bankası kavramına Hayes'in önerdiği tanımla açıklamak olasıdır. Hayes veri bankasını "genellikle çok büyük ölçekte olup, çok kullanıcı amaçlı olan bir veri tabanı"¹¹ olarak tanımlamaktadır. Buna göre veri bankası bir tür veri tabanıdır, ancak veri tabanından ayrıldığı nokta nicel olarak daha büyük olması, yani daha çok öge hakkında veri içermesi, ve birden fazla kullanıcı tarafından erişilebilir ve işlenebilir olmasıdır.

Ancak geçen zaman içinde gelişen veri tabanı teknolojisi bu ayrımı ortadan kaldırmıştır. Artık neredeyse her veri tabanı birden fazla kullanıcı tarafından erişilebilir ve işlenebilir durumdadır; ve örneğin Paradox for Windows ile yaratılan veri tabanı dosyalarına iki milyar adet civarında öge hakkında veri girilebilmektedir. Bu nedenle de veri tabanı terimi veri bankası kavramını da artık içerdüğinden

10 Dbase 3 Plus. Introduction to Dbase 3 Plus [bilgisayar dosyası]. — [y.y.] : Ashton-Tate, [1986?]

11 Hayes, Robert Mayo...

özellikle günümüzde kullanılan veri tabanları ile veri bankası kavramının ayrımı ortadan kalkmıştır. Buna koşut olarak da veri bankası teriminin bir geçerliliği kalmamıştır.

Bu arada veri tabanı terimi kimi zaman da belirli konularda veri içeren bilgi sistemi anlamında da kullanılmaktadır. Terimin bu yanlış kullanımının yaygınlığına karşın ortaya çıkabilecek yanlış anlaşılmalara önlenmesi için veri tabanı ve bilgi sistemi terimlerinin birbirlerinden ayrılmasında yarar vardır.

Veri Tabanı Yönetim Sistemi:

Veri tabanı yönetim sistemi veri tabanı kavramının aksine özellikle bilgisayara yönelik bir kavramdır. Tanımı fazla tartışmalı olmayan kavram kısaca bilgisayar ortamındaki veri tabanlarını yaratarak, biçimlendirerek, değiştirerek, onlara eklemeler ve çıkartmalar yaparak hepsinden önemlisi sorgulayarak yöneten, bilgisayar işletmeninin veri tabanından yararlanmasını sağlayan, çoklukla ticari bir genel uygulama yazılımı olarak tanımlanabilir.

Burada üstünde titizlikle durulması gereken bir nokta vardır: Aslında çoğu özel amaçlı uygulama yazılımı bir tür özel veri tabanı yönetim sistemidir. Bunların genel amaçlı veri tabanı yönetim sistemlerinden ayrılan yanı, özel amaçlı uygulama yazılımlarında kullanılan veri tabanı yapılarının belirli ve çoklukla değiştirilemez olması ve yapılabilecek işlemlerin de buna koşut olarak belirli ve kısıtlı olmaları, ancak genel amaçlı veri tabanı yönetim sistemlerinde bunun aksine gerek veri tabanı yapılarının gerekse işlemlerin çeşitlerinin sınırsız olmaları, yani tamamen kullanıcının gereksinimlerine göre düzenlenebilir olmalarıdır.

Veri Tabanı Uygulama Yazılımı:

Veri tabanı uygulama yazılımları veri tabanı yönetim sistemlerinde kimi özel uygulamalarda sıkça tekrarlanan yordamları oluşturan komutların kullanıcı tarafından belirlenen üst başlıklar altında bir araya getirilmesinden oluşan yazılımlardır. Bu yazılımların tek amacı veri tabanı yönetim sistemleri ile gerçekleştirilen belirli işlemleri kolaylaştırmak ve tümleşik bir yapıya oturtmaktır.

Bu tür yazılımların veri tabanı yönetim sistemlerinin işleyişinden hiç bir farkları yoktur. Belirlenmiş olan komutlar veri tabanı yönetim sistemine ait komutlardır. Uygulama yazılımı işlerken kullanıcı aslında veri tabanı yönetim sisteminin işlediğinin ayrımında değildir ve sistemi uygulama yazılımının olanakları içinde kendi ihtiyaçlarına göre, ayrıntılı olarak yapılan işlemleri bilmeksizin ve görmeksizin, yönlendirir.

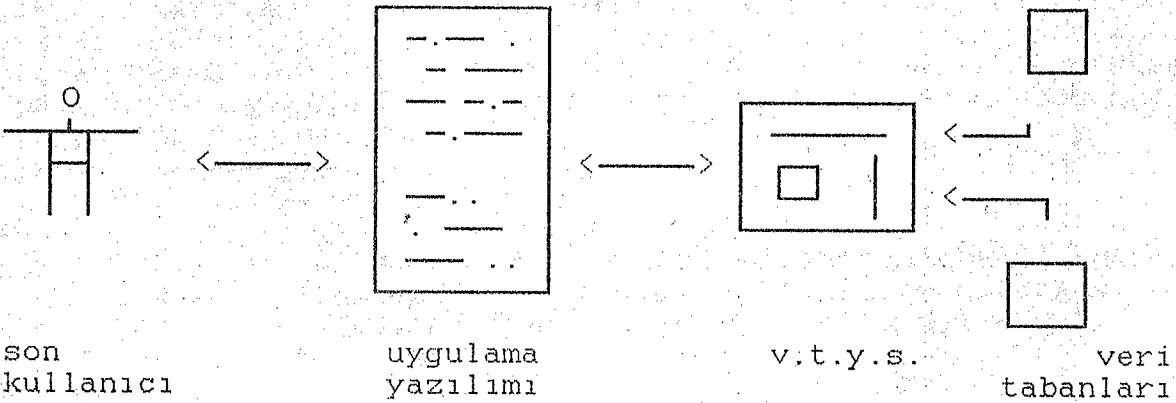
Uygulama yazılımları çoklukla kendi başlarına çalışmazlar; ancak bir yorumlayıcı yazılım, yani veri tabanı yönetim sistemi altında işlerler. Buna karşın son zamanlarda bu tür yazılımları kendi başlarına işler biçime dönüştürecek derleyiciler de, fazla yaygınlık kazanmamakla beraber, pazarda yerlerini almışlardır.

Özellikle yoğun bir bilgi işleme ihtiyacı olan işletmelerde bu tür uygulama yazılımlarının derlenmeden veri tabanı yönetim sisteminde yorumlatılarak kullanılması daha akılcı bir çözümdür, çünkü derlenmemiş bir yazılımın kaynak kodlarına girerek zaman içinde gerek duyulabilecek değişikliklerin, ekleme ve çıkartmaların yapılması olasıdır, ancak bu, derlenmiş olan yazılımlar için, yazılımın kaynak kodları makine koduna çevrilmiş olduğundan, söz konusu değildir. Yani derlenmiş bir yazılımda yapılabilecek işlemler, kurulabilecek olan veri tabanı dosyalarının yapıları, belirli ve sabitken, kaynak kodundaki derlenmemiş yazılımlarda bunları değiştirme olanakları sınırsızdır.

Bu kavramların çözümlenmesinden şu sonuçlara ulaşılabilir:

1. Veri tabanı örgütlü bir veri dermesini içeren bir bilgisayar dosyasıdır.
2. Veri tabanı yönetim sistemi veri tabanı dosyalarını yönetmeye yarayan genel amaçlı ticari yazılımdır.
3. Veri tabanı uygulama yazılımları veri tabanı yönetim sistemlerinde kimi özel uygulamalarda sıkça tekrarlanan yordamları oluşturan komutların kullanıcı tarafından belirlenen üst başlıklar altında bir araya getirilmesinden oluşan yazılımlardır.

Bu sonuçları Sauer aşağıkine benzer bir çizimle ifade etmiştir¹²:



12 Sauer, Hermann. Relationale Datenbanken : Theorie und Praxis, inklusive SQL-2 / Hermann Sauer. — 2. korrig. u. erw. Aufl. — Bonn... [ve öte.] : Addison-Wesley, 1992. — s. 11'de

Veri tabanı yaklaşımı 1960'ların sonlarına doğru¹³, gelişmekte olan bilgisayar teknolojisinde anlık işlemler ve hesaplamalar yerine belli mantıkal yapılar içinde saklanacak bilgilere dayanarak yüksek düzeyli, karmaşık ve her şeyden önce planlamaya/karar vermeye veya denetimi sağlamaya yönelik olarak¹⁴ sorgulamaların yapılabileceği sistemler anlayışının ağırlık kazanmasından sonra, gündeme gelmeye başlamıştır.

Veri tabanı düşüncesinin o zamana kadar geçerliliğini korumuş olan "geleneksel dosya yaklaşımı"ndan¹⁵ temel farkı her hangi bir konudaki verinin "ayrı dosyalarda saklanan bağımsız birimlerde değil de tek birimde"¹⁶ saklanması düşüncesidir. Bu düşünce doğrultusunda ortaya çıkan sonuç ise verinin derlenerek belli bir örgütlenme içinde tek dosyada, kısaca aynı belli tabanda, yani yukarda da sözü edilen veri tabanı dosyasında saklanmasıdır.

Burada özellikle belirtilmesi gereken nokta geleneksel dosya yaklaşımında verinin saklandığı birbirlerinden bağımsız dosyaların kendilerini yönetecek olan yazılıma fiziksel olarak sıkı sıkıya bağımlılıklarıdır. Bu yaklaşımın en büyük yitirimi aynı verinin ayrı işlevleri yüklenmiş olan değişik yazılımlarla birlikte, yazılımlar ancak birbirlerinin ardı sıra çalıştırılabildiğinden, gereksiz yere tekrar edilmesidir¹⁷. Oysa devrim niteliğini taşıyan veri tabanı yaklaşımında, aynı veri tabanı dosyası ile birden çok uygulama yapılabilmektedir. Sözelimi geleneksel dosya yaklaşımında kütüphane yazılımları sağlama, kataloglama, dolaşım ve işbirliği gibi ayrı yazılımlardan oluşmaktadır. Sağlamaya ait veri sadece sağlama yazılımında kullanılabildiğinden aynı veri kataloglama yazılımı için bilgisayara tekrar yüklenmekte, eğer kataloglanan kitabın verisi dolaşım ya da işbirliği yazılımlarında kullanılacaksa aynı veri tekrar tekrar bilgisayara girilmekte gereksiz yere bellek harcanmaktadır. Oysa veri tabanı yaklaşımında katalog verisi sadece katalog veri tabanında saklanmakta, sağlama, kataloglama, dolaşım ve işbirliği yazılımları, ya da çağdaş programlama yaklaşımıyla yordamları, bu veritabanı dosyasını ortaklaşa kullanmaktadırlar.

13 Sauer... — s. [1]'de

14 Oxborrow, Elizabeth A. Databases and database control systems : concepts and issues / Elizabeth Oxborrow. — 2nd ed. — Bromley, Kent, England : Chartwell-Bratt, c. 1989. — s. 3'de

15 Oxborrow, Elizabeth A...

16 Oxborrow, Elizabeth A...

17 Sauer... — s. 13'de

1975'te yayınlanan, ANSI/SPARC Modeli olarak da anılan konuya ilişkin standart, veri tabanı yaklaşımını birbirleri ile dikey bağlantılı üç temel katmanda ele almaktadır¹⁸:

1. Dış katman	Kullanıcı bakışı	uygulama yazılımı
2. Kavramsal katman	Mantıksal genel bakış	v.t.y.s.
3. İç katman	Fiziksel bakış	veri tabanı

ANSI/SPARC Modeli'ne göre bu üç katman da birbirlerinden fiziksel olarak tamamen bağımsız birimler olmak zorundadırlar¹⁹. Böylelikle verinin uygulama yazılımından tamamen bağımsız olması sağlanmış, kavramsal katman olarak ele alınan veri tabanı yönetim sistemi ile kullanıcıya mantıksal açıdan bir genel bakış sağlanarak buradan fiziksel olarak ayrı yerleşimlere sahip olan değişik veri tabanı dosyalarına, kurulan mantıksal bağlantılarla, merkezi erişim olanağı verilmiştir.

Kurulan mantıksal bağlantılarla sağlanan bir diğer olanak ise, uygulama yazılımlarının son kullanıcıya veri tabanı yönetim sisteminin olanaklarını, veri tabanı yönetim sisteminden bağımsız olarak sağlamasıdır.

ANSI/SPARC Modelinin özellikle Türkiye'ye sağladığı, yabancı literatürde gereksinim duyulmadığı için hiç ele alınmayan, çok önemli başka bir olanak ise son kullanıcıyı dil sorunundan kurtarmasıdır. Yazılım pazarında kullanıcıya sunulan veri tabanı yönetim sistemleri, sözcük işlemciler, hesap tabloları v.b. genel amaçlı uygulama yazılımlarının büyük bir bölümü, yurt dışında üretildiklerinden, Türkçe değildirler. Bu da son kullanıcıyı bu uygulama yazılımlarından yararlanabilmek için yabancı dil, özellikle de İngilizce bilmek zorunda bırakmaktadır. Ancak ANSI/SPARC Modeline göre geliştirilecek olan özel uygulama yazılımının dili Türkçe olabileceğinden bu yazılımı herkes kullanabilecektir.

18 Sauer... — s. 12

19 Sauer...

Bunun en güzel kanıtı yazılım pazarında son kullanıcıya sunulan özel amaçlı Türkçe uygulama yazılımlarının bazılarının derlenerek veri tabanı yönetim sisteminden dil bakımından tamamen bağımsız bir biçime çevrilmiş yazılımlar olmalarıdır.

Veri tabanı yaklaşımının getirdiği başlıca yenilikler ise özetle şöyle sıralanabilir:

1. Gereksiz tekrarlar önlenmiş, bellek tasarrufu sağlanmıştır.
2. Değişik veri tabanı dosyalarına veri tabanı yönetim sisteminden merkezi erişim sağlanmıştır.
3. Merkezi denetim sonucu veri güvenliği sağlanmıştır.
4. Veri tabanı yönetim sisteminden merkezi erişim sağlanarak basit işlemler için yazılım geliştirme gereği ortadan kalkmıştır.
5. Veri tabanı yapısı nedeniyle erişim hızı artmıştır.
6. Veri tek dosya yapısına yüklendiğinden verilerin arasındaki tutarsızlıklar tamamen ortadan kalkmıştır.
7. Veri dosyalarının yapısında standartlaşma sağlandığından özellikle kütüphaneler gibi kuruluşların arasında veri alışverişi kolaylaşmış, birbirlerinden bağımsız sistemler telefon hattıyla birbirlerine veri aktarmanın ötesinde birbirlerinin veri tabanlarına erişerek, hatta onlarla işlem yaparak, zaman ve veri taşıma masraflarından büyük kazançlar sağlanmıştır.

1.3 Veri Tabanı Geliştirme Süreci

Oxborrow çeşitli araştırmacılarca önerilen veri tabanı geliştirme yöntemlerini özet olarak aşağıda amaçları ve sonuçta elde edilmek istenen son ürünleri belirtilen beş alt süreçten oluşan şu sürece oturtmuştur²⁰:

1. Veri araştırması:

Amaç : Verinin doğasının ve kullanımının tanımlanması
Son ürün : İskelet, veri sözlüğü

2. Veri örnekleme:

Amaç : Gerçekliğe ait olguların kendilerini yansıtacak bir örneğe yüklenmesi
Son ürün : Genellenmiş mantıksal örnek

3. Veri tabanı tasarımı:

Amaç : Mantıksal örneğin v.t.y.s.'ne özgü veri
örneğine yansıtılması
Son ürün : v.t.y.s.'ne özgü mantıksal örnek

4. Veri tabanı işletimi:

Amaç : Olgusal verinin fiziksel veri tabanına
yüklenmesi
Son ürün : İç katman ve veri tabanı dosyası

5. Veri tabanı ayarlaması:

Amaç : Veri tabanının kullanımının düzenlenmesi ve
gerek duyulan durumlarda veri tabanının
yeniden yapılandırılması
Son ürün : Elverişli veri tabanı

Oxborrow tarafından oraya konan, sistem analizi sürecini andıran bu süreci ortaya çıkaran alt süreçleri incelemek konuya yaklaşımı daha da anlaşılır kılacaktır.

1.3.1 Veri Araştırması

Veri araştırması alt süreci her yönden veri tabanı geliştirme sürecinin en önemli alt sürecidir. Bu önemin nedeni gerçekleştirilecek veri araştırmasının tutarlılığı, doğruluğu, konuya ve amaca uygunluğu, gerçekliği yansıtırma derecesi v.b. niteliklerinin veri tabanının niteliklerini, verimini ve etkinliğini doğrudan etkileyecek olmasıdır. Yukarıda da ele alındığı gibi amacı verinin doğasının ve kullanımının tanımlanması olan veri araştırması, bu nedenle kesin olarak geliştirilmek istenen veri tabanı hangi konuda ise o konunun, araştırılacak verinin sistem içindeki konumunu saptayabilecek sistem analizi bilgisi olan, uzmanınca gerçekleştirilmelidir.

Araştırılarak derlenecek "bilgi, sistem içinde yer alacak nesnelere özgü ayrıntıları, nesnelere yönelik kısıtlamaları, birbirleriyle kurdukları ilişkileri, fiziksel niteliklerini v.b.'ni içermelidir"²¹.

Buna ek olarak kullanılacak verinin sistem içinde nasıl işleneceği de belirlenmelidir²². Böylelikle bir sonraki alt

21 Oxborrow...

22 Oxborrow... — s. 18'de

süreçte belirlenecek olan veri örneğinin sisteme uygunluğunun sağlanmasına da olanak tanınacaktır.

Bu alt süreçte derlenecek olan bilgi, ya da yukarda da belirlenmiş olan iki kavramın ayrımı göz önünde bulundurulur. veri, veri tabanında yer alacak verinin doğasına özgü veridir. Bu tür veri ise metaveri (ing. metadata), yani veri ötesi olarak adlandırılmaktadır²³.

Sonuç olarak ortaya çıkacak olan iki son üründen ilki veri tabanının iskeletinden başka bir şey olmayacaktır. Diğer ise veri sözlüğü olarak adlandırılan veri tabanı yönetim sisteminin veriyi tanımasını sağlayacak olan ve doğrudan doğruya metaveriyi içerecek olan üründür²⁴. Veri sözlüğü ikinci alt süreçte oluşturulacak mantıksal veri örneğinin de yapısını belirleyecektir.

1.3.2 Veri örnekleme

Veri örnekleme alt sürecinin amacı veri araştırmasında derlenen veriyi, yani metaveriyi, gerçekliğin mantıksal bir örneğine yansıtmaktır.

Veri örneklemesinin temellerinin neo-pozitivist felsefe akımının üyelerinden Ludwig Wittgenstein'in (1889-1951) bilgi kuramı (epistemoloji) konusundaki çalışmalarında yatabileceği konusunda bazı spekülasyon göstermeler dışında şu anda her hangi bir bilimsel kanıt olmamasına karşın, her iki konunun birbiriyle örtüşme derecesinde ilişkili olmaları tartışmasız bir gerçektir.

Wittgenstein yedi temel felsefi önerme ve bunların alt önermelerinden oluşan "Tractatus logico-philosophicus"²⁵ adlı incelemesinde önce gerçekliğin ve olguların ne olduğunu tartışır.

Sonuçta şu önermeye ulaşır:

"Bizler olguların resimlerini yaparız." (önerme 2.1).

Buna bağlı olarak da "resmin gerçekliğin bir örneği" olduğunu belirler (önerme 2.12).

23 Oxborrow...

24 Oxborrow...

25 Wittgenstein, Ludwig. Tractatus logico-philosophicus = Logisch-philosophische Abhandlung / Ludwig Wittgenstein. — Frankfurt am Main : Suhrkamp Verlag, c. 1963, 1989. — s. 16-18'de

Wittgenstein daha sonra şu önermeyi saptar:

"Nesneleri resimde resmin ögeleri karşılar" (önerme 2.13).

Buna göre gerçekliğe ait nesneleri mantıksal örnekte örneğin ögeleri karşılamaktadır.

"Resim, ögelerinin birbirleriyle belli bir tarz ve biçimde ilişki kurmalarından oluşur" (önerme 2.14).

Yani mantıksal örnek kendisini oluşturan ögelerin belli bir tarz ve biçimde birbirleriyle ilişki kurmalarından oluşur. Wittgenstein saptamalarına devam eder:

"Ögelerin bu ilişkisi resmin yapısını, [resmin] olasılığı da resimlemenin biçimini ifade eder" (önerme 2.15).

"Buna göre resme bir de onu resim yapan resimleyen ilişki aittir" (önerme 2.1513).

"Resimleyen ilişki nesneleri resmin ögelerinin karşılamasından oluşur" (önerme 2.1514).

"Olgu resim olabilmek için resimlenen ile ortak bir şeye sahip olmalıdır" (önerme 2.16).

"Resim ve resimlenende, birinin diğerinin resmi olabilmesi için, bir şeyin özdeş olması gerekir" (önerme 2.161).

"Resmin, gerçekliği kendi tarz ve biçiminde -doğru ya da yanlış- resimleyebilmesi için, gerçeklik ile ortak sahip olması gereken şey resimlemenin biçimidir" (önerme 2.17).

"Hangi biçimde olurlarsa olsunlar, bütün resimlerin gerçekliği -doğru ya da yanlış- resimleyebilmeleri için, gerçeklikle ortak olarak sahip olmaları gereken şey, mantıksal biçimdir, ki bu da gerçekliğin biçimidir" (önerme 2.18).

Kısacası mantıksal örneğin yapısı doğrudan doğruya gerçekliğin mantıksal biçimini yansıtmalı, hatta ona koşut olmalıdır. Bu noktada veri örneğini oluşturan ögeleri tanımakta yarar vardır:

1.3.2.1.2 Varlıklar ve Yüklemler

"Varlıklar gerçeklikte var olan, bizim ilgimizi çeken çeşitli nitelikleri taşıyan nesnelerdir. Yüklemler ise bir varlığın niteliklerinin her biri olarak tanımlanabilirler."

Varlıklar Wittgenstein'in ele aldığı gerçekliğin nesneleridirler.

1.3.2.1.3 "Domain" ve Tanımlayıcılar

"Domain" sözcüğünün ne yazık ki tam bir Türkçe karşılığı yok. Ancak varlıkların ve yüklemelerin ait oldukları özel sınıfın anlamına geldiği söylenebilir. "Domain"ler özellikle veri tabanı geliştirme sürecinin bir sonraki alt süreci olan veri tabanı tasarımı verinin doğasının belirlenmesinde büyük önem taşırlar; yani veri bir tarih mi olacaktır, bir sayı mı yoksa bir kişi adı mı, v.b.

Tanımlayıcılar ise gerçekliğin nesnelerinin mantıksal örnekteki soyut karşılığı olacaklardır.

1.3.2.2 Veri Örneğinin Saptanmasının Aşamaları

Veri örneği tüm bu öğelere dayanılarak şu akış içinde saptanır:

1. Varlık kümeleri ve ilişkiler tanımlanır
2. Yüklemler tanımlanır
3. İlişki kümeleri tanımlanır
4. Tanımlayıcılar saptanır
5. "Domain"ler belirlenir
6. Bütün bu öğeler bir araya getirilerek tam mantıksal örnek kurulur

1.3.3 Veri Tabanı Tasarımı

Veri tabanı tasarımı yukarıda kurulan mantıksal veri örneğinin, oluşturulmak istenen veri tabanının geliştirileceği veri tabanı yönetim sisteminde yeni baştan tasarlanması, sistemin olanakları göz önünde bulundurularak mantıksal sistemin veri tabanı yapısına uyarlanmasıdır. Yukarıda da belirtildiği gibi elde edilecek son ürün veri tabanı dosyasının bilgisayar ortamındaki mantıksal yapısıdır.

27. Oxborrow... — s. 25'te

28. Oxborrow... — s. 26'da

29. Oxborrow... — s. 27-37'de

1.3.2.1 Veri Örneğinin Öğeleri

1.3.2.1.1 İlişkiler

Çeşitli araştırmacılar tarafından otaya konmuş olan ilişki gruplamaları vardır. Ancak bunların ötesinde veri örnekleme için etkili olabilecek nokta ilişkilerin değişik türleridir.

İlişki türlerinin iki belirleyici özelliği vardır: ilişkinin derecesi ve varlığı²⁶.

İlişkinin derecesi ya tanımsızdır, ya bire bir ilişki söz konusudur, ya bire çok ilişki ya da çoka çok ilişki.

Tanımsız ilişkiler bu araştırmanın konusu dışıdır.

Bire bir ilişkiye verilebilecek en güzel örnek öğrenci öğrenci numarası ilişkisi. Her öğrencinin sadece bir öğrenci numarası vardır ve her öğrenci numarası sadece bir öğrenciye aittir.

Bire çok ilişki anne (ya da baba) ile çocukları arasında vardır. Her çocuğun mutlaka bir annesi (ya da babası) vardır, ancak her annenin (ya da babanın) birden çok çocuğu olabilir.

Çoka çok ilişki ise öğrenciler ile dersler arasında vardır. Her öğrenci birden çok dersi alır ve her dersi birden çok öğrenci alır.

İlişkinin varlığı ise ya zorunludur, ya seçmeli ya da olası.

Zorunlu ilişki gene öğrenci öğrenci numarası ilişkisiyle açıklanabilir. Her öğrencinin sadece bir öğrenci numarası olmalıdır ve her öğrenci numarası sadece bir öğrenciye ait olmalıdır.

Seçmeli ilişki ise öğrenci ile yüksek okul arasında vardır: her öğrenci bir yüksek okula gidebilir ve her yüksek okula değişik öğrenciler gidebilir. Görüleceği gibi burada bir zorunluluk değil, aksine bir seçim vardır.

Olası ilişki bunların en karmaşık olanıdır. Burada bir taraf için bir zorunluluk varken, diğeri için seçim vardır: her kitabın mutlaka bir yazarı vardır ancak her yazarın mutlaka bir kitap yazmış olması gerekmez.

26 Oxborrow... — s. 23'te

1.3.4 Veri Tabanı İşletimi

Veri tabanı işletimi alt sürecinde, veri tabanında dermesi oluşturulacak veri, artık fiziksel olarak bilgisayar ortamındaki veri tabanına girilmeye başlanır. Bu süreç içinde gerekli görülen girdi düzeltmeleri yapılır. Son ürün olarak ise veri tabanı dosyası elde edilir.

Bu arada geliştirilebilecek diğer bir şey ise bu dosyaya ait sorgulama yordamlarıdır.

1.3.5 Veri Tabanı Ayarlaması

Bu alt süreçte artık veri tabanının yapısına ilişkin son ayarlamalar yapılır. Bu aşamada yapıya ufak tefek müdahaleler yapılabilir gibi başlı başına yeniden yapılandırma çalışmalarına da gidilebilir.

Veri tabanı geliştirme süreci aslına bakılacak olursa hiç bitmeyecek bir süreçtir. Zaten veri tabanının yapısının ve içeriğinin güncel şartlara ve gereksinimlere uyum sağlayabilmesi için en sağlıklı yaklaşım da bu doğrultudadır.

1.4 Veri Tabanlarının Türleri

Veri tabanları örgütlenme yapıları açısından zaman içinde farklılıklar göstermişler ve böylece de veri tabanlarının değişik türleri ortaya çıkmıştır. Ancak tarihsel açıdan konuya yaklaşıldığında, yaklaşım olarak veri tabanı düşüncesinin değişmediği, aksine değişimin yapısal nitelikli olduğu görülür. Bu yüzdendir ki yeni bir tür ortaya çıktığında eskisi etkisini yitirmektedir.

Veri tabanlarında yaşanan bu yapısal değişikliklerin başlıca nedeni veri tabanının yapısının giderek gerçekliğin bir örneğinin bilgisayar ortamındaki yansıması olmaktan, uygulama amacına yönelik olarak yeniden biçimlendirilmiş bir yapıya yönelmesidir. Örneğin ilk türlerden olan hiyerarşik veri tabanlarında veri tabanı yapısı uygulama (özellikle de sorgulama) kaygısı taşımaksızın, gerçekliğin felsefi anlamda tamamen koşut bir yansımasıdır. Ağ yapılı veri tabanlarında bu yapı biraz daha uygulamaya yönelik bir biçime sokulmaya çalışılmış, ancak elde edilen hiyerarşik veri tabanının yapısındakinden biraz daha büyük bir karmaşa olmuştur.

Veri tabanı dünyasında ilk devrim niteliğini taşıyan gelişme yapı bakımından diğerlerinden oldukça yalın, özellikle sorgulamalarda büyük kolaylıklar sağlayan ilişkisel veri tabanlarının ortaya çıkmasıdır.

İlişkisel veri tabanlarındaki kimi aksaklıklar ve kısıtlamaların giderilmeye çalışılmasıyla da özellikle 90'lı yıllara damgasını vuracak gibi görünen nesne yönelimli veri tabanları ortaya çıkmıştır.

Bu noktada tarihsel açıdan başlı başına eğilimleri temsil eden bu dört türü özet niteliğinde genel olarak ele almak, diğerlerine de sadece isim olarak da olsa değinmekte yarar vardır.

1.4.1 Hiyerarşik Veri Tabanları

Oldukça karmaşık bir yapıya sahip olan hiyerarşik veri tabanlarında veri tabanının yapısı, bir kökteki yüklemle bir ağacın dallarını andıracak biçimde³¹ hiyerarşik olarak ilişkilendirilmiş başka yüklemelerden oluşur. Kendi için de tutarlı olan bu yapılanmada yüklemeler başka bir köke de ait yüklemeler oldukları zaman, yani birden çok hiyerarşik yapı içinde yer almaya başladıkları zaman, veri tabanının yapısı doğrudan doğruya ağ yapısına dönüşür.

1.4.2 Ağ Yapılı Veri Tabanları

Ağ yapılı veri tabanları özellikle CODASYL (COMmittee on DATA SYstems Languages = Veri Sistemleri Dilleri Komitesi) adlı gurubun çalışmaları sonucunda 1960'ların sonlarında ortaya çıkmıştır³².

Ağ yapılı veri tabanında veri tabanının fiziksel örgütlenmesi yüklemeler ve varlıkların ilişkilerinden kurulan kümelerin bir ağ biçiminde bir araya getirilmesinden oluşmaktadır. Böylesine karmaşık bir yapı beraberinde karmaşık yönetim işlemlerini de getirmekte, bu da sistemin uygulamalarda zorluk çıkarmasına neden olmaktadır³³.

Gerek ağ yapılı gerekse hiyerarşik veri tabanlarında karşılaşılan en önemli sorun ise bu karmaşık yapının öğelerinin manyetik ortamda fiziksel olarak değişik adreslere dağıtılmış olmaları ve bu nedenle ileride yapılmak istenebilecek bir yeniden yapılandırmanın çok güç olmasıdır³⁴.

31 İlişkisel sistemler ve diğerleri... — s. 38'de

32 Oxborrow... — s. 68'de

33 Oxborrow... — s. 68-77'de

34 Sauer... — s. 14-15'te

1.4.3 İlişkisel Veri Tabanları

İlişkisel veri tabanı düşüncesi E. F. Codd'un 1970 yılında ortaya koyduğu kuramsal yaklaşımın bir ürünüdür. İlişkisel veri tabanlarının örgütlenmesi tablolar biçiminde gerçekleşir.

Örgütlenme yapısı ilişkisel öğeler, ilişkisel bütünlük kuralları ve ilişkisel işlemlerin tanımlarıyla belirlenmiştir.

İlişkisel öğeler kısaca şöyle ele sıralanabilir:

"domain"ler
ilişkiler
derece
yüklemler
tutanaklar
birincil anahtar
ikincil anahtar
aday anahtar
yabancı anahtarlar

Bunlar bir tabloya yansıtılacak olursa:

Birincil anahtar

yüklem /\		
öğrenci #	Soyadı	Adı
9008DE005	Arıkan	Aykut
9008KÜ031	Doğanay	Hakan
9008KÜ054	Ekici	Aynur

} 3 değişik "domain"den
} 3 değişik yüklem

} tutanak

<----->
derece = 3

35 Sauer... — s. 19'da

36 Paradox for Windows. Getting started...

37 Sauer... — s. 19-20'de

38 Sauer... — s. 20-27'de

"Domain"ler, ilişkiler ve yüklemeler, ki çoğu veri tabanı yönetin sisteminde bunlara "alan" da denir, ilişkisel veri tabanlarında da diğerleriyle aynı işlevleri taşırlar, ancak ilişkisel yapıda ilişkilerin yapılandırılmasına, hiyerarsik ve ağ yapılı veri tabanlarının aksine, önem verilmemiş, yüklemeler arası ilişkiler doğrudan yüklemelerin dilsel mantığında gerçekleştirilmiştir. Bunun sonucunda her varlık için yapı bakımından doğrusal ilişkiler kurulmuş, aynı doğrudan yer alan yüklemeler o varlık için ortaya çıkan tabloda bir satırı oluşturmışlar, buna da tutanak adı verilmiştir. Böylelikle bir veri tabanını oluşturan aynı sınıftan ya da türden farklı varlıklara ait tutanaklar, yapı bakımından birbirleriyle özdeş hale gelmişlerdir. Bunun sonucunda ortaya çıkan veri tabanı yapısı, yukarki örnekte de görüldüğü gibi son derece yalın ve kolay anlaşılır bir biçime girmiş, buna koşut olarak da sorgulama işlemleri hiç görülmediği kadar kolaylaşmış ve dolayısıyla da hızlanmışlardır.

Bunun yalı sıra tabloların şu 4 özelliği ilişkisel yapı açısından çok önemlidir³⁷:

1. Aynı tabloda, birbirinin aynı olan, yani yüklemelerinin değerleri aynı olan, iki tutanak bulunamaz. Sözelimi aynı öğrenci için iki tutanak tutulamayacağı gibi.
2. Tutanakların sırası belirsizdir. Buna göre tutanakların hangi sıra ile tabloya girileceği tamamen kullanıcının tutumuna bağlıdır.
3. Yüklemelerin sırası belirsizdir. anki yüklem hangi sırada tabloda yer alacağı gene tamamen kullanıcının tutumuna bağlıdır.
4. Yüklemelerin değerleri bölümsüzdür. Yani bir yüklem alacağı değer iki ayrı değerden oluşamaz.

Bu özelliklerin ışığında ilişkisel öğelerden anahtarları anlamak daha kolaydır.

Birincil anahtar yukarda anılan ilk özelliğin bir sonucudur. Tabloda yer alacak her tutanak birbirinden farklı olmak zorundaysa, tutanağının en az bir yüklemının değerinin o yüklem diğer tutanaklar için içerdiği değerlerden farklı olması gerekmektedir. Bu yüklem ise birincil anahtarı oluşturur. Birincil anahtarın alacağı değerler tek ve "biricik" (ing. unique) olmalıdır.

İlişkisel yapı gereği aynı tabloda hepsi birbirinden farklı olan kayıtlar bulunur, ancak bu kayıtların hangi yüklem değerinde farklılık göstereceği bilinemez. Örneğin

bir nüfus tutanakları tablosunda kişilerin ya adları, ya soyadları, ya doğum tarihleri, ya anne-baba adları, ya doğum yerleri v.b. özellikleri farklı olacaktır. Aynı değeri taşıyan bir insanın varlığı mantıksal olarak olası değildir. Ancak farklılığın nerede olabileceğini değil bilmek kestirmek dahi olanaksızdır.

Özellikle belli öğelere ait verinin arandığı sorgulamalarda, o öğenin tutanağının, her değeri diğer değerlerden farklı olan bir yüklemi olması gerekmektedir. Bu, yükleme istenilen belli tutanağa erişimde anahtar görevini üstlenmiş olan yükleme, ilişkisel yapıda birincil anahtar adı verilmiştir. Birincil anahtarların değerleri bir okulda öğrenci numarası, bir ürün için seri numarası, değişik alanlardaki sicil numaraları, kişiler için ulusal düzeyde ulusal ya da sosyal güvenlik numaraları olabilirler.

İkincil anahtarlar, ki bunlara kimi zaman seçmeli anahtar da denir, gene bir tablo içinde birincil anahtar dışında anahtar özelliğini taşıyan yüklemelerdir. Aday anahtarlar da bu özelliği taşırlar ancak ikincil anahtardan farklı olarak, aynı ilişkilere başka bir bakış açısı verilip bazı yüklemeler bu bakış açısına göre başka tabloları oluşturduklarında, aday anahtarlar bu tablolarda birincil anahtar rolünü üstlenirler.

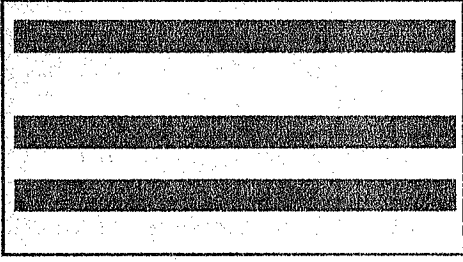
Yabancı anahtarlar ise, bir tabloda yer alan, ancak başka bir tablonun da birincil anahtarı olan aday anahtarlardır.

Görüldüğü gibi ilişkisel veri tabanında yüklemeler verliliklerinin temelini ille de olgusal gerçeklikten almazlar. Sorgulma gibi tamamen uygulamaya yönelik konular da veri tabanının yapısını etkiler.

İlişkisel veri tabanlarında şu iki ilişkisel bütünlük kuralına dikkat etmek gerekir³⁹:

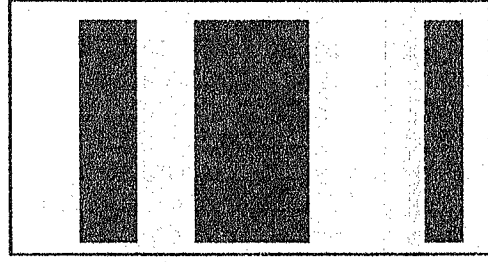
1. Eger bir yüklem birincil anahtarsa, bunun değeri asla boş bırakılamaz (varlık bütünlüğü kuralı).
2. Yabancı anahtara sahip olan bir tabloda bu yabancı anahtar eğer başka bir tablonun birincil anahtarına yönlendirme yapıyorsa ilk tablonun yabancı anahtarının değerleri diğer tablonun birincil anahtarlarının değerleriyle ya aynı olmalı ya da boş bırakılmalıdır (yönlendirme bütünlüğü kuralı).

İlişkisel veri tabanlarında aşağıda grafiklerle de ifade edilen şu işlemler özellikle büyük önem taşırlar⁴⁰:



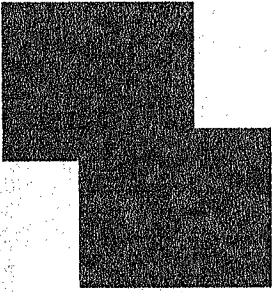
Seçme (restrict)

tablodan belli ölçütlere uygun tutanakların çekilmesi



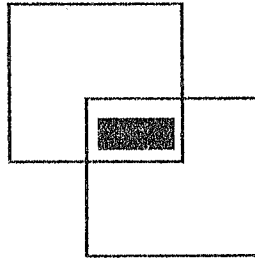
İzdüşüm (Projection)

tablodan belirli yüklemelerin çekilmesi



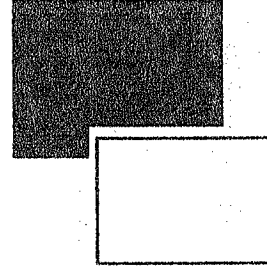
Birleştirme
(Join)

İki tablodaki tutanakların birleştirilmesi



Kesişim
(Intersection)

Her iki tabloda birden bulunan tutanakların çekilmesi



Ayrım
(Difference)

Bir tabloda bulunup diğerinde bulunmayanların çekilmesi

Nesne yönelimlilik yaklaşımı özellikle 1990'lı yılların başlarından itibaren bilgi işlem çevrelerindeki güçlü bir yönelimin adıdır. Doğrudan veri tabanlarına da yansıyan yaklaşım, ilişkisel veri tabanlarının bir takım aksaklıkları karşısında akılcı ve uygulanabilir çözümler sunduğundan, ilişkisel sistemlere henüz son verememekle birlikte kimi çevrelerde çok çabuk kabul görmüştür.

İlişkisel veri tabanlarının en büyük yitirimlerinden biri hiç kuşkusuz basit tablo yapısı içinde özellikle karmaşık sorgulara yönelik karmaşık yapıları barındırmamalarıdır⁴¹.

Örneğin katalog dosyası içinde konu başlığı yüklemi için tutanak yapısında birden çok değer alamamaktadır. Oysa kataloglanacak olan kitap genelde birden çok konu başlığıyla donatılır. Bu durumda ya tutanak bileşik bir konu başlığıyla donatılacak, ya da diğer konu başlıkları için tutanaktaki diğer yüklemelerin değerleri ikinci bir tutanakta diğer konu başlıklarıyla donatılarak veri tabanı yaklaşımının tam aksine gereksiz yere yinelenenlerdir. Basit bir hesaplama bir katalog tutanağı ortalama 3 konu başlığı ile donatılacaksa, veri tabanının harcadığı bellek alanı 3 katına çıkar.

Ancak bu durum nesne yönelimli veri tabanları için sözkonusu değildir. Çünkü "nesne yönelimli veri tabanlarında nesnelerin niteliklerini belirlemek için standart veri yapılarının dışında bunların tekrarlanabilir öğeleri de vardır"⁴².

Yani ilişkisel veri tabanında şu biçimde yer alacak katalog tutanağı:

Yazar adı	Kitap adı	Konu başlığı
Heuer, Andreas	Objektorientierte...	VERİ TABANLARI
Heuer, Andreas	Objektorientierte...	NESNE YÖNELİMLİLİK

41. Heuer, Andreas. Objektorientierte datenbanken : Konzepte, Modelle, Systeme / Andreas Heuer. — Bonn... [ve öte.] : Addison-Wesley, 1992. — s. 52'de

42. Heuer, Andreas...

nesne yönelimli veri tabanlarında şu biçimde yer alacaktır:

Yazar adı	Kitap adı	Konu başlıkları
Heuer, Andreas	Objektorientierte...	VERİ TABANLARI NESNE YÖNELİMLİLİK

Yani kısacası ilişkisel veri tabanlarında tutanaklar alanlardan oluşurken nesne yönelimli veri tabanlarında tutanaklar nesne ile doğrudan ilişki kurabilen ve sayıca sınırsız olabilen alt alanlardan da oluşabilir.

Nesne yönelimli veri tabanlarında bu nedenle, nesnenenin yapısında zamanın akışı içinde ortaya çıkabilecek değişiklikler, tutanak yapısı diğer nesneler için bozulmaksızın veri tabanına yansıtılabilmektedir⁴³. Bu nedenle de nesne yönelimli olarak adlandırılmaktadırlar. Hatırlanacağı gibi ilişkisel veri tabanlarında ilişki tablolarındaki tutanakların yapısında bu tür bireysel değişikliklere gitmek olanaksızdı.

1.4.5 Diğer

Bu türlerin dışında bir de artık geçerlilikleri kalmamış ve kullanılmayan, Rastgele Erişimli, Sıralı Erişimli, Dizimli ve Sırasız Liste olarak adlandırılan veri tabanları da vardır.

1.5 Veri Tabanı Yönetim Sistemlerinin İşlevleri

1.5.1 Tasarım

Tasarım, doğrudan doğruya veri tabanı geliştirme sürecinin veri tabanı tasarımı alt sürecinde gerçekleştirilen işlemleri üstlenen bir işlevdir. Yani mantıksal veri örneği bu işlev sayesinde veri tabanının manyetik ortamdaki yapısına dönüşür.

Yapılan işlemler sırasıyla şunlardır:

1. Yükleme (alan) adlarının belirlenmesi
2. Yüklemlerin (alanların) cinslerinin belirlenmesi (sayı, tarih, alfanumerik dizi v.b.)

43 Heuer, Andreas... — s. 56'da

3. Yükleme değerlerinin uzunluklarının belirlenmesi (alfanumerik diziler için)
4. Veri tabanı dosyasının yapısının bir ad ile saklanması

Sonuçta ortaya içi boş bir veri tabanı çıkar.



1.5.2 Güncelleştirme

Veri tabanı işletimi alt sürecini gerçekleştiren işlemdir. Bu işlevde ya yapısal ya da içeriksel değişikliklere gidilerek veri tabanı güncel tutulur. Veri tabanına yeni tutanakların eklenmesi işlemi de, veri tabanında içeriksel değişikliğe yol açtığı için bir güncelleştirme işlemidir.

1.5.3 Sorgulama ve Raporlama

Bu işlev veri tabanı yönetim sisteminin en önemli işlevidir. Veri tabanı yönetim sisteminin birincil amacı ne veri tabanlarını kurmak, ne de onları güncelleştirmektir; veri tabanı yönetim sistemlerinin birincil amacı kullanıcıya bilgi sağlamaktır. Bunun yolu da sorgulamadan geçer.

Sorgulama işlevi ne kadar etkinse veri tabanı da aslına bakılacak olursa o kadar etkindir. Bu nedenle sorgulama kesin olarak çok amaçlı bir yapıya sahip olmalıdır. Yani örneğin ilişkisel sistemlerde sorgulama yalnızca anahtarlar üzerinden değil, diğer yüklemelerin alacağı değerler üzerinden de yapılabilmelidir. Tıpkı kütüphane kataloglarının kişi adı, başlık, konu başlığı, dizi adı v.b. öğelere ait girişlerden taranması gibi. Sorgulama bunun yanı sıra hata paylarını karşılayacak derecede esnek olmalıdır.

Raporlama sorgulamanın bir ilerki aşamasıdır. Sorgulamada elde edilen karşılıklar bu aşamada kullanıcıya istğine ve gereksinimlerine bağlı olarak her hangi bir biçimde ekran ya da yazıcı çıktısı olarak kullanıcıya sunulurlar. Yazıcıdan alınacak raporlar için kimi veri tabanı yönetim sistemleri grafik ve değişik font seçenekleri gibi olanakları da kullanıcıya sağlamaktadırlar.

1.5.4 Dağıtık Veri Yönetimi

Günümüzde çoğu veri tabanı yönetim sistemi bilgisayar ağlarında da çalışabilme özelliğini göstermektedirler. Yani bir bilgisayarda işleyen bir veri tabanı yönetim sistemi, diğer bir bilgisayarda işleyen uyumlu olduğu bir veri tabanına ya yerel ağ bağlantıları ya da modemler üstünden telefon veya veri iletişimi hatlarını kullanarak erişebilmekte, onu sorgulayabilmekte, verilerinin paylaşabilmektedir.

Günümüzde Dağıtık Veri Yönetimi (Distributed Data Management) yaklaşımıyla değişik bilgisayar sistemlerinin bir tabanda bir araya getirilebilmekte olması⁴⁴ özellikle veri tabanı yönetim sistemlerini bu yaklaşıma uyumlu hale gelmeye ya da en azından desteklemeye yöneltmektedir. Bu yaklaşım özellikle bilgisayara dayalı kütüphaneler arası işbirliği alanında yaşanan bazı sorunları da çözebilecek gibi görünmektedir.

1.5.5 Diğer işlevleri

Veri tabanı yönetim sistemlerinin diğer işlevleri işe çoğaltma, silme, yeni ad verme gibi yönetim sisteminin işlevleridir. Ancak bu işlevlerin veri tabanı yönetim sistemi içinde yer alması kullanıcıya büyük rahatlık verdiğinden çoğunlukla tercih edilen bir durumdur. Aksi hallerde kullanıcı veri tabanı yönetim sistemi yazılımında yönetim sistemine çıkmak zorunda kalacaktır.

1.6 Veri Tabanı Uygulama Yazılımları

Veri tabanı uygulama yazılımları geliştirildikleri veri tabanı yönetim sistemiyle uyumlu, onun tarafından yorumlanabilen bir özel uygulama dili ile kodlanmışlardır:

1.6.1 Veri tabanı Uygulama Dilleri

Bu bölümde özellikle yazılım pazarında satışa sunulan ticari yazılımlar hakkında genel bilgi verilecektir.

1.6.1.1 Dbase/Clipper

Dbase* günümüzde işletmelerde en çok kullanılan veri tabanı yönetim sistemidir. Bu kullanım çokluğunun nedeni sistemin gelişmişliği değil aksine az gelişmişliğidir. Bu arada özellikle belirtmekte yarar vardır ki yoğun olarak kullanılan Dbase sürümü son sürümlerden olan Dbase IV 2.0 değil bundan yaklaşık 9 yıl öncesinin sürümü olan Dbase III Plus'tır.

Geniş bir kullanıcı kesimi tarafından kullanılan bu sürüm çoklukla kendisiyle tam uyumlu olmayan Clipper adlı derleyici ile derlenmiş, ortaya çıkan uygulamalar kişisel

* Dbase, Borland International Corporation'ın tescilli markasıdır.

44 Inside IBM's Distributed Data Management architecture / by R. A. Demers... [ve öte.]. — s. 459-487. — IBM Systems Journal. — vol 31, No. 3 (1992). —463'de

telefon rehberi gerksinimlerini başarıyla karşılamış, ancak kütüphane uygulamaları gibi karmaşık uygulamalarda sonuç doğal olarak başarısızlık ve hayal kırıklığı olmuştur; Mikrobeta Kutup örneğinde olduğu gibi.

1.6.1.2 SQL (Yapılandırılmış Sorgulama Dili)

IBM tarafından geliştirilen SQL zaman içinde uluslararası bir veri tabanı uygulama dili standardı olmuştur. Bunda özellikle kişisel bilgisayarlardan, iş istasyonları ve büyük ağ sistemlerinde uygulanabilir olması⁴⁵ yatmaktadır. Kimi üreticiler ürünlerinde SQL standardını desteklemekte, kimileri de kendi uygulama dillerini SQL ile uyumlu hale getirmektedirler.

1.6.1.3 ObjectPAL

ObjectPAL, Object Based Paradox Application Language adlı uygulama dilinin kısaltılmış adıdır. ObjectPAL Windows işletim sistemi altında çalışan Paradox for Windows veri tabanı yönetim sisteminde Windows temelli tümleşik veri tabanı uygulamalarının geliştirilebileceği bir dildir⁴⁶. ObjectPAL ile geliştirilecek uygulamalar ürünün üreticisi tarafından geliştirilen ve e satışa sunulan ürünle tam olarak uyumlu RUNTIME adlı derleyici ile derlenerek bağımsız bir uygulama yazılımı biçimine de dönüştürülebilir.

SQL standardını sorgulamaları ile destekleyen sistem özellikle Windows gibi bir grafik kullanıcı arayüzünün verdiği olanaklarla ilişkisel sistemin basit yapısını birleştirmiş ve her düzeyde kullanıcı için Paradox for Windows gibi güçlü bir sistemi kullanılabilir kılmıştır.

1.6.2 Uygulama Yazılımlarının Geliştirilmesinde Dikkat Edilmesi Gerekenler

1.6.2.1 Dil

Veri tabanı uygulama yazılımlarında dilin oynadığı rol aslında diğer yazılımların geliştirilmesindekiyle aynıdır. Kullanıcı-sistem diyalogunda kullanılacak dil sade ve anlaşılır olmalıdır. Ancak özel konulu uygulamalarda genel terimlerin yerine konuya yönelik özel (kimi zaman da teknik) terimler kullanılabilir.

45 Sauer... — s.43'te

46 Paradox for Windows. Guide to ObjectPAL. — Scotts Valley, California : Borland International, 1992. — s. 1'de

Ancak veri tabanı yönetim sistemine yönelik özel terimler kullanıcıya yansıtılmamalı, kavramlar herkesin anlayacağı genel terimler ile ifade edilmeye çalışılmalıdır: örneğin yükleme yerine alan ya da hane, tutanak yerine de kayıt terimlerinin kullanılması gibi.

1.6.2.2 Kullanıcı Dostluğu

Tüm yazılımların kullanıcı dostu olması çok önemlidir. Yazılımda ekran görüntüleri gözü rahatsız edecek türden olmamalı, parlak renklerin yerine gözü daha az yoracak, sade ve uyumlu renklerin ve renk bileşimlerinin kullanılmalıdır.

Yazılımın işeyişi kullanıcının gereksinimleri doğrultusunda olmalı, yordamlarda gereksiz işlemler ya da tekrarlar bulunmamalıdır.

Yazılım kesin olarak fare destekli olmalı kullanıcı sadece sayısal ya da alfabetik değer girişlerine duyarlı menülerden kurtarılmalı, gerekiyorsa "pop-up" ve "pull-down" menüler kullanılmalıdır.

Bu nedenle yazılımlar iyi bir grafik kullanıcı arayüzü olan Windows altında çalışarak, pencere mantığı, düğme menüsü gibi grafik kullanıcı arayüzlerinden yararlanabilmelidir.

Yazılımlarda özellikle dikkat edilmesi gereken nokta ise yazılımın yapısının tümleşik olmasıdır. Kullanıcı gereksinimlerini tek yazılımdan karşılayabilmeli, değişik işlemler için başka yazılımları çalıştırmak zorunda kalmamalıdır. Bu ise diğer yazılımların alt yordamlar biçiminde ana yazılımın yapısında tümleşik olarak bulunmasıyla olur. Yordamlar arası geçişler kolay, dolaysız ve çabuk olmalıdır.

2 Kütüphanelerde Veri Tabanı Yönetim Sistemi Uygulamaları ve Bir Çözüm Önerisi

2.1 Veri Tabanı Yönetim Sistemi Uygulamalarının Kütüphanelere Getirdikleri Yenilikler

2.1.1 Tümlşik Yapı

Gerek arşivlerin, gerek kütüphanelerin gerekse belge-bilgi merkezlerinin yapısı sistemattiktir. Bu sistemi birbirleriyle tümlşik olarak işleyen alt sistemler oluşturur: sağlama, kataloglama, yararlanđırma v.b. Bu nedenle veri tabanı yönetim sistemleri ve bu sistemlerde geliştirilecek uygulama yazılımlarının yukarda da ele alındığı gibi kullanıcılarına sunduđu tümlşik yaklaşım söz konusu kuruluşlar için yaşamsal önem taşır. Hele ki Türkiye gibi otomasyon konusunda sistemsizliklerin yaşandığı içinü dünya ülkelerinde.

Veri tabanı uygulaması ile geliştirilecek olan bir sağlama katalogu veri tabanı sağlama yordamının verisi olduđu gibi asıl katalog veritabanına aktarılabilecek, bu veri tabanı ise okur tarafından sorgulanarak bibliyografik tarama yapılabilecektir.

Bu kolaylık ise söz konusu kuruluşlara işlerini tümlşik bir yapı içinde esgüdümlü bir tarzda yapmaları olanağını verecektir.

2.1.2 Standardizasyon ve İşbirliği Olanakları ve MARC Biçimine Duyulan Gereksinimin Ortadan Kalkması

Veri tabanı yönetim sistemlerinin sağladığı önemli bir gelişim kurulan ağ yapıları ile farklı bilgisayarlardaki farklı veri tabanlarına erişimdir. Bu sayede sistemler başka sistemlere ait veri tabanlarını sorgulayabilmekte ve onların üstünde işlemler yapabilmektedirler. Bu işe makinaca okunabilir bibliyografik verinin bibliyografik kuruluşlar arası değişimini amaçlayan MARC biçimine gereksinimi teorik olarak ortadan kaldıracaktır.

MARC yaklaşımının ortaya atıldığı yıllarda her kütüphane kendi kayıtlarını kendi özgün veri yapılarında tutmaktadır. Bu kayıtların diğer kütüphaneler tarafından kullanılabilmeleri için hepsinin ortak bir iletişim ya da değişim biçimine oturtulması gerekmektedir. MARC bu gereksinime yanıt vermek için tasarlanmış bir ortak iletişim formatıdır.

Ancak kütüphanelerde yetkin bir üst kuruluş (örneğin Kütüphaneler Genel Müdürlüğü) tarafından ortak bir veri

tabanı yapısı belirlenirse kütüphanelerin kullanacakları veri tabanı standartlarını destekleyen veri tabanı yönetim sistemleri ya da bunlara dayalı uygulama yazılımları ile kütüphaneler arası veri değişimi zaten kendiliğinden sağlanabilecektir. Dolayısıyla MARC biçimine gerek kalmayacaktır.

Burada hemen belirtmek gerekir ki MARC yaklaşımı Türkiye'de henüz tam anlamıyla kavranamamıştır. MARC Biçimi Bir İletişim Biçimidir, KUTUPHANENİN VERİ TABANININ YAPISININ Biçimi Değildir. BU NEDENLE KUTUPHANELERİN KATALOGLARINI OLUŞTURAN VERİ TABANLARINI MARC Biçimine Dayandırmak MANTIKSIZLIKTIR. KUTUPHANELERİN VERİ TABANLARI DİĞER KURULUŞLARIN VERİ TABANLARI GİBİ YARATILDIKLARI VERİ TABANI YÖNETİM SİSTEMİNDE EN İYİ NASIL SORGULANABİLECEKLERSE O Biçimde Kurulmalıdırlar. Ancak ne yazık ki kütüphane bilimcilerimiz dahi bu yanlış görememektedirler. Örneğin Jale Baysal "MARC I" ve "MARC II" için "veri tabanı der"⁴⁷. Gene, örneğin Anadolu Üniversitesi Kütüphanesi'nde uygulama yazılımı "SQL/DS veri tabanı yönetim sistemi ile geliştirilmiştir" ancak gene de "MARC standart alan kodları kullanılmıştır"; Kütüphane ve bilgi işlem yetkilileri de bundan Türk Kütüphaneciliği dergisinde kıvançla söz etmektedirler⁴⁸.

Ancak bu konuda özellikle yazılımcıların bu hataya düşmesi genel olarak Türkiye'de yazılımcıların içler acısı durumunu göstermektedir: aklıevvel yazılımcılarımız ya veri tabanı kurma sürecini bilmemektedirler, ya da kendilerine çok zor gelen, aslında çok da zor olan, veri araştırması ve veri örneklemesini yapmaya üşenerek karşılarına ilk çıkan kaynak olan MARC ile soruna sağlıksız bir şekilde yaklaşmaktadırlar.

2.1.3 Kütüphanecinin Değişen Konumu

Kütüphanelerde veri tabanı yönetim sistemlerine dayalı uygulanamların artması ile, ki genel eğilim artacağı yönünde, özellikle yönetici konumundaki kütüphaneci veri tabanı yöneticisi olmaya adaydır.

47 Baysal, Jale. Kütüphanecilik alanında yeni kavramlar, araçlar, yöntemler / Jale Baysal. — 2. bs. — İstanbul : İstanbul Üniversitesi Edebiyat Fakültesi Yayınları, 1987. — s. 160'da

48 Yılmaz, Adnan. Veri tabanına dayalı Türkçe kütüphane otomasyonu yazılımı : KYBELE / haz. Adnan Yılmaz, Hürya Aslan. — s. [10]-17. — Türk Kütüphaneciliği. — [c.] 6, [s.] 1 (1992). — s. [10]-11'de

Bu nedenle geleceğin kütüphanecileri öncelikle kütüphane sisteminin tümleşik yapısını özümsemek zorundadırlar. Buna koşut olarak ileri bir veri tabanı yönetim sistemi bilgisi de temel şartlardan biri olacaktır. Özellikle kendi uygulamasını geliştirmek isteyen kütüphaneciler veri tabanı geliştirme sürecini ayrıntılı bir şekilde tanımak ve bunu gerçekleştirebilmek zorundadırlar.

2.2 Kütüphanelerde Veri Tabanı Yönetim Sistemine dayalı Uygulamalar için Paradox for Windows Veri Tabanı Yönetim Sistemi Altında Geliştirilebilecek Bir Çözüm Önerisi

2.2.1 Paradox for Windows'a Dayalı Çözümlerin Uygulanabilirliği

2.2.1.1 Veri Tabanı Yapısı Ve Tasarım

Sistem özellikle tasarım konusunda sanki kütüphaneler için tasarlanmış gibidir. Diğer veri tabanı yönetim sistemlerinde alfa numerik alanların uzunlukları önceden belirlenen değerde sabit kalırken, Paradox for Windows bu sorunu "Memo" adlı değişken uzunluklu alfanumerik alan türüyle çözmüştür⁴⁹. Memo alanlarının uzunlukları, eğer dosyanın içinde saklanacaklarsa 1-240 karakter arasındadır. Eğer dosya dışında saklanacaklarsa, bu sayı sadece manyetik bellek ünitesindeki, yani kişisel bilgisayarların hard disk'lerindeki bellek alanı ile sınırlanabilmektedir. Kullanıcı sistem tarafından kurulan mantıksal ilişkiler sonucunda alan dosya dışında saklı olsa bile bunu fark etmeyecek ve sistemin ilişkiselliği bozulmayacaktır.

Böylelikle katalog verisinin değişken uzunluk sorunu da kendiliğinden ortadan kalkmaktadır.

Gene Paradox for Windows'a özgü bir alan türü olan "OLE" (Object Linking and Embedding) alanında ise başlı başına görüntüsel belgeler (resim, çizim, fotoraf v.b.) saklanabilmektedir⁵⁰. Bu ise belge hizmeti veren arşivlere, özel kütüphanelere v.b. kuruluşlara veri tabanında oluşturulmuş katalogda yapılacak tarama sonucunda istenen belgenin görüntüsünü ekrana yansıtabilme olanakını verecektir.

49 Paradox for Windows. User's guide. — Scotts Valley, California : Borland International, 1992. — s. 15'te

50 Paradox for Windows. User's guide... — s. 16'da

2.2.1.2 İşletim ve Denetim

Windows'a daya grafik kullanıcı arayüzü sayesinde Paradox for Windows'da veri tabanı işletimi yapmak son derece basitleşmiştir. Paradox for Windows'un kullanıcı tanımlı veri girişi "form"ları özellikle bir takım rutin işlevlerin yüklenebileceği düğmelerle donatılabilmektedir⁵¹, bu ise veri tabanı yönetim sisteminden uygulama yazılımına geçmeden basit işlemlerin yapılmasını inanılmayacak kadar kolaylaştırabilmektedir.

Sistem belli bir "form"dan birden çok veri tabanına giriş yapmaya olanak tanımakta böylelikle kullanıcı, veri tabanı değişik dosyalara pratik nedenlerle bölünmüş olsa bile bunları bir arada tek veri tabanında işletim yapar gibi işletebilecektir.

Bu tümleşik yapı sayesinde kütüphaneci katalog betimlemesi ve sınıflama gibi değişik işleri aynı ekrandan yapabilme olanağına kavuşmuştur.

2.2.1.3 Sorgulama ve Raporlama

Sistemin en can alıcı işlevi olan sorgulama hem güçlü sorgulama parametreleri⁵² ile hem de QBE (query by example = örneğe dayalı sorgu) yaklaşımı⁵³ ile bibliyografik veri tabanında en karmaşık sorgulara herhangi bir uygulama yazılımına gerek kalmadan daha veri tabanı yönetim sistemindeyken çok esnek bir tarzda olanak vermektedir.

Raporlamada kütüphanecilik açısından en büyük kazanım sistemin ASCII dizilimini kullanan çoğu sistemin aksine Türkçe alfabetik dizilimi desteklemesi ve raporlarını bu doğrultuda alfabetik olarak üretebilmesidir. Bunun yanı sıra raporlar bir çok font ve sayfa düzeni seçeneğine sahiptir⁵⁴. Bu ise sisteme gerekli donanım eşliğinde katalog kartından bibliyografyaya kadar bir çok çıktı olanağını sağlamaktadır.

Sistem bunun yanı sıra manyetik ortamda saklı bulunan çeşitli belge görüntülerinden de kullanıcıya çıktı olanağı vermektedir.

-
- 51 Paradox for Windows. User's guide... — s. 431'de
 - 52 Paradox for Windows. User's guide... — s. 219-221'de
 - 53 Paradox for Windows. User's guide... — s. 148'de
 - 54 Paradox for Windows. User's guide... — s. 467-534'de

2.2.2 Çözüm Önerisi : B.B.V.T.Y. *

B.B.V.T.Y., yani Bölünmüş Bibliyografik Veri Tabanı Yapısı özellikle Paradox For Windows veri tabanı yönetim sistemi için tasarlanmış bir katalog veri tabanı önerisidir.

Katalog yapısı için Anglo-Anmerikan Kataloqlama Kuralları 2'de⁵⁵ ele alınan bibliyografik betimleme kuralları temel alınmıştır, yani yapı Anglo-Amerikan standardına uygundur.

Veri tabanının, değişik veri tabanı dosyalarına bölünmüş olmasının altında pratik bir düşünce yatmaktadır: ilişkisel yapıdan kaynaklanan aksaklıkları ortadan kaldırmak.

Bilindiği gibi kütüphane katalogları çok karmaşık veri tabanlarıdır. Bunun nedeni her alan için sayısı belli olan ancak kaçının ya da hangilerinin kullanılacağı hiçbir zaman için belli olmayan değişik alt alanların varlığıdır (kimi zaman da yokluğu). Bu alanları ilişkisel yapının tutanaklarında doğrusal olarak birbirleri ardına başlı başına alanlar gibi sıralamak olasıdır. Ancak bu sefer de kullanılamayan alanlar gereksiz bellek alanı kullanımı gibi sorunlar doğuracaklardır.

Buna bulunan çözüm ise basittir. Her alan için başlı başına bir veri tabanı dosyası açılmıştır. Bu dosyaya tüm alt alanlar başlı başına farklı tutanaklarda girilir. Tutanak yapısı içinde alt alan adı alanının yanı sıra bir anahtar alanı, bir de yönlendirne alanı eklenmiştir; bunlara bazı yerlerde gerek duyulan ek alanlar da eklenmiştir, ki bununla da kimi yerde veri tabanının kendi yapısı içinde yetke dizinlerinin oluşması sağlanmıştır.

Anahtar alan bir yerde kütüphanwlerde kullanılan demirbaş numarasıdır. Bu numaranın özelliği kataloglanan kitabı temsil etmesidir. Anahtar alanın değeri kataloglanan bir kitap için bütün veri tabanlarında aynı olduğundan bu numara üstünden bağlantı kurulacaktır. Yapı ilişkisel temele dayanmakla beraber birincil ya da ikincil anahtar alanlarının olmaması ilişkisel yapıyı ortadan kaldırmakta anahtar olarak

* Çözümün hukuksal çerçevesi için lütfen araştırmanın başlangıcında yer alan "YASAL UYARILAR" bölümüne bakınız.

55 Anglo-American cataloguing rules / prep. under the dir. of the Joint Steering Committee for Revision of AACR ; ed. by Michael Gorman and paul Winkler. — 2nd ed., 1988 rev. — Chicago ; Ottawa ; London : ALA ; CLA ; LA, c 1988, 1989

kullanılan öge doğrudan doğruya kataloglanan belgenin nesnesine yönelik olduğundan nesne yönelimlilik benzeri bir yapı ortaya çıkmaktadır. Bu yapı çerçevesinde anahtar alanın değeri dolaşım dosyasında da da kullanıcı numarası ile ilişkilendirilerek hangi belgenin hangi okurda olduğu bu ilişki ile saptanabilecektir.

Yönlendirme alanında yer alacak değer doğrudan doğruya alt alan ögesinin hangi alt alan olduğunu belirleyecektir.

Kütüphaneci bu arada bütün bu karmaşık yapıdan uzak kalarak kendi iş akışına göre belirlenmiş bir ekran "form"undan girişlerini yapacak, geliştirilecek uygulama yazılımı kataloglanan her kitaba bir anahtar numarası vererek ve bütün alt alanları gerekli veri tabanı dosyalarına bu anahtar numarası hangi alt alan olduğunu belirleyen yönlendirme kodu ile dağıtacaktır.

Sorgulmalarda ise Paradox for Windows'un güçlü sorgulama seçenekleri yardımıyla uygulama yazılımı, belli bir sorgulama "form"u ekranından girilen bir ya da bir den fazla sorgulama ögesini denk düştükleri veri tabanından tarayacak, yanıt olarak aldığı anahtar numaraları üstünden QBE yöntemiyle bulunduğu diğer alt alanları ve alanları anahtar numarası aynı olanları yönlendirme koduna göre ekranda geleneksel katalog kartı görüntüsü altında ya da tercih edilebilecek başka bir biçimde bir araya getirecektir. Aynı yöntem raporlama çıktılarında da kullanılabilecektir.

Bu aşamada her veri tabanını dosyasını tek tek ele almak gereklidir:

B.B.V.T.Y. 11 ayrı veri tabanı dosyasından oluşur bunlar kişi adlarının bulunduğu KISI.DB, başlıkların bulunduğu BASL.DB, konu başlıklarının bulunduğu KONUB.DB, belgenin doğasına ilişkin verinin bulunduğu BELGE.DB, sınıflama kodlarının bulunduğu SINIF.DB, dizi kayıtlarının bulunduğu DIZI.DB, yayın ya da baskı yerlerinin bulunduğu YER.DB, yayıncı ya basımevinin bulunduğu YABA.DB, yayın ya da baskı tarihleini içeren TARİH.DB, notları içeren NOT.DB, ve ISBN ya da ISSN numaralarını içeren STAND.DB, veri tabanı dosyalarıdır.

KISI.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
YÖNLENDİRME	N
İŞLEV	MEMO
KİŞİ ADI	MEMO
SAN	MEMO
DENETİM	N
TEKBİÇİM ADI	MEMO

Görüldüğü gibi ANAHTAR adlı alan sayısal bir alandır. Bunun nedeni yazılımın kataloglardan kayıt eksiltilmesi durumunda bütün numaraları düzeltebilmesi ve yeni numaraları da basitçe sayarak kendi kendine verebilmesi içindir.

YÖNLENDİRME adlı sayısal alana önceden belirlenmiş sayısal kodları yazılımın kataloglama yapılırken alt alanlara göre kaydederek gerekli hallerde gene değerlendirebilecektir. Burada özellikle belirtmek gerekir ki kişi adları, sorumlu adı, emeğigeçen adı, basım sorumlusu adı v.b. alt alanlara ait olabilirler; hatta notlar alanından buraya gönderme bile yapılabilir. Buna göre yönlendirme kodu önceden Anglo-Amerikan kurallarının ilgili maddelerine göre belirlenir ve yazılımda saptanır.

İŞLEV alanı sorumluluk alanında yer alan işlev "yayına hazırlayan", "resimleyen", "derleyen" v.b. türü bildirimlerin kısaltmalarını içerir. Uzunluğu belirsiz olduğundan "memo" türü bir alan olarak belirlenmiştir.

KİŞİ ADI alanı değişken uzunluklu bir alandır. Bu alan doğrudan doğruya kişi adının gövdede bulunan biçimini içerir.

SAN alanında ilgili kural gereği kişi adı İngiliz soyluluk sanı taşıyorsa bu içerilir.

DENETİM alanında kişi adının tüzel kişiye mi ait yoksa kişiye mi ait olduğu belirlenir.

TEKBİÇİM ADI alanı ise kiş adının özellikle temel giriş için alacağı biçimi içerir. Bu alan ile kişi adları için kendiliğinden bir yetke dizini de oluşur.

BASL.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
BASLIK	MEMO
YÖNLENDİRME	N

BASLIK değişken alanı ilgili kataloglama kuralı gereği başlık, bilgi başlığı, koşul başlık v.b. alt alanları içerir.

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar, YÖNLENDİRME alanı alt alanın türünü belirler.

KONUB.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
KONU BAŞLIĞI	MEMO

KONU BAŞLIĞI değişken alanı konu başlığını içerir.

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar.

BELGE.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
GBB	MEMO
ÖZEL AYRINTI	MEMO
UZUNLUK	MEMO
FİZİKSEL ÖZELLİKLER	MEMO
BOYUTLAR	MEMO
BİRLİKTEKİ BELGELER	MEMO

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar.

Diğer değişken uzunluklu alanlar Kataloqlama Kuralları'nın ilgili maddeleri gereği belgenin doğasına ilişki veri içerirler.

SINIF.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
YER #	MEMO
SINIFLAMA #	MEMO

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar.

YER # değişken uzunluklu alanı D.D.C., L.C.C. v.b. yerleştirme sistemi kodlarını içerir.

SINIFLAMA # değişken uzunluklu kodları U.D.C., I.T.C. v.b. sınıflama sistemi kodlarını içerirler.

DIZI.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
DİZİ ADI	MEMO
YÖNLENDİRME	MEMO
TEKBIÇİM DİZİ	MEMO

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar, YÖNLENDİRME alanı alt alanın türünü belirler.

DİZİ ADI değişken uzunluklu alanı dizi alanlarını içerir.

TEKBIÇİM DİZİ değişken uzunluklu alanında zaman içinde değişiklik gösterebilecek dizi adları bir tekbiçimde toplanır. Böylelikle de veri tabanının yapısında kendiliğinden bir yetke dizini oluşur.

YER.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
YÖNLENDİRME	N
YER	MEMO
ÜLKE	MEMO

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar, YÖNLENDİRME alanı alt alanın türünü belirler.

YER değişken uzunluklu alanına belgenin yayın ya da basım yeri/yerleri alt alanları bulunur.

ÜLKE değişken uzunluklu alanına belgenin yayınlandığı ülke girilir. Bu alanın ilgili Kataloqlama Kuralında yeri vardır, ve ülke adına göre yapılacak taramalar için gerekli görülmüştür.

YABA.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
YÖNLENDİRME	N
İŞLEV	N
YAYBAS	MEMO
TEKBEÇİM	MEMO

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar, YÖNLENDİRME alanı alt alanın türünü belirler.

İŞLEV sayısal alanı Kataloqlama Kuralları gereği olup kuruluşun işlevini belirler.

YAYBAS değişken uzunluklu alanı yayıncının ya da basımevinin adını alt alanlarını içerir. Bu ad TEKBEÇİM adlı değişken uzunluklu alanı zaman içindeki değişiklikler göz önüne alarak bu adı bir tekbeçime oturtur.

TARİH.DB

<u>Alan Adı</u>	<u>Türü</u>
ANAHTAR	N
YÖNLENDİRME	N
BASLAMA	MEMO
SON	MEMO
CPRIGHT	N

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar, YÖNLENDİRME alanı alt alanın türünü belirler.

BASLAMA değişken uzunluklu alanı özellikle süreli yayınlar için yayına başlama tarihinin girileceği alt alana yöneliktir. SON değişken uzunluklu alanı ise gene süreli yayınlar için yayın bitiş tarihinin girileceği alt alana yöneliktir. Dizi yayın olmayalar BASLAMA alanına girilebilirler.

CPRIGHT alanında girilen tarihin copyright tarihi olup olmadığı belirlenir.

Alan AdıTürü

ANAHTAR

N

NOT

MEMO

ÖNCELİĞİ

N

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar. Alt alanların sıralanışı Kataloglama Kutalları gereği ÖNCELİĞİ alanında belirtildiği için yönlendirmeye gerek yoktur.

NOT değişken uzunluklu alanına notlar alanının 22 değişik alt alanı girilir.

STAND.DB

Alan AdıTürü

ANAHTAR

N

STANDART #

A13

ANAHTAR sayısal alanı diğer veri tabanları ile bağlantıyı sağlar.

STANDART # alanı ISBN numarası tireler dahil 13 basamaklı olduğu için 13 karakterlik bir alfanumerik alan olarak belirlenmiştir. 9 basamaklı ISSN numarası da bu alanda belirtilebilir.

Günümüzde gerek kütüphanecilik gerekse bilgisayar konulu yayınlarda veri tabanı kavramı çoğunlukla kendisinden başka bir kavram olan veri tabanı yönetim sistemi kavramının yerine, sanki onun bir kısaltmasıymış gibi, yanlış olarak kullanılmaktadır. Oysa veri tabanı kavramı, aşağıda da ele alınacağı gibi, özellikle bilgisayar ortamındaki uygulamalarda, verilerin yüklü bulunduğu örgütlü teknik düzenegi (kütüphanecilik bakış açısıyla bir belgeyi) belirtirken, veri tabanı yönetim sistemi kavramı bu düzenek ile işlem yapmaya, onu söz konusu örgütlenme içinde yönetmeye yarayan bir tür uygulama yazılımını belirtmektedir.

Veri tabanı uygulama yazılımları veri tabanı yönetim sistemlerinde kimi özel uygulamalarda sıkça tekrarlanan yordamları oluşturan komutların kullanıcı tarafından belirlenen üst başlıklar altında bir araya getirilmesinden oluşan yazılımlardır.

Veri tabanı yaklaşımının getirdiği başlıca yenilikler ise özetle şöyle sıralanabilir:

1. Gereksiz tekrarlar önlenmiş, bellek tasarrufu sağlanmıştır.
2. Değişik veri tabanı dosyalarına veri tabanı yönetim sisteminden merkezi erişim sağlanmıştır.
3. Merkezi denetim sonucu veri güvenliği sağlanmıştır.
4. Veri tabanı yönetim sisteminden merkezi erişim sağlanarak basit işlemler için yazılım geliştirme gereği ortadan kalkmıştır.
5. Veri tabanı yapısı nedeniyle erişim hızı artmıştır.
6. Veri tek dosya yapısına yüklendiğinden verilerin arasındaki tutarsızlıklar tamamen ortadan kalkmıştır.
7. Veri dosyalarının yapısında standartlaşma sağlandığından özellikle kütüphaneler gibi kuruluşların arasında veri alışverişi kolaylaşmış, birbirlerinden bağımsız sistemler telefon hattıyla birbirlerine veri aktarmanın ötesinde birbirlerinin veri tabanlarına erişerek, hatta onlarla işlem yaparak, zaman ve veri taşıma masraflarından büyük kazançlar sağlanmıştır.

Veri tabanı geliştirmesi şu beş alt süreçten oluşan bir süreçtir:

1. Veri araştırması
2. Veri örnekleme
3. Veri tabanı tasarımı
4. Veri tabanı işletimi
5. Veri tabanı ayarlaması

Veri tabanları, örgütlenme yapıları açısından zaman içinde farklılıklar göstermişler ve böylece de veri tabanlarının değişik türleri ortaya çıkmıştır. Ancak tarihsel açıdan konuya yaklaşıldığında, yaklaşım olarak veri tabanı düşüncesinin değişmediği, aksine değişimin yapısal nitelikli olduğu görülür. Bu yüzdendir ki yeni bir tür ortaya çıktığında eskisi etkisini yitirmektedir.

Veri tabanlarında yaşanan bu yapısal değişikliklerin başlıca nedeni veri tabanının yapısının giderek gerçekliğin bir örneğinin bilgisayar ortamındaki yansıması olmaktan, uygulama amacına yönelik olarak yeniden biçimlendirilmiş bir yapıya yönelmesidir. Örneğin ilk türlerden olan hiyerarşik veri tabanlarında veri tabanı yapısı uygulama (özellikle de sorgulama) kaygısı taşımaksızın, gerçekliğin felsefi anlamda tamamen koşut bir yansımasıdır. Ağ yapılı veri tabanlarında bu yapı biraz daha uygulamaya yönelik bir biçime sokulmaya çalışılmış, ancak elde edilen hiyerarşik veri tabanının yapısından biraz daha büyük bir karmaşa olmuştur.

Veri tabanı dünyasında ilk devrim niteliğini taşıyan gelişme yapı bakımından diğerlerinden oldukça yalın, özellikle sorgulamalarda büyük kolaylıklar sağlayan ilişkisel veri tabanlarının ortaya çıkmasıdır.

İlişkisel veri tabanlarındaki kimi aksaklıklar ve kısıtlamaların giderilmeye çalışılmasıyla da özellikle 90'lı yıllara damgasını vuracak gibi görünen nesne yönelimli veri tabanları ortaya çıkmıştır.

Kütüphane alanında veri tabanlarının getirdiği yenilikler, bilgisayara dayalı uygulamaların tümleşik bir yapıya kavuşması, veri tabanlarının işlevleri bedeniyle ulusal düzeyde bir MARC biçimine ihtiyaç duyulmaması, ve kütüphanecinin zamanla bir veri tabanı yöneticisi konumuna geleceği edgilimidir.

Paradox for Windows adlı veri tabanı yönetim sistemi özellikle kütüphanelerde gerçekleştirilebilecek uygulamalar için sağladığı geniş olanaklar, kolay kullanımı ve güçlü yapısı ile çözüm olanakları sağlayacak bir sstemdir.

Bölünmüş Bibliyografik Veri Tabanı Yapısı adlı çözüm ise özellikle Paradox for Windows altındaki uygulamalar için geliştirilmiş, sistemin olanaklarını kullanıcının en üst düzeyde hizmetine sokan bir çözümdür.

American heritage dictionary. — 2nd college ed. — Boston : Houghton Mifflin Company, c. 1982

Anglo-American cataloguing rules / prep. under the dir. of the Joint Steering Committee for Revision of AACR ; ed. by Michael Gorman and Paul Winkler. — 2nd ed., 1988 rev. — Chicago ; Ottawa ; London : ALA ; CLA ; LA, c. 1988, 1989

Baysal, Jale. Kütüphanecilik alanında yeni kavramlar, araçlar, yöntemler / Jale Baysal. — 2. bs. — İstanbul : İstanbul Üniversitesi Edebiyat Fakültesi Yayınları, 1987

Dbase 3 Plus. Introduction to Dbase 3 Plus [bilgisayar dosyası]. — [y.y.] : Ashton-Tate, [1986?]

Hayes, Robert Mayo. Handbook of data processing for libraries / Robert M. Hayes. — 2nd ed. — Los Angeles : Melville Publishing Company, 1974

Heuer, Andreas. Objektorientierte datenbanken : Konzepte, Modelle, Systeme / Andreas Heuer. — Bonn... [ve öte.] : Addison-Wesley, 1992

Inside IBM's Distributed Data Management architecture / by R. A. Demers... [ve öte.]. — s. 459-487. — IBM Systems Journal. — vol 31, No. 3 (1992)

İlişkisel sistemler ve diğerleri. — s. 38-44 . — Bilgisayar . — (Temmuz 1993)

Oxborrow, Elizabeth A. Databases and database control systems : concepts and issues / Elizabeth Oxborrow. — 2nd ed. — Bromley, Kent, England : Chartwell-Bratt, c. 1989

Paradox for Windows. Getting started. — Scotts Valley, California : Borland International, 1992

Paradox for Windows. Guide to ObjectPAL. — Scotts Valley, California : Borland International, 1992

Paradox for Windows. User's guide. — Scotts Valley, California : Borland International, 1992

Sauer, Hermann. Relationale Datenbanken : Theorie und Praxis, inklusive SQL-2 / Hermann Sauer. — 2. korrig. u. erw. Aufl. — Bonn... [ve öte.] : Addison-Wesley, 1992

Türkçe sözlük. — genişl. 7. bs. — [Ankara] : Türk Dil Kurumu, 1983

Wittgenstein, Ludwig. Tractatus logico-philosophicus =
Logisch-philosophische Abhandlung / Ludwig Wittgenstein. —
Frankfurt am Main : Suhrkamp Verlag, c. 1963, 1989

Yılmaz, Adnan. Veri tabanına dayalı Türkçe kütüphane
otomasyonu yazılımı : KYBELE / haz. Adnan Yılmaz, Hüryasa
Aslan. — s. [10]-17. — Türk Kütüphaneciliği. — [c.] 6,
[s.] 1 (1992)

Yontar, Aysel. Bilgi bilime giriş [ders notları]. —
[İstanbul] : [y.y.], [1992]. — Not : Metin teksirdir