

## E - DÖNÜŞÜMLERİN TEMELİNİ OLUŞTURAN İŞLEMLERİN DEĞİŞİK SUNUCULAR ÜZERİNDE ÇALIŞMA SÜRELERİNİN KARŞILAŞTIRILMASI

Benian TEKİNDAL\*

Fırat ERDOĞAN\*\*

### Öz:

Klasik web sayfalarında kullanılan statik içeriklerin dinamiğe dönüştürülmesi ve tüm sektörlerin işlerini bu platform üzerinde sürdürmesi tüm maliyetleri ve harcanan zamanı azaltmaktadır. Bu çalışmada, nesne tabanlı programlama, standart programlama ve fonksiyon-prosedür tabanlı programlama teknikleri ile geliştirilen ve e-dönüşümlerde sıklıkla kullanılan veritabanlarıyla etkileşime geçme, dosya sistemleri, anlamlı grafikler üretme işlemlerini kapsayan kod düzenleri, dinamik programlama yöntemi çerçevesinde ve PHP web programlama diliyle hazırlanarak çeşitli platformlarda performans testine tabi tutulmuştur. Her bir işlem için geliştirilen kod düzenleri, donanımı ve sistem yazılımları farklı olan üç adet sunucu bilgisayarda ve farklı zamanlarda 1000'er kez çalıştırılmıştır. Kod düzenlerinin çalıştırılması sonucunda, performansı belirlemek için her adımda kodların çalışma süreleri mikro saniye cinsinden elde edilmiş ve sunucu bilgisayarlarda depolanmıştır. Elde edilen bu verilere öncelikle homojenlik testi uygulanmıştır. Kodların çalışma sürelerinin, sunucu bilgisayarlara ve oluşturulan kod düzenine ne kadar bağlı olduğu ve oluşturulan kod düzenlerinden hangisinin daha iyi sonuç verdiği faktöriyel düzende varyans analizi tekniği ile değerlendirilmiştir. Değerlendirme sonucunda istatistik olarak önemli olan ortalama farklılıklarının hangi alt grupların lehine olduğu, ortalamalar arasındaki farklılıkları ortalamaların sıralamadaki konumunu dikkate alarak değerlendirmeyi amaçlayan Duncan çoklu karşılaştırma testi ile saptanmıştır. Yapılan analizler sonucunda, e-dönüşümler için hazırlanmış web sayfalarında nesne tabanlı kodlamanın kullanılması ve sunucu bilgisayarların Linux işletim sistemine sahip olması gerçekleştirilen işlemin süresini kısalttığı görülmüştür.

---

\* Yrd. Doç.Dr, Gazi Üniv, End. Sanatlar Eğitim Fakültesi, Bilgisayar Eğitimi, [benian@gazi.edu.tr](mailto:benian@gazi.edu.tr)

\*\* Öğretmen, Kırıkkale Gazi Endüstri Meslek Lis., Bilişim Teknolojileri [firat\\_erdogan@msn.com](mailto:firat_erdogan@msn.com)

**Anahtar Kelimeler:** E-dönüşüm, kod düzeni, web sayfası, nesne tabanlı kodlama

**THE COMPARISON OF WORKING TIMES BELONGING TO THE PROCEDURES WHICH FORM THE FOUNDATION ON VARIOUS SERVERS**

**Abstract:**

*Transformation of static contents used on the classic web pages and all sectors' maintaining their studies on that platform lessen all the costs and used time. Users have problems, during the usage of internet platform, on the order of knowledge settlement and on the reaching to the knowledge in a fast way. On the success of e-transformation platform, the role of code project is important because of this point of view. At this project, the schedule based on object, standart programme and interaction with databases used oftenly for e-transformation and improved with function-procedure based programme techniques, file systems, code orders covering meaningful graphic production procedures have been subjected to performance test on various platforms by being prepared with PHP web programme language and at the frame of dynamic programme method. Code orders improved for each operation were operated 1000 (a thousand) times at three server computers whose rigging and system software were different from each other. As a result of operation of code orders, working times of codes have been examined in terms of micro-seconds and stored in server computers. In order to make that clear, internet usage density and in order to bring the obtained results to safe mode, sample procedures were operated to the code orders two times, totally, at the time which is the most probably dense usage times of internet 09:00-16:00 hours and the most probably rare usage times of internet 02:00-06:00 hours. Firstly, homogeneity test has been operated to the obtained results. How about the dependence on mode code order and to server computer of codes' working times and the which code order has given the better result issues were examined on factorial order by the help of variance analyze technique. On the consequence of calculation made by variance analyze technique on the factorial order, which sub-groups are advantageous in terms of important average varieties has been obtained statically with Duncan alternative comparison test aiming to examine the differences between averages by analyzing the place of averages at the enumeration. As the result of made analyze, on the issues of connection to MySQL database and the operation of SQL inquiry, it has been seen that Linux operating system's servers were the fastest. During the realization of the procedure, the hardware qualities of server computers are not as effective as used system and appliance softwares. Again, at the SQL inquiries' operation, while web pages were programmed, the usage of object-*

*based code order caused to be operated faster than the other code orders. On the procedure of file system and interaction with servers, it has been seen that hardware qualities of server computer were not as effective as used system and appliance software. It has been seen that the usage of each code order is suitable for this procedure. In web placement, at the procedure of solutionary graphics production, Linux operating system servers have been the fastest server types. On the programming of this procedure, the usage of object-based code order or functional code order increased the speed of realization of procedure. All the servers for the registration adding treatment to the database were labelled as direct proportional with the density of internet usage. For installing file treatment to the servers, the servers having Windows 2003 Server operating system were labelled as direct proportional with the density of internet usage. It was found that internet usage was unimportant in order to produce solutionary graphic on the web environment. As the result of this examination, it has been seen that the usage of object-based coding on the web - pages and server computers possessed Linux operating system shortens the time of procedure.*

**Keywords:** E-transformation, code order, web page, object oriented coding

## **GİRİŞ**

Bilgi ve iletişim teknolojilerini etkin kullanarak, mevcut kültürün iş modelinin, iş süreçlerinin, ürün ve hizmetlerin; tüm sosyal statülerin yararına, bir bütünlük içerisinde değiştirilmesi sürecine e-dönüşüm denilmektedir. E-dönüşüm, birey, iş, ticaret, kurum ve devlet dönüşümlerini kapsamaktadır.

E-dönüşüm aşamasını tamamlamış yapılarda, tüm hizmetler elektronik ortamda çevrimiçi olarak sağlanır, kurum veya kuruluş odaklı olarak yürütülen her hizmet, vatandaş odaklı olarak yürütülmeye başlanır. Bilgiye ve hizmetlere web üzerinde tek bir kaynaktan erişilir.

Bir toplumun, bilgi toplumuna geçtiğini gösteren ölçütlerden bir tanesi bilgisayar kullanımı ile toplam nüfus oranıdır. İkinci ölçüt internet kullanım düzeyidir. Üçüncü ölçüt ise bilgi kesiminde çalışan işgücünün toplam nüfusa oranıdır (Oktay, Balkanlı, Salepçioğlu; 2004 : 157). Teknik altyapı göstergeleri kadar, ülkelerin hizmetlerin sunumu ve geliştirilmesindeki durumu, teknoloji üretme yetenekleri, insan sermayesi, hukuki düzenlemeleri gibi pek çok ölçüt mevcuttur (DPT; 2005).

Yaygın olarak hemen her işlevsel nesne “e” öneki ile anılmaya başlanmıştır; e-sağlık, e-vatandaş, e-kurum vb. işlevinde bilgi ve iletişim teknolojilerini yaygın olarak kullanmaya aday her nesne “e” öneki ile anılabilir. Ancak “e” ile karşılaşıldığında anlaşılması gereken olgu, yalnızca elektronik ya da bilgi ve iletişim teknolojileri değil,

bu alt yapının, toplumun her kesiminde yaygın olarak kullanımının oluşturacağı sosyal, ekonomik ve kültürel değişimler de anlaşılmalıdır.

World Wide Web ortamı, e-dönüşüm platformu olarak karşımıza çıkmaktadır. Bu ortamda web sayfaları yoğun olarak kullanılmaktadır. Web sayfalarının etkili, verimli tasarlanmış ve programlanmış olması çok önemlidir. E-dönüşüm yol haritasında son adım budur. E-dönüşümün başarısını direkt etkileyebilecek iki faktörden biri web sayfalarının verimli programlanması diğeri de web sayfalarındaki estetik belirleyicilerdir (Yılmaz; 2001 : 108).

E-dönüşümün gerekliliği üç temel unsurla açıklanabilir;

- Bilgi Ekonomisi: Bilgiye dayalı ekonomi, ekonomik getiriye dönüşen ve dönüştürülecek bilgi ve teknolojik yeniliğin üretilmesidir (Arifoğlu; 2004 : 9).
- Bilişimsizlik Maliyeti: Bilişim teknolojilerinin hiç kullanılmamasından ya da etkin ve verimli olarak kullanılmamasından doğan maddi kayıplar bilişimsizlik maliyeti olarak ifade edilmektedir (Arifoğlu; 2004 : 16).
- Sayısal Uçurum: Sayısal uçurum, farklı coğrafi alanlarda bulunan ve farklı sosyo-ekonomik düzeyde olan kişiler, yerleşim merkezleri ve iş ortamlarının bilgi ve iletişim teknolojilerini ve interneti, kendi yaşamlarında kullanım düzeyleri arasındaki farklılık olarak tanımlanmaktadır (Arifoğlu; 2004 : 40) ve Telekomünikasyon Kurumu, 2005). Sayısal uçurum, yeni teknolojilerin yavaş yayılımından dolayı oluşur (Closing The Dijital Divide; 1999).

E-dönüşümde güvenlik dijital imzaların yaygınlaştırılmasıyla sağlanabilir. Dijital ortamda gönderilen belgelerin hangi kuruma ya da kişiye ait olduğunu doğrulayan dijital imza, ıslak imza ile eşit statüye sahip bir uygulamadır (DPT; 2006). Sertifika otorite yazılımları dijital imzaların uluslararası geçerliliğini sağlamaktadır (TBD Kamu-Bib 2. Çalışma Grubu; 2005).

## **I) WEB SİTELERİNİ DİNAMİK PROGRAMLAMA TEKNİKLERİ**

### **A) Standart Programlama Tekniği**

Standart programlama tekniğinde yazılımlar, ardı ardına ilerleyen ve yazılımın geliştirildiği dilin gömülü fonksiyon ve yordamlarını kullanmaya yönelik olarak geliştirilir. Basit işlem ve tekrar kullanımı söz konusu olmayan yazılımlarda pratik olması bakımından tercih edilmektedir. Yazılım bileşenleri, daha çok akış kontrol deyimleri, döngüler ve bol miktarda normal ve bir veya çok boyutlu dizi değişkenlerden oluşmaktadır. Kullanılan değişkenlerin tamamı global değişkenler olup hafızada fazlaca yer kaplayabilmektedir.

Geliştirilen yazılımlar yeniden kullanma, kolay hata giderme, verimli olma ve yazılım zorluğu özellikleri bakımından zayıftır. Buna karşın karmaşık olmayan kod yapısı özelliği bakımından güçlüdür. İşlemin gerçekleşmesi için hazırlanması gereken

kod satırları, fonksiyonel ve nesne tabanlı programlama tekniklerine göre daha uzun olabilmektedir.

### **B) Fonksiyonel – Prosedürel Programlama Tekniği**

Fonksiyonel programlamada yazılımlar birbirini çağıran bir dizi yordam ve işlevler topluluğu olarak geliştirilir. Her yordam ve işlev kendi yerel verisini kendi yerel değişkenlerinde tutar. Paylaşılması gereken veriler yordam çağırma komutlarında parametre olarak yordamdan yordama geçirilir. Parametrelere sığmayacak kadar büyük veriler ise genel (global) değişkenler içerisinde herkesin kullanımına açılır (Baransel, Mumcuoğlu; 2003 : 5).

Fonksiyonel programlamanın zayıf yönlerinden birisi genel kullanıma açılan verileri tümüyle korumasız bırakmasıdır. Verinin kullanım amacı veri üzerinde yapılabilecek işlemleri hiçbir biçimde sınırlamaz. Veriyi tutan değişken genel kullanıma açıldıktan sonra o değişken türünün desteklediği her türlü işlem veriyeye uygulanabilir. Amaç dışı kullanımdan kaynaklanan yanlışlıklar ortaya çıktıktan sonra, yanlışlığa neden olan program kesiminin saptanması genellikle zordur. Bunun için söz konusu veriye erişen tüm programların belirlenmesi ve tek tek incelenmesi gerekir. Fonksiyonel programlamada, eldeki kodun yeniden kullanımına ilişkin altyapı zayıftır.

### **C) Nesne Tabanlı Programlama Tekniği**

Nesneye yönelik programlama tekniği, fonksiyon ve prosedür kullanımına dayalı programlama tekniğine alternatif olarak geliştirilmiş bir yöntemdir. Nesneye yönelik programlamada uygulamalar birbirleriyle iletişim kurabilen nesnelere topluluğu olarak tasarlanır ve gerçekleştirilir. Kod işletimi nesnelere içinde yapılır ve her nesne bir diğer nesneye ileti göndererek ondan hizmet alabilir (Baransel, Mumcuoğlu; 2003 : 9).

Nesnelere, metotlar ve niteliklerden oluşur. Nitelikler nesnenin sahip olduğu verileri, metotlar ise bunlar üzerinde yapılabilecek işlemleri temsil etmektedir. Nesnelere genel özellikleri şunlardır (Baransel, Mumcuoğlu; 2003 : 9);

Nesnelere sınıflardan oluşmaktadır. Sınıflar, nesnenin oluşmasını sağlayan bilgileri içeren yapının adıdır. Bu nedenle sınıflar nesne şablonu olarak adlandırılır.

- 1) Nesnelere kalıcı ve geçici nitelikte olabilir. Nesnelere ana bellekte oluşan ve burada kullanılan yapılardır. Ana bellekte işleri tamamlandığında önlem alınmazsa yok olurlar.
- 2) Nesnelere niteliklerinde tutulan değerlerle tanımlanan durumları vardır. Aynı sınıfa bağlı nesnelere birbirinden ayıran unsur işleyiş biçimleri ya da yetenekleri değil içlerinde tuttukları veri değerleridir.
- 3) Nesnelere yaşam döngüleri vardır.

- 4) Nesnelere arayüzleriyle bilinir ve kullanılırlar. Arayüzler de nesnelere hangi hizmetleri sağladığını belirtir, ancak bunları nasıl yaptıkları konusunda bilgi içermez.

Yazılım platformunun kullanıcı sayısının artması durumunda yeni isteklerin ortaya çıkması kaçınılmazdır. Bu nedenle, yazılımın uzun süre ve çok sayıda kullanıcı tarafından kullanılması ve yazılımla ilgi eklenti, değişiklik, iyileştirme istekleri yazılımın iyi yöndeki sağlık göstergeleridir. Nesneye yönelik programlamada, yapılan değişikliklerin ana kalıp üzerindeki etkilerini kısıtlamak üzere aşağıdaki özellikler kullanılmaktadır.

- 1) *Kalıtım (inheritance)*: Eldeki sınıfları kullanarak yeni sınıflar oluşturabilme özelliğidir. Kalıtım özelliği eldeki kodun yeniden kullanımı ve değiştirilmesi konularında fonksiyonel programlamaya göre daha esnek bir yapı sunmaktadır. Örneğin; üst sınıftaki metotta değişiklik yapıldıkça tüm alt sınıf nesnelere bundan otomatik olarak yararlanır. Kalıtım mekanizması, üst sınıf nesnelere kullanıldığı her yerde alt sınıf nesnelere izin verir. Bu olanak Liskov'un yer değiştirme kuralı olarak bilinir. Bu kuralın uygulanabilmesi, alt sınıf nesnelere üst sınıf nesnelere ilişkin tüm metod çağrılarına yanıt verebilmesi ile sağlanabilmektedir (Baransel, Mumcuoğlu; 2003 : 11).
- 2) *Çok biçimlilik (polymorphism)*: Alt sınıf nesnelere üst sınıf nesnelere aynı temel özelliklere sahip nesnelere. Bu özellik çok biçimlilik olarak adlandırılır.
- 3) *Sarmalama (encapsulation)*: Veri ve işlevlerin tek bir nesne biçiminde tanımlanabilmesi özelliğidir.

## II) MATERYAL VE METOT

Bu çalışmada, e-dönüşümlerin temelini oluşturan ve sıklıkla kullanılan işlemlerin sunucular üzerinde çalışma süreleri incelenmiştir. Aşağıda ana başlıklar altında, e-dönüşümlerde sıklıkla kullanılan işlemler verilmiştir. Bu işlemler için oluşturulacak kod satırları hızlı ve verimli çalışma, pratik yazılmış olma, tekrar kullanılabilir olma ve güvenli olma özelliklerini taşımaktadır. Ana başlıklar altındaki bu işlemler şunlardır:

### A) Web Sayfalarından Veritabanlarıyla Etkileşime Geçme İşlemi

E-dönüşümde, sıklıkla kullanılan işlemlerden web sayfalarından veritabanlarıyla etkileşime geçmede veritabanlarına kayıt ekleme işlemi örnek işlem olarak tercih edilmiştir. Sunucu bilgisayarlar için MySQL host adı, kullanıcı adı ve şifre bilgileri tanımlanmıştır. Burada kullanılan şifrelerin uzunluğu ve veritabanı isimlerinin uzunluğu eşit tutulmuştur. Veritabanına erişene kadar geçen sürenin, böylece eşit

olması sağlanmıştır. Sunucular üzerinde oluşturulan veritabanına ait tablonun kayıt desenini oluşturan SQL kod düzeni şu şekildedir:

```
CREATE TABLE `islem1` (  
  `sira_no` int(5) NOT NULL auto_increment,  
  `sure` varchar(8) NOT NULL default '0',  
  PRIMARY KEY (`sira_no`)) TYPE=MyISAM AUTO_INCREMENT=1;
```

Örnek işlemin ilk aşamada, veritabanına erişim gerçekleştirilmiş, daha sonra her bir adımda veritabanına 63 bitlik bir verinin SQL komutları aracılığıyla sorgu halinde eklenmesi sağlanmıştır. İşlemin en sonunda veritabanı bağlantısı kesilmiştir. Kodların çalışma sürelerinin hesabına, veritabanı bağlantı işlemleri ve dosya açma ve kapatma işlemleri dahil edilmemiştir.

### **B) Dosya Sistemi ve Sunucularla Etkileşime Geçme İşlemi**

Dosya sistemi ve sunucularla etkileşime geçme işlemi için örnek işlem olarak, sunuculara dosya yükleme işlemi tercih edilmiştir. Kullanıcıların bilgisayarlarından web sunucularına dosya aktarılması, güvenlik, sunucuların hantallaşması gibi birçok istenmeyen durumu beraberinde getirecektir. Bu noktada, kullanıcıların göndereceği dosyaları test etmek gerekmektedir.

Bu işlemler gerçekleştirilirken İnternet Explorer programında bir HTML arayüz kullanılması gerekmektedir. Kullanıcı bilgisayara göndermek istediği dosyayı seçmek durumundadır. Bu noktada kullanıcının göndereceği dosyaları kontrol altında tutmak, dosya büyüklüğünü ve tipini kontrol etmek ve kullanıcıyı yönlendirmek güvenlik ve sunucu trafiği açısından önemsenmiştir. Sunucuya gönderilecek olan dosya tipi olarak text dosyalar tercih edilmiştir. Çünkü text dosyalar üzerinde işlemler hem kolay olmakta hem de dosya büyüklükleri az olmaktadır. Dosya büyüklüğü olarak 1000 byte değeri seçilmiştir. Buradaki amaç, sunucu trafiğini azaltmaktır ve internet bağlantı hızının, dosya transferine olan etkisini azaltmaktır.

Yapılan ilk işlem, HTML arayüzünden kullanıcıların 1000 byte değerinden büyük olmayan bir text dosyası seçmesidir. Bu aşamadan sonra, boyut ve tip kontrolleri gerçekleştirilmiştir. Kod çalışma süresi, dosyanın kullanıcının yerel bilgisayarından alınıp sunucu bilgisayara yüklenmesi işlemi için geçen süreyi temsil etmektedir. Burada, dosya kontrolleri kod çalışma süresine dahil edilmemiştir.

### **C) Web Ortamında Çözümsel Grafikler Üretme İşlemi**

Web ortamında çözümsel grafikler üretme işlemi, e-dönüşümlerde üzerinde hassasiyetle durulması gereken bir konudur. Veritabanlarından alınan verileri anlamlı hale getirmek, web ortamında grafikler üretme işlemine iyi bir örnektir. Burada, PNG formatlı grafikler oluşturma işlemi tercih edilen örnek işlemidir.

PNG grafikleri üretilirken, ilk olarak bir firmanın 12 ay boyunca cari hesap bilgileri tam sayı tipli dizi değişkene sabit olarak girilmiş ve değerler veritabanı bilgisi olarak alınmış daha sonra grafiklerin üretilmesi işlemine geçilmiştir. Veritabanı bilgisi olarak, dizi değişkenlerin tercih edilmesindeki neden ölçülecek bilginin grafik üretme süresi olmasındandır. Zaten kod çalışma süresine, dizi değişkenden verilerin çekilmesi işlemi dahil edilmemiştir. Bu istendiği takdirde gerçek bir veritabanı veya bir dosya kullanılabilir. Grafik üretme işleminde 280X230 piksel büyüklüğünde bir çalışma tuvali oluşturulmuştur. Bu tuvalin üzerinde dizi değişkenlerden alınan bilgilere göre çeşitli büyüklükte ve mavi renkte 12 adet dikdörtgen yerleştirilmiştir. Daha sonra oluşturulan grafiğin x eksenine 12 ayın bilgisini içerecek şekilde bu dikdörtgenler yerleştirilmiş ve her bir barın altına ilgili ayın numarası yazılmıştır. Y eksenini ise 4 eşit bölüme ayrılmış ve burada 0, 50, 100, 150, 200 bilgileri uygun yerler yazdırılmıştır. Her bölme için tuvale bir çizgi çizilmiştir.

#### **D) Ağ Ve Protokol Fonksiyonları**

Ağ ve protokol fonksiyonlarında örnek işlem olarak, eklentili e-postalar gönderme işlemi tercih edilmiştir. Bu tercihte FTP sunucularının yanıt sürelerinin uzunluğu ve e-dönüşümlerde e-posta gönderme kadar sık kullanılan bir işlem olmaması etkili olmuştur. E-dönüşümlerde, kullanıcılarla etkileşime geçmenin en kolay yolu e-postadır. Haber grupları, forumlar vb. işlemler için en önemli e-dönüşüm iletişim platformu da e-postadır.

E-posta gönderme işleminde, gönderilecek e-posta mesajlarına, 48X49 piksel büyüklüğünde ve 2,61 kilobyte dosya büyüklüğünde bir GIF resmi eklenmiştir. E-postanın, eklenti olarak aldığı dosya, daha önceden sunucu bilgisayara yüklenmiş ve kişisel bilgisayarlardan bu dosyanın transferi için geçecek süre ortadan kaldırılmıştır. E-posta gönderme işleminde postanın kimden geldiği, kime gideceği, konusu, metni ve postanın öncelik seviyesi bilgilerine dikkat edilmiştir. Her bir iterasyonda e-posta, aynı ve kullanılmakta olan bir e-posta adresine yollanmıştır. E-posta gönderme işleminde, sunucunun kodun çalışmasına ayırdığı süreler hesap edilirken, postanın kimden geldiği, kime gideceği, konusu, metni ve postanın öncelik seviyesi belirlenmesi işlemlerine dikkat edilmemiştir. Kodların çalışma süresine, e-postanın gönderilmesi için sunucunun harcadığı süre süre dahildir.

Bu çalışmada, materyal seçiminde; kod sistemlerinin yazılmasında, PHP, kod sistemlerinin çalıştırılmasında, Apache web sunucu programı, veritabanı sunucusu olarak, MySQL kullanılmıştır. Apache, MySQL ve PHP üçlüsünün tercih edilmesinde hem lisanslarının ücretsiz olması hem de tüm dünyada artan kullanımları nedeniyle tercih edilmiştir (Netcraft; 2006).

Alınan her örnek için üçer farklı kodlama sistemi geliştirilmiştir. Geliştirilen kodlama sistemleri üç farklı özellikteki sunucu bilgisayarlar üzerinde çalıştırılmıştır. Sunucu bilgisayarların birbirinden farklı seçimindeki amaç, üretilen kodlama



sisteminden elde edilecek sonuçların sadece kod düzenine bağlı olmasını sağlamaktır. Oluşturulan kodlar, Internet Explorer 6.0.2900.2180 128 bit sürümde çalıştırılacaktır ve sunucu bilgisayarlar erişim Microsoft Windows XP Service Pack 2 işletim sistemi yüklü bilgisayar ile gerçekleştirilecektir. Bu bilgisayarda ürün satıcısının belirttiği tüm güncelleştirmeler yüküldür. Sunucu bilgisayarlar erişim internet ile World Wide Web ortamında gerçekleştirilecektir. İnternet erişimi ADSL (Asymmetric Digital Subscriber Line) ile ve 512 Kbps (Kilobit Per Second) hızında yapılmaktadır.

Sunucu bilgisayarların detaylı özellikleri şunlardır:

- 1) Tip 1: Birinci sunucu bilgisayar Linux Redhat işletim sistemine sahip, 2.8 Ghz Dual Xeon işlemcili, 2 GB DDR2 RAM, SATA Disk, 100 Mbit hat çıkışı özelliklerine sahiptir.
- 2) Tip 2: İkinci sunucu bilgisayar Windows Server 2003 işletim sistemine sahip, 3.2 Ghz Dual Xeon 64 Bit Extreme Edition işlemcili, 2GB DDR2 RAM, SATA Disk, 100 Mbit hat çıkışı özelliklerine sahiptir.
- 3) Tip 3: Üçüncü sunucu bilgisayar Windows XP işletim sistemine sahip, 650 Mhz Pentium 3 işlemcili, 192 MB SDRAM, IDE Disk özelliklerine sahiptir.

Kodların çalışma sürelerinin, sunucu bilgisayarlar ve oluşturulan kod düzenine ne kadar bağlı olduğu ve oluşturulan kod düzenlerinden hangisinin daha iyi sonuç verdiği faktöriyel düzende varyans analizi tekniği ile değerlendirilmiştir. İstatistik analiz sonucunda, elde edilen ortalama değerleri mikro saniye cinsinden elde edildiği için küçük değerlikli ortalamalar daha hızlı işlem yapıldığının göstergesidir.

Ana başlıklardan örnek olarak seçilen dört adet işlemden her bir işlem için, kod düzeni faktörünün;

- Standart kod düzeni
- Fonksiyonel kod düzeni
- Nesne tabanlı kod düzeni olmak üzere 3 seviyesi vardır.

Sunucu bilgisayar tipi faktörünün ise;

- Linux işletim sistemli (Tip 1)
- Windows 2003 Server işletim sistemli (Tip 2)
- Windows XP işletim sistemli (Tip 3) olmak üzere 3 seviyesi vardır.

Seçilen tüm örnek işlemler için standart kod düzeninde, fonksiyonel kod düzeninde ve nesne tabanlı kod düzeninde olmak üzere 3 farklı kod düzeni geliştirilmiştir. Her bir kod düzeni, İnternet Explorer programında bir HTML arayüz aracılığıyla 1000'er kez çalıştırılmıştır. Tekrarlar, üç farklı özellikteki sunucu bilgisayarlar üzerinde, internet kullanımının en yoğun olduğu varsayılan (Birinci zaman aralığı: hafta içi gündüz saat 09:40) ve en az olduğu varsayılan zaman aralığında (İkinci zaman aralığı: hafta içi gece saat 02:00) toplam iki defa yapılmıştır. Her bir tekrarın ardından sunucu bilgisayarlar 10 mikro saniyelik gecikme süresi ayrılmıştır.

Gecikme süreleri işlemin gerçekleşme süresine dahil edilmemiştir. Elde edilen çalışma süresi verisi, ilgili kod düzenine ait bir metin dosyasına eklenmiştir. İki zaman aralığında elde edilen veriler sunucu bilgisayarlar üzerinde tutulmuştur. Bu dosyalardaki veriler analiz kurallarına uygun olarak MINITAB 14 paket programına aktarılmıştır.

Bu çalışmada istatistik analiz yöntemlerinden biri olan, varyans analizi kullanılmıştır. Varyans analizi deney düzenine bağlı olarak, elde edilen verilerin analizinde faktör sayısına ve faktörün seviyesine göre farklı şekillerde yararlanılan bir yöntemdir (Özdamar; 2002 : 420). Faktöriyel düzende varyans analizi tekniği ile yapılan hesaplamalar sonucunda istatistik olarak önemli olan ortalama farklılıklarının hangi alt grupların lehine olduğu, ortalamalar arasındaki farklılıkları ortalamaların sıralamadaki konumunu dikkate alarak değerlendirmeyi amaçlayan Duncan çoklu karşılaştırma testi ile saptanmıştır. Alt gruplardaki ortalama farklarının değerlendirilmesinde güven aralığı olarak  $p < 0,01$  seçilmiştir. Faktöriyel düzende varyans analizi tekniği ile yapılan hesaplamalar sonucunda, hangi ortalamalar arasında farklılığın olduğu ve Duncan testi sonucunda, muamele kombinasyonlarında, yani alt gruplarda hangi ortalamalar arasında farklılığın olduğu, betimsel istatistikler üzerinde aşağıdaki kurallara uygun olarak gösterilmiş ve yorumlanmıştır. Ortalama değerlerinin yanında yer alabilecek harfler:

- a harfi: En hızlı işlem süresini temsil etmektedir
- b harfi: Orta derecede hızlı işlem süresini temsil etmektedir
- c harfi: En yavaş işlem süresini temsil etmektedir

Her ortalama değerinin yanında yer alan aynı harf, ortalamalar arasında istatistik olarak önemli bir farklılığın olmadığını göstermektedir. Farklı harfler ise ortalamalar arasında istatistik olarak önemli bir farklılığın olduğunu göstermektedir. Bu farklılığın, hangi ortalama veya alt grup lehine olduğu ise harfin alfabeadaki önceliğine göre belirlenmektedir.

Ortalamaların yanında yer alan a ile b harflerinin oluşturduğu harf grubuna ait ortalama değer (ab), diğer iki ortalama değerleriyle (a ve b) istatistik olarak önemli bir farklılığın olmadığını ifade etmektedir. İki ortalama değeri arasındaki istatistik olarak önemli bir farklılık varsa, bu farklılığın, hangi ortalama veya alt grup lehine olduğu ise yine alfabeadaki önceliğine göre belirlenmektedir.

### **III) ARAŞTIRMA BULGULARI VE TARTIŞMA**

#### **A) Web Sayfalarından Veritabanlarıyla Etkileşime Geçme İşlemi (Veritabanına Kayıt Ekleme Örneği)**

Birinci ve ikinci zaman aralıklarında, ilk işlem olan veritabanına SQL komutlarının uygulanması işleminde, varyans analizi tekniğine ilişkin hesaplamalar sonunda sadece sunucu bilgisayar tipi faktörünün seviyelerinin ortalamaları arasındaki

farklar istatistik olarak önemli bulunmuştur ( $p<0,01$ ). Kod düzeni faktörünün seviyeleri arasındaki farklılıklar ve kod düzeni x sunucu bilgisayar tipi etkileşimleri ise istatistik olarak önemli değildir (Tablo : 1).

**Tablo: 1**  
**Veritabanına Kayıt Ekleme Örneği İçin Elde Edilen Varyans Analizi Sonuçları**

	Varyasyon Kaynağı	SD	Kareler Toplamı	Kareler Ortalaması	F	P
1. Zaman Aralığı	Kod Düzeni	2	49736590	24868295	1,59	0,204
	Sunucu Bilgisayar Tipi	2	356867579	178433790	11,40	0,000*
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	39953513	9988378	0,64	0,635
	Hata	8991	1,40689E+11	15647803	-	-
2. Zaman Aralığı	Kod Düzeni	2	18733376	9366688	1,27	0,280
	Sunucu Bilgisayar Tipi	2	592508769	296254384	40,24	0,000*
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	27056344	6764086	0,92	0,452
	Hata	8991	66197921632	7362687	-	-

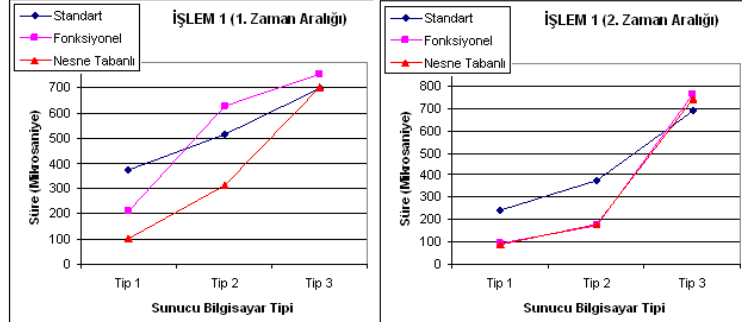
\* $p<0,01$

**Tablo: 2**  
**Veritabanına Kayıt Ekleme Örneği İçin Sunucu Bilgisayar Faktörüne Göre Oluşan Betimsel İstatistikler**

	Sunucu Bilgisayar Tipi	Tekrar Adedi	Ortalama	Ortalamaların Standart Hatası	Standart Sapma	En Düşük	En Yüksek
1. Zaman Aralığı	Tip 1	3000	230,0a	98,5	5393,0	76,8	280997,0
	Tip 2	3000	484,0b	72,3	3961,0	163	206295,0
	Tip 3	3000	717,3c	26,9	1474,0	500	50561,2
2. Zaman Aralığı	Tip 1	3000	142,0a	50,3	2756,0	77,0	199230
	Tip 2	3000	244,0a	66,4	3634,0	83,0	27521,9
	Tip 3	3000	730,2b	20,7	1132,3	85,1	35113,1

Tablo : 2'de ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere; Tip 1 sunucu bilgisayar diğer sunucu bilgisayar tiplerine göre örnek işlemi daha hızlı gerçekleştirmektedir. Sunucu bilgisayarların özellikleri dikkate alındığında Tip 1 sunucu bilgisayar, Tip 2 sunucu bilgisayara oranla işlemci hızlarının paralelinde bir

sonuç üretmemektedir. Daha yavaş olan Tip 1 sunucu bilgisayar, daha hızlı işlem gerçekleştirmektedir. Tip 3 sunucu bilgisayar ise, beklendik şekilde diğer sunucu bilgisayar tiplerine göre daha yavaş çalıştığı gözlenmiştir.



**Grafik: 1**

### Veritabanına Kayıt Ekleme Örneğinden Elde Edilen Ortalamaların Faktörlere Göre Değişimi

İkinci zaman aralığında tüm işlemlerin hızı, internet kullanımının yoğunluğuyla alakalı olarak beklendik şekilde artmıştır (Tablo : 2 ve Grafik : 1). MySQL veritabanıyla etkileşime geçme işleminde, kullanılan veya tercih edilen yazılımın, donanımdan daha etkili olabileceğidir. Tip 1 sunucu bilgisayarda kullanılan sistem yazılımı Linux, Tip 2 sunucu bilgisayarda kullanılan sistem yazılımı ise Windows 2003 Server'dır. Elde edilen sonuçlara göre MySQL veritabanlarıyla etkileşime geçme işleminde, Linux işletim sistemine sahip sunucular daha hızlıdır (Genel Ortalama Tip 1  $\bar{x} = 230$  ve  $\bar{x} = 142$  - Tip 2  $\bar{x} = 484$  ve  $\bar{x} = 244$ ).

### B) Dosya Sistemi ve Sunucularla Etkileşime Geçme (Sunuculara Dosya Yükleme Örneği)

Birinci ve ikinci zaman aralıklarında, varyans analizi tekniğine ilişkin hesaplamalar sonucunda sadece sunucu bilgisayar tipi faktörünün ortalamaları arasındaki farklar istatistik olarak önemli olmuşlardır ( $p < 0,01$ ). Kod düzeni faktörünün seviyeleri arasındaki farklılıklar ve kod düzeni x sunucu bilgisayar tipi etkileşimleri ise istatistik olarak önemli değildir (Tablo : 3).

**Tablo: 3**  
**Sunuculara Dosya Yükleme Örneği İçin Elde Edilen Varyans Analizi Sonuçları**

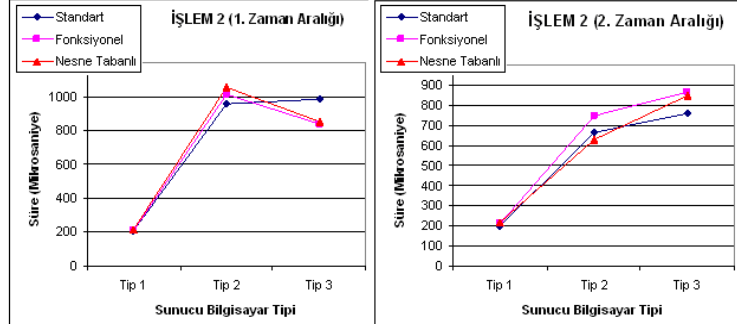
	Varyasyon Kaynağı	SD	Kareler Toplamı	Kareler Ortalaması	F	P
1. Zaman Aralığı	Kod Düzeni	2	1561821	780910	0,39	0,679
	Sunucu Bilgisayar Tipi	2	1106155284	553077642	273,83	0,000*
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	17003920	4250980	2,10	0,077
	Hata	8991	18159856525	2019782	-	-
2. Zaman Aralığı	Kod Düzeni	2	6601398	3300699	3,94	0,019
	Sunucu Bilgisayar Tipi	2	615592497	307796248	367,45	0,000*
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	6365426	1591357	1,90	0,108
	Hata	8991	7531436347	837664	-	-

\*p<0,01

**Tablo: 4**  
**Sunuculara Dosya Yükleme Örneği İçin Sunucu Bilgisayar Faktörüne Göre Oluşan Betimsel İstatistikler**

	Sunucu Bilgisayar Tipi	Tekrar Adedi	Ortalama	Ortalamaların Standart Hatası	Standart Sapma	En Düşük	En Yüksek
1. Zaman Aralığı	Tip 1	3000	210,7a	0,9	48,1	200,99	1679,9
	Tip 2	3000	1005,8c	32,5	1779,9	446,1	36447,1
	Tip 3	3000	889,2b	31,0	1700,3	624,9	34874,0
2. Zaman Aralığı	Tip 1	3000	210,7a	0,3	18,8	197,9	27802,9
	Tip 2	3000	679,0b	19,1	1045,9	203,9	25579,9
	Tip 3	3000	823,4c	21,8	1192,2	211,0	31588,1

Yapılan analiz sonucunda ve Tablo : 4'te ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere, ortalamalar arasındaki fark, iki zaman aralığında da her üç sunucu bilgisayar tipinde istatistik olarak önemli bir farklılık göstermektedir. Tip 1 sunucu bilgisayar, örnek işlemi, diğer sunucu bilgisayar tiplerine göre daha hızlı yapmaktadır. Tip 2 sunucu bilgisayar ikinci zaman aralığında daha hızlı işlem yapmıştır.



**Grafik:2**  
**Sunuculara Dosya Yükleme Örneğinden Elde Edilen Ortalamaların**  
**Faktörlere Göre Değişimi**

Grafik : 2’de görüldüğü gibi, sunuculara dosya yükleme işleminde Tip 1 sunucu bilgisayar tüm kod düzenlerinde Tip 2 ve Tip 3 sunucu bilgisayarlara göre daha hızlı çalışmaktadır (Genel Ortalama Tip 1  $\bar{x} = 210,7$  ve  $\bar{x} = 210,7$  - Tip 2  $\bar{x} = 1005,8$  ve  $\bar{x} = 679,0$ - Tip 3  $\bar{x} = 889,2$  ve  $\bar{x} = 823,4$ ). Tip 2 sunucu bilgisayar, sunuculara dosya yükleme işleminde internet kullanım yoğunluğundan etkilenmektedir. Tip 1 sunucu bilgisayar, bu işlem için her iki zaman aralığında da aynı ortalamayı üretmiştir. Bunun anlamı, MySQL veritabanıyla etkileşime geçme işleminde, kullanılan veya tercih edilen yazılımın, donanımdan daha etkili olabileceğidir. Dosya sistemi ve diğer sunucularla etkileşime geçme işleminde, Linux işletim sistemine sahip sunucular, sunucu bilgisayarların işlemci hızlarına bakılmaksızın daha hızlı çalıştığı söylenebilir.

### **C) Web Ortamında Çözümsel Grafikler Üretme İşlemi (PNG Formatlı Grafikler Üretme Örneği)**

Birinci ve ikinci zaman aralıklarında, üçüncü işlem olan web ortamında çözümsel grafikler üretilmesi işleminde, varyans analizi tekniğine ilişkin hesaplamalar sonucunda kod düzeni x sunucu bilgisayar tipi etkileşimi Tablo : 5’te de görüleceği gibi istatistik olarak önemli bulunmuştur ( $p < 0,01$ ).

**Tablo: 5**  
**PNG Formath Grafikler Üretme Örneği İçin Elde Edilen**  
**Varyans Analizi Sonuçları**

	Varyasyon Kaynağı	SD	Kareler Toplamı	Kareler Ortalaması	F	P
1. Zaman Aralığı	Kod Düzeni	2	10472456704	5236228352	1057,1	0,000
	Sunucu Bilgisayar Tipi	2	70170324211	350885162105	7083,07	0,000
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	6375044812	1593761203	321,75	0,000*
	Hata	8991	44535848598	4953381	-	-
2. Zaman Aralığı	Kod Düzeni	2	11347736399	5673868200	899,22	0,000
	Sunucu Bilgisayar Tipi	2	87507995471	43753997735	6934,33	0,000
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	6015731090	1503932772	238,35	0,000*
	Hata	8991	56731127380	6309768	-	-

\*p<0,01

**Tablo: 6**  
**Birinci Zaman Aralığında PNG Formath Grafikler Üretme İşlemi**  
**İçin Sunucu Bilgisayar Tipi ve Kod Düzeni Etkileşimine Göre Oluşan**  
**Betimsel İstatistikler (Faktör: Kod Düzeni)**

Faktör	Sunucu Bilgisayar Tipi	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Standart Kod Düzeni	Tip 1	1000	2339,9a	75,2	2378,4	2155,1	76731,0
	Tip 2	1000	2264,9a	39,8	1258,1	1884,0	29687,9
	Tip 3	1000	10696,0b	127,0	4009,0	9020,0	58119,0
Fonksiyonel Kod Düzeni	Tip 1	1000	925,4a	1,1	34,3	906,9	1604,1
	Tip 2	1000	1690,2b	40,4	1276,7	1077,2	22298,1
	Tip 3	1000	5772,1c	97,9	3094,6	4639,9	54333,0
Nesne Tabanlı Kod Düzeni	Tip 1	1000	1037,9a	4,3	136,3	992,1	1833,0
	Tip 2	1000	1358,6b	40,3	1274,4	1115,8	34272,0
	Tip 3	1000	6088,7c	91,7	2901,3	4866,8	41971,9

Tablo : 6'dan anlaşılacağı üzere; standart kod düzeninde, işlemin gerçekleşme süresi Tip 1 ve Tip 2 sunucu bilgisayarlarda istatistik olarak önemsizdirler. Beklendik şekilde Tip 3 sunucu bilgisayar hem Tip 1 hem de Tip 2 sunucu bilgisayara oranla işlemi daha yavaş yapmaktadır ve bu durum istatistik olarak önemlidir (Tip 1  $\bar{x}$  = 2339,9 – Tip 2  $\bar{x}$  = 2264,9 – Tip 3  $\bar{x}$  = 10696,0). Fonksiyonel kod düzeninde ortalamalar, her üç sunucu tipinde de istatistik olarak önemli farklılık göstermektedir ve Tip 1 sunucu bilgisayar örnek işlemi daha hızlı yapmaktadır (Tip 1  $\bar{x}$  = 925,4 – Tip 2  $\bar{x}$  = 1690,2 – Tip 3  $\bar{x}$  = 5772,1). Nesne tabanlı kod düzeninde de, ortalamalar arasında her üç sunucu bilgisayar tipinde de istatistik olarak önemli farklılık bulunmaktadır ve Tip 1 sunucu bilgisayar örnek işlemi daha hızlı yapmaktadır (Tip 1  $\bar{x}$  = 1037,9 – Tip 2  $\bar{x}$  = 1358,6 – Tip 3  $\bar{x}$  = 6088,7).

**Tablo: 7**  
**Birinci Zaman Aralığında PNG Formatlı Grafikler Üretme İşlemi İçin**  
**Sunucu Bilgisayar Tipi Ve Kod Düzeni Etkileşimine Göre Oluşan Betimsel**  
**İstatistikler (Faktör: Sunucu Bilgisayar Tipi)**

Faktör	Kod Düzeni	Tekrar	Ortalama	Ortalama. Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Tip 1	Standart	1000	2339,9b	75,2	2378,4	1884,0	76731,0
	Fonksiyonel	1000	925,4a	1,1	34,3	906,9	54333,0
	Nesne Tabanlı	1000	1037,9a	4,3	136,3	992,1	41971,9
Tip 2	Standart	1000	2264,9c	39,8	1258,1	1884,0	29687,9
	Fonksiyonel	1000	1690,2b	40,4	1276,7	1077,2	22298,1
	Nesne Tabanlı	1000	1358,6a	40,3	1274,4	1115,8	34272,0
Tip 3	Standart	1000	10696,0c	127,0	4009,0	9020,0	58119,0
	Fonksiyonel	1000	5772,1a	97,9	3094,6	4639,9	54333,0
	Nesne Tabanlı	1000	6088,7b	91,7	2901,3	4866,8	41971,9

Tablo : 7'den de anlaşılacağı üzere; Tip 1 sunucu bilgisayarda standart kod düzeni, hem fonksiyonel hem de nesne tabanlı kod düzenlerine göre daha yavaş çalışmaktadır. Bu durum istatistik olarak önemlidir. Nesne tabanlı kod düzeni ve fonksiyonel kod düzeni ortalamaları arasında istatistik olarak önemli bir farklılık yoktur (Standart  $\bar{x}$  = 2339,9 – Fonksiyonel  $\bar{x}$  = 925,4 – Nesne Tabanlı  $\bar{x}$  = 1037,9). Tip



2 sunucu bilgisayarda, her üç kod düzeni ortalamaları arasındaki fark istatistik olarak önemli bulunmuştur ve nesne tabanlı kod düzeniyle örnek işlem daha hızlı yapılmıştır (Standart  $\bar{x}$  =2264,9 – Fonksiyonel  $\bar{x}$  =1690,2 – Nesne Tabanlı  $\bar{x}$  =1358,6). Tip 3 sunucu bilgisayarda da her üç kod düzenine ait ortalamalar arasındaki fark istatistik olarak önemli bulunmuştur ve fonksiyonel kod düzeniyle örnek işlem daha hızlı yapılmaktadır (Standart  $\bar{x}$  =10696,0 – Fonksiyonel  $\bar{x}$  =5772,1 – Nesne Tabanlı  $\bar{x}$  =6088,7).

**Tablo: 8**  
**İkinci Zaman Aralığında PNG Formatlı Grafikler Üretme İşlemi İçin**  
**Sunucu Bilgisayar Tipi Ve Kod Düzeni Etkileşimine Göre Oluşan Betimsel**  
**İstatistikler (Faktör: Kod Düzeni)**

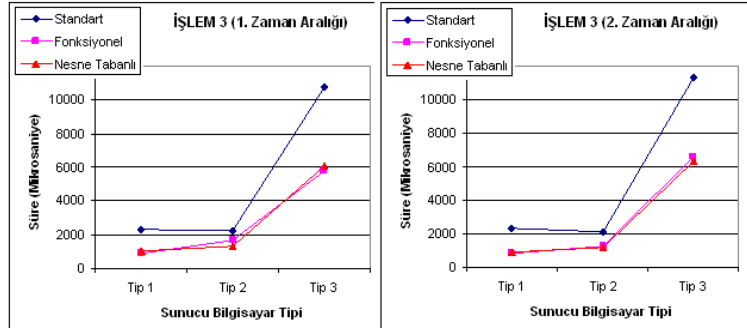
Faktör	Sunucu Bilgisayar Tipi	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Standart Kod Düzeni	Tip 1	1000	2361,9b	80,8	2554,9	2131,0	81550,1
	Tip 2	1000	2130,5a	34,8	1101,3	1869,0	18592,1
	Tip 3	1000	11285,0c	132,0	4185,0	8940,0	49997,0
Fonksiyonel Kod Düzeni	Tip 1	1000	875,7a	0,34	10,6	864,0	1096,0
	Tip 2	1000	1294,2b	24,8	785,1	1074,1	12646,0
	Tip 3	1000	6614,0c	147,0	4639,0	4624,0	67870,0
Nesne Tabanlı Kod Düzeni	Tip 1	1000	951,1a	0,3	10,3	940,1	1178,0
	Tip 2	1000	1192,1b	5,6	177,3	1117,0	3011,0
	Tip 3	1000	6344,5c	96,8	3060,3	4802,0	41035,9

İkinci zaman aralığında, Tablo : 8'den de anlaşılacağı üzere; standart kod düzenine ait ortalamalar arasındaki farklılıklar, her üç sunucu bilgisayar tipinde de istatistik olarak önemli bulunmuştur ve Tip 2 sunucu bilgisayar örnek işlemi daha hızlı yapmıştır (Tip 1  $\bar{x}$  = 2361,9 – Tip 2  $\bar{x}$  =2130,5 – Tip 3  $\bar{x}$  =11285,0). Fonksiyonel kod düzeninde de her üç sunucu bilgisayar tipine ait ortalama değerleri arasında istatistik olarak önemli bir farklılık bulunmuştur ve Tip 1 sunucu bilgisayar örnek işlemi daha hızlı yapmıştır (Tip 1  $\bar{x}$  = 875,7 – Tip 2  $\bar{x}$  =1294,2 – Tip 3  $\bar{x}$  =6614,0). Nesne tabanlı kod düzeninde de her üç sunucu bilgisayar tipine ait ortalama değerleri arasında istatistik olarak önemli farklılık bulunmuştur ve Tip 1 sunucu bilgisayar örnek işlemi diğer sunucu bilgisayar tiplerine göre daha hızlı yapmıştır (Tip 1  $\bar{x}$  = 951,1 – Tip 2  $\bar{x}$  =1192,1 – Tip 3  $\bar{x}$  =6344,5).

**Tablo: 9**  
**İkinci Zaman Aralığında PNG Formath Grafikler Üretme İşlemi İçin**  
**Sunucu Bilgisayar Tipi Ve Kod Düzeni Etkileşimine Göre Oluşan Betimsel**  
**İstatistikler (Faktör: Sunucu Bilgisayar Tipi)**

Faktör	Kod Düzeni	Tekrar	Ortalama	Ortalama. Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Tip 1	Standart	1000	2361,9b	80,8	2554,9	2131,0	81550,1
	Fonksiyonel	1000	875,7a	0,3	10,6	864,0	1096,0
	Nesne Tabanlı	1000	951,1a	0,3	10,3	940,1	1178,0
Tip 2	Standart	1000	2130,5b	34,8	1101,3	1869,0	18592,1
	Fonksiyonel	1000	1294,2a	24,8	785,1	1074,1	12646,0
	Nesne Tabanlı	1000	1192,1a	5,6	177,3	1117,0	3011,0
Tip 3	Standart	1000	11285,0c	132,0	4185,0	8940,0	49997,0
	Fonksiyonel	1000	6614,0b	147,0	4639,0	4624,0	67870,0
	Nesne Tabanlı	1000	6344,5a	96,8	3060,3	4802,0	41035,9

İkinci zaman aralığında, Tablo : 9'dan da anlaşılacağı üzere; Tip 1 sunucu bilgisayarda, standart kod düzeni hem fonksiyonel hem de nesne tabanlı kod düzenlerine göre daha yavaş çalışmaktadır (Standart  $\bar{x}$ =2361,9 – Fonksiyonel  $\bar{x}$ =875,7 – Nesne Tabanlı  $\bar{x}$ =951,1). Bu durum istatistik olarak önemlidir. Nesne tabanlı kod düzeni ve fonksiyonel kod düzeni ortalamaları arasında, istatistik olarak önemli bir farklılık yoktur. Tip 2 sunucu bilgisayarda da aynı durum söz konusudur (Standart  $\bar{x}$ =2130,5 – Fonksiyonel  $\bar{x}$ =1294,2 – Nesne Tabanlı  $\bar{x}$ =1192,1). Tip 3 sunucu bilgisayarda, her üç kod düzenine ait ortalama değerleri arasındaki fark istatistik olarak önemli bulunmuştur ve nesne tabanlı kod düzeni örnek işlemi daha hızlı yapmıştır (Standart  $\bar{x}$ =11285,0 – Fonksiyonel  $\bar{x}$ =6614,0 – Nesne Tabanlı  $\bar{x}$ =6344,5).



**Grafik: 3**  
**PNG Formatlı Grafikler Üretme Örneğinden Elde Edilen Ortalamaların Faktörlere Göre Değişimi**

Birinci zaman aralığında, PNG formatlı grafikler üretme örnek işleminde Tip 1 sunucu bilgisayar (Linux işletim sistemine sahip sunucu bilgisayar), daha hızlı işlem yapmaktadırlar. Linux işletim sisteminde bu örnek işlem fonksiyonel kod düzeniyle gerçekleştirildiğinde diğer kod düzenlerine göre daha hızlı işlem yapmaktadır. Şayet, Tip 2 sunucu bilgisayar (Windows 2003 Server işletim sistemine sahip sunucu bilgisayar) kullanılacak olursa, nesne tabanlı kod düzeni diğer kod düzenlerine göre daha iyi sonuç verdiği söylenebilir (Grafik : 3). İkinci zaman aralığında, Tip 1 sunucu bilgisayar (Linux işletim sistemine sahip sunucu bilgisayar), daha hızlı işlem yapmaktadırlar. Linux işletim sisteminde bu örnek işlem fonksiyonel kod düzeniyle veya nesne tabanlı kod düzeniyle gerçekleştirildiğinde, standart kod düzenine göre daha hızlı işlem yapmaktadırlar (Grafik : 3).

#### **D) Ağ ve Protokol Fonksiyonları (Eklentili E-Postalar Gönderme Örneği)**

Birinci ve ikinci zaman aralıklarında, dördüncü işlem olan ağ ve protokol fonksiyonları işleminde varyans analizi tekniğine ilişkin hesaplamalar sonucunda, kod düzeni x sunucu bilgisayar tipi interaksyonu Tablo : 10'da da görüleceği gibi istatistik olarak önemli bulunmuştur ( $p < 0,01$ ).

**Tablo: 10**  
**Eklentili E-Postalar Gönderme Örneği İçin Elde Edilen Varyans Analizi**  
**Sonuçları**

	Varyasyon Kaynağı	SD	Kareler Toplamı	Kareler Ortalaması	F	P
1. Zaman Aralığı	Kod Düzeni	2	6,98346E+12	3,49173E+12	27,81	0,000
	Sunucu Bilgisayar Tipi	2	5,48652E+13	2,74326E+13	218,50	0,000
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	2,17423E+12	5,43557E+11	4,33	0,002*
	Hata	8991	1,12880E+15	1,25548E+11	-	-
2. Zaman Aralığı	Kod Düzeni	2	2,53462E+12	1,26731E+12	16,38	0,000
	Sunucu Bilgisayar Tipi	2	3,57674E+13	1,78837E+13	231,08	0,000
	Kod Düzeni x Sunucu Bilgisayar Tipi	4	6,82036E+12	1,70509E+12	22,03	0,000*
	Hata	8991	6,95831E+14	77391942469	-	-

\*p&lt;0,01

**Tablo: 11**  
**Birinci Zaman Aralığında Eklentili E-Postalar Gönderme İşlemi İçin**  
**Sunucu Bilgisayar Tipi Ve Kod Düzeni Faktörlerine Göre Oluşan Belirtici**  
**İstatistikler (Faktör: Kod Düzeni)**

Faktör	Sunucu Bilgisayar Tipi	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Standart Kod Düzeni	Tip 1	1000	235344,0c	8283,0	261945,0	77109,0	3420095,0
	Tip 2	1000	64833,0a	7933,0	250869,0	12064,0	4726613,0
	Tip 3	1000	144939,0b	1869,0	59114,0	46796,0	1169753,0
Fonksiyonel Kod Düzeni	Tip 1	1000	290980,0b	15915,0	503289,0	81416,0	8443738,0
	Tip 2	1000	94004,0a	25290,0	799743,0	11885,0	24031936,0
	Tip 3	1000	261758,0b	2743,0	86748,0	89212,0	772116,0
Nesne Tabanlı Kod Düzeni	Tip 1	1000	250805,0c	7492,0	236927,0	77941,0	2279741,0
	Tip 2	1000	59745,0a	6034,0	190810,0	12133,0	2682039,0
	Tip 3	1000	204789,0b	1390,0	43962,0	156773,0	504283,0

Birinci zaman aralığında, Tablo : 11’de ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere; standart kod düzenine ait ortalama değerleri, her üç sunucu bilgisayar tipinde de istatistik olarak önemli bir farklılık göstermiştir ve örnek işlem Tip 2 sunucu bilgisayarda daha hızlı yapılmıştır (Tip 1  $\bar{x} = 253344,0$  – Tip 2  $\bar{x} = 64833,0$  – Tip 3  $\bar{x} = 144939,0$ ). Fonksiyonel kod düzeninde ortalamalar arasındaki farklar, Tip 1 ve Tip 3 sunucu bilgisayarda istatistik olarak önemsizdir; Tip 2 sunucu bilgisayar, hem Tip 1 hem de Tip 3 sunucu bilgisayara göre istatistik olarak önemli farklılık göstermektedir ve Tip 2 sunucu bilgisayar örnek işlemi daha hızlı yapmaktadır (Tip 1  $\bar{x} = 290980,0$  – Tip 2  $\bar{x} = 94004,2$  – Tip 3  $\bar{x} = 261758,0$ ). Nesne tabanlı kod düzeninde, ortalamalar arasında her üç sunucu bilgisayar tipinde de istatistik olarak önemli farklılık bulunmaktadır ve Tip 2 sunucu bilgisayar örnek işlemi daha hızlı yapmaktadır (Tip 1  $\bar{x} = 250805,0$  – Tip 2  $\bar{x} = 59745,0$  – Tip 3  $\bar{x} = 204789,0$ ).

**Tablo: 12**

**Birinci Zaman Aralığında Eklentili E-Postalar Gönderme İşlemi İçin Sunucu Bilgisayar Tipi Ve Kod Düzeni Faktörlerine Göre Oluşan Belirtici İstatistikler (Faktör: Sunucu Bilgisayar Tipi)**

Faktör	Kod Düzeni	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Tip 1	Standart	1000	235344,0a	82830,0	261945,0	77109,0	3420095,0
	Fonksiyonel	1000	290980,0b	15915,0	503289,0	81416,0	8443738,0
	Nesne Tabanlı	1000	250805,0a b	7492,0	236927,0	77941,0	2279741,0
Tip 2	Standart	1000	64833,0ab	7933,0	250869,0	12064,0	4726613,0
	Fonksiyonel	1000	94004,0b	25290,0	799743,0	11885,0	24031936,0
	Nesne Tabanlı	1000	59745,0a	6034,0	190810,0	12133,0	2682039,0
Tip 3	Standart	1000	144939,0a	1869,0	261945,0	77109,0	3420095,0
	Fonksiyonel	1000	261758,0c	2743,0	86748,0	89212,0	772116,0
	Nesne Tabanlı	1000	204789,0b	1390,0	43962,0	156773,0	504283,0

Birinci zaman aralığında, Tablo : 12’de ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere; Tip 1 sunucu bilgisayarda elde edilen ortalama değerleri, nesne tabanlı kod düzeninin, fonksiyonel ve standart kod düzenleriyle

istatistik olarak önemli bir farklılığının olmadığını göstermektedir. Buna karşın, standart kod düzeni ile fonksiyonel kod düzenine ait ortalama değerleri arasında istatistik olarak önemli farklılık bulunmaktadır ve standart kod düzeni örnek işlemi daha hızlı yapmaktadır (Standart  $\bar{x}$  =235344,0 – Fonksiyonel  $\bar{x}$  =290980,0 – Nesne Tabanlı  $\bar{x}$  =250805,0). Tip 2 sunucu bilgisayarda elde edilen ortalama değerleri, standart kod düzeninin, fonksiyonel ve nesne tabanlı kod düzenleriyle istatistik olarak önemli bir farklılığının olmadığını göstermektedir. Buna karşın, nesne tabanlı kod düzeni ile fonksiyonel kod düzenine ait ortalama değerleri arasında istatistik olarak önemli farklılık bulunmaktadır ve nesne tabanlı kod düzeni örnek işlemi daha hızlı yapmaktadır (Standart  $\bar{x}$  =64833,0 – Fonksiyonel  $\bar{x}$  =94004,0 – Nesne Tabanlı  $\bar{x}$  =59745,0). Tip 3 sunucu bilgisayarda da her üç kod düzeni ortalamaları arasındaki fark istatistik olarak önemli bulunmuştur ve standart kod düzeni örnek işlemi daha hızlı yapmaktadır (Standart  $\bar{x}$  =144939,0 – Fonksiyonel  $\bar{x}$  =261758,0 – Nesne Tabanlı  $\bar{x}$  =204789,0).

**Tablo: 13**

**İkinci Zaman Aralığında Eklentili E-Postalar Gönderme İşlemi İçin Sunucu Bilgisayar Tipi Ve Kod Düzeni İnteraksiyonuna Göre Oluşan Belirtici İstatistikler (Faktör: Kod Düzeni)**

Faktör	Sunucu Bilgisayar Tipi	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Standart Kod Düzeni	Tip 1	1000	163860,0b	3427,0	108381,0	80393,0	1610063,0
	Tip 2	1000	61373,0a	21636,0	684198,0	11770,0	17636419,0
	Tip 3	1000	144121,0b	2486,0	78630,0	34608,0	949091,0
Fonksiyonel Kod Düzeni	Tip 1	1000	286995,0c	8060,0	254878,0	94349,0	3258769,0
	Tip 2	1000	52254,0a	5894,0	186371,0	11850,0	2491603,0
	Tip 3	1000	149971,0b	2136,0	67558,0	34898,0	421835,0
Nesne Tabanlı Kod Düzeni	Tip 1	1000	178880,0b	4498,0	142238,0	83932,0	1529888,0
	Tip 2	1000	58151,0a	8925,0	282222,0	12031,0	4575540,0
	Tip 3	1000	167155,0b	2516,0	79561,0	49715,0	1572182,0

İkinci zaman aralığında, Tablo : 13'te ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere; standart kod düzenine ait ortalama değerleri arasındaki farklılıklar, Tip 1 ve Tip 3 sunucu bilgisayarda istatistik olarak önemsiz bulunurken;

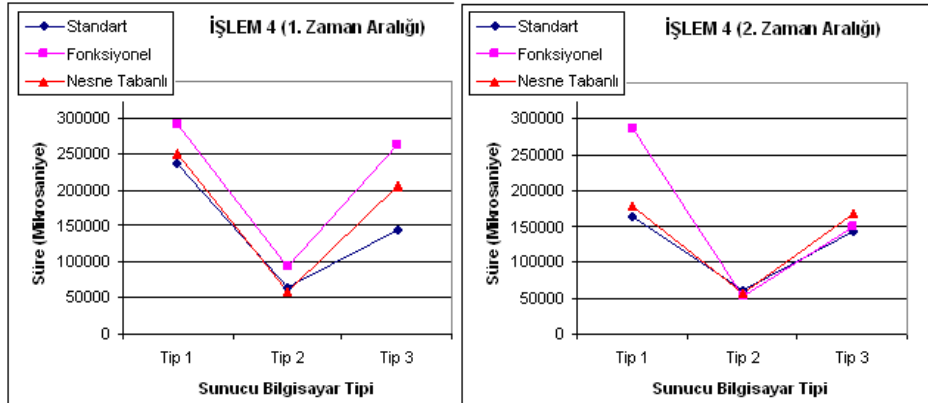
Tip 2 sunucu bilgisayara ait ortalama, hem Tip 1 hem de Tip 3 sunucu bilgisayarlarda elde edilen ortalamalar ile istatistik olarak önemli farklılık göstermiştir ve Tip 2 sunucu bilgisayar örnek işlemi daha hızlı yapmıştır (Tip 1  $\bar{x} = 163860,0$  – Tip 2  $\bar{x} = 61373,0$  – Tip 3  $\bar{x} = 144121,0$ ). Fonksiyonel kod düzeninde elde edilen ortalamalar arasındaki fark, her üç sunucu bilgisayar tipinde de istatistik olarak önemli bulunmuştur ve Tip 2 sunucu bilgisayar örnek işlemi daha hızlı yapmıştır (Tip 1  $\bar{x} = 286995,0$  – Tip 2  $\bar{x} = 52254,0$  – Tip 3  $\bar{x} = 149971,0$ ). Nesne tabanlı kod düzeninde elde edilen ortalamalar, Tip 1 ve Tip 3 sunucu bilgisayarda istatistik olarak önemsizdir. Buna karşın Tip 2 sunucu bilgisayar hem Tip 1 hem de Tip 3 sunucu bilgisayara göre örnek işlemi daha hızlı yapmıştır ve bu durum istatistik olarak önemli bulunmuştur (Tip 1  $\bar{x} = 178880,0$  – Tip 2  $\bar{x} = 58151,0$  – Tip 3  $\bar{x} = 167155,0$ ).

**Tablo: 14**  
**İkinci Zaman Aralığında Eklentili E-Postalar Gönderme İşlemi İçin**  
**Sunucu Bilgisayar Tipi Ve Kod Düzeni İnteraksiyonuna Göre Oluşan Belirtici**  
**İstatistikler (Faktör: Sunucu Bilgisayar Tipi)**

Faktör	Kod Düzeni	Tekrar	Ortalama	Ortalama Standart Hatası	Standart Sapma	En Düşük	En Yüksek
Tip 1	Standart	1000	163860,0a	3427,0	108381,0	80393,0	1610063,0
	Fonksiyonel	1000	286995,0b	8060,0	254878,0	94349,0	3258769,0
	Nesne Tabanlı	1000	178880,0a	4498,0	142238,0	83932,0	1529888,0
Tip 2	Standart	1000	61373,0a	21636,0	684198,0	11770,0	17636419,0
	Fonksiyonel	1000	52254,0a	5894,0	186371,0	11850,0	2491603,0
	Nesne Tabanlı	1000	58151,0a	8925,0	282222,0	12031,0	4575540,0
Tip 3	Standart	1000	144121,0a	2486,0	78630,0	34608,0	949091,0
	Fonksiyonel	1000	149971,0a	2136,0	67558,0	34898,0	421835,0
	Nesne Tabanlı	1000	167155,0a	2516,0	79561,0	49715,0	1572182,0

İkinci zaman aralığında, Tablo : 14'te ortalama değerlerinin yanında yer alan harflerden de anlaşılacağı üzere; Tip 1 sunucu bilgisayarda elde edilen ortalama değerleri arasında, standart ve nesne tabanlı kod düzenlerinde, istatistik olarak önemli bir fark bulunmuştur. Fonksiyonel kod düzeni, hem standart hem de nesne tabanlı kod düzenlerine göre örnek işlemi daha yavaş yapmıştır ve bu durum istatistik olarak önemli bulunmuştur (Standart  $\bar{x} = 163860,0$  – Fonksiyonel  $\bar{x} = 286995,0$  – Nesne

Tabanlı  $\bar{x}=178880,0$ ). Tip 2 sunucu bilgisayarda elde edilen ortalama değerleri arasında her üç kod düzeninde de istatistik olarak önemli bir farklılık bulunmamıştır (Standart  $\bar{x}=61373,0$  – Fonksiyonel  $\bar{x}=52254,0$  – Nesne Tabanlı  $\bar{x}=58151,0$ ). Tip 3 sunucu bilgisayarda da aynı durum söz konusudur (Standart  $\bar{x}=144121,0$  – Fonksiyonel  $\bar{x}=149971,0$  – Nesne Tabanlı  $\bar{x}=167155,0$ ).



Grafik: 4

#### Eklentili E-Postalara Göndermen Örneğinden Elde Edilen Ortalamaların Faktörlere Göre Değişimi

Grafik : 4’de görüleceği gibi, her iki zaman aralığında da, eklentili e-postalar gönderme örnek işleminde, Tip 2 sunucu bilgisayar (Windows 2003 Server işletim sistemine sahip sunucu bilgisayar) daha hızlı işlem yapmaktadır. Windows 2003 Server işletim sisteminde bu örnek işlem nesne tabanlı kod düzeniyle gerçekleştirildiğinde diğer kod düzenlerine göre daha hızlı işlem yapmaktadır. Eklentili e-postalar gönderme işleminde, sunucu bilgisayarların işlemci hızı ve kullanılan e-posta sunucusunun o anki işlem yoğunluğu, kullanılan filtreler ve güvenlik duvarı sistemleri dikkate alınmalıdır.

#### SONUÇLAR

Elde edilen bulgular ve yapılan istatistik değerlendirmeler sonucunda, e-dönüşümler için web sayfalarının programlanmasında işlemi yapan kod düzenlerinin ve kullanılacak sunucu tiplerinin işlemin gerçekleşmesine olan etkisine ilişkin aşağıdaki sonuçlar elde edilmiştir:

1) MySQL veritabanına bağlanma ve SQL sorgularının çalıştırılmasında en hızlı sunucu tipi Linux işletim sistemini kullanan sunuculardır. İşlemin gerçekleşme süresinde, sunucu bilgisayarın donanım özellikleri, kullanılan sistem ve uygulama



yazılımları kadar etkili değildir. Yine SQL sorgularının çalıştırılmasında, web sayfaları programlanırken nesne tabanlı kod düzeninin kullanılması diğer kod düzenlerine göre daha hızlı işlem yapılmasına neden olmaktadır.

2) Dosya sistemi ve sunucularla etkileşime geçme işleminde, en hızlı sunucu tipi Linux işletim sistemini kullanan sunuculardır. İşlemin gerçekleşme süresinde, sunucu bilgisayarın donanım özellikleri, kullanılan sistem ve uygulama yazılımları kadar etkili değildir. Bu işlemin programlanmasında kod düzenlerinin işlemin gerçekleşmesine olan etkisi yoktur. Her üç kod düzeninin de bu işlem için kullanılması uygundur.

3) Web ortamında çözümsel grafikler üretme işleminde en hızlı sunucu tipi Linux işletim sistemine sahip sunuculardır. İşlemin gerçekleşme süresinde, sunucu bilgisayarın donanım özellikleri, kullanılan sistem ve uygulama yazılımları kadar etkili değildir. Bu işlemin programlanmasında nesne tabanlı kod düzeninin kullanılması veya fonksiyonel kod düzeninin kullanılması işlemin gerçekleşme hızını arttırmaktadır.

4) Ağ ve protokol fonksiyonları işlemlerinde en hızlı işlem yapan sunucular Windows 2003 Server işletim sistemine sahip sunuculardır. İşlemin gerçekleşme süresinde, sunucu bilgisayarın donanım özellikleri, kullanılan sistem ve uygulama yazılımları kadar etkilidir. Kod düzenlerinin işlemin gerçekleşme süresine olan etkisi incelendiğinde; nesne tabanlı kod düzeninin kullanılması diğer kod düzenlerinden daha hızlı işlem yapılmasını sağlamaktadır.

5) Genel olarak, e-dönüşümlerde tercih edilmesi gereken sistemler üç katmanlı ve web tabanlı mimarilere uyan sistemlerdir. Bu sistemler bir istemci, bir sunucu, bir veritabanı sunucusundan oluşmaktadır. Üç katmanlı mimaride sunucu katmanında sistem yazılımı olarak Linux işletim sisteminin kullanılması, oluşturulan web programlarında kod düzeni olarak nesne tabanlı kod düzeninin tercih edilmesi işlemlerin gerçekleşme hızını arttırmaktadır.

6) E-dönüşümler için web sayfalarının tasarlanmasında modüler yapı tercih edilmelidir. Hiyerarşik düzeni önceden belirlenmiş, dengelenmiş organizasyon modellemesi tercih edilmelidir. Kullanıcılar, internet platformunun kullanımında bilgi yerleşim düzeninde ve bilgiye erişimde zorluk çekmektedir. Bir e-dönüşüm platformunun başarısında tasarımın rolü bu bakımdan önemlidir.

#### **KAYNAKÇA**

ARİFOĞLU, A. (2004), E-Dönüşüm Yol Haritası, Dünya, Türkiye, SAS Bilişim Yayınları, ss. 9-13, 16-26, 40-48, Ankara.

BARANSEL, C. ve MUMCUOĞLU, A. (2003), Web Tabanlı, Üç Katmanlı Yazılım Mimarileri, SAS Bilişim Yayınları, ss. 5-11.

CLOSING THE DIGITAL DIVIDE: AN INITIAL REVIEW. (1999),  
<http://www.interpolicy.org/briefing/ErnestWilson0700.htm> (10.12.2005).

DEVLET PLANLAMA TEŞKİLATI, BİLGİ TOPLUMU DAİRESİ BAŞKANLIĞI. (2005),  
<http://www.bilgitoplumu.gov.tr/edtr.asp> (10.12.2005).

DEVLET PLANLAMA TEŞKİLATI, BİLGİ TOPLUMU DAİRESİ BAŞKANLIĞI, E-İMZA  
KANUNU (23.01.2004 tarih ve 25355 sayılı Resmi Gazete), (2006)  
[http://www.bilgitoplumu.gov.tr/mevzuat/20040123\\_eimza.pdf](http://www.bilgitoplumu.gov.tr/mevzuat/20040123_eimza.pdf) (15.03.2006).

E-DÖNÜŞÜM TÜRKİYE İCRA KURULU STK İZLEME KOMİTESİ (2004). “E-dönüşüm  
Türkiye İcra Kurulu STK İzleme Komitesi Raporu”, *TBD, TBV, TUBİSAD Rapor No:1*, ss.  
1-8, İstanbul.

NETCRAFT (2006), [http://news.netcraft.com/archives/web\\_server\\_survey.html](http://news.netcraft.com/archives/web_server_survey.html) (10.04.2006).

OKTAY, E., BALKANLI, A.O. ve SALEPÇİOĞLU, A. (2004), “Bilgi Toplumunda Yeni  
Ekonomi ve E-dönüşüm Stratejileri”, *3. Ulusal Bilgi, Ekonomi ve Yönetim  
Kongresi*, ss. 157 – 159, İzmir.

ÖZDAMAR, K. (2002), *Paket Programlar ile İstatistiksel Veri Analizi*, Kaan Kitabevi, ss. 420-  
421, Eskişehir,

TELEKOMÜNİKASYON KURUMU. (2005), “Sayısal Uçurumun Önlenmesi: Stratejik Plan”  
[http://www.tk.gov.tr/Yayin/Raporlar/pdf/Sayisal\\_Ucurumun\\_onlenmesi.pdf](http://www.tk.gov.tr/Yayin/Raporlar/pdf/Sayisal_Ucurumun_onlenmesi.pdf), ss.25  
(20.08.2005)

TÜRKİYE BİLİŞİM DERNEĞİ KAMU – BİB 2. ÇALIŞMA GRUBU (2005), “E-İmzanın  
Toplumsal Boyutu”, *Kamu Bilişim Platformu VII*, ss.11-12, Antalya.

YILMAZ, A.E. (2001), *Web Sayfalarında Ana Sayfaların (Home Pages) Estetik Belirleyicileri*,  
Ondokuz Mayıs Üniversitesi Sosyal Bilimler Enstitüsü Yayınlanmamış Yüksek Lisans  
Tezi, Samsun.