

# XML VE VERİTABANI ENTEGRASYONU İLE ESNEK UYGULAMA GELİŞTİRME

Hale AS, Engin ERDEM, Ata ÖNAL

Ege Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 35100-Bornova/İzmir

## ÖZET

Bilişim alanında son yılların gözde konularından biri Kurumsal Uygulama Tümeleşimi'dir. Kurumsal Uygulama Tümeleşimi basit bir şekilde, bir organizasyonun bilgi işlem ortamını oluşturan farklı sistemler arasında standart köprüler oluşturulması süreci olarak tanımlanabilir. Kurumsal Uygulama Tümeleşimi'nin popüler olmasının nedeni, şirket birleşme, satın alma ve evlilikleri olmuştur. Bu tür ilişkiler içinde olan şirketlerde çalışmakta olan benzer sistemlerin bir süre daha çalışması sağlanırken, hedef en kısa vadede şirketler arasındaki uygulama entegrasyonunun çözülmesi olmuştur. XML teknolojisi ile hem kurum içindeki hem de kurumlar arasındaki uygulama entegrasyon sorunlarına çözüm bulunabilir.

**Anahtar Kelimeler :** XML, Veritabanı, Entegrasyon

## DEVELOPING FLEXIBLE APPLICATIONS WITH XML AND DATABASE INTEGRATION

### ABSTRACT

In recent years the most popular subject in Information System area is Enterprise Application Integration (EAI). It can be defined as a process of forming a standart connection between different systems of an organization's information system environment. The incorporating, gaining and marriage of corporations are the major reasons of popularity in Enterprise Application Integration. The main purpose is to solve the application integrating problems while similar systems in such corporations continue working together for a more time. With the help of XML technology, it is possible to find solutions to the problems of application integration either within the corporation or between the corporations.

**Key Words :** XML, Database, Integration

## 1. GİRİŞ

Bilgisayarların doğuşundan itibaren verilerin saklanması gerekliliği önemli bir problem teşkil etmiştir. Basit dosyalama yöntemleri ile başlayan bu tarihsel süreç, günümüzde mükemmel özellikleri olan, çok kuvvetli veritabanlarına kadar uzanmaktadır. Yakın tarihte veri saklama teknolojilerinde yaşanan olumlu gelişmeler, veritabanı sistemlerinin kullanımını yaygınlaştırmıştır. Günümüzde birçok işletme, kurum veya kuruluş kendi bünyesinde farklı uygulamalarda değişik veri saklama teknolojilerini

kullanmaktadır. Farklı veri saklama teknolojileri beraberinde entegrasyon sorununu da gündeme getirmekte, örneğin bu ve benzeri sistemlerde birbiriyle veri paylaşımı söz konusu olduğunda önemli sorunlar ortaya çıkmakta, beraberinde yüksek maliyeti de getirmektedir. Bilişim alanında son yılların gözde konularından birisi de Kurumsal Uygulama Tümeleşimi'nin getirdiği sorunlar olmuştur. Burada hazırladığımız çalışma ile hem kurum içindeki hem de kurumlar arasındaki uygulama entegrasyon sorunlarına çözüm bulmayı amaçlamaktayız.

## 2. XML

XML, eXtensible Markup Language sözcüklerinden türetilmiş bir kısaltmadır (Anon., 2002). Yani Genişletilebilir “Markup” Dili demektir. XML, bağımsız bir markup dili olmaktan ziyade bir meta dildir, yani kullanıcıların kendi markup bileşenlerini oluşturmalarını sağlayacak bir araçtır. Web üzerinde giderek artan bir hızla veri alışveriş standardı haline gelen XML, metin-tabanlı bir markup dildir. Veriler tag’lar kullanılarak betimlenir (betimleyiciler <...> şeklinde büyük küçük sembolleri arasında yer alır. Genellikle bu tag’lar, “markup” olarak bilinir. Ancak HTML’den farklı olarak, XML tag’ları verinin ne şekilde görüntüleneceğini değil, verinin kendisini betimler. XML tag’ları program kodundaki bir saha ismi gibi davranır. Veriyi betimleyici bir etiket işlevi görür (örneğin: <mesaj>...</mesaj>). Verinin betimlenmesi, verinin ne anlama geldiğini yolunda bize ipuçları verdiğinden, XML verinin anlamını belirleme mekanizması olarak da nitelenebilir.

Bir veri yapısı için saha isimleri tanımlarken olduğu gibi, belirli bir uygulama için bize anlamlı gelen istediğimiz her XML tag’ını kullanmakta özgürüz. Eğer, bir çok uygulama aynı XML verisini ortak olarak kullanmak durumundaysa, kullanacakları ortak tag isimlerini belirlemek zorundadırlar. Örneğin; bir mesajlaşma uygulaması için kullanacağımız XML verisi şu şekilde olabilir:

```
<mesaj>
  <kime>sen@seninAdresin.com</kime>
  <kimden>ben@benimAdresim.com</kimden>
  <konu>XML</konu>
  <metin>
XML, gerçekten çok etkileyici!
  </metin>
</mesaj>
```

Bu örnekteki taglar, mesajı tümüyle betimler; kimden, kime, konu, metin. Taglar arasındaki veri, XML verisinin bir “element”ini tanımlar. Aynı zamanda, <kime> tagı, <mesaj>..</mesaj> tagları tarafından kapsanır. Bu yöntem, XML’in verileri hiyerarşik veri yapıları şeklinde bulundurmasını getirmiştir.

XML’in getirilerinden en önemlisi; veriye kendi kendini tanımlayabilir bir yapı kazandırmasıdır. XML verisi belirli tanımlama tabloları, farklı veri tanımlama araçları gibi unsurlara gerek duymaz çünkü verinin tanımı XML’in kendi içerisinde bulunmaktadır. XML, verinin sadece görüntülenmesinin yanısıra tamamı ile kullanılabilir kılınmasını sağlamaktadır. Bu özelliği ile XML, özellikle verinin işlenmesine yönelik iş uygulamaları

için önemli bir avantaj sunmaktadır. Ayrıca kullanıcıların kendi taglarını oluşturma imkanına sahip olmaları, XML’in genişleyebilirliğinin önemli bir göstergesidir. XML dokümanları, metin bazlı verilerden ses ve görüntü içeren çoklu ortam verilerine kadar bir çok farklı veri tipini içerebilmektedir. Bu veri tiplerinin birlikte bulunduğu veri depolarında, özellikle iş amaçlı veriler içerisinde etkin ve çok detaylı arama işlemlerini hızlı şekilde gerçekleştirmek XML sayesinde mümkün olacaktır. Bunların dışında, XML’in en önemli özelliklerinden biri de W3C tarafından bir standart olarak kabul edilmesidir (Anon., 2000). Özellikle iş dünyasında, birbirleri ile işlem yapacak dolayısıyla veri değişimi gerçekleştirecek işletmelerin veya kurumların birbirleri ile aynı dili konuşmaları sadece standart bir dilin oluşması ile mümkün olacaktır. Aksi halde SGML’de (Standard Generalized Markup Language) görülen sorunlar ortaya çıkabilecektir.

XML, genel yapısı ve özellikle dünyada çok yaygın kabul görmesinin yanı sıra standart olarak kabul edilmesi ile birlikte işletmelere çok daha verimli etkileşim imkanı verecektir. XML’in kökleri doküman yönetimine dayansa da, birçok farklı alanda uygulamaları mevcuttur. Tedarikçiler, müşteriler, iş ortakları ve B2B zincirindeki diğer kurumlar arasında bilgi paylaşmak isteyen şirketlere XML büyük avantaj sağlar (Anon., 2002). Birçok kurum ve kuruluş değişik amaçlar için XML’i kullanmaya ve avantajlarını görmeye başlamıştır. XML’in belirli bir sahibi yoktur. XML, herkese açık bir kod olmasından dolayı belli bir lisans sahibi veya geliştiricisi olmadığından, herkesin kolaylıkla ve isteyerek kullanabileceği bir yapıya sahiptir. Kullanıcıya yakın yapısı ve kullanımının son derece kolay olması da geniş kabul görmesinde önemli rol oynayan faktörlerden birisidir. Artan rekabet koşullarında yapılan yatırımın geri dönüşünün ve bundan da öncelikli, doğru yatırımın gerçekleştirilmesinin önemi son derece büyüktür. XML, gerçekleştiriminden kullanıma kadar sunduğu bir çok avantaj ve yürütülmesi açısından EDI (Electronic Data Interchange) gibi çok karmaşık ve maliyeti yüksek sistemlere ihtiyaç duymadığından yapılan yatırımın çok daha etkin geri dönüşüne imkan vermektedir.

Özetle;

XML bilgi alışverişi için „tek“ açık standarttır. XML her tür bilgiyi tanımlamak için tasarlanmıştır. XML teknolojisinin öncülüğü son kullanıcılar tarafından yapılmakta ve bilişim sektöründeki büyük firmaların çoğu tarafından desteklenmektedir. XML dokümanları, uygulamalar ve kullanıcılar tarafından kolaylıkla anlaşılabilir bir özelliktedir.

XML tanımlamaları baz alınarak işletme sınırları aşılabildiği bilgisi sağlanmıştır.  
XML programlama dili ve platformdan bağımsızdır.

### 3. VERİTABANI SİSTEMLERİ

Veritabanı bilginin hammaddesinin depolandığı yer olarak tanımlanabilir. Bilgiyi oluşturan bileşenleri sınıflara ayırıp aralarındaki ilişkileri de dikkate alarak depolar, istendiği zaman birleştirerek sunar. Bilgi teknolojileri ile ilgili hemen her konunun bileşenlerinden birini veritabanı oluşturmaktadır. Kişisel faks yazılımlarından, en gelişmiş sunucu sistemlerine, kurumsal kaynak planlaması (Enterprise Resources Planning- ERP) yazılımlarından, doküman arşiv sistemlerine, CAD/CAM uygulamalarından, veri ambarı uygulamalarına, bankacılıktan, elektronik sektörüne, üniversitelerin kayıtlanma sistemlerinden, hastane otomasyon sistemlerine ve internet üzerindeki arama motorlarına kadar her sistemin içerisinde mutlaka bir veri tabanı bulunmaktadır. Hatta işletim sistemleri bile sistem bilgilerini özel yapıları olan veritabanlarında tutmaktadır (Anon., 2002).

Veritabanı, içeriği kolaylıkla erişilebilir, yönetilebilir ve güncellenebilir bir şekilde düzenlenmiş veri topluluğu olarak da tanımlanabilir. Diğer bir ifadeyle; veritabanı, birbiriyle bağlantılı tablolardan oluşan ve tüm verilerin sadece tablolar olarak saklanabileceği sistemdir. Veritabanı sistemleri kullanım kolaylıklarını da beraberinde getirmektedir. Gelenekselleşmiş dosyalama sistemlerine göre yedekleme ve bakım kolaylığının yanı sıra güvenlik ve performans ile ilgili pek çok avantajlar sunmaktadır.

Veritabanları genel anlamda üç değişik grupta toplanabilir. Birincisi kişisel diyebileceğimiz veritabanı sistemleridir. Bunlar içerisinde Microsoft Office Professional paketi içerisinde bulunan Access ve bunun dışında dBase, FoxPro, Paradox sayılabilir. Bu veritabanı sistemleri diğerleri ile karşılaştırıldığında dezavantajları olduğu tartışma götürmez bir gerçektir. Ancak hataları nedeniyle değil kapasitelerinin yetersizliği nedeniyle böyle bir durum söz konusudur. İkinci grup ise İlişkisel Veritabanları (Relational Database) adıyla bilinen uygulamalardır (İlişkisel, 2001). İlişkisel veritabanları bilgiyi saklama, işleme, yedekleme, raporlama ve geri getirme konularında çözümler getirmektedir. Kurumsal firmaların tercihi bu sistemlerdir. Bugün bilinen en popüler ilişkisel veritabanları arasında Oracle, DB2, Sysbase, Informix, Progress ve Microsoft SQL Server sayılabilir. Veritabanı ile ilgili üçüncü ve son grup,

kurumların çok büyük boyutlu veri tabanı analizlerine dayalı gereksinimlerini karşılamak amacıyla kurduğu Veri Ambarı (Datawarehouse) türü teknolojilerdir. Karmaşık veriler ve bu veriler arasındaki analize dayalı teknolojiye OLAP (Online Analytical Processing) işlemleri veri ambarında kullanılmaktadır. İlişkisel veri tabanı kuruluşlarının neredeyse tamamının, veri ambarı çözümü bulunmaktadır. Geleneksel anlamdaki veri ambarı projeleri yalnızca veritabanı sistemlerini değil, donanım ve işletim sistemini de içine almaktadır. Ancak günümüze gelindiğinde tüm eğilimler, veri ambarı kavramının ortam bağımsız bir şekilde günden güne internet'e dayalı teknolojiler üzerinde çalıştığını göstermektedir.

Geçmişten günümüze yazılım istekleri değişime uğramış olsa bile, onlardan beklenen hizmet hep aynı kalmıştır. O da "Hemen şimdi bilgi" olarak ifade edilebilir. Veritabanı dendiğinde yalnızca verilerin depolandığı bir kavram algılamak çok doğru bir yaklaşım olmayacaktır. Veritabanı, verilerin bilgiye dönüştürülerek diğer verilerle birlikte işlendiği kocaman bir fabrikadır denilebilir.

Kurumsal firmaların kullandıkları veri tabanları bazen gömülü bazen de açık olmaktadır. Kuruluşların ihtiyaçlarına göre veritabanı üzerinde yeni yazılımlar geliştirmek mümkündür. Çok gelişmiş bir uygulama geliştirme aracınız olabilir ama veritabanı sisteminiz iyi değildir ya da veritabanı sisteminiz çok iyi olmasına rağmen uygulama geliştirme aracınız istediklerinizi karşılayamayabilir.

### 4. İLİŞKİSEL VERİTABANI YÖNETİM SİSTEMLERİ

İlişkisel Veritabanı Yönetim Sistemleri, büyük miktarlardaki verilerin güvenli bir şekilde tutulabildiği, bilgilere hızlı erişim imkanlarının sağlandığı, bilgilerin bütünlük içerisinde tutulabildiği ve birden fazla kullanıcıya aynı anda bilgiye erişim imkanının sağlandığı programlardır.

İlişkisel veritabanlarının genel özellikleri şunlardır:

- Büyük miktarlarda veri tutabilmekte ve verilerin depolandığı alanları ayarlama imkanı vermektedir.
- Aynı anda çok sayıda kullanıcıya verilerin bütünlüğünü bozmadan hizmet verebilmektedir.
- Günün 24 saati ve haftalar boyu aralıksız çalışabilmektedir.

- Yetkisiz erişimleri engelleme ve kontrol edebilme gibi güvenlikle ilgili imkanları sağlamaktadır.
- İstemci/Sunucu mimarisinin bütün avantajlarını kullanabilmektedir.
- Birçoğu kendi uygulama geliştirme araçlarına sahiptir.

## 5. ENTEGRASYON SORUNLARI VE ESNEK UYGULAMA GELİŞTİRME

Farklı veri saklama teknolojileri entegrasyon sorununu gündeme getirmiştir. Örneğin farklı sistemlerin birbiriyle veri paylaşımı söz konusu olduğunda önemli sorunlar ortaya çıkmakta, beraberinde yüksek maliyeti de getirmektedir. Bilişim alanında son yılların önemli konularından biri de “Kurumsal Uygulama Tümüleşimi” olmuştur. Kurumsal Uygulama Tümüleşimi basit bir şekilde, bir organizasyonun bilgi işlem ortamını oluşturan farklı sistemler arasında standart köprüler oluşturulması süreci olarak tanımlanabilir. Kurumsal Uygulama Tümüleşimi'nin gündeme oturmasındaki temel neden, şirket birleşme, satın alma ve evlilikleri olmuştur. Bu tür ilişkiler içinde olan şirketlerde çalışmakta olan benzer sistemlerin bir süre daha çalışması sağlanırken, hedef en kısa vadede şirketler arasındaki uygulama entegrasyonunun çözülmesi olmuştur. XML teknolojisi ile hem kurum içindeki hem de kurumlar arasındaki uygulama entegrasyon sorunlarına çözüm bulunabilir. Kurum içinde, farklı veri saklama teknolojilerini kullanan uygulamalar gelecekte meydana gelebilecek entegrasyon sorunlarına farklı seviyelerde tepki göstermektedir. Ortaya çıkan bu tepki, uygulamada kullanılan veri saklama teknolojisinin birbirlerine verdiği destek çerçevesinde değişken olmaktadır. Günümüzde böyle bir uygulamayı tasarlarken, farklı veri saklama teknolojilerinin sunduğu bu desteklere dikkat etmeliyiz. XML bu açıdan birçok olumlu desteğe sahip olan bir teknolojidir. Saklanan verilerin durağan olduğu durumlarda, veri depolama, veritabanı sistemleri aracılığıyla yapılırken, uygulamaların birbiriyle entegre olması gerektiği durumlarda veriler XML yapılarında tutulmalıdır. Hem entegrasyonun hem de durağanlığın ön plana çıktığı durumlarda ise ikisinin de kullanımı kaçınılmaz olarak görülmektedir. Böylece farklı uygulamaların veri paylaşımı kolaylaşarak karşılaşılan entegrasyon sorunları minimuma indirilmiş olur.

## 6. SBS (SATINALMA BİLGİ SİSTEMİ)

Yukarıda belirtilen entegrasyon sorunlarıyla, her kurum içi veya kurumlar arası entegrasyonda olduğu gibi Ege Üniversitesi Satın Alma Bilgi Sistemi'nin (E. Ü. SBS) tasarlanması sırasında da karşılaşılmıştır. Satın Alma Bilgi Sistemi, hizmet ya da malzeme taleplerinin ortaya çıkışı, bu taleplere ilişkin bütçe ve stok kontrolü, gerektiği durumlarda ihaleye çıkılması, ardından da talep sahiplerine hizmet ya da malzemenin arz edilmesini kapsayan bir sistemdir. Satın alma işlevi, alt sistem bazında da olsa kurulup işletilmesi gereken bir sistemdir. Önemli olan ortaya konacak sistemin gelecekte kurulacak büyük sistemlere kolayca entegre olabilecek bir yapıda oluşturulmasını sağlamaktır. E.Ü. SBS, bu ilkelerle hareket etmektedir. Satın Alma Bilgi Sistemi, diğer tüm kurumsal bilgi sistemleri ile entegre bir şekilde çalışabilmelidir. Bunun gerçekleştirilebilmesi için bilgi sistemlerindeki teknolojinin sunduğu olanaklardan yararlanılmalıdır. Sadece teknolojik yenilikleri sistemlere dahil etmek değil, aynı zamanda genişletilebilir yapıda olmalarını sağlamak da önemlidir (Cavkaytar, 2002).

Gelecekte SBS'yi büyük bir bilgi sisteminin küçük bir bileşeni olarak göreceğiz. Bu, şu şekilde de ifade edilebilir; sistemler birbirleri ile öylesine entegre bir şekilde çalışıyor olacaklar ki, onları alt sistemler halinde soyutlamak oldukça güçleşecektir. Ege Üniversitesi Satın Alma Bilgi Sistemi, bildiride ele alınan kapsamı açısından tam anlamıyla bir B2B (Business to Business - Kurumlar arası iş) uygulaması olarak nitelenirse de, proje bir bütün olarak ele alındığında E.Ü. SBS'nin bir B2B uygulaması olarak nitelendirilmesi gerektiği rahatlıkla söylenebilir (Anon., 2003). Kısaca, E.Ü. SBS büyük bir resmin küçük bir parçasıdır. Örneğin, Satın Alma Bilgi Sistemi'nin, gelecekte Stok sistemiyle entegrasyonu kaçınılmaz bir durumdur. Bu sistem tasarlanırken verilerin XML yapılarında saklanması uygun görülmüştür. Ancak XML, güçlü bir veritabanı teknolojisini sunduğu olanakları, bazı güçlü uygulamalar olmadan verememektedir. Yani XML teknolojisi, bir veritabanı teknolojisi değildir. XML'e bu özelliği katabilmek için bazı teknolojilerin yardımını almak gerekmektedir. Bu kapsamda sadece COM+ teknolojisi ele alınacaktır.

## 7. COM+

COM+, COM (Component Object Model) ve MTS (Microsoft Transaction Server) programlama modellerinin birleşimi ile yeni özellikler eklenerek

oluşturulmuş Microsoft'un yeni nesil bileşen tabanlı uygulama geliştirme metodolojisidir (Anon., 2002). COM ilk tanımlandığı sırada iş istasyonu bazında çalışması planlanmıştı. DCOM'un (Distributed COM) piyasaya sürülmesi ile ağ üzerinden nesne yaratma ve metod çağırma olanaklı hale getirildi. Daha sonra MTS, sunucu tarafı bileşen yönetimi özelliklerini sağlamak ve DCOM'un bazı yetersizliklerini gidermek amacı ile geliştirildi. DCOM'un yetersizliklerine örnek olarak güvenlik ile ilgili konular, bileşen yönetimi ve konfigürasyonu ile ilgili zorluklar gösterilebilir. Microsoft COM+ ile tüm bu teknolojilerin birleştirilmesinin yanı sıra kuruluş bazında uygulama geliştirme metodolojisine de bir standart getirmektedir.

COM+ ve Component Object Model (COM) birbirine paraleldir. COM+, geliştiricilerin işini kolaylaştırmak için, herhangi bir dilde, herhangi bir aracın kullanımı içinde yazılım öğeleri oluşturmak ve kullanmak için, COM'un bütünleşmiş hizmetleri ve özellikleri üzerine inşa edilir.

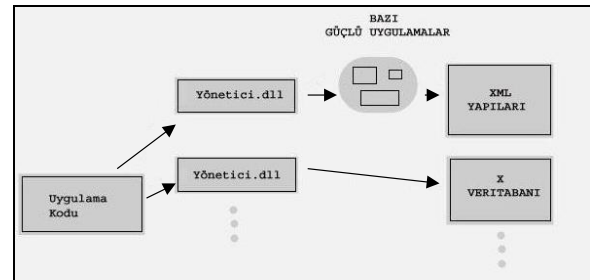
Satınalma Bilgi Sistemine esnek bir yapı kazandırabilmek için verilerin XML yapılarında saklanması ve bu işin başlı başına veritabanı mantığında yürütülmesi uygun görülmüştür. İşte bu kapsamda COM+ teknolojisinin kullanımı söz konusu olmuştur. Çünkü XML tek başına, güçlü veritabanı sistemlerinin sunduğu olanakları içermemektedir. Örneğin; SBS'nin işlem akışı sırasında karşılaşılan en önemli sorun, bir XML dosyasının kilitlemesi olmuştur. İşlemler sırasında, değiştirilen dosyanın kilitlemesi gerekmektedir. Veritabanı yönetim sistemleri kilitleme işlemini otomatik olarak gerçekleştirir. XML dosyası temelde bir metin tabanlı dosya niteliği taşıdığı ve XML okuma yazma işlemlerinde DOM (Document Object Model) metodu kullanıldığı için bir kullanıcı bu dosyaya eriştiği zaman diğer kullanıcıların erişiminin engellenmesi gerekmektedir. Sistemin işleyişi sırasında ilgili COM+ nesnesi çalıştırılarak gerekli isteme yanıt verilir ve dosyanın kilitlemesi sağlanır ve gelen istemler başka bir COM+ nesnesi sayesinde sıraya alınarak teker teker cevap verilebilmektedir. Aynı anda birçok kullanıcının verileri işleme isteğini karşılamak çeşitli sorunları da beraberinde getireceğinden COM+ teknolojisi gibi bir teknolojinin kullanılması kaçınılmazdır.

## 8. SONUÇ

Bir yandan ilerde karşılaşılabilecek kesin olan entegrasyon hali, diğer yandan da verilerin sistemde hızlı ve güvenilir bir şekilde işlenebilir olması gerekliliği bizi farklı arayışlara itmiştir. Çözüm

olarak verilerin farklı veri saklama teknolojileri üzerinde minimum maliyetle saklanması fikri doğmuştur. Şekil 1'i incelediğimizde bu yaklaşımın temellerini görebiliriz. Yani veriler aynı anda hem bir veritabanı sistemi üzerinde hem de XML yapılarında saklanabilecektir.

Sistem işleyişi süresince, uygulama kodu kısmı sabit bir şekilde kalacak, sistem yerel olarak çalışacağı zaman X Veritabanını, entegrasyon halinde ise XML yapılarını kullanacaktır. Böylelikle aynı anda hem veritabanı sisteminin hem de XML yapılarının işleyişi sağlanabilir. İkinci katmanda görülen DLL'ler (Dynamic Link Libraries - Dinamik Link Kütüphaneleri) sisteme yeni veritabanı sistemleri eklense bile ana çatılarını koruyacaklardır. Ana çatı kavramından kasıt sahip oldukları metod isimlerinin aynı olmasından başlayıp, bu metodların aldığı parametrelerin, bu parametrelerin tiplerinin ve bu metodlardan geriye dönen değerlerin tiplerinin bire bir olmasına kadar uzanan geniş bir yelpazedir. Sistemde dinamik olan bir başka yapı da XML yapılarıdır. Entegrasyon durumlarında en önemli rolü XML yapıları üstlenirler. Bazı güçlü uygulamalar COM+ gibi teknolojilerin yardımı ile, XML yapılarının işlenişi sırasında hızlı işlenebilirlik, güvenilirlik vb. kavramlar açısından XML yapılarına destek vermektedir. XML'e günümüzde birçok veritabanı sistemi tarafından destek verilmektedir. Eğer, uygulamaya yeni veritabanı sistemleri de eklenmek istenirse, ikinci katmandaki DLL'lere yeni bir DLL eklenerek, uygulama kodunu değiştirmeden bu soruna çözüm getirilebilir.



Şekil 1. Farklı veri ortamlarının entegrasyonu

Ege Üniversitesi Satınalma Bilgi Sisteminin tasarlanması sırasında ortaya çıkan entegrasyon probleminin çözümüne yönelik olarak geliştirilen bu çalışma diğer kurum, kuruluş ve şirketlerde kullanılan uygulamalar tarafından, herhangi bir kurum içi veya kurumlar arası entegrasyon sırasında karşılaşılabilecek muhtemel sorunlara önceden çözüm üretilen olacaktır.

## 9. KAYNAKLAR

- Anonim, 2002. Veritabanı Yönetim Sistemleri, <http://www.bilgiyonetimi.org>.
- Anonim, 2001. İlişkisel Veritabanı Yönetim Sistemleri, <http://www.programcilik.net>.
- Anonim, 2000. W3C, XML Specifications, <http://www.w3c.org/XML>.
- Anonim, 2002. COM+, <http://www.aptech-turkiye.com>.
- Anonymous, 2002. XML, <http://www.softwareag.com>.
- Anonymous, 2003. B2B Fundamentals, <http://www.communityb2b.com/library/fundamentals.cfm>.
- Cavkaytar, H., Önal, A. 2002. XML & Java Teknolojilerini kullanarak gerçekleştirilen bir B2B projesi, Ege Üniversitesi Fen Bilimleri Enstitüsü (Yüksek Lisans Tezi).